Prediksi Kemungkinan Sesorang Dapat Menderita Stroke menggunakan algoritma K-Nearest Neighbor (K-NN) dan Multi-layer Perceptron

Muhamad Rendi

Program Studi Informatika, Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta

Abstract-Stroke adalah kondisi yang terjadi ketika pasokan darah ke otak terganggu atau berkurang akibat penyumbatan (stroke iskemik) atau pecahnya pembuluh darah (stroke hemoragik). Tanpa darah, otak tidak akan mendapatkan asupan oksigen dan nutrisi, sehingga sel-sel pada sebagian area otak akan mati. Kondisi ini menyebabkan bagian tubuh yang dikendalikan oleh area otak yang rusak tidak dapat berfungsi dengan baik. Faktor yang menyebabkan terjadinya stroke antara lain adalah faktor kesehatan, faktor gaya hidup, faktor usia bahkan faktor keturunan. Berdasarkan laporan dari Pusat Data dan Informasi Kementrian Kesehatan RI, secara nasional prevalensi stroke di Indonesia tahun 2018 berdasarkan diagnosis dokter pada penduduk berumur diatas 15 tahun sebesar 10,9%, atau diperkirakan sebanyak 2.120.362 orang. Maka dari itu diperlukan kesadaran bagi semua orang untuk menghindari resiko-resiko yang dapat menyebabkan penyakit stroke. Untuk menyadarkan semua orang akan bahaya stroke perlu adanya keterlibatan teknologi modern sebagai pencegahan penderita mengalami stroke yang tiba-tiba. Salah satunya menggunakan model Machine Learning untuk menginformasi lebih awal tanda-tanda seseorang akan terkena stroke atau tidak. Dengan adanya model ini diharapkan semua orang dapat mampu menjaga kesehatan agar tidak beresiko terkena penyakit stroke dengan melakukan gaya hidup sehat dan selalu menjaga kesehatan tubuh. Pada penelitian ini menggunakan algoritma K-Nearest Neighbor (K-NN) dan Multi-layer Perceptron untuk pembuatan modelnya. Model yang dirancang menggunakan K-NN menghasikan skor akurasi 89%, precision 90%, recall 89%, dan f1-score 89%. Sedangkan model yang dirancang menggunakan Multi-layer Perceptron menghasilkan skor akurasi 88%, precision 88%, recall 88%, dan f1-score 88%. Skor yang ditunjukkan menandakan bahwa hasil prediksi dari model yang dirancang dengan algoritma K-NN memiliki hasil yang lebih baik dari pada model yang dirancang dengan algoritma Multi-layer Perceptron.

Keywords—Stroke, Kesehatan, K-Nearest Neighbor, Multi-Layer Perceptron, dan WHO.

I. PENDAHULUAN

Menurut WHO (The World Health Organization) definisi stroke adalah berkembang pesat tanda-tanda klinis gangguan fungsi otak fokal (atau global), dengan gejala berlangsung 24 jam atau lebih atau menyebabkan kematian, tanpa penyebab yang jelas selain dari vaskular[1]. Stroke adalah kondisi yang terjadi ketika pasokan darah ke otak terganggu atau berkurang akibat penyumbatan (stroke iskemik) atau pecahnya pembuluh darah (stroke hemoragik). Tanpa darah, otak tidak akan mendapatkan asupan oksigen dan nutrisi, sehingga sel-sel pada sebagian area otak akan mati. Kondisi ini menyebabkan bagian tubuh yang dikendalikan oleh area otak yang rusak tidak dapat berfungsi dengan baik. Faktor yang menyebabkan terjadinya stroke antara lain adalah faktor kesehatan, faktor gaya hidup, faktor usia bahkan faktor keturunan.

Diperkirakan 17,9 juta orang meninggal karena CVD (Penyakit kardiovaskular) pada 2019, mewakili 32% dari semua kematian global. Dari kematian tersebut, 85% disebabkan oleh serangan jantung dan stroke menurut laporan dari WHO (The World Health Organization)[2]. Berdasarkan laporan dari Pusat Data dan Informasi Kementrian Kesehatan RI, secara nasional prevalensi stroke di Indonesia tahun 2018 berdasarkan diagnosis dokter pada penduduk berumur diatas 15 tahun sebesar 10,9%, atau diperkirakan sebanyak 2.120.362 orang[3]. Maka dari itu diperlukan kesadaran bagi semua orang untuk menghindari resiko-resiko yang dapat menyebabkan penyakit stroke.

