

# IMPLEMENTASI ALGORITMA *SUPPORT VECTOR MACHINE* (SVM) UNTUK PREDIKSI PENYAKIT *STROKE* DENGAN ATRIBUT BERPENGARUH

Ulfa Amelia

Universitas Buana Perjuangan Karawang  
Karawang, Indonesia  
If17.ulfaamelia@ubpkarawang.ac.id

Jamaludin Indra

Universitas Buana Perjuangan Karawang  
Karawang, Indonesia  
jamaludin.indra@ubpkarawang.ac.id

Anis Fitri Nur Masruriyah

Universitas Buana Perjuangan Karawang  
Karawang, Indonesia  
anis.masruriyah@ubpkarawang.ac.id

**Abstract**— Strok menjadi penyakit yang menduduki peringkat ketiga di Indonesia setelah jantung dan kanker. Seringkali manusia lengah dalam menyadari adanya penyakit Strok. Kurangnya tenaga medis di Indonesia membuat masyarakat sulit untuk mendeteksi dini penyakit strok. Oleh karena itu penelitian ini dilakukan untuk memprediksi adanya penyakit Strok menggunakan Algoritma *Support Vector Machine* (SVM) untuk klasifikasi himpunan data yang menggunakan Metode *Confusion Matrix*. Pengujian algoritma SVM menggunakan Kernel *Linear* untuk mendapatkan hasil terbaik. Penelitian menggunakan algoritma *Support Vector Machine* (SVM) dengan *Relief-f*. Data yang menggunakan 3426 Baris dan lima kolom. Hasil pengujian menghasilkan akurasi data sebesar 100%.

**Kata kunci** — Algoritma *Support Vector Machine* (SVM), *Data Mining*, *Strok*

## I. PENDAHULUAN

*World Health Organization* (WHO) mendefinisikan penyakit strok sebagai kerusakan mendadak pada fungsi otak yang terjadi dalam 24 jam atau lebih [1]. Data statistik penyakit strok dunia menyatakan sekitar 15 juta orang di dunia terkena strok setiap tahunnya dan satu dari enam orang di dunia akan terkena Strok [2]. Kasus strok di Indonesia terdiagnosis sebanyak 43,1% pada usia 75 tahun keatas dan 0,2% pada usia rentang dari 15-24 tahun oleh tenaga kesehatan [3]. Jika penyakit strok tidak ditangani dengan baik, maka tingkat kematian akan semakin bertambah.

Telah dilakukan diagnosa penyakit strok *Transient Ischaemic Attack* (TIA) menggunakan Metode *Dempster Shafer* [4]. Penelitian tersebut menggunakan data gejala dengan total 10 objek kemudian diolah dengan metode *Dempster Shafer*. Hasil dari penelitian tersebut algoritma *Dempster Shafer* berhasil melakukan diagnosa penyakit strok jenis TIA. Selanjutnya penelitian menggunakan algoritma *Probabilistic Symmetric* untuk mendiagnosa penyakit Strok [5]. Pada kasus tersebut ditemukan bahwa dokter spesialis Strok di Indonesia masih kurang. Sehingga, membuat masyarakat sulit untuk mendapatkan pemeriksaan yang cepat. Penelitian tersebut membangun sistem pakar dengan metode *Case Based Reasoning* dan Algoritma *Probabilistic Symmetric* yang dapat membantu mendiagnosa penyakit Strok dengan cepat dan mudah. Hasil menunjukkan bahwa web aplikasi yang dirancang dapat mendiagnosa awal gejala Strok tanpa menemui seorang dokter. Akurasi yang didapatkan dari hasil penelitian ini mendapatkan akurasi sebesar 66.66%.

Telah dilakukan diagnosis penyakit stroke menggunakan *K-Nearest Neighbor* dan *Naïve Bayes* [6]. Stroke penyebab kecacatan nomor satu di Indonesia, penelitian menggunakan atribut kategorik dan numerik. Algoritme KNN digunakan untuk proses data numerik dan algoritma *Naïve Bayes* digunakan untuk data kategorik. Hasil akurasi pada data seimbang adalah 96.67% dan akurasi pada data tidak seimbang menghasilkan akurasi sebesar 100%. Telah dilakukan penerapan optimasi dengan metode *Adaboost* menggunakan algoritma *Naïve Bayes* [7]. Penyakit stroke memerlukan perhatian khusus karena stroke merupakan penyakit nomor dua mematikan di dunia. Oleh karena itu penelitian dilakukan menggunakan algoritma *Naïve Bayes* menggunakan metode *Adaboost*. Data menggunakan *split data* dalam melakukan pengujian. Pengujian menggunakan *Naïve Bayes* menghasilkan akurasi sebesar 0.976. pengujian kedua menggunakan *Naïve Bayes* dan metode *Adaboost* menghasilkan akurasi sebesar 0.981.

