Jurnal TRANSFORMATIKA
Vol.23, No.2, Januari 2026, pp. 202 - 212
P-ISSN: 1693-3656, E-ISSN: 2460-6731
https://journals.usm.ac.id/index.php/transformatika/

Klasifikasi Akun Palsu Pada Pengikut Akun Rental Cosplay Averentcos Di Instagram Menggunakan Metode *K-Nearest Neighbour (KNN)*

Hafidz Mufrodi, S.Kom.[1], Nur Wakhidah, S.Kom., M.Cs.[2*], Prind Triajeng Pungkasanti, S.Kom, M.Kom.[3]
[1]Fakultas Teknologi Informasi dan Komunikasi, Universitas Semarang
Jl. Soekarno Hatta, RT.7/RW.7, Tlogosari Kulon, Kec. Pedurungan, Kota Semarang, Jawa Tengah 50196, e-mail: hafidzmufrodi@gmail.com
[2]Fakultas Teknologi Informasi dan Komunikasi, Universitas Semarang
Jl. Soekarno Hatta, RT.7/RW.7, Tlogosari Kulon, Kec. Pedurungan, Kota Semarang, Jawa Tengah 50196, e-mail: ida@usm.ac.id
[3]Fakultas Teknologi Informasi dan Komunikasi, Universitas Semarang
Jl. Soekarno Hatta, RT.7/RW.7, Tlogosari Kulon, Kec. Pedurungan, Kota Semarang, Jawa Tengah 50196, e-mail: prind@usm.ac.id

ARTICLE INFO

History of the article :
Received 24 Oktober 2025
Received in revised form 14 November 2025
Accepted 21 Januari 2026
Available online 30 Januari 2026

Keywords:

Social Media; K-Nearest Neighbor (KNN) Algorithm; Fake Account Analysis

*** Correspondece:**

Telepon:
-

E-mail:
ida@usm.ac.id

ABSTRACT

Technological advances have given business people several options to develop their business. One of them is using Instagram social media as a promotional tool. However, using social media as a promotional medium has its own problems. Fake accounts spread on Instagram can reduce the reach of business accounts. Until now, there are more than millions of fake accounts. Instagram continues to increase its efforts to detect and delete these fake accounts. This study was conducted to be able to classify accounts suspected of being fake accounts on the followers of the averentcos account using the k-NN method. The data used were 500 follower accounts with various backgrounds. This study used several variables, namely Profile Photo, Username Length, Number of Name Words, Similarity of Name to Username, Bio Length, External Links, Public Accounts, Number of Posts, Number of Followers, Number of Followed so that accuracy of 86,666% can be achieved.

INTRODUCTION

Platform media sosial online beberapa tahun terakhir telah menjadi elemen utama dalam perubahan ini. Latar belakang puncak dari fenomena ini adalah pertumbuhan eksponensial pengguna internet di Indonesia. Pada saat yang sama, populasi pemuda yang besar di Indonesia, yang sangat terhubung dengan teknologi dan media sosial, telah memperkuat peran penting platform-platform ini dalam kehidupan sehari-hari [1]. Manfaat platform media sosial online di

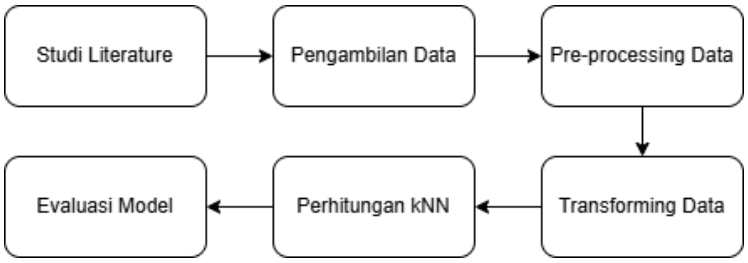
Indonesia sangat bervariasi dan mencakup berbagai aspek kehidupan. Salah satu manfaat utamanya adalah kemudahan dalam berkomunikasi dan berinteraksi dengan orang lain [2]. Media sosial memungkinkan individu untuk tetap terhubung dengan teman, keluarga, dan rekan bisnis mereka, terlepas dari jarak geografis. Terutama ketika keluarga atau teman berada di luar kota atau bahkan luar negeri. Oleh karena itu, platform media sosial juga memberikan peluang bisnis yang besar di Indonesia [3]. Dalam era digital yang terus berkembang, media sosial dan platform online Instagram telah menjadi sarana komunikasi dan pertukaran informasi yang sangat vital dalam kehidupan sehari-hari. Banyak orang bergantung pada platform ini untuk berinteraksi, berbagi gagasan, bisnis, dan mendapatkan informasi [4].