Berdasarkan pernyataan diatas muncul ide peneliti untuk membuat model machine learning tentang Prediksi Kemungkinan Sesorang Dapat Menderita Stroke menggunakan algoritma K-Nearest Neighbor (K-NN) dan Multi-layer Perceptron yang dimana tujuan dari proyek ini yaitu untuk mengklasifikasikan orang-orang yang beresiko terkena penyakit stroke atau tidak. Dengan adanya model ini diharapkan semua orang dapat mampu menjaga kesehatan agar tidak beresiko terkena penyakit stroke dengan melakukan gaya hidup sehat dan selalu menjaga kesehatan tubuh. Implementasinya model ini dapat dijalankan pada sebuah platform aplikasi web ataupun android.

Dalam penelitian ini, selain membuat model machine learning untuk memprediksi stroke, peneliti juga mengevaluasi dan membandingkan performa dari 2 algoritma yang digunakan sebagai model prediksi stroke yaitu K-Nearest Neighbor dan Multi-Layer Perceptron. Penelitian ini bertujuan menawarkan model arsitektur machine learning yang memiliki akurasi lebih baik.

II. STUDI LITERATURE

A. K-NN Overview

K-Nearest Neighbor atau yang dikenal sebagai K-NN adalah suatu metode yang menggunakan algoritma supervised dimana hasil dari query instance yang baru diklasifikan berdasarkan mayoritas dari kategori pada KNN (Sikki, 2009). Prinsip kerja KNearest Neighbor adalah mencari jarak terdekat antara data yang akan di evaluasi dengan k tetangga (neighbor) dalam data pelatihan (Whidhiasih et al., 2013). Tujuan dari algoritma KNN adalah untuk mengklasifikasi objek baru berdasarkan atribut dan training samples. Dimana hasil dari sampel uji yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada KNN (Krisandi et al, 2013). Jarak yang digunakan adalah jarak Euclidean Distance. Jarak Euclidean adalah jarak yang paling umum digunakan pada data numerik (Krisandi et al, 2013). Euclidean distance didefinisikan sebagai berikut (Krisandi et al., 2013).

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (a_r(x_i) - a_r(x_j))^2} \quad (1)$$

Dimana:

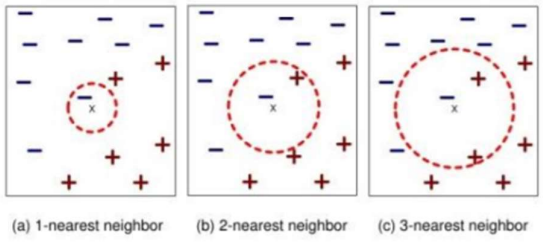
$d(x_i, x_j)$: Jarak Eunclidean (Euclidean Distance)

(x_i) : record ke-i

(x_j) : record ke-j

(a_r) : data ke-r

i,j : 1,2,3,….n



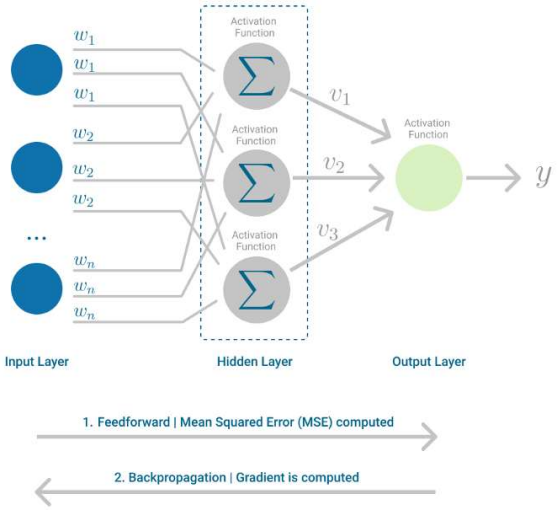
Gambar 1. Ilustrasi K-Nearest Neighbor

Setelah mengambil k tetangga terdekat pertama kemudian dihitung jumlah data yang mengikuti kelas yang ada dari k tetangga tersebut. Kelas dengan data terbanyak yang mengikutinya menjadi kelas pemenang yang diberikan sebagai label kelas pada data X.

