P-ISSN: 1411-3635
E-ISSN: 2962-5645
https://doi.org/10.37817/Tekinfo.v26i1

Penerapan Algoritma *K-Nearest Neighbor* (KNN) untuk Memprediksi *Stroke* pada Rumah Sakit Pusat Otak Nasional Prof. Dr. dr. Mahar Mardjono Jakarta

[1]Annisa Cintakami Firdaus, [2]Ionia Veritawati
[1,2]Program Studi Teknik Informatika, Universitas Pancasila, Jakarta

E-mail: [1]cintakamiannisa@gmail.com, [2]ionia.veritawati@univpancasila.ac.id

ABSTRAK

Kasus *Stroke* sekarang banyak terjadi dengan didominasi oleh orang berusia 40 tahun ke atas dan banyak kasus *stroke* yang dirujuk ke RS PON sudah dalam keadaan terlambat yang menyebabkan peluang kesembuhan semakin rendah. Algoritma KNN merupakan salah satu algoritma *machine learning* yang dapat digunakan untuk mengklasifikasikan jenis *Stroke*. KNN digunakan untuk mengklasifikasi dalam menentukan dua jenis kelas *Stroke* yaitu *Cerebral Infarction* (*Stroke* Iskemik) dan *Intracerebral Haemorrhage* (*Stroke* Hemoragik). Secara garis besar tahapan penelitian ini dimulai dengan mengumpulkan data secara langsung pasien *Stroke*, melakukan pengolahan data dengan memanfaatkan bahasa pemrograman *python*, *preprocessing data*, normalisasi data, membagi data uji dan data latih yang dapat diproses ke dalam model dari data yang jenis *Stroke* sudah diketahui sehingga jenis *Stroke* dapat diprediksi. Dari proses evaluasi diperoleh hasil klasifikasi optimal pada komposisi data latih 80% dan data uji 20% mendapatkan hasil klasifikasi yang optimal pada K bernilai 3 dengan *accuracy* 70%.

Kata kunci : *K-Nearest Neighbor*, prediksi, *stroke*,

ABSTRACT

Stroke cases are now increasingly common, predominantly affecting individuals aged 40 and above. Many stroke cases referred to RS PON arrive in a delayed state, reducing the chances of recovery. The K-NN algorithm is a machine learning algorithm that can be used to classify types of strokes. K-NN is utilized to classify two stroke categories are Cerebral Infarction (Ischemic Stroke) and Intracerebral Hemorrhage (Hemorrhagic Stroke). In general, this research follows several stages, beginning with directly collecting stroke patient data, processing the data using the Python programming language, performing data preprocessing and normalization, and splitting the dataset into training and testing sets. The model is trained using data with known stroke types, enabling stroke type prediction. The evaluation process yielded optimal classification results with a training data composition of 80% and a test data composition of 20%. The highest classification accuracy was obtained when K was set to 3 achieving 70%.

Keyword : *K-Nearest Neighbor, prediction, stroke*

1. PENDAHULUAN

Stroke merupakan suatu penyakit defisit neurologis yang disebabkan oleh perdarahan atau sumbatan pada bagian otak yang dapat menimbulkan cacat atau kematian. Otak merupakan organ yang sangat kompleks yang terdiri dari kumpulan sel saraf (*nerve cell*) yang berperan pada semua sinyal dan sensasi

P-ISSN: 1411-3635
E-ISSN: 2962-5645

dalam membuat tubuh manusia dapat berpikir, bergerak, dan bereaksi dari suatu kejadian atau keadaan. Suplai oksigen dan nutrisi diperlukan oleh otak secara kontinu karena organ otak yang tidak mampu untuk menyimpan energi. Pendistribusian nutrisi dan oksigen dilakukan melalui darah dari jantung melalui arteri menuju otak (Putri Ayundari Setiawan, 2021).

Penyakit *Stroke* menduduki peringkat nomor tiga penyebab kematian di Indonesia setelah penyakit jantung dan kanker. *Stroke* dapat terjadi mendadak tanpa timbul gejala yang pasti. *Stroke* merupakan penyakit pada serebrovaskuler yang terjadi akibat aliran darah pembawa oksigen ke otak berkurang dengan ditandai dengan kematian jaringan otak (infarkserebral). Aliran darah pembawa oksigen berkurang disebabkan oleh terjadinya adanya penyempitan, penyumbatan atau pecahnya pembuluh darah (Wahab dkk., 2019).

Faktor penyebab terjadinya stroke yang tidak dapat diubah antara lain umur, keturunan, serta gender. Ada faktor lain dalam pemicu terjadinya *stroke* yang bisa diubah yaitu dengan adanya penyakit lain yang diderita seperti penyakit jantung, hipertensi, ginjal, diabetes, dan obesitas (Laela Tusifaiyah dkk., 2022).