Akun palsu di media sosial yang mengincar pelaku usaha bisa menjadi ancaman yang cukup serius. Akun palsu sering digunakan untuk tujuan penipuan dan penggelapan [5]. Jasa sewa rental kostum cosplay saat ini sedang berkembang pesat melalui media sosial. Seluruh transaksi yang dilakukan hanya melalui media online sehingga pelaku usaha rentan mendapatkan orderan dari pihak yang tidak bertanggung jawab. Akun palsu tanpa kredibilitas yang sangat banyak kerap kali digunakan untuk mendapatkan kepercayaan kepada pihak rental kostum dengan cara order sampling melalui banyak akun palsu supaya pemilik rental cosplay lengah dan memberikan orderan jasa sewa kostumnya. Oleh karena itu, deteksi dan identifikasi akun palsu menjadi sangat penting dalam menjaga integritas dan keamanan platform online [6]. Pemanfaatan teknik *machine learning*, terutama algoritma *K-Nearest Neighbour (kNN)*, telah terbukti efektif dalam mengatasi masalah klasifikasi dan deteksi, banyak penelitian menggunakan algoritma ini untuk menyelesaikan berbagai macam masalah dari data yang mudah hingga data yang kompleks [7] termasuk dalam mengidentifikasi akun palsu di media sosial. *kNN* adalah algoritma yang mampu menghasilkan model yang baik dalam memisahkan dua kelas berbeda dalam data, dalam hal ini, antara akun yang asli dan akun palsu [8].

Penelitian ini bertujuan untuk mengembangkan dan meningkatkan metode deteksi akun palsu pada akun Instagram Averentcos dengan menggunakan *kNN*. Dengan fokus pada aplikasi yang relevan dalam platform media sosial tertentu. Penelitian ini akan menyelidiki fitur-fitur yang relevan untuk deteksi akun palsu, seperti jenis nama pengguna, status pengguna, pola posting, dan lainnya. Dengan demikian, penelitian ini diharapkan akan memberikan kontribusi dalam upaya menjaga keamanan dan integritas platform online serta membantu pengguna untuk mengidentifikasi akun palsu dengan lebih efektif. Penelitian ini memiliki potensi untuk memberikan manfaat nyata dalam melindungi pengguna media sosial dari potensi ancaman yang ditimbulkan oleh akun palsu, serta meningkatkan kepercayaan dan keamanan dalam berpartisipasi dalam platform- platform online yang semakin penting dalam kehidupan saat ini.

RESEARCH METHODS

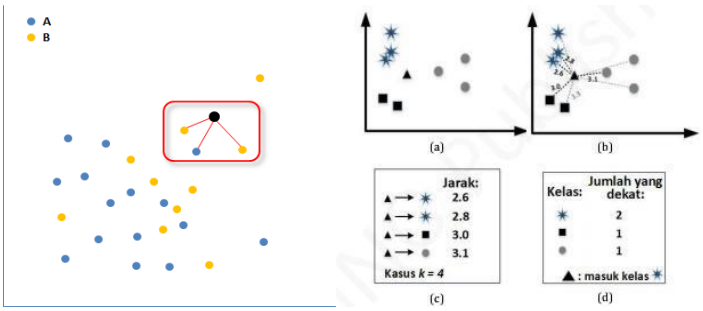
Metodologi penelitian digunakan untuk mendapatkan data dengan tujuan tertentu. Metode ini mencakup langkah-langkah sistematis dan logis yang dapat diikuti oleh peneliti untuk mengumpulkan dan menganalisis data guna mencapai kesimpulan yang valid. Penelitian ini menggunakan algoritma *k-NN*. *k-Nearest neighbor* atau *k-NN* adalah metode yang bekerja dengan mencari sejumlah *k* objek data atau pola yang paling dekat dengan pola masukan, kemudian memilih *class* dengan jumlah terbanyak diantara *k* pola tersebut [9]. Langkah-langkah yang dilakukan adalah sebagai berikut:



Gambar 1. Tahapan Penelitian

1. Studi Literature

Algoritma *K-Nearest Neighbor (kNN)* adalah salah satu metode *machine learning* yang digunakan untuk mengklasifikasikan suatu data baru berdasarkan data-data yang sudah ada (*train set*) [10]. Algoritma ini akan mencari *k* data terdekat dari data baru tersebut, lalu melihat kelas dari *k* data terdekat itu. Kemudian data baru akan diklasifikasikan ke dalam kelas yang paling banyak muncul diantara *k* data terdekat tersebut.