Kemudian penelitian yang mendiagnosa penyakit Strok dengan tipe yang berbeda menggunakan *Intuitionistic Fuzzy* telah dilakukan oleh Djatna [8]. Penelitian tersebut menggunakan data pasien penyakit stroke dengan 114 atribut dengan total data 20 variabel. Proses penelitian dimulai dengan mengkonversi data dengan *intuitionistic fuzzy*, kemudian diproses dengan pohon keputusan berbasis *intuitionistic fuzzy*. Hasil akurasi penelitian tersebut mencapai 90,59% dalam mengklasifikasi penyakit Strok. Lalu penelitian analitik prediktif menggunakan algoritma *Artificial Neural Network* (ANN) penyakit Strok dilakukan oleh Masruriyah [9]. Penelitian menggunakan data catatan medis pasien berjumlah 18,425 dan memakai 10 variabel teratas untuk mendapatkan estimasi terbaik. Implementasi menggunakan metode *K-Fold Cross Validation* untuk membagi data *training* dan data *testing*. Hasil yang didapat dalam penelitian menghasilkan tingkat akurasi sebesar 95.15%. Penelitian prediksi penyakit Strok juga dilakukan menggunakan metode *Naïve Bayes-Certainly Factor* [10] dan metode *Learning Vector Quantization* [11].

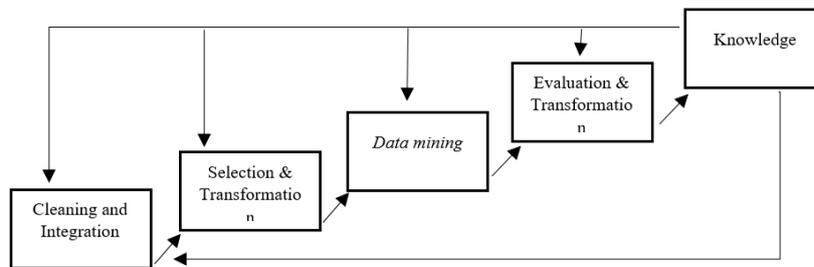
II. DATA DAN METODE

A. Strok

Strok merupakan penyakit *cerebrovascular* atau kerusakan otak yang menyumbat pembuluh darah sehingga menghambat asupan suplai darah ke otak [1]. Tersumbatnya pembuluh darah mengakibatkan nutrisi dan oksigen tidak masuk ke jaringan otak [1]. Menurut *World Health Organizatin* (WHO) pada tahun 2012 kematian dari penyakit strok sebesar 51% diseluruh dunia yang disebabkan oleh tekanan darah tinggi [3]. Gaya hidup yang salah merupakan faktor risiko pada Strok berulang adalah sama dengan faktor risiko pada Strok pertama [2].

B. Data Mining

*Data mining* didefinisikan sebagai pencarian data yang memilih sebuah pola dengan metode tertentu. Teknik algoritma metode atau cara yang tepat dalam pengolahan data sangat mempengaruhi dalam *data mining*, dengan memanfaatkan ilmu lain seperti Matematika, Statistik dan pengenalan pola [10]. *Data mining* juga mendefinisikan analisis otomatis pada suatu data yang yang berjumlah besar untuk menentukan pola penting yang cenderung tidak disadari keberadaanya [11].



Gambar 1 Tahapan *Data mining* [11]

C. Algoritma *Relief-f*

Algoritma *Relief-f* merupakan metode seleksi fitur yang dikembangkan dari algoritma *relief* dan dipublikasikan oleh Kononenko pada tahun 1994 [12]. Metode yang umumnya digunakan dalam *relief-f* adalah *Entropy-Based Ranking*, *Term Strength*, *Term Contribution*, *Document Frequency Based Selection* [13]. Algoritma *Relief-f* tidak membatasi karakteristik serta memiliki tingkat efisiensi yang tinggi [12].