Penanganan *stroke* dengan cepat akan mengurangi tingkat kerusakan yang terjadi pada otak dan kemungkinan terjadinya komplikasi. Salah satu cara untuk memprediksi jenis penyakit *stroke* yaitu dengan menggunakan klasifikasi. Jenis penyakit *stroke* diperlukan klasifikasi agar dapat memprediksi penyakit dengan cepat dan akurat. Hasil prediksi secara akurat dapat membantu tenaga medis dalam mengambil keputusan tindakan yang tepat(Ulfatul dkk., 2022).

Penyakit *stroke* dapat dilakukan pemeriksaan dengan alat radiologi yang berperan penting dalam mendiagnosis *stroke*. Meskipun *Magnetic Resonance Imaging* (MRI) dengan *Diffusion-Weighted Imaging* (DWI) memiliki sensitivitas yang baik dalam mendeteksi *Stroke* Iskemik atau *Cerebral Infarction*, prosedur ini dianggap kurang efisien dari segi biaya. Di sisi lain, *CT scan* tanpa kontras cukup efektif untuk mendeteksi *Cerebral Infarction* dan mengeksklusi *Stroke* Hemoragik atau *Intracerebral Haemorrhage*, dengan ketersediaan dan biaya yang lebih terjangkau. Pemeriksaan MRI dengan DWI memiliki sensitivitas yang lebih tinggi dan lebih mampu mendeteksi *Cerebral Infarction*, namun biaya yang dibutuhkan lebih mahal (Lie Michael, 2020).

K-Nearst Neighbor merupakan metode klasifikasi yang efisien dalam mengenali pola, mengkategorikan teks, memproses objek dan dapat melakukan *training data* dalam jumlah besar (Ulfatul dkk., 2022). Pada kasus klasifikasi penyakit jantung dengan menerapkan algoritma KNN bekerja dengan sangat baik dengan menggunakan K = 3, nilai akurasi yang didapatkan sebesar 92% (Akbarollah dkk., 2023). Pada penelitian sebelumnya *K-Nearst Neighbor* juga pernah digunakan untuk kasus penyakit diabetes (Argina, 2020) dan klasifikasi penyakit gagal jantung (Pratama dkk., 2022).

Rumah Sakit Pusat Otak Nasional dr. Mahar Mardjono atau RS PON merupakan rumah sakit rujukan nasional untuk penyakit saraf dalam penanganan khusus kesehatan otak dan saraf. RS PON memiliki layanan rujukan *Comprehensive Stroke Care* untuk pengobatan *stroke* secara efektif dan tepat dalam menangani gejala *stroke*.

Berdasarkan pendahuluan di atas, maka diperlukan adanya suatu sistem diagnosis prediksi jenis penyakit *Stroke* menggunakan metode Algoritma *K-Nearest Neighbor*. Di Rumah Sakit Pusat Otak Nasional Prof. Dr. dr. Mahar Mardjono Jakarta (RS PON) masih dilakukan pengecekan dan pencatatat diagnosis secara manual. Dengan adanya faktor seperti riwayat hipertensi, riwayat jantung, riwayat diabetes, dan lainnya

P-ISSN: 1411-3635
E-ISSN: 2962-5645
https://doi.org/10.37817/Tekinfo.v26i1

akan digunakan dalam perhitungan untuk menentukan prediksi dari penyakit *stroke* yang dialami oleh pasien. Dalam penelitian sebelumnya, metode *K-Nearest Neighbor* memiliki akurasi yang cukup baik pada penyakit diabetes (Argina, 2020) dan penyakit gagal jantung (Pratama dkk., 2022). Diharapkan dengan penelitian pada studi kasus ini dapat membantu dalam menentukan penyakit *stroke* pasien yang ada pada Rumah Sakit Pusat Otak Nasional Prof. Dr. dr. Mahar Mardjono Jakarta (RS PON).

2. LANDASAN TEORI

Stroke

Stroke merupakan suatu keadaan dapat dialami seseorang yang disebabkan dengan adanya kerusakan pada otak yang terjadi secara tidak terduga, progresif dan cepat yang disebabkan oleah adanya gangguan peredaran darah otak non traumatik. Penyakit *stroke* tidak hanya terjadi pada seseorang berusia lanjut tetapi dapat juga menyerang pada usia produktif antara usia 20 sampai dengan 44 tahun. S*troke* yang terjadi pada usia muda dapat menyebabkan beberapa masalah seperti terjadi kecacatan fisik, depresi, penurununan fungsi kognitif, dan menganggu produktifitas. Hipertensi dan diabetes melitus dapat menjadi faktor resiko penyebab yang mempengaruhi terjadinya *stroke*. Hipertensi dapat menyebabkan kerusakan arteri seluruh tubuh yang mengakibatkan pembuluh darah pecah dan sumbatan ateri di otak. Diabetes mellitus dapat menyebabkan peningkatan lemak atau pembekuan dinding darah, gumpalan atau lemak yang dapat menyumbat pembuluh darah sehingga menyebabkan *stroke*. Faktor resiko penting lain yaitu merokok, dan penaykit jantung (Utama & Nainggolan, 2022).