Gambar 2. Gambaran Algoritma *k-NN*

Gambar 2.2.a sebagai contoh akan dicari apakah data tes □ akan akan dicari apakah data tes masuk * atau □ atau o? Gambar 2.2b) Metode *kNN* menghitung jarak antara data tes dengan semua data latih. Pada gambar terlihat perhitungan yang menghasilkan jarak 2.6, 2.8, 3.0, 3.1, dan 3.3. Gambar 2.2c) pada contoh ini digunakan nilai k=4 sehingga diurutkan empat jarak yang dekat sesuai rankingnya. Terlihat bahwa jarak paling dekat adalah 2.6. Gambar 2.2d) Pada langkah ini dihitung berapa jumlah label yang dekat pada kelas-kelas tersebut. Terlihat ada 2 label pada kelas *, 1 label pada kelas □, dan 1 label pada kelas o. Dari sini terlihat bahwa data tes diklasifikasikan ke kelas bintang *. Demikian gambaran metode *kNN*.

Penentuan nilai *k* dalam algoritma klasifikasi *K-Nearest Neighbor* dapat dicari berdasarkan nilai sampel *k* terdekat *k1, k2, ks*. Semakin banyak data yang ada semakin kecil jumlah *k* yang dipilih, sedangkan jika ukuran dimensi data yang ada lebih besar, jumlah *k* yang dipilih harus lebih tinggi. Dalam menentukan nilai *k*, lebih baik menggunakan angka ganjil seperti $k = 1, 3, 5, \ldots$, dst. Nilai *k* harus memenuhi syarat yaitu $k < N$ dimana N merupakan jumlah dari dataset latih, karena nilai *k* digunakan untuk mencari jumlah mayoritas dari kelas/label pada data latih maka nilai *k* tidak boleh lebih dari jumlah dataset latih. Mencari tetangga terdekat atau jarak pada algoritma *k-NN* terdapat 3 cara umum yaitu Jarak *Euclidean*, Jarak *Manhattan*, dan Jarak *Minkowski* [11].

2. Data Collection

Data yang digunakan dalam penelitian ini bersumber dari website *opensource* Kaggle https://www.kaggle.com/datasets/free4ever1/instagram-fake-spammer-genuine-accounts untuk digunakan sebagai data training. Kemudian bersumber dari pengikut akun Instagram https://www.instagram.com/averentcos/ yang digunakan untuk data test pada model *KNN* di google collab. Data tersebut menggunakan berbagai variable yang mencerminkan sebuah akun di Instagram. Data training yang digunakan sebanyak 576 data, sedangkan data test yang digunakan sebanyak 120 yang dikumpulkan melalui proses scraping menggunakan tool extension chrome IG Exporter. Variabel data yang digunakan adalah berikut:

1) Profile pic = gambar profil atau tidak? (0 = tidak punya; 1 = punya)
2) nums/lenght username = jumlah karakter numerik dalam nama pengguna
3) fullname words = jumlah kata dalam nama lengkap
4) num/lenght fullname = jumlah karakter numerik dalam nama lengkap

5) name==username = nama pengguna sama dengan nama lengkap (1 = benar; 0 = salah)
6) description length = jumlah karakter dalam Bio
7) external URL = memiliki URL eksternal dalam Bio (1 = ada; 0 = tidak ada)
8) private = profil bersifat pribadi (1 = ya; 0 = tidak)
9) #posts = jumlah publikasi
10) #followers = jumlah pengikut
11) #follows = jumlah profil yang diikuti pengguna
12) fake = variabel target (0 = tidak palsu; 1 = palsu)

3. Data Preprocessing

Sebelum data bisa digunakan, dilakukan beberapa *preprocessing* data. Tujuan utama dari tahap ini adalah untuk memastikan bahwa data memiliki sifat yang akurat, lengkap, dan terstruktur dengan baik [12]. Data yang tidak dilakukan preprocessing akan menghasilkan kinerja yang kurang akurat [13]. Pemilihan metode *preprocessing* yang akan digunakan dapat berpengaruh pada kualitas data keluaran proses. Pada penelitian ini, tahapan yang dilakukan dalam data preprocessing adalah melakukan *tokenizing*, *cleaning*, dan *adding* data. *Tokenizing* adalah penghitungan beberapa variabel pada data sehingga data menjadi lebih akurat [14]. *Tokenizing* dilakukan dengan cara menggunakan tools *tokenizer* yang cukup popular yaitu website *openai tokenizer platform.* Kemudian *cleaning* dan *adding* data adalah pembersihan data duplikat serta penambahan data yang kosong.

4. Transforming Data

Pada tahapan data transform, data yang didapatkan akan diubah terlebih dahulu untuk menyesuaikan dengan penggunaan algoritma atau metode yang digunakan, sehingga pola pada data dapat ditemukan dengan cepat[15]. Pada tahapan ini, teknik transform yang digunakan adalah *Min-Max Normalization*.

$$z = \frac{x - \min(x)}{[\max(x) - \min(x)]}$$

Z adalah hasil normalisasi, x adalah nilai x asli, min(x) adalah nilai minimal untuk variabel x, dan max(x) adalah nilai maksimal untuk variabel x. Kemudian data diubah kedalam bentuk format csv sehingga data bisa dibaca oleh program yang sudah dibuat sebelumnya terutama library pandas pada google collab.