Tahap-tahap untuk memilih atribut yang akan dipilih oleh algoritma *Relief-f* yaitu [12]:

1. Penentuan jumlah iterasi dan inisiasi seluruh nilai bobot fitur = 0
2. Pemilihan data yang akan digunakan sebagai titik pusat
3. Mencari *hit* dan *miss* terdekat dengan cara perhitungan antara data yang memiliki kelas yang sama dengan titik pusat. *Hit* merupakan kelas positif antara jarak dan titik pusat. *Miss* merupakan kelas negatif antara jarak dan titik pusat.
4. Melakukan pembaruan bobot pada setiap fiturnya. Fitur dengan data kategori dihitung menggunakan persamaan 1.

$$diff(A, Ri, HM) = \begin{cases} 0; & \text{Value}(A, Ri) = \text{value}(A, H, M) \\ 1; & \text{otherwise} \end{cases} \quad 1$$

5. Sedangkan fitur dengan data numerik menggunakan persamaan 2.

$$diff(A, Ri, HM) = \frac{|value(A, H, M)|}{Max(A) - min(A)} \quad 2$$

6. Sehingga perbaruan bobot dihitung menggunakan persamaan 3.

$$W[A] = W[A] - \frac{diff(A, Ri, H)}{m} + \frac{diff(A, Ri, M)}{m} \quad 3$$

7. Dilanjut dengan iterasi yang dimulai dari langkah satu sampai dengan bobot baru yang telah didapatkan.

D. Algoritma *Support Vector Machine* (SVM)

*Support Vector Machine* (SVM) adalah sistem pembelajaran dengan menggunakan teori berupa fungsi linear dalam suatu fitur yang dilatih menggunakan Algoritma yang didasari oleh teori yang optimal [14]. SVM dikembangkan oleh Boser, Guyon, Vapnik, dan pertama kali dipublikasikan pada tahun 1992 di *Annual Workshop on Computational Learning Theory*. Teori dasar SVM diperoleh dari percampuran teori komputasi yang telah ada sebelumnya. Prinsip dasar algoritma ini adalah Klasifikasi *Linear*, kemudian dikembangkan agar dapat berfungsi pada Klasifikasi *Non-Linear* [15]. Prinsip dasar SVM adalah pengembangan yang mengklasifikasikan *linear* agar dapat diproses pada masalah *non-linear*. Prinsip dasar ini memakai metode *kernel trick* pada fitur berdimensi tinggi [15]. Hasil akurasi data yang dihasilkan algoritma SVM ditentukan oleh parameter dan fungsi *kernel* yang digunakan. Beberapa tipe *kernel* yang dapat digunakan ditunjukkan pada Tabel 2.1 [14].

Tabel 2.1 Tabel Rumus *Kernel* SVM

Nama <i>Kernel</i>	Fungsi <i>Kernel</i>
<i>Linear</i> (Dot)	$G(x_1, x_2) = x_1'x_2$
<i>Radial Basis Function</i> (RBF)	$G(x_1, x_2) = exp$
<i>Polynomial</i>	$G(x_1, x_2) = \gamma x_1x_2 + c$

Regresi *non-linear* menggantikan nilai awal prediksi  $X^T Z$  dengan fungsi yang sesuai dengan matriks gram ( $g_i$ ) Pada rumus ganda Algoritma SVM. SVM regresi *non-linear* dapat diminimalisir pada persamaan 4:

$$L(a) = \frac{1}{2} \sum_{i=1}^w \sum_{j=1}^w (\alpha_i - \alpha_j)(\alpha_j - \alpha_j)G(x_i - x_j) + \epsilon \sum_{i=1}^N (\alpha_i + \alpha_i) - \sum_{i=1}^N y_i (\alpha_i + \alpha_2) \quad 4$$

Di mana:

$$\sum_{n=1}^N (\alpha_n - \alpha_n) = 0 \quad 5$$

$$\forall sn: \leq \alpha_n \leq C \quad 6$$

$$\forall sn: \leq \alpha_n \leq C (4) \quad 7$$

Keadaan *Karush-Kuhn-Tucker* (KKT) wajib melengkapi kondisi seperti persamaan 8, 9, 10, 11.