Berdasarkan penyebab, terdapat dua jenis *stroke* yaitu:

1. *Cerebral Infarction* atau *Stroke* Iskemik merupakan *stroke* terjadi akibat tersumbatnya atau ada hambatan dalam aliran darah ke otak.
2. *Intracerebral Haemorrhage* atau *Stroke* hemoragik merupakan *stroke* akibat pecahnya pembuluh darah pada otak yang menyebabkan pendarahan pada bagian intraserebral dan ruang subaraknoid.

Stroke memiliki faktor risiko yang dapat diubah dan tidak dapat diubah. Faktor yang tidak dapat diubah antara lain usia, gender, berat badan, genetik, dan ras. Faktor risiko yang dapat dimodifikasi diantaranya tekanan darah tinggi (hipertensi), jantung, dan diabetes (Husada dkk., 2020).

Terjadinya penyempitan pembuluh darah pada jantung dapat menyebabkan penyakit jantung. Jika memiliki riwayat penyakit jantung mempunyai risiko 5,440 kali lebih besar mengalami *stroke* iskemik (Hisni dkk., 2022)

Diabetes melitus termasuk faktor risiko *stroke* yang disebabkan oleh terjadinya peningkatan kadar lemak darah yang mengganggu transformasi lemak tubuh. Peningkatan kadar lemak darah dalam tubuh akan meningkatkan risiko akan terjadinya *stroke*. Diabetes mempercepat terjadinya penyempitan dan pengerasan akibat plak (aterosklerosis) pada pembuluh darah besar dan kecil yang ada pada jantung dan otak. Kadar glukosa darah yang tinggi akan memperluas area infark (sel mati) yang terjadi akibat metabolisme glukosa dengan kadar oksigen yang sedikit *(anaerob)* yang akan merusak jaringan otak. Penyebab diabetes melitus menjadi *stroke* karena ada proses aterosklerosis yang merusak dinding pembuluh darah besar dan pembuluh darah perifer yang meningkatkan gumpalan trombosit (agregat platelet) yang menyebabkan terjadinya aterosklerosis. Terjadinya hiperglikemia dapat meningkatkan kekentalan darah (viskositas) yang menyebabkan naiknya tekanan darah yang mengakibatkan terjadinya *stroke* iskemik (Karmila Sari dkk., 2021).

https://doi.org/10.37817/Tekinfo.v26i1
P-ISSN: 1411-3635
E-ISSN: 2962-5645

Algoritma KNN

Algortima *K-Nearest Neighbor* (KNN) merupakan metode algoritma yang dapat digunakan klasifikasi objek baru berdasarkan nilai K atau tetangga terdekat. KNN merupakan algoritma *supervised learning* yang menghasilkan klasifikasi dari *query instance* (data baru yang ingin diprediksi) berdasarkan kelas terbanyak dari kategori pada KNN setelah dilakukan pengurutan dari jarak *euclidian* terkecil. Kelas terbanyak atau mayoritas y akan menjadi kelas hasil klasifikasi. Algoritma KNN sebagai berikut:

1. Menentukan Nilai K (jumlah tetangga terdekat)
2. Menghitung Nilai dari jarak *Euclidian* tiap objek terhadap data sampel yang diberikan seperti pada Persamaan (1):

$$d(x,y) = \sqrt{\sum_{i=0}^{n}(x_i - y_i)^2} \qquad (1)$$

Keterangan :

d(x,y) : Jarak *Euclidean* antara dua vektor x dan y

n : Jumlah atribut atau dimensi pada setiap vektor

x_i dan y_i : nilai atribut ke-i pada vektor x dan y

3. Mengurutkan jarak *Euclidean* setiap dari yang terkecil atau terdekat.
4. Menentukan jarak terdekat sebanyak K yang telah ditentukan sebelumnya.
5. Menentukan kelas data yang ingin diprediksi dengan cara menggunakan kelas mayoritas yang telah diurutkan, sehingga mendapatkan kelas prediksi (M. Syukri Mustafa & I Wayan Simpen, 2019).

Normalisasi

Normalisasi *z-score* dilakukan untuk menghindari bias dan menjaga nilai atribut lebih stabil terhadap nilai baru yang lebih kecil dari minimal dan nilai baru yang lebih besar dari makasimal. Normalisasi *z-score* dapat dihitung dengan persamaan (2):

$$z_{il} = \frac{(x_{il} - \mu_{il})}{\sigma_{il}} \qquad (2)$$

Keterangan:

z_{il} : Hasil normalisasi *z-score* (objek ke-i dengan variabel atribut ke- l)

x_{il} : Nilai data asli (objek ke-i dengan variabel atribut ke- l)

μ_{il} : Nilai rata-rata atau mean atribut ke-l

σ_{il} : Standar deviasi atribut ke- l

(Safitri dkk., 2024).