5. Perhitungan *kNN*

Perhitungan *kNN* dilakukan dengan menggunakan Google Collab. Program di rancang untuk mempelajari pola data dari data-data yang sudah tersedia, sehingga hasil pada program tersebut menghasilkan pengeluaran sesuai dengan yang diharapkan. Library yang digunakan dalam program adalah *Pandas*, *NumPy*, dan *Scikit-learn*. Library *Pandas* adalah salah satu library populer di python yang digunakan untuk analisis dan manipulasi data, terutama data yang berbentuk tabel seperti spreadsheet atau database. Library *NumPy* digunakan untuk komputasi numerik di python. Sedangkan library *Scikit-learn* digunakan untuk komputasi algoritma *kNN* sekaligus evaluasi model.

6. Evaluasi Model

Evaluasi model yang digunakan adalah metode *Euclidean Distance*. Metode *Euclidean Distance* dipillih karena memiliki beberapa keunggulan yaitu perhitungan jarak menggunakan *Eucidean Distance* lebih umum digunakan dalam metode *kNN* dan *Euclidean Distance* mempunyai hasil yang lebih optimal dibanding metode yang lain.

RESULTS

Pengumpulan data dilakukan dengan mengambil data training dari website *opensource Kaggle* https://www.kaggle.com/datasets/free4ever1/instagram-fake-spammer-genuine-accounts

serta scrapping dari instagram menggunakan tools *IG Exporter* untuk digunakan sebagai data test. Kemudian diperoleh data sebagai berikut ini:

Tabel 1. Hasil Data Scrapping

number	profile	full_name	username
1	https://www.instagram.com/ceii_1088	Ceinaa	ceii_1088
2	https://www.instagram.com/morisa_uwu	Hailey	morisa_uwu
3	https://www.instagram.com/gw_ardhi2	ard	gw_ardhi2
4	https://www.instagram.com/dear.daphnyy	Ur daphy	dear.daphnyy
5	https://www.instagram.com/wanapwiu_	Rere	wanapwiu_
6	https://www.instagram.com/cia.cos_	Cia シア	cia.cos_
7	https://www.instagram.com/noyspace_	NOY꧁•ᴥ•?	noyspace_
8	https://www.instagram.com/el.skut_	biasa dipanggil om om	el.skut_
9	https://www.instagram.com/rafif_0_0	Rafif0_0	rafif_0_0
10	https://www.instagram.com/vaniaadnv_	⊹₹ⓐⓝi̇i `	vaniaadnv_
...	...	...	...
576	https://www.instagram.com/the_smollghost	SmollymollyRiley	the_smollghost

Peneliti masuk satu persatu dalam instagram masing-masing follower untuk menganalisis lebih lanjut apakah akun follower tersebut masih aktif atau tidak, mengetahui jumlah postingan, gambar profile, follower, following, private, dan link bio. Peneliti menggunakan 120 follower untuk data testing. Data yang sudah siap untuk dianalisis selanjutnya disimpan dalam format csv yaitu train.csv dan test.csv dan diupload ke http://www.dropbox.com. File train.csv dan test.csv yang sudah diupload tersebut sudah siap digunakan dan dinalisis menggunakan Metode *kNN*. Berikut data training yang diperoleh dari website *Kaggle*:

Tabel 2. Data Training

No	profile pic	nums/length username	fullname words	nums/length fullname	username	description length	external URL	private	#posts	#followers	#follows	fake
1	1	0,27	0	0	0	53	0	0	32	1000	955	0
2	1	0	2	0	0	44	0	0	286	2740	533	0
3	1	0,1	2	0	0	0	0	1	13	159	98	0
4	1	0	1	0	0	82	0	0	679	414	651	0
5	1	0	2	0	0	0	0	1	6	151	126	0
....	....	....	....	....	....	....	....	....	....	....	....	....
576	1	0,27	1	0	0	0	0	0	2	150	487	1

Selanjutnya dilakukan data processing pada hasil scrapping yang tertera pada tabel 1., maka diperoleh hasil sebagai berikut:

Tabel 3. Data Test

No	profile pic	nums/length username	fullname words	nums/length fullname	username	description length	external URL	private	#posts	#followers	#follows	fake
1	1	0,3	1	0,2	0	0,54	0	0	3	86	248	1
2	1	0,4	2	0,27	0	0,85	1	0	29	2572	597	0
3	1	0,3	1	0,27	0	0,41	1	0	12	179	172	0