Pada kNN, nilai k dapat memberikan pengaruh terhadap performa klasifikasi yang dihasilkan. Jika nilai k terlalu kecil

B. Multi-Layer Perceptron Overview

Jaringan saraf tiruan lapis banyak atau disebut multilayer perceptron merupakan pengembangan lebih lanjut dari perceptron lapis tunggal. Pembelajaran menggunakan algoritma delta yang disebut error backpropagation training algorithm, argument dimasukkan diumpamakan secara arah maju sedangkan proses pembelajaran selain melakukan perambatan arah maju juga memanfaatkan perambatan arah balik. Apabila hasil tidak sesuai dengan target maka bobot diperbaharui selama proses siklus pembelajaran hingga tercapai nilai kemelesetan minimum yang diharapkan atau keluaran sama dengan target. (Muis, 2006)



Gambar 2. Ilustrasi Multi-Layer Perceptron

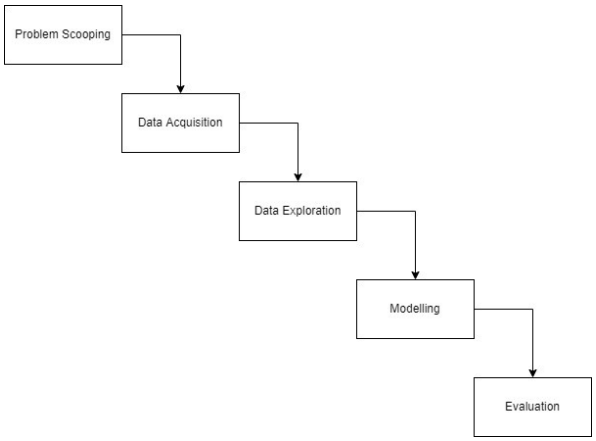
Dimana penjelasan cara kerja MLP sebagai berikut:

i. Input didorong maju melalui MLP dengan mengambil dot product dari input tersebut dengan bobot-bobot yang ada di antara input layer dan hidden layer. Dot product ini menghasilkan nilai pada lapisan tersembunyi.
ii. MLP menggunakan fungsi aktivasi di setiap lapisan yang dihitung. Dorong keluaran terhitung pada lapisan saat ini melalui salah satu fungsi aktivasi.
iii. Setelah output yang dihitung pada lapisan tersembunyi telah didorong melalui fungsi aktivasi, dorong ke lapisan berikutnya di MLP dengan mengambil dot product dengan bobot yang sesuai.
iv. Mengulangi langkah dua dan tiga sampai lapisan output tercapai.
v. Pada lapisan output, perhitungan akan digunakan untuk algoritma backpropagation yang sesuai

dengan fungsi aktivasi yang dipilih untuk MLP (dalam kasus pelatihan) atau keputusan akan dibuat berdasarkan output (dalam kasus pengujian).

III. METODOLOGI PENELITIAN

Penelitian ini menggunakan beberapa langkah penelitian, yaitu : problem scoping, data acquisition, data exploration, modelling dan evaluation. Tahap pada penelitian ini juga dapat dilihat pada gambar berikut :



Gambar 3. Diagram Penelitian

A. Problem Scooping

Problem scoping adalah tahap dimana dilakukan analisis terhadap masalah yang sedang terjadi. Pada penelitian ini dilakukan analisis terhadap permasalahan pemilihan sepeda gunung oleh komunitas penggemar sepeda dan juga masyarakat yang mulai menggemari olahraga sepeda. Untuk mendapatkan hasil analisis yang lebih akurat dilakukan paper review. Paper review juga digunakan untuk mendapatkan mengetahui metode apa yang paling cocok digunakan untuk menyelesaikan permasalahan.