Confusion Matrix

Confusion Matrix dilakukan untuk mengevaluasi kinerja setelah diterapkan model klasifikasi yang bertujuan memberikan informasi nilai perbandingan hasil klasifikasi yang didapatkan dari perhitungan algoritma. *Confusion Matrix* digambarkan pada tabel 1.

Tabel 1 *Confusion Matrix*

	Actual	
Predicted	*TRUE*	*FALSE*
TRUE	TP	FP
FALSE	TN	FN

Keterangan :

- TP (*True Positive*) : data positif, diprediksi benar.
- TN (*True Negative*) : data negatif, diprediksi benar.
- FP (*False Positive*) : data negatif, diprediksi data positif.
- FN (*False Negative*) : data positif, diprediksi data negatif

Dari *Confusion Matrix* dapat dilakukan perhitungan nilai akurasi, presisi, *recall* dan *F1-Score* dengan cara:

1. Akurasi, mengukur seberapa akurat model melakukan klasifikasi.

Akurasi = (TP+TN) / (TP+FP+FN+TN)

2. Presisi, Perhitungan perbandingan antara jumlah data yang diprediksi dengan jumlah semua data yang diprediksi positif.

Presisi = (TP) / (TP + FP)

3. *Recall*, Perhitungan perbandingan antara jumlah data yang diprediksi dengan jumlah semua data yang memiliki kondisi aktual positif.

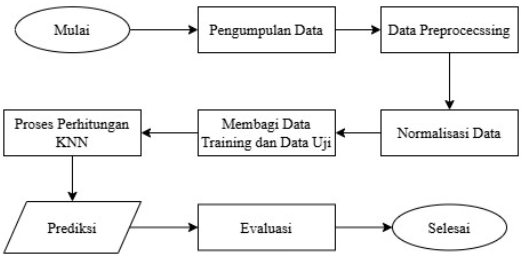
Recall = TP / (TP + FN)

P-ISSN: 1411-3635
E-ISSN: 2962-5645
https://doi.org/10.37817/Tekinfo.v26i1

4. *F1-Score*, merupakan perbandingan rata - rata presisi dengan *recall* yang dibobotkan
F1-score = 2*(*Recall**Presisi)/(*Recall* +Presisi) (Pristian Luthfy Romadloni dkk., 2022).

3. METODOLOGI

Tahapan yang dilakukan dalam penelitian ini mencakup tahapan yang teroganisir. Gambar 1 merupakan *flowchart* dari penelitian ini.



Gambar 1 *Flowchart* Penelitian

1. Pengumpulan Data
Dilakukan pengumpulan data secara langsung dan pemahaman terhadap data yang dibutuhkan melalui studi literatur.
2. *Data Preprocessing*
Preprocessing data dengan melakukan *encoding* terhadap data yang memiliki tipe data objek.
3. Normalisasi Data
Setelah itu, dilakukan normalisasi data dengan *z-score* bertujuan untuk memastikan data tidak terdistorsi oleh skala yang berbeda.
4. Membagi *data training* dan *data testing*
Setelah data siap untuk digunakan, lalu dilakukan pembagian data menjadi *data training* dan *data testing* dengan *single train test split*.
5. Proses Perhitungan KNN
Setelah itu, dilakukan proses pemodelan dengan perhitungan algoritma KNN.
6. Evaluasi
Tahap terakhir yaitu evaluasi dengan melakukan evaluasi terhadap model dengan melakukan perhitungan nilai akurasi, nilai *recall*, nilai presisi, dan *F1-score*.

4. HASIL DAN PEMBAHASAN

Skenario Penerapan Metode Yang Digunakan

Pada skenario penerapan penelitian ini digunakan sampel data sebanyak 8 data, seperti pada Tabel 2. Langkah pertama yang dilakukan adalah dengan melakukan *encoding* sesuai dengan nilai bobot pada Tabel 3.

Tabel 2 Sampel Data

No.	ID	Hipertensi	Jantung	Diabetes	Riwayat *Stroke*	Gender	Umur	Diagnosa
1	0013-88-37	ya	tidak	ya	ya	L	73	*Cerebral infarction*
2	0014-76-31	ya	tidak	tidak	tidak	L	45	*Cerebral infarction*
3	0014-75-38	ya	tidak	ya	tidak	P	61	*Cerebral infarction*
4	0015-52-33	tidak	tidak	ya	tidak	P	57	*Cerebral infarction*
5	0013-70-95	tidak	tidak	tidak	tidak	L	62	*Intracerebral haemorrhage*
6	0014-53-01	ya	tidak	tidak	tidak	L	49	*Intracerebral haemorrhage*
7	0015-44-35	ya	tidak	ya	ya	P	63	*Intracerebral haemorrhage*
8	0015-82-14	ya	tidak	tidak	tidak	L	70	*Intracerebral haemorrhage*

Tabel 3 Nilai Bobot

Nama Atribut	Sub-Atribut	Bobot
Riwayat Hipertensi	ya	1
	tidak	0
Riwayat Jantung	ya	1
	tidak	0
Riwayat Diabetes	ya	1
	tidak	0
Riwayat *Stroke*	ya	1
	tidak	0
Gender	L	1
	P	0
Diagnosa	*Cerebral infarction*	0
	Intracerebral haemorrhage	1

Untuk melakukan perhitungan skenario, digunakan data lain untuk melakukan prediksi diagnosa yang mempunyai riwayat medis pada Tabel 4 yang berisi data pasien yang akan diprediksi jenis *stroke* yang dialami.