No	profile pic	nums/length username	fullname words	nums/length fullname	nums/length username	description length	external URL	private	#posts	#followers	#follows	fake
4	1	0,27	4	0,7	0	0,16	0	1	6	250	2708	1
5	1	0,3	1	0,13	0	0,15	1	0	2	157	561	1
....	....	....	....	....	....	....	....	....	....	....	....	....
120	1	0,3	1	0,2	0	0,09	0	0	0	10	37	1

Kemudian dilakukan perhitungan manual menggunakan metode Euclidean Distance. Data nomor 1 sampai nomor 120 dari test.csv akan diprediksi apakah akun instagram tersebut palsu atau tidak dengan menggunakan rumus:

$$\text{Dist (X,Y)} = \sqrt{\sum_{i=1}^{D}(X_i - Y_i)^2}$$

Keterangan:
Dist (X,Y) = jarak antar objek (Euclidean Distancing)
Xi = sampel data
Yi = data uji

D = dimensi data
i = variabel data

a. Obyek nomor 1 dari test.csv

Nomor 1 dari test.csv dengan nomor 1 dari train.csv

Ed 1-1 = $\{(1 - 1)^2 + (0.27 - 0.30)^2 + (0 - 1)^2 + (0 - 0.20)^2 + (0 - 0)^2 + (53 - 81)^2 + (0 - 0)^2 + (0 - 0)^2 + (32 - 3)^2 + (1000 - 86)^2 + (955 - 248)^2\}^{1/2} = 1156.23$

Nomor 1 dari test.csv dengan nomor 2 dari train.csv

Ed 1-2 = $\{(1 - 1)^2 + (0.27 - 0.30)^2 + (0 - 1)^2 + (0 - 0.20)^2 + (0 - 0)^2 + (53 - 81)^2 + (0 - 0)^2 + (0 - 0)^2 + (286 - 3)^2 + (2740 - 86)^2 + (533 - 248)^2\}^{1/2} = 2684.47$

.... dengan cara yang sama sampai nomor 576 dari train.csv

Nomor 1 dari test.csv dengan nomor 576 dari train.csv
Ed 1-576 = $\{(1 - 1)^2 + (0.27 - 0.30)^2 + (1 - 1)^2 + (0 - 20)^2 + (0 - 0)^2 + (0 - 81)^2 + (0 - 0)^2 + (0 - 0)^2 + (2 - 3)^2 + (150 - 86)^2 + (487 - 248)^2\}^{1/2} = 260.34$

b. Obyek nomor 2 dari test.csv

Nomor 2 dari test.csv dengan nomor 1 dari train.csv

Ed 2-1 = $\{(1 - 1)^2 + (0.27 - 0.40)^2 + (0 - 2)^2 + (0 - 0.27)^2 + (0 - 0)^2 + (53 - 128)^2 + (0 - 1)^2 + (0 - 0)^2 + (32 - 29)^2 + (1000 - 2572)^2 + (955 - 597)^2\}^{1/2} = 1614.00$

Nomor 2 dari test.csv dengan nomor 2 dari train.csv

Ed 2-2 = $\{(1 - 1)^2 + (0.27 - 0.40)^2 + (0 - 2)^2 + (0 - 0.27)^2 + (0 - 0)^2 + (53 - 128)^2 + (0 - 1)^2 + (0 - 0)^2 + (286 - 29)^2 + (2740 - 2572)^2 + (533 - 597)^2\}^{1/2} = 324.69$

.... dengan cara yang sama sampai nomor 576 dari train.csv

Nomor 2 dari test.csv dengan nomor 576 dari train.csv
Ed 2-576 = $\{(1 - 1)^2 + (0.30 - 0.27)^2 + (1 - 6)^2 + (0.20 - 1.03)^2 + (0 - 0)^2 + (0.09 - 0.00)^2 + (0 - 1)^2 + (0 - 1)^2 + (0 - 35)^2 + (10 - 841)^2 + (37 - 570)^2\}^{1/2} = 2428.02$

DOI : https://doi.org/10.26623/transformatika.v23i2.12287

c. Obyek nomor 120 dari test.csv

Nomor 120 dari test.csv dengan nomor 1 dari train.csv

Ed 120-1 = $\{(1 - 1)^2 + (0.27 - 0.30)^2 + (0 - 1)^2 + (0 - 0.20)^2 + (0 - 0)^2 + (53 - 14)^2 + (0 - 0)^2 + (0 - 0)^2 + (32 - 0)^2 + (1000 - 10)^2 + (955 - 37)^2\}^{1/2} = 1351.06$