B. Data Acquisition

Data acquisition adalah tahap dimana dilakukan pengumpulan data apa yang diperlukan. Pengumpulan dataset dari penelitian ini diperoleh melalui situs kaggle (https://www.kaggle.com/fedesoriano/stroke-prediction-dataset). Kumpulan dataset tersebut akan digunakan untuk melatih dan mengembangkan model AI yang dapat memprediksikan Kemungkinan Sesorang Dapat Menderita Stroke

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   id                 5110 non-null   int64
 1   gender             5110 non-null   object
 2   age                5110 non-null   float64
 3   hypertension       5110 non-null   int64
 4   heart_disease      5110 non-null   int64
 5   ever_married       5110 non-null   object
 6   work_type          5110 non-null   object
 7   Residence_type     5110 non-null   object
 8   avg_glucose_level  5110 non-null   float64
 9   bmi                4909 non-null   float64
 10  smoking_status     5110 non-null   object
 11  stroke             5110 non-null   int64
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
```

Gambar 4. dataset penelitian

C. Data Exploration

Data exploration adalah tahap dimana dilakukan pemahaman terhadap data. Pada penelitian ini dilakukan pembersihan dataset untuk memahami data. Hasil dari pengumpulan dataset melalui situs kaggle masih bersifat mentah dan belum teratur, maka dilakukan preprocessing data. Preprocessing dilakukan dengan beberapa tahap sebagai berikut:

i. *Menkonversi label yang bertipe data kategori ke dalam label numerik biner menggunakan teknik one-hot encoding*

Dikarenakan data pada dataset stroke prediction memiliki beberapa data kategorikal atau teks maka harus diubah menjadi data numerik dan tetap membuat algoritma atau model untuk dapat memahaminya. Untuk mengubah data kategorikal atau teks disini menggunakan teknik one-hot encoding, teknik one-hot encoding adalah teknik yang merubah setiap nilai di dalam kolom menjadi kolom baru dan mengisinya dengan nilai biner yaitu 0 dan 1. Proses ini diperlukan karena model tidak bisa memproses data teks melainkan data numerik.

ii. *Melakukan standarisasi data pada semua fitur data*

Standarisasi data membuat semua fitur numerik berada dalam skala data yang sama dan dapat membuat komputasi dari pembuatan model dapat berjalan lebih cepat karena rentang datanya hanya antara 0-1. Untuk melakukan proses tersebut digunakan fungsi MinMaxScaler yang rumusnya dapat dilihat seperti dibawah ini:

$$x^I = \frac{x - \min(x)}{\max(x) - \min(x)}$$

iii. *Mengatasi data yang kosong pada dataset dengan menggunakan metode KNN Imputation.*

Terdapat dataset yang kosong pada kolom bmi, maka untuk mengatasinya terdapat dua pilihan yaitu metode dengan menghapus data atau metode dengan menambahkan data. Pemilihan metode dengan menghapus data bukanlah hal yang bijak karena akan mengakibatkan model yang nantinya akan dibuat kehilangan banyak informasi. Sehingga dipilihlah cara untuk memanipulasi datanya, dengan mengisi data yang kosong dengan nilai rata-rata dari tetangga terdekat yang diukur dengan jarak Euclidean. Pada tahap ini digunakan fungsi KNNImputer untuk mengganti data yang kosong.

iv. *Mengatasi masalah yang data yang jumlahnya tidak seimbang dengan menggunakan teknik SMOTE (Synthetic Minority Oversampling Technique)..*

Dataset yang tidak seimbang pada data kategori akan menyebabkan model yang dibuat menjadi bias terhadap suatu kategori yang memiliki data lebih banyak. Oleh karena itu diperlukan teknik manipulasi data, dan yang digunakan di sini adalah teknik SMOTE (Synthetic Minority Oversampling Technique). SMOTE mengadopsi teknik K-Nearest Neighbors dalam membuat instance data baru jadi untuk prosesnya kurang lebih hampir sama dengan KNN Imputation.