Tabel 4 Data Pasien akan Diprediksi

No.	ID	Hipertensi	Jantung	Diabetes	Riwayat *Stroke*	Gender	Umur	Diagnosa
0	0000-00-00	tidak	ya	ya	tidak	P	48	?

Hasil *encoding* dari data pada Tabel 2 sesuai dengan nilai bobot pada Tabel 3 dapat dilihat pada Tabel 5. Pada Tabel 5 pada baris 9 merupakan data hasil *encodiing* yang ingin diprediksi dari Tabel 4.

https://doi.org/10.37817/Tekinfo.v26i1
P-ISSN: 1411-3635
E-ISSN: 2962-5645

Tabel 5 Hasil *Encoding*

No.	ID	Hipertensi	Jantung	Diabetes	Riwayat *Stroke*	Gender	Umur	Diagnosa
1	0013-88-37	1	0	1	1	1	73	0
2	0014-76-31	1	0	0	0	1	45	0
3	0014-75-38	1	0	1	0	0	61	0
4	0015-52-33	0	0	1	0	0	57	0
5	0013-70-95	0	0	0	0	1	62	1
6	0014-53-01	1	0	0	0	1	49	1
7	0015-44-35	1	0	1	1	0	63	1
8	0015-82-14	1	0	0	0	1	70	1
9	0000-00-00	0	1	1	0	0	48	?

Lalu dilakukan perhitungan kuadrat jarak *Euclidian* (*Euclidean distance*) masing - masing objek terhadap data sampel sesuai persamaan (1). Hasil dari perhitungan jarak *Euclidian* (*Euclidean distance*) terdapat pada Tabel 6. Pada Tabel 6 kolom diagnosa 0 merupakan *Cerebral infarction* dan 1 merupakan *Intracerebral haemorrhage.*

Tabel 6 Hasil Perhitungan *Euclidian*

No.	**ID**	**Diagnosa**	**Jarak *Euclidian***
1	0013-88-37	0	25,079
2	**0014-76-31**	**0**	**3,6055**
3	0014-75-38	0	13,0767
4	**0015-52-33**	**0**	**9,0553**
5	0013-70-95	1	14,1067
6	**0014-53-01**	**1**	**2,236**
7	0015-44-35	1	15,0996
8	0015-82-14	1	22,0907
9	0000-00-00	?	

Setelah itu ditentukan nilai K berdasarkan jarak *Euclidian* terkecil. Mencoba dengan nilai K=3, lalu diambil 3 teratas nilai jarak *Euclidian* terkecil, yaitu nomor 6, nomor 2, nomor 4 dari Tabel 6 yang diurutkan seperti pada Tabel 7.

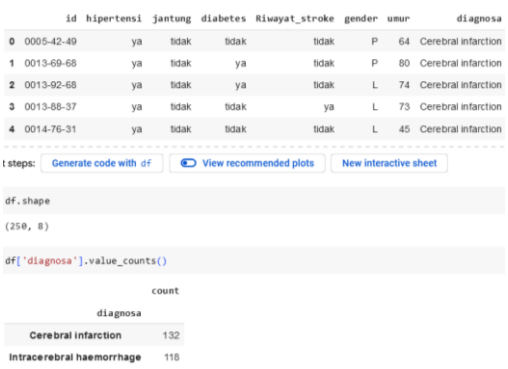
Tabel 7 Jarak *Euclidian* Terkecil

ID	**Diagnosa**	**Jarak *euclidian***	**Urutan Jarak *Euclidian* terkecil**
0014-53-01	1	2,236	1
0014-76-31	0	3,6055	2
0015-52-33	0	9,0553	3

Dari 3 titik data terdekat pada Tabel 7, dapat dilihat bahwa kelas mayoritas adalah 0 yaitu *Cerebral infarction* untuk dua data. Dengan demikian, pasien yang ingin melakukan diagnosa pada Tabel 6 nomor 9 dengan id 0000-00-00 akan terprediksi terdiagnosa *Cerebral infarction*.