$$\forall n: \alpha_n (\epsilon + \xi_n - y_n + f(x_n)) = 0 \quad 8$$

$$\forall n: \alpha_n (\epsilon + \xi_n + y_n - f(x_n)) = 0 \quad 9$$

$$\forall n: \xi_n (C - \alpha_n) = 0 \quad 10$$

$$\forall n: \xi_n (C - \alpha_n) = 0 \quad 11$$

Persamaan diatas menunjukkan bahwa epsilon menggunakan fungsi ganda *Lagrange*  $an = 0$  dan  $an = 0$ . Dukungan vector atau support vector merupakan pengganda *Lagrange* yang bukan nol. Prediksi nilai baru pada SVM dapat menggunakan fungsi pada persamaan 12 berikut:

$$f(x) - \sum_{n=1}^N (\alpha_n - \alpha_n)Gx_n, x+b \quad 12$$

E. *Confusion Matrix*

Data yang akan diuji tingkat akurasi menggunakan *Confusion Matrix*. Metode *Confusion Matrix* digunakan sebagai pengukur proses suatu metode klasifikasi. *Confusion Matrix* mengandung empat istilah yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *False Negative* (FN) [16]. Tabel 1 merupakan dasar untuk melakukan perhitungan *Confusion Matrix* [17].

Tabel 1 Tabel Metode *Confusion Matrix*

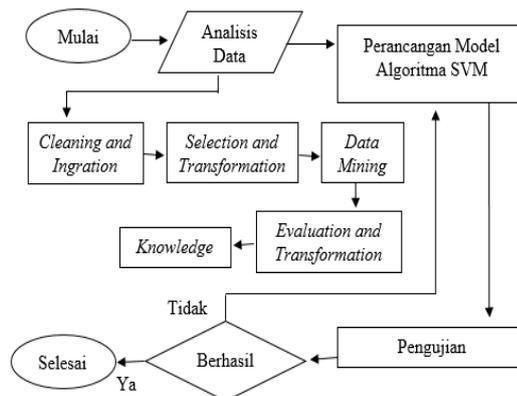
		<i>Observed</i>	
		<i>True</i>	<i>False</i>
<i>Predicted Class</i>	<i>True</i>	<i>True Positive (TP)</i>	<i>False Positive (FP)</i>
	<i>False</i>	<i>False Negative (FN)</i>	<i>True Negative (TN)</i>

*True Positive* (TP) adalah akumulasi data positif yang terklasifikasi dengan tepat oleh sistem. Kemudian *True Negative* (TN) adalah akumulasi data negatif yang terklasifikasi dengan tepat oleh sistem. *False Negative* (FN) adalah akumulasi data negatif yang terklasifikasi salah oleh sistem. *False Positive* (FP) adalah akumulasi data positif yang terklasifikasi salah oleh sistem. Untuk menghitung jumlah klasifikasi, dapat dilakukan dengan cara perhitungan Akurasi. Akurasi merupakan hasil dari suatu proses yang menunjukkan nilai data antara data yang terklasifikasi dengan benar dari keseluruhan data dalam bentuk persentase untuk memprediksi diagnosa penyakit Strok. Rumus akurasi dapat digunakan dengan persamaan 4.

$$Akurasi = \frac{TN + TP}{TP + TN + FP + FN} \quad 13$$

F. Prosedur Penelitian

Prosedur Penelitian ini dimulai dengan analisis *dataset* dengan rincian penelitian sebagai berikut:



Gambar 2 Gambar Prosedur Peneliiian

III. HASIL DAN PEMBAHASAN

A. Analisis Data

Data diperoleh dari situs *website* kaggle sebagai acuan data dalam penelitian ini dan data diproses menjadi beberapa proses. Data yang diperoleh terdapat 11 kolom (Jenis Kelamin, Umur, Hipertensi, Jantung, Pekerjaan, Jenis Pekerjaan, Perumahan, Glukosa, Massa Indeks Tubuh, Perokok, dan Diagnosa Strok) dan 3427 baris. Data terbagi menjadi dua bagian yaitu Data *Testing* dan Data *Training*. Data *Training* merupakan data asli yang sudah menghasilkan diagnosa. Data *Testing* merupakan data untuk menguji Algoritma tanpa menginput data diagnosa.