v. *Melakukan pembagian dataset menjadi dua bagian dengan rasio 80% untuk data latih dan 20% untuk data uji.*

Agar dapat menguji performa dari model pada data sebenarnya, maka perlu dilakukan pembagian dataset kedalam dua atau tiga bagian. Pada proyek ini dilakukan dua bagian saja yakni pada data latih dan data uji dengan rasio 80:20.

D. Modeling

Modelling merupakan tahap pembuatan model dari sistem prediksi yang dibuat. Pada penelitian ini peneliti menggunakan K-Nearest Neighbor dan Multi-layer Perceptron. Pemilihan algoritma ini didasarkan pada dataset yang dimiliki peneliti memiliki data berkategori dan data numerik sehingga cocok menggunakan 2 algoritma tersebut. Implementasi algoritma tersebut dilakukan melalui Google Colab dengan library dari scikit-learn.

E. Evaluation

Dalam penelitian ini evaluasi dilakukan dengan menggunakan metriks akurasi, f1-score, recall dan precision. Indeks-indeks tersebut digunakan untuk menentukan apakah model mempunyai performa baik atau tidak. Penjelasannya sebagai berikut:

i. *Akurasi*

Akurasi adalah ukuran keakuratan model saat menggunakan data aktual untuk memprediksi data. Akurasi dapat dihitung dengan rumus di atas. Kelebihan dari metrik ini adalah sering digunakan untuk membuat model klasifikasi, baik itu klasifikasi dua kelas maupun kategori. Kerugian dari indikator ini adalah dapat "menyesatkan" data yang tidak seimbang.

$$\texttt{accuracy}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} 1(\hat{y}_i = y_i)$$

ii. *Precision*

Precision adalah metrik dalam kasus klasifikasi, yang digunakan untuk menghitung efek model dalam memprediksi label positif terhadap semua label positif model. Jadi bagaimana cara menghitungnya, pertama kita perlu memahami istilah TP, TN, FP, FN. Deskripsi singkat ditunjukkan pada tabel di bawah ini.

Tabel 1. Tabel Confusion Matriks

	Actual class (observation)	
Predicted class (expectation)	tp (true positive) Correct result	fp (false positive) Unexpected result
	fn (false negative) Missing result	tn (true negative) Correct absence of result

$$Precision = \frac{tp}{tp + fp}$$

iii. *Recall*

Recall adalah metrik dalam kasus klasifikasi, yang digunakan untuk menghitung efek model dalam memprediksi label positif untuk semua label data positif. Cara menghitungnya bisa dilihat pada rumus di bawah ini.

$$recall = \frac{tp}{tp + fn}$$

iv. *F1-Score*

f1-score merupakan metrik dalam kasus klasifikasi yang digunakan untuk menghitung seberapa baik hasil prediksi model (precision) dan seberapa lengkap hasil prediksinya (recall). Cara menghitungnya dapat dilihat pada rumus dibawah ini.

$$F_\beta = (1 + \beta^2) \frac{Precision \times recall}{\beta^2 precision + recal}$$

IV. HASIL DAN PEMBAHASAN

Setelah melakukan pra-pemrosesan data yang baik pada tahap modeling akan dilakukan pembuatan dua model sekaligus, yakni model dengan MLPClassifier dan KNeighborsClassifier. Pada tahapan dalam pembuatan model pertama yakni membuat model dengan MLPClassifier yakni dilakukan kompilasi model dan memanggil fungsi fit untuk memulai proses pelatihan dengan parameter yaitu data X_Train dan Y_Train yang sudah dipersiapkan sebelumnya, setelah itu dilakukan pengujian dari model. Setelah

pembuatan model pertama dilanjutkan membuat model yang kedua sebagai pembanding dari model MLPClassifier dengan membuat model KNeighborsClassifier. Pada pembuatan model kedua prosesnya mirip dengan pembuatan model yang pertama yaitu melakukan kompilasi model dan memanggil fungsi fit untuk memulai proses pelatihan dengan parameter yaitu data X_Train dan Y_Train yang sudah dipersiapkan sebelumnya, setelah itu dilakukan pengujian dari model.