Analisis Distribusi Nilai

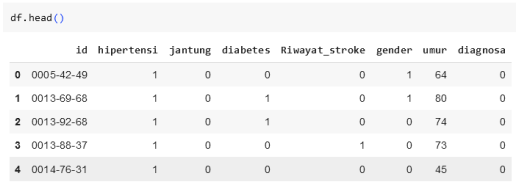
Berikut adalah hasil *processing* dataset lengkap.



Gambar 2 Data Mentah

Dataset adalah data pasien *stroke* yang terlihat pada Gambar 2 berjumlah 250 baris dan 8 kolom. Atribut yang dimiliki pada dataset adalah id pasien, riwayat hipertensi, riwayat Jantung, riwayat diabetes, riwayat *stroke*, gender, umur, dan diagnosa. Diagnosa memilki dua label nama yaitu *cerebral infarction* dan *intracerebral haemorrhage* yang berjumlah masing – masing 132, dan 118 data pasien.

Tahap *preprocessing* data



Gambar 3 *Hasil Encoding*

Setelah melihat jumlah baris, kolom, dan tipe data selanjutnya dilakukan *encoding* terhadap data yang memiliki tipe data *object*. Tipe data *object* diubah menjadi integer. Hasil enconding yang telah dilakukan, menghasilkan perubahan seperti pada gambar 3.

P-ISSN: 1411-3635
E-ISSN: 2962-5645
https://doi.org/10.37817/Tekinfo.v26i1

```
df.isna().sum() #cek missing value
```

	0
id	0
hipertensi	0
jantung	0
diabetes	0
Riwayat_stroke	0
gender	0
umur	0
diagnosa	0

Gambar 4 *Missing Value*

Selanjutnya dilakukan pemeriksaan terhadap *missing value* pada data seperti pada gambar 4. Hasilnya, tidak ditemukannya *missing value* pada data.

```
df.duplicated().sum()
print(f"Jumlah data duplikat: {df.duplicated().sum()}")
```

```
Jumlah data duplikat: 0
```

Gambar 5 Duplikat Data

Selanjutnya dilakukan pemeriksaan terhadap ada atau tidak duplikasi data pada data seperti pada gambar 5. Hasilnya, tidak ditemukannya duplikasi pada data.

	id	hipertensi	jantung	diabetes	Riwayat_stroke	gender	umur	diagnosa
0	0005-42-49	0.642207	-0.340693	-0.568112	-0.410152	1.266557	0.302275	0
1	0013-69-68	0.642207	-0.340693	1.760216	-0.410152	1.266557	1.581744	0
2	0013-92-68	0.642207	-0.340693	1.760216	-0.410152	-0.789542	1.101943	0
3	0013-88-37	0.642207	-0.340693	-0.568112	2.438123	-0.789542	1.021976	0
4	0014-76-31	0.642207	-0.340693	-0.568112	-0.410152	-0.789542	-1.217095	0

Gambar 6 Normalisasi Data

Tahap selanjutnya adalah melakukan normaliasi data pada kolom riwayat hipertensi, riwayat jantung, riwayat diabetes, riwayat *Stroke*, gender, dan umur. Hasil dari normaslisai terlihat pada gambar 6. Normalisasi data *z-score* dilakukan untuk memastikan data terorganisir dengan baik dan data terdistribusi dengan stabil.

```
# memetakan x dan y
x = df.iloc[:,1:-1]
y = df.iloc[:,-1]

# membagi data latih 80% dan data uji 20%
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.20,random_state=0)

len(x_test) #data
```

```
50
```

Gambar 7 Pembagian Data

Tahap selanjutnya adalah membagi data latih dan data uji dengan pembagian 80% data latih yaitu sebanyak 200 data latih dan 20% data uji yaitu sebanyak 50 data uji seperti pada Gambar 7.

Penggunaan algoritma KNN

```
scaler = StandardScaler()
x_train = scaler.fit_transform(x_train)
x_test = scaler.transform(x_test)
selector = SelectKBest(score_func=f_classif, k=2)
x_train_selected = selector.fit_transform(x_train
y_train)
x_test_selected = selector.transform(x_test)
selector.get_support(indices=True))
k_range = range(1,len(x_test)+1)
scores = {}
scores_list = []
for k in k_range:
  knn = KNeighborsClassifier(n_neighbors=k)
  knn.fit(x_train,y_train)
  y_pred = knn.predict(x_test)
  scores[k] = accuracy_score(y_test,y_pred)
```

Gambar 8 *Coding* Penggunaan Algoritma KNN

Setelah melakukan pembagian data latih dan data uji, algoritma KNN disiapkan seperti pada kode Gambar 8 yang menggunakan fitur *selection* terlebih dahulu yang dilanjutkan dengan algoritma KNN.

Setelah memasukkan algoritma KNN, ditampilkan akurasi dari masing - masing nilai K pada Tabel 8.