Nomor 120 dari test.csv dengan nomor 2 dari train.csv

Ed 120-2 = $\{(1 - 1)^2 + (0.27 - 0.30)^2 + (0 - 1)^2 + (0 - 0.20)^2 + (0 - 0)^2 + (53 - 14)^2 + (0 - 0)^2 + (0 - 0)^2 + (286 - 0)^2 + (2740 - 10)^2 + (533 - 37)^2\}^{1/2} = 2786.55$

.... dengan cara yang sama sampai nomor 576 dari train.csv

Nomor 120 dari test.csv dengan nomor 576 dari train.csv
Ed 120-576 = $\{(1 - 1)^2 + (0.27 - 0.30)^2 + (1 - 1)^2 + (0 - 20)^2 + (0 - 0)^2 + (0 - 14)^2 + (0 - 0)^2 + (0 - 0)^2 + (2 - 0)^2 + (150 - 10)^2 + (487 - 37)^2\}^{1/2} = 471.49$

Dari perhitungan Euclidean Distance secara manual, maka diperoleh hasil sebagai berikut:

Tabel 4. Hasil Euclidean Distance

No	Data 1	Sort	Data 2	Sort	….	Data 120	Sort
1	1156.23	450	1614.00	49	…	1351.06	450
2	2684.47	507	324.69	4	…	2789.55	507
3	185.72	118	2467.43	234	…	162.14	226
4	852.63	418	2254.88	148	…	1002.93	418
5	160.25	83	2469.82	235	…	167.44	228
6	669901.09	569	667415.23	569	…	669977.10	569
7	86.42	5	2487.00	244	…	183.57	234
8	1010.50	436	1587.41	45	…	1069.31	424
9	3016.91	514	2245.45	146	…	3234.19	515
10	12873.18	547	10377.25	547	…	12960.03	547
….	….	….	….	….	….	….	….
576	260.34	282	2428.02	210	…	471.49	327

Kemudian hasil data tersebut diurutkan secara *Ascending* (data terkecil ke data terbesar). Dan diperoleh hasil sebagai berikut:

Tabel 5. Hasil Sortir Euclidean Distance Ascending

Sort	Data 1	Fake	Data 2	Fake	Data …	Data 120	Fake
1	51.34	0	242.47	0	…	14.63	1
2	66.85	0	257.98	0	…	14.73	1
3	77.10	0	267.12	0	…	15.30	1
4	82.23	1	324.69	0	…	15.46	1
5	86.42	0	489.45	0	…	15.85	1
6	87.96	1	491.09	0	…	16.97	1
7	88.53	1	504.95	1	…	17.18	1
…	…		…		…	…	
576	15338452.00	0	15335966.01	0	…	15338528.00	0

Menggunakan nilai K = 3 dengan menggunakan tabel 5. (data yang diambil adalah 3 data teratas, maka diperoleh hasil sebagai berikut:

Tabel 6. Hasil Kategorisasi KNN, dengan K = 3

Data test	**Fake**		
	0	**1**	**Prediksi**
Data 1	3	0	0
Data 2	3	0	0
Data 3	3	0	0
Data 4	0	3	1
Data 5	0	3	1
Data 6	0	3	1
Data 7	1	2	1
Data 8	0	3	1
Data 9	3	0	0
Data 10	3	0	0
...	...	...	...
Data 120	0	3	1

Menggunakan nilai K = 4 dengan menggunakan tabel 5. (data yang diambil adalah 4 data teratas, maka diperoleh hasil sebagai berikut:

Tabel 7. Hasil Kategorisasi kNN, dengan K = 4

	Fake		
Data test	**0**	**1**	**Prediksi**
Data 1	3	1	0
Data 2	4	0	0
Data 3	5	0	0
Data 4	0	4	1
Data 5	1	3	1
Data 6	0	4	1
Data 7	1	3	1
Data 8	1	3	1
Data 9	4	0	0
Data 10	4	0	0
...	...	...	...
Data 120	0	4	1

Menggunakan nilai K = 5 dengan menggunakan tabel 5. (data yang diambil adalah 5 data teratas, maka diperoleh hasil sebagai berikut:

Tabel 8. Hasil Kategorisasi kNN, dengan K = 5

Data test	**Fake**		
	0	**2**	**Prediksi**
Data 1	2	3	1
Data 2	1	4	1
Data 3	1	4	1
Data 4	4	1	0
Data 5	1	4	1
Data 6	0	5	0
Data 7	2	3	1
Data 8	2	3	1
Data 9	4	1	0
Data 10	5	0	1
...	...	...	...
Data 120	0	5	0