Tabel 2 Tabel Data Keterangan

Variabel Kolom	Nomor	Keterangan
<i>Gender</i>	1	Jenis Kelamin
<i>Age</i>	2	Umur
<i>Hypertension</i>	3	Riwayat Hipertensi
<i>Heart Disease</i>	4	Riwayat Jantung
<i>Ever Married</i>	5	Pernah Menikah
<i>Work Type</i>	6	Jenis Pekerjaan
<i>Residence Type</i>	7	Tipe Perumahan
<i>Average Glucose Level</i>	8	Tingkat Gula
<i>BMI</i>	9	Indeks Massa Tubuh
<i>Smoking Status</i>	10	Status Perokok
<i>Stroke</i>	11	Diagnosa Strok

Tabel 3 Tabel Data *Training*

1	2	3	4	5	6	7	8	9	10	11
Male	55	0	0	Yes	Private	Urban	89.17.00	31.05.00	Never Smoked	0
Female	42	0	0	No	Private	Urban	98.53.00	18.05	Never Smoked	0
Female	24	0	0	No	Private	Urban	97.55.00	26.02.00	Never Smoked	0
Female	33	0	0	Yes	Private	Rural	86.97	42.02.00	Never Smoked	0
Female	20	0	0	No	Private	Rural	84.07.00	27.06.00	Smokes	0

1) *Preprocessing Data*

Data yang tersedia masih perlu melalui tahap *preprocessing* agar data menjadi konsisten dan mudah untuk dibaca oleh bahasa pemrograman Python. Pengolahan data *preprocessing* dilakukan untuk mengubah inkonsisten karakter data dari Huruf menjadi Numerik. Kolom data yang diubah kedalam tahap *preprocessing* yaitu *Residence*, *Average Glucose Level*, *Gender*, *Work Type*, *Ever Married*, *Smoking Status* dan *BMI*. Data penyakit strok sebelum dilakukan *preprocessing* dapat dilihat pada tabel 4.

Tabel 4 Tabel data sebelum *preprocessing*

1	2	3	4	5	6	7	8	9	10	11
Male	55	0	0	Yes	Private	Urban	89.17.00	31.05.00	Never Smoked	0
Female	42	0	0	No	Private	Urban	98.53.00	18.05	Never Smoked	0
Female	24	0	0	No	Private	Urban	97.55.00	26.02.00	Never Smoked	0
Female	33	0	0	Yes	Private	Rural	86.97	42.02.00	Never Smoked	0
Female	20	0	0	No	Private	Rural	84.07.00	27.06.00	Smokes	0

Keterangan hasil *preprocessing* data pada prediksi penyakit strok dijelaskan pada tabel 5.

Tabel 5 Tabel Hasil Tahapan *Preprocessing Data*

1	2	3	4	5	6	7	8	9	10	11
1	55	0	0	1	2	1	2439	154	1	0
0	42	0	0	0	2	1	2783	27	1	0
0	24	0	0	0	2	1	2754	101	1	0
0	33	0	0	1	2	0	2302	260	1	0
0	20	0	0	1	2	0	2231	115	2	0

Kemudian, penjelasan hasil *preprocessing* dapat dilihat pada tabel 6.

Tabel 6 Tabel Keterangan *preprocessing*

<b>Gender (1)</b>	= Male (1), Female (0)
<b>Hypertension (3), Heart Disease (4), Ever Married (5), Stroke (11)</b>	= Yes (1), No (0)
<b>Work Type (6)</b>	= Government (0), Self-Employed (1), Private (2)
<b>Residence Type (7)</b>	= Urban (1), Rural (0)

<b>Smoking Status (10)</b>	= Never Smoked (1), Smokes (2), Formerly Smoked (0)
----------------------------	---

B. Hasil Perancangan Model Algoritma SVM dengan Relief-f

Algoritma SVM dengan Relief-f bertujuan untuk mengetahui tingkat akurasi yang diperoleh dari perancangan model algoritma SVM dengan Relief-f. Hasil akurasi menghasilkan presentase sebesar 100%. Penentuan model Relief-f menggunakan atribut yang berpengaruh dalam data penyakit stroke. Cara kerja Relief-f mengurangi beberapa data yang tidak diperlukan dan hanya mengambil data yang memiliki peringkat tertinggi. Fungsi dari Algoritma Relief-f ini untuk menguji Algoritma dengan memakai dataset penting dan menguji tingkat akurasinya.