Tabel 8 Nilai Akurasi dari K

K Value	*Testing Accuracy*
1	0.54
2	0.60
3	**0.70**
4	0.66
5	0.68
6	0.66
7	0.66
8	0.64
9	0.60
10	0.68

Nilai *testing accuracy* pada masing - masing K ditunjukkan pada Tabel 8 yang menunjukkan K=3 memiliki nilai akurasi tertinggi dengan nilai 70%.

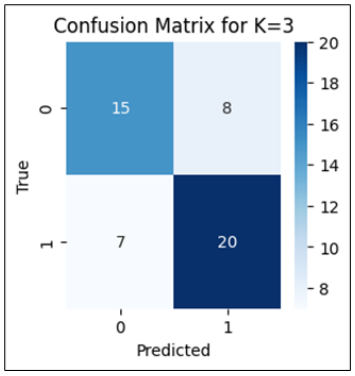
https://doi.org/10.37817/Tekinfo.v26i1
P-ISSN: 1411-3635
E-ISSN: 2962-5645

Gambar 9 Hasil Perhitungan Nilai K = 3

Tahap selanjutnya memasukkan nilai K = 3 seperti pada Gambar 9. Hasil yang ditunjukkan pada Gambar 9 data prediksi tidak semuanya sesuai dengan data asli.

Evaluasi

Evaluasi dilakukan menggunakan metrik evaluasi yaitu akurasi untuk mengukur kinerja model secara keseluruhan, presisi untuk mengukur banyaknya prediksi positif yang benar, *recall* untuk mengukur kembali banyaknya prediksi data positif yang benar terdeteksi, dan *F1-score* untuk menggabungkan presisi dan *recall* menjadi satu metrik yang seimbang.



Gambar 10 *Confusion Matrix*

Dengan K=3 didapatkan *Confusion Matrix* seperti yang ditampilkan pada gambar 10. Berdasarkan *Confusion Matrix*, nilai akurasi, presisi, dan *recall*, sebagai berikut:

a. Akurasi

Akurasi = (TN+TP) / (TN+TP+FN+ FP)

Akurasi = (15+20) / (15+20+7+8) = 0,70

b. Presisi

Presisi = (TP) / (FP + TP)

Presisi = (20) / (8+ 20) = 0,71

c. *Recall*

Recall = TP / (TP + FN)

Recall = (20) / (20 + 7) = 0,74

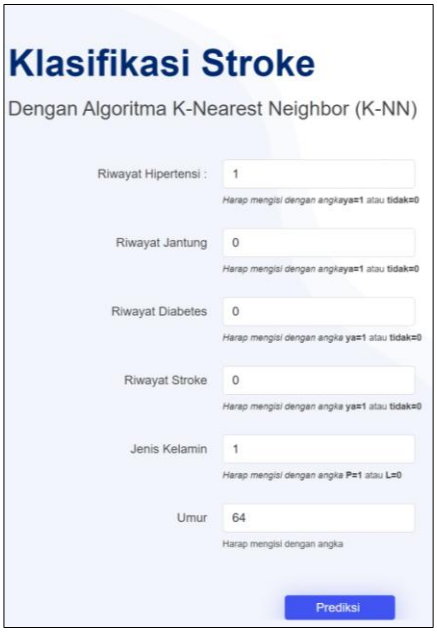
d. *F1-score*

F1-score=2*(*Recall**Presisi)/(*Recall*+Presisi)

F1-score = 2*(0.74*0,71)/(0.74+0,71) = 0,72

Implementasi

Implementasi dilakukan dengan membuat *website* yang dapat diakses untuk melakukan klasifikasi.



Gambar 11 Tampilan *Input*

Gambar 12 Tampilan *Output*

Gambar 11 merupakan tampilan *input* dari sistem prediksi *stroke*. Untuk memprediksi diagnosa *stroke* diperlukan *input* berupa beberapa data terlebih dahulu yang kemudian akan dilakukan perhitungan menggunakan algoritma KNN. *User* hanya perlu mengisi data yang

sesuai dan menekan tombol Prediksi. Gambar 12 merupakan tampilan *ouput* dari sistem prediksi *Stroke* yang akan menampilkan hasil prediksi dari perhitungan menggunakan algoritma KNN.

5. KESIMPULAN

Berdasarkan hasil yang diperoleh pada penelitian ini mengenai penerapan algoritma *K-Nearest Neighbor* (KNN) dalam memprediksi *Stroke*, maka dapat disimpulkan bahwa algoritma *K-Nearest Neighbor* (KNN) dapat diterapkan untuk memprediksi penyakit *Stroke* dengan nilai kinerja K yang optimal dengan menggunakan K = 3 dengan nilai akurasi sebesar 70%, presisi 71%, *recall* 74%, *F1-score 72%*.

6. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan dan membantu dalam penelitian ini. Penulis mengucapkan terima kasih khusus disampaikan kepada Rumah Sakit Pusat Otak Nasional Prof. Dr. dr. Mahar Mardjono Jakarta atas fasilitas dan sumber daya yang diberikan.