DOI : https://doi.org/10.26623/transformatika.v23i2.12287

Untuk perhitungan menggunakan program pada google collab, dilakukan pada link berikut: https://colab.research.google.com/drive/1iPMQCSEdBeMiqsDQmV3vCZvypgJDtgGp?usp=sharing.

```
model = KNeighborsClassifier(n_neighbors=K , weights="distance")
model.fit(X_train,y_train)
predicting = model.predict(X_test)
print("The average accuracy of the model was:" , accuracy_score(y_test , predicting))
print(classification_report(y_test , predicting))
```

Dan menghasilkan nilai akurasi yang berbeda-beda untuk tiap nilai K:
K = 2. The average accuracy of the model was: 0.7916666666666666
K = 3. The average accuracy of the model was: 0.85
K = 4. The average accuracy of the model was: 0.8666666666666667
K = 5. The average accuracy of the model was: 0.85
Dari hasil pengujian diatas dapat disimpulkan bahwa akurasi tertinggi pada K = 4 yaitu sebesar 0.8666.
Hasil akurasi selanjutnya akan di evaluasi dengan menggunakan metode Confusion Matrix. Akurasi merupakan rasio prediksi benar dari positif dan negatif dengan keseluruhan data. Pengukuran akurasi dapat menggunakan rumus:

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

TP = *True positives*
TN = *True negatives*
FP = *False positives*
FN = *False negatives*

Perhitungan K=3:

True Positives (TP): 52 (Prediksi kelas 1 dan benar kelas 1)
False Positives (FP): 8 (Prediksi kelas 1, tetapi sebenarnya kelas 0)
False Negatives (FN): 10 (Prediksi kelas 0, tetapi sebenarnya kelas 1)
True Negatives (TN): 50 (Prediksi kelas 0 dan benar kelas 0)
Akurasi dihitung dengan rumus:

$$\frac{TN + TP}{TP + TN + FP + FN} = \frac{102}{120} = 0{,}85$$

Perhitungan K=4:
True Positives (TP): 54 (Prediksi kelas 1 dan benar kelas 1)
False Positives (FP): 6 (Prediksi kelas 1, tetapi sebenarnya kelas 0)
False Negatives (FN): 10 (Prediksi kelas 0, tetapi sebenarnya kelas 1)
True Negatives (TN): 50 (Prediksi kelas 0 dan benar kelas 0)
Akurasi dihitung dengan rumus:

$$\frac{TN + TP}{TP + TN + FP + FN} = \frac{104}{120} = 0{,}8666$$

Perhitungan K=5:

True Positives (TP): 51 (Prediksi kelas 1 dan benar kelas 1)
False Positives (FP): 9 (Prediksi kelas 1, tetapi sebenarnya kelas 0)

False Negatives (FN): 9 (Prediksi kelas 0, tetapi sebenarnya kelas 1)
True Negatives (TN): 51 (Prediksi kelas 0 dan benar kelas 0)
Akurasi dihitung dengan rumus:

$$\frac{TN + TP}{TP + TN + FP + FN} = \frac{102}{120} = 0{,}85$$

Perhitungan menggunakan program:

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, predicting)
print('Confusion matrix\n\n', cm)
print('\nTrue Positives(TP) = ', cm[0,0])
print('\nTrue Negatives(TN) = ', cm[1,1])

