KOMPARASI PENERAPAN METODE *BAGGING* DAN *ADABOOST* PADA ALGORITMA C4.5 UNTUK PREDIKSI PENYAKIT STROKE

SKRIPSI

UIN SUNAN AMPEL
S U R A B A Y A

DISUSUN OLEH :

NUR DIANA SAPUTRI

H76217043

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN AMPEL
SURABAYA

2021

PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini, Nama

: Nur Diana Saputri

NIM : H76217043

Program Studi : Sistem Informasi

Angkatan 2017

Menyatakan bahwa saya tidak melakukan plagiat dalam penulisan skripsi saya yang berjudul : "KOMPARASI PENERAPAN METODE BAGGING DAN ADABOOST PADA ALGORITMA C4.5 UNTUK PREDIKSI PENYAKIT STROKE". Apabila suatu saat nanti terbukti saya melakukan tindakan plagiat, maka saya bersediamenerima sanksi yang telah ditetapkan.

Demikian pernyataan keaslian ini saya buat dengan sebenar - benarnya.

Surabaya, 21 Oktober 2021

Yang Menyatakan,

(Nur Diana Saputri)
NIM : H76217043

LEMBAR PERSETUJUAN PEMBIMBING

Skripsi oleh

NAMA : NUR DIANA SAPUTRI

NIM : H76217043

JUDUL : KOMPARASI PENERAPAN METODE *BAGGING* DAN *ADABOOST* PADA ALGORITMA C4.5 UNTUK KLASIFIKASI PENYAKIT STROKE

Ini telah diperiksa dan disetujui untuk diujikan

Surabaya, 26 Agustus 2021

Menyetujui,

Dosen Pembimbing I	Dosen Pembimbing II
Dwi Rolliawati, M.T	Khalid, M.Kom
NIP. 197909272014032001	NIP. 197906092014031002

PENGESAHAN TIM PENGUJI SKRIPSI

Skripsi Nur Diana Saputri ini telah dipertahankan
di depan tim penguji skripsi
di Surabaya, 6 September 2021

Mengesahkan,
Dewan Penguji

Penguji I

Dwi Rolliawati, M.T
NIP. 197909272014032001

Penguji II

Khalid, M.Kom
NIP. 197906092014031002

Penguji III

Nurrissaidah Ulinnuha, M.Kom
NIP. 199011022014032004

Penguji IV

Mujib Ridwan, S.Kom.,M.T
NIP. 198604272014031004

Mengetahui,

Dekan Fakultas Sains dan Teknologi
UIN Sunan Ampel Surabaya

Prof. Dr. Evi Fatimatur Rusydiyah, M. Ag
NIP. 197312272005012003

KEMENTERIAN AGAMA
UNIVERSITAS ISLAM NEGERI SUNAN AMPEL SURABAYA
PERPUSTAKAAN
Jl. Jend. A. Yani 117 Surabaya 60237 Telp. 031-8431972 Fax.031-8413300
E-Mail: perpus@uinsby.ac.id

LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika UIN Sunan Ampel Surabaya, yang bertanda tangan di bawah ini, saya:

Nama : NUR DIANA SAPUTRI
NIM : H76217043
Fakultas/Jurusan : SAINS DAN TEKNOLOGI / SISTEM INFORMASI
E-mail address : nurdianasaputrii@gmail.com

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Perpustakaan UIN Sunan Ampel Surabaya, Hak Bebas Royalti Non-Eksklusif atas karya ilmiah :
■ Sekripsi ☐ Tesis ☐ Desertasi ☐ Lain-lain (……)
yang berjudul :
KOMPARASI PENERAPAN METODE BAGGING DAN ADABOOST
PADA ALGORITMA C4.5 UNTUK PREDIKSI PENYAKIT STROKE

beserta perangkat yang diperlukan (bila ada). Dengan Hak Bebas Royalti Non-Ekslusif ini Perpustakaan UIN Sunan Ampel Surabaya berhak menyimpan, mengalih-media/format-kan, mengelolanya dalam bentuk pangkalan data (database), mendistribusikannya, dan menampilkan/mempublikasikannya di Internet atau media lain secara ***fulltext*** untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan atau penerbit yang bersangkutan.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Perpustakaan UIN Sunan Ampel Surabaya, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini yang saya buat dengan sebenarnya.