DAFTAR PUSTAKA

Akbarollah, M. F., Wiyanto, W., Ardiatma, D., & Zy, A. T. (2023). Penerapan Algoritma K-Nearest Neighbor Dalam Klasifikasi Penyakit Jantung. *Journal of Computer System and Informatics (JoSYC)*, *4*(4), 850–860. https://doi.org/10.47065/josyc.v4i4.4071

Argina, A. M. (2020). *Indonesian Journal of Data and Science Penerapan Metode Klasifikasi K-Nearest Neigbor pada Dataset Penderita Penyakit Diabetes*. *1*(2), 29–33.

Hisni, D., Evelianti Saputri, M., & Surjani. (2022). *Faktor - Faktor yang Berhubungan Dengan Kejadian Stroke Iskemik Di Instalasi Fisioterapi Rumah Sakit Pluit Jakarta Utara Periode Tahun 2021*. https://doi.org/https://doi.org/10.59894/jpkk.v2i1.333

Husada, S., Riview, L., & Puspitasari, P. N. (2020). *Hubungan Hipertensi Terhadap Kejadian Stroke Association Between Hipertension and Stroke Artikel info Artikel history*. *12*, 922–926. https://doi.org/10.35816/jiskh.v10i2.435

Karmila Sari, E., Agata, A., & Studi Keperawatan, P. (2021). Korelasi Riwayat Hipertensi dan Diabetes Mellitus dengan Kejadian Stroke. Dalam *Jurnal Ilmu Keperawatan Indonesia (JIKPI)* (Vol. 2, Nomor 2).

Laela Tusifaiyah, A., Anindhita, N., Saptono, Y., & Kunci, K. (2022). Penerapan Metode Foward Chaning Untuk Diagnosa Penyakit Penyebab Stroke. Dalam *Information System Journal (INFOS)* | (Vol. 5, Nomor 1).

Lie Michael. (2020). *Sistem Skoring Alberta Stroke Program Early CT Score untuk Evaluasi Kasus Stroke Iskemik*.

M. Syukri Mustafa, & I Wayan Simpen. (2019). Implementasi Algoritma K-Nearest Neighbor (KNN) Untuk Memprediksi Pasien Terkena Penyakit Diabetes Pada Puskesmas Manyampa Kabupaten Bulukumba. Dalam *Februari* (Vol. 2019, Nomor 1).

Pratama, Y., Prayitno, A., Azrian, D., Aini, N., Rizki, Y., & Rasywir, E. (2022). Klasifikasi Penyakit Gagal Jantung Menggunakan Algoritma K-Nearest Neighbor. *Bulletin of Computer Science Research*, *3*(1), 52–56. https://doi.org/10.47065/bulletincsr.v3i1.203

Pristian Luthfy Romadloni, Wiga Maulana Baihaqi, & Bagus Adhi Kusuma. (2022). Komparasi Metode Pembelajaran Mesin Untuk Implementasi Pengambilan Keputusan Dalam Menentukan Promosi Jabatan Karyawan. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *6*. https://doi.org/https://doi.org/10.36040/jati.v6i2.5238

Putri Ayundari Setiawan. (2021). *Diagnosis dan Tatalaksana Stroke Hemoragik*. http://jurnalmedikahutama.com

Safitri, N., Kusnandar, D., & Martha, S. (2024). Implementasi Algoritma K-Nearest Neighbor dengan Normalisasi

Z-Score dalam Klasifikasi Penerima Bantuan Sosial Desa Serunai. Dalam *Buletin Ilmiah Math. Stat. dan Terapannya (Bimaster)* (Vol. 13, Nomor 1).

Ulfatul, D., Rachmad, M., Oktavianto, H., & Rahman, M. (2022). Perbandingan Metode K-Nearest Neighbor Dan Gaussian Naive Bayes Untuk Klasifikasi Penyakit Stroke Comparison Of K-Nearest Neighbor And Gaussian Naive Bayes Methods For Stroke Disease Classification. Dalam *Jurnal Smart Teknologi* (Vol. 3, Nomor 4). http://jurnal.unmuhjember.ac.id/index.php/JST

Utama, Y. A., & Nainggolan, S. S. (2022). Faktor Resiko yang Mempengaruhi Kejadian Stroke: Sebuah Tinjauan Sistematis. *Jurnal Ilmiah Universitas Batanghari Jambi*, *22*(1), 549. https://doi.org/10.33087/jiubj.v22i1.1950

Wahab, A., Samarinda, S., Lishania, I., Goejantoro, R., & Nasution, Y. N. (2019). Perbandingan Klasifikasi Metode Naive Bayes dan Metode Decision Tree Algoritma (J48) pada Pasien Penderita Penyakit Stroke di RSUD Comparison of the Classification for Naive Bayes Method and the Decision Tree Algorithm (J48) for Stroke Patients in Abdul Wahab Sjahranie Samarinda Hospital. *Jurnal EKSPONENSIAL*, *10*(2).