Cara untuk mendapatkan ranked attributes menggunakan algoritma Relief-f yang dimulai dengan:

**Input:**  $M$  learning instances  $x_k$  described by  $N$  features; sampling parameter  $m$

**Output:** for each feature  $F_i$  a quality weight  $-1 \leq W[i] \leq 1$

for  $i=1$  to  $N$  do  $W[i] = 0.0$ ;

end for;

for  $l = 1$  to  $m$  do

randomly pick an instance  $x_k$ ; find its nearest hit  $x_h$  and nearest miss  $x_m$ ;

for  $i=1$  to  $N$  do

$$W[i] = \frac{W[i] - \text{diff}(i, x_k, x_h)}{m} + \frac{\text{diff}(i, x_k, x_m)}{m};$$

end for;

end for;

return(W);

Hasil analisa data terhadap atribut peringkat dataset dijelaskan pada tabel 7.

Tabel 7 Tabel Peringkat Dataset

Ranked Attributes	Selected Attributes	Variable
0.02892	2	Age
0.02194	4	Heart Disease
0.0192	5	Ever Married
0.01614	3	Hypertension
0.01131	10	Smoking Status
0.0088	1	Gender
0.00796	7	Residence Type
0.0073	6	Work Type
0.00385	9	BMI
0.00153	8	Average Glucose Level

Dari tabel yang sudah ditentukan, data lima teratas dijadikan acuan penelitian untuk proses selanjutnya seperti pada tabel 8.

Tabel 8 Tabel Data Relief-f

No	Age	Heart Disease	Ever Married	Hypertension	Smoking Status
1	55	0	1	0	1
2	42	0	0	0	1
3	24	0	0	0	1
4	33	0	1	0	1
5	20	0	1	0	2
...	...	...	...	...	...
3421	46	0	1	0	0
3422	52	0	1	0	1
3423	47	0	1	0	0
3424	57	0	1	1	2
3425	45	0	1	0	0

C. Pengujian

Implementasi Algoritma Relief-f dengan metode Rasio dan Confusion Matrix menggunakan 3.426 Baris dan lima kolom variabel. Data training yang digunakan berjumlah 2.398 dan data testing berjumlah 1.028, menghasilkan akurasi data sebesar 100%. Hasil akurasi Algoritma Relief-f ditunjukkan pada gambar 4.

```

Confusion Matrix
[[976  0]
 [ 0  52]]
Test Set: 1028
Accuracy = 100.0 %

```

Gambar 4 Gambar Akurasi Data *Relief-f*

#### IV. KESIMPULAN

Hasil yang disimpulkan pada penelitian prediksi penyakit Stroke menggunakan Algoritma SVM adalah model prediksi yang dibangun menggunakan algoritma SVM dengan *Relief-f*. Menggunakan 2.398 data *training* dan 1.028 data *testing* menghasilkan akurasi data sebesar 100%. Metode yang digunakan menggunakan *Rasio* dan *Confusion Matrix* dengan total jumlah data sebanyak 3426 baris dan lima kolom. Penerapan Algoritma SVM dengan *Relief-f* menggunakan Kernel Linear untuk mendapatkan hasil terbaik. Saran yang dapat diambil dari penelitian ini, agar hasil dapat dirancang menggunakan aplikasi berbasis *mobile*. Penggunaan aplikasi *mobile* dalam memprediksi penyakit stroke akan mempermudah pengguna untuk mengetahui diagnosa dini penyakit stroke.

#### PENGAKUAN

Naskah Ilmiah ini adalah sebagian dari penelitian Tugas Akhir milik Ulfa Amelia, dengan judul "Implementasi Algoritma *Support Vector Machine* (SVM) untuk Prediksi Penyakit *Stroke*" yang dibimbing oleh Indra Jamaludin dan Anis Fitri Nur Masruriyah.