A. Hasil Model K-NN

Dari hasil model menggunakan algoritma K-Nearest Neighboar atau K-NN didapatkan sebagai berikut.

	precision	recall	f1-score	support
Healthy	0.964200	0.828718	0.891340	975.000000
Stroke	0.849142	0.969072	0.905152	970.000000
accuracy	0.898715	0.898715	0.898715	0.898715
macro avg	0.906671	0.898895	0.898246	1945.000000
weighted avg	0.906819	0.898715	0.898228	1945.000000

Gambar 5. Performa Model dengan Algoritma K-NN

B. Hasil Model MLP

Dari hasil model menggunakan algoritma Multi-Layer Perceptron atau MLP didapatkan sebagai berikut.

	precision	recall	f1-score	support
Healthy	0.918051	0.850256	0.882854	975.000000
Stroke	0.859885	0.923711	0.890656	970.000000
accuracy	0.886889	0.886889	0.886889	0.886889
macro avg	0.888968	0.886984	0.886755	1945.000000
weighted avg	0.889043	0.886889	0.886745	1945.000000

Gambar 6. Performa Model dengan Algoritma MLP

C. Hasil pengujian Model

Setelah dilakukan pengujian, model yang dirancang menggunakan K-NN menghasikan skor akurasi 89%, precision 90%, recall 89%, dan f1-score 89%. Sedangkan model yang dirancang menggunakan Multi-layer Perceptron menghasilkan skor akurasi 88%, precision 88%, recall 88%, dan f1-score 88%. Skor yang ditunjukkan menandakan bahwa hasil prediksi dari model yang dirancang dengan algoritma K-NN memiliki hasil yang lebih baik dari pada model yang dirancang dengan algoritma Multi-layer Perceptron.

		Healthy			Stroke		
	accuracy	precision	recall	f1-score	precision	recall	f1-score
Model Multi-Layer Perceptron	0.886889	0.918051	0.850256	0.882854	0.859885	0.923711	0.890656
Model K-Nearest Neighbors	0.898715	0.964200	0.828718	0.891340	0.849142	0.969072	0.905152

V. KESIMPULAN

Pada penelitian ini, peneliti membandingkan sekaligus mengusung Model untuk memprediksi dan mengklasifikasi orang yang berkemungkinan stroke. Tujuan peneliti adalah meminimalisir terjadinya kematian akibat stroke secara tiba-tiba.

Hasil dari pengujian model memperlihatkan bahwa model yang dirancang menggunakan K-NN menghasikan skor akurasi 89%, precision 90%, recall 89%, dan f1-score 89%. Sedangkan model yang dirancang menggunakan Multi-layer Perceptron menghasilkan skor akurasi 88%, precision 88%, recall 88%, dan f1-score 88%. Skor yang ditunjukkan menandakan bahwa hasil prediksi dari model yang dirancang dengan algoritma K-NN memiliki hasil yang lebih baik dari pada model yang dirancang dengan algoritma Multi-layer Perceptron.

DAFTAR PUSTAKA

[1] [1] Truelsen, T. dan Begg, S. (2006), The Global Burden Of Cerebrovascular Disease, World Health Organization.

[2] Cardiovascular diseases (CVDs).(2021, Juni 11). who.int. Medium. https://www.who.int/en/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds) [Diakses 15 januari 2022].

[3] Infodatin Stroke.(2019, November 29). pusdatin.kemkes.go.id. Medium. https://pusdatin.kemkes.go.id/article/view/20031000003/infodatin-stroke.html [Diakses 15 Januari 2022].

[4] Harrison, O. (2019, July 14). Machine Learning Basics with the K-Nearest Neighbors Algorithm. Medium. https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761 [Diakses 15 Januari].

[5] Lambrou, A dan Harris, P. (2010). Assessment of Stroke Risk Based on Morphological Ultrasound Image Analysis with Conformal Prediction. IFIP International Federation for Information Processing. No. 146–153.

[6] Akter, S. dan Amin, A. (2021). Stroke prediction analysis using machine learning classifiers and feature technique. International Journal of Electronics and Communications System. Volume 1, Issue 2, 17-22