print('\nFalse Positives(FP) = ', cm[0,1])
print('\nFalse Negatives(FN) = ', cm[1,0])
```

Hasil evaluasi model menggunakan metode Confusion Matrix menyatakan bahwa K=4 memiliki akurasi terbaik dengan nilai sebesar 0,8666

CONCLUSIONS AND RECOMMENDATIONS

Metode klasifikasi dengan algoritma K-Nearest Neighbour (kNN) dalam mengidentifikasi Instagram dengan akun palsu di media sosial Instagram sangat bergantung pada kemiripan suatu akun dengan akun-akun lain yang sudah diklasifikasikan. Jika akun palsu memiliki pola yang berbeda dari akun asli, kNN dapat mengidentifikasi akun tersebut dengan akurasi yang cukup baik. K-Nearest Neighbour (kNN) cukup efektif untuk klasifikasi Instagram dengan akun palsu di media sosial Instagram dengan tingkat akurasi 86,666%. Kualitas Dataset yang sudah dibuat terhadap proses klasifikasi K-Nearest Neighbour (kNN) dalam memprediksi akun palsu mempunyai pengaruh yang cukup besar bergantung pada pemilihan fitur yang tepat. Berdasarkan penelitian yang telah dilakukan maka dapat dikemukakan saran untuk penelitian selanjutnya tidak hanya sebatas klasifikasi, tetapi dapat merekomendasi akun instagram mana yang dicurigai sebagai akun palsu dan digunakan untuk tindakan yang melanggar hukum atau berkemungkinan melakukan penipuan. Penelitian selanjutnya diharapkan dapat menggunakan metode lain seperti korelasi, N-gram, atau SVM sebagai pembandingnya, karena pada penelitian ini akurasi tertinggi yang diperoleh hanya sebesar 86,666 %.

REFERENCES

[1] I. Alinan, *Perkembangan Teknologi media Sosial Mengubah mental Dan Karakter remaja Dan Pemuda DITENGAH pandemi covid-19*, Dec. 2021. doi:10.31219/osf.io/v38re
[2] D. Prasetya and R. Marina, "Studi Analisis Media Baru: Manfaat Dan Permasalahan Dari media Sosial Dan Game Online," *Telangke:Jurnal Telangke Ilmu Komunikasi*, vol. 4, no. 2, pp. 01–10, Oct. 2022. doi:10.55542/jiksohum.v4i2.357
[3] A. Widya Astuti, S. Sayudin, and A. Muharam, "Perkembangan Bisnis di era Digital," *Jurnal Multidisiplin Indonesia*, vol. 2, no. 9, pp. 2787–2792, Sep. 2023. doi:10.58344/jmi.v2i9.554
[4] S. Herdiyani, C. Safa'atul Barkah, L. Auliana, and I. Sukoco, "Peranan Media Sosial Dalam Mengembangkan Suatu bisnis: Literature review," *Jurnal Administrasi Bisnis*, vol. 18, no. 2, pp. 103–121, Dec. 2022. doi:10.26593/jab.v18i2.5878.103-121
[5] P. P. Sudin, R. Magdalena, E. S. Priowirjanto, and D. Soeikromo, "Penyalahgunaan Akun Instagram perihal penipuan jual beli secara online Ditinjau Dari UU ite Dan Pasal 378 KUHP

Tentang penipuan," *Journal of Education, Humaniora and Social Sciences (JEHSS)*, vol. 5, no. 1, pp. 20–26, Jul. 2022. doi:10.34007/jehss.v5i1.842

[6] 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, pp. 191–200, Dec. 2021. doi:10.54706/senastindo.v3.2021.139

[7] M. Yunus and N. K. Pratiwi, "Prediksi status gizi Balita Dengan Algoritma K-nearest neighbor (KNN) Di Puskemas cakranegara," *JTIM : Jurnal Teknologi Informasi dan Multimedia*, vol. 4, no. 4, pp. 221–231, Feb. 2023. doi:10.35746/jtim.v4i4.328

[8] A. Syafi'i, M. Afdal, E. Saputra, and R. Novita, "Analisis Sentimen Ulasan Pengguna aplikasi Penjualan Pulsa Menggunakan algoritma naïve Bayes classifier," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 7, no. 3, pp. 1300–1308, Jul. 2024. doi:10.32493/jtsi.v7i3.41364

[9] N. Wakhidah and S. N. Rochmah, "Klasifikasi kualitas Mutu Susu Pasteurisasi menggunakan metode klasifikasi K-nearest neighbor," *AITI*, vol. 21, no. 1, pp. 58–71, Apr. 2024. doi:10.24246/aiti.v21i1.58-71

[10] M. Suyal and P. Goyal, "A review on analysis of K-nearest neighbor classification machine learning algorithms based on supervised learning," *International Journal of Engineering Trends and Technology*, vol. 70, no. 7, pp. 43–48, Jul. 2022. doi:10.14445/22315381/ijett-v70i7p205

[11] T. M. Ghazal *et al.*, "Performances of K-means clustering algorithm with different distance metrics," *Intelligent Automation & Soft Computing*, vol. 29, no. 3, pp. 735–742, 2021. doi:10.32604/iasc.2021.019067

[12] "Optimizing data preprocessing: The data preprocessing interface," *International Research Journal of Modernization in Engineering Technology and Science*, Dec. 2023. doi:10.56726/irjmets46515

[13] Mr. S. Kumar and Mr. N. D.C, "A survey on improving classification performance using data pre processing and machine learning methods on NSL-KDD data," *International Journal Of Engineering And Computer Science*, Apr. 2016. doi:10.18535/ijecs/v5i4.17

[14] P. Kharismadita and F. Rahutomo, "Implementasi tokenizing plus Pada Sistem Pendeteksi kemiripan Jurnal Skripsi," *Jurnal Informatika Polinema*, vol. 2, no. 1, p. 24, Nov. 2015. doi:10.33795/jip.v2i1.50

[15] A. Jadhav, D. Dhaulakhandi, S. K. Shandilya, L. Malviya, and A. Mewada, "Data transformation: A preprocessing stage in machine learning regression problems," *Artificial Intelligence Techniques in Power Systems Operations and Analysis*, pp. 183–194, Jun. 2023. doi:10.1201/9781003301820-10