#### DAFTAR PUSTAKA

- [1] N. Permatasari, "Perbandingan Stroke Non Hemoragik dengan Gangguan Motorik Pasien Memiliki Faktor Resiko Diabetes Melitus dan Hipertensi," *Jurnal Ilmiah Kesehatan Sandi Husada*, pp. 298-304, 2020.
- [2] P. A. W. Suwaryo, W. T. Widodo and E. Setianingsih, "Faktor Risiko yang Mempengaruhi Kejadian Stroke," *Jurnal Keperawatan*, pp. 251-260, 2019.
- [3] B. Tamam, "Faktor Risiko Terhadap Kejadian Stroke di RSUD Dr. Koesnadi Bondowoso," pp. 1-20, 2020.
- [4] M. Lestari and S. , "Sistem Pakar Mendiagnosa Penyakit Stroke Transient Ischaemic Attack (TIA) Dengan Menggunakan Metode Dempster Shafer," *Jurnal Sistem Informasi dan Teknologi Jaringan*, pp. 25-30, 2021.
- [5] R. Rachman, "Implementasi Case Based Reasoning Mendiagnosa Penyakit Stroke Menggunakan Algoritma Probabilistic Symmetric," *Jurnal Informatika*, pp. 10-16, 2021.
- [6] A. Puspitawuri, E. Santoso and C. Dewi, "Diagnosis Tingkat Risiko Penyakit Stroke Menggunakan Metode K-Nearest Neighbor dan Naïve Bayes," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, pp. 3319-3324, 2019.
- [7] A. Byna and M. Basit, "Penerapan Metode Adaboost Untuk Mengoptimasi Prediksi Penyakit Stroke Dengan Algoritma Naïve Bayes," *Jurnal SISFOKOM*, vol. 09, pp. 407-411, 2020.
- [8] T. Djatna, M. K. D. Hardhienata and A. F. N. Masruriyah, "An intuitionistic fuzzy diagnosis analytics for stroke disease," *Big Data*, pp. 1-14, 2018.
- [9] A. F. N. Masruriyah, T. Djatna , M. K. D. Hardhienata, D. Wahiddin and H. H. Handayani, "Predictive Analytics For Stroke Disease," *Big Data*, 2020.
- [10] R. S. Hutama, N. Hidayat and E. Santoso, "Sistem Pakar Deteksi Dini Penyakit Stroke Menggunakan Metode Naive Bayes-Certainly Factor," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, pp. 4333-4339, 2018.
- [11] S. Atmaja, "Sistem Pendukung Keputusan untuk Deteksi Dini Resiko Penyakit Stroke menggunakan Learning Vektor Quantization," *INDEXIA*, pp. 29-35, 2019.
- [12] G. "Data Mining di Bidang Pendidikan Anak untuk Analisa Prediksi Kinerja Mahasiswa dengan Komparasi 2 Model Klasifikasi pada STMIK Jabar," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, pp. 23-30, 2019.
- [13] J. D. Z. Abidin, S. Nurmaini and R. F. Malik, "Penerapan Metode K-Nearest Neighbor dalam Memprediksi Masa Studi Mahasiswa (Studi Kasus : Mahasiswa STIKOM Dinamika Bangsa)," *Computer Science and ICT*, pp. 133-138, 2017.
- [14] R. N. Yusra, O. S. Sitompul and S. , "Kombinasi K-Nearest Neighbor (KNN) dan Relief-F untuk Meningkatkan Akurasi pada Klasifikasi Data," *InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan*, pp. 16-21, 2021.
- [15] H. Sammir, "Metode Feature Selection pada Naive Bayes menggunakan Struktur Data Redis," *Jurnal J-Click*, pp. 28-34, 2019.
- [16] B. Sugara and A. Subekti, "Penerapan Support Vector machine (SVM) pada Small Dataset untuk Deteksi Dini Gangguan Autisme," *Jurnal PILAR Nusa Mandiri*, pp. 177-182, 2019.
- [17] E. H. Harahap, L. Muflikah and B. Rahayudi, "Implementasi Algoritma Support Vector Machine (SVM) untuk Penentuan Seleksi Atlet Pencak Silat," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, pp. 3843-3848, 2018.
- [18] K. and S. Susanti, "Klasifikasi Kelayakan Peserta Pengajuan Kredit Rumah dengan Algoritma Naïve Bayes di Perumahan Azzura Residencia," *Jurnal Teknologi Pelita Bangsa*, pp. 43-48, 2019.
- [19] B. P. Pratiwi, A. S. Handayani and S. , "Pengukuran Kinerja Sistem Kualitas Udara Dengan Teknologi WSN Menggunakan Confusion Matrix," *JURNAL INFORMATIKA UPGRIS*, pp. 66-75, 2020.