Surabaya, 21 Oktober 2021

Penulis

(NUR DIANA SAPUTRI)

ABSTRAK

KOMPARASI PENERAPAN METODE *BAGGING* DAN *ADABOOST* PADA ALGORITMA C4.5 UNTUK PREDIKSI PENYAKIT STROKE

Oleh :
Nur Diana Saputri

Penyakit stroke merupakan gangguan fungsional otak yang disebabkan oleh tersumbatnya aliran peredaran darah. Secara global, stroke menjadi penyebab dari kematian dan kecacatan karena keterbatasan sumber daya dan ahli saraf. Oleh karena itu, diperlukan sistem klasifikasi data mining agar dapat membantu ahli saraf dalam menangani penyakit stroke. Penelitian ini bertujuan untuk memprediksi stroke agar dapat mengurangi risiko kematian dengan cara menerapkan metode klasifikasi data mining. Data yang digunakan untuk penelitian ini merupakan dataset stroke disease yang akan dianalisa menggunakan algoritma C4.5 dengan menerapkan metode bagging dan adaboost dari *Ensemble Learning* untuk memprediksi stroke. Hasil dari penelitian ini menunjukkan bahwa prediksi stroke menggunakan algoritma C4.5 mendapatkan nilai akurasi sebesar 92.87%. Sedangkan setelah dikombinasikan dengan metode bagging nilai akurasinya meningkat menjadi 95.02% dan setelah dikombinasikan dengan metode adaboost akurasinya meningkat menjadi 94.63%. Sehingga dapat disimpulkan bahwa penerapan metode bagging dan adaboost dapat meningkatkan kinerja dari algoritma C4.5. Namun peningkatan tersebut kurang signifikan karena hanya meningkat 3%.

Kata Kunci : Klasifikasi, Stroke Disease, Decision Tree, Bagging, Adaboost.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

ABSTRACT

COMPARATIVE APPLICATION OF BAGGING AND ADABOOST METHODS IN C4.5 ALGORITHM FOR STROKE DISEASE PREDICTION

Oleh :
Nur Diana Saputri

Stroke is a functional disorder of the brain caused by blockage of blood circulation. Globally, stroke is the leading cause of death and disability due to limited resources and neurologists. Therefore, a data mining classification system is needed in order to assist neurologists in dealing with stroke. This study aims to predict stroke in order to reduce the risk of death by applying data mining classification methods. The data used for this study is a stroke disease dataset which will be analyzed using the C4.5 algorithm by applying the bagging and adaboost methods from Ensemble Learning to predict stroke. The results of this study indicate that stroke prediction using the C4.5 algorithm gets an accuracy value of 92.87%. Meanwhile, after being combined with the bagging method, the accuracy value increased to 95.02% and after being combined with the adaboost method the accuracy increased to 94.63%. So it can be concluded that the application of bagging and adaboost methods can improve the performance of the C4.5 algorithm. However, the increase was less significant because it only increased by 3%.

Keywords: *Classification, Stroke Disease, Decision Tree, Bagging, Adaboost.*

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

DAFTAR ISI

ABSTRAK ..... **vi**

ABSTRACT ..... **vii**

DAFTAR ISI ..... **viii**

DAFTAR GAMBAR ..... **xi**

DAFTAR TABEL ..... **xii**

BAB 1 PENDAHULUAN **1**

1.1 Latar Belakang 1

1.2 Rumusan Masalah 4

1.3 Batasan Masalah 4

1.4 Tujuan Penelitian 4

1.5 Manfaat Penelitian 5

BAB 2 TINJAUAN PUSTAKA **6**

2.1 Tinjauan Penelitian Terdahulu 6

2.2 Teori Dasar yang Digunakan 9

2.2.1 Stroke 9

2.2.2 Faktor dan Gejala Stroke 10

2.2.3 Data Mining 11

2.2.4 Klasifikasi 13

2.2.5 Algoritma Decision Tree 14

2.2.6 Algoritma C4.5 14

2.2.8 Metode Bagging 16

2.2.9 Metode Adaboost 18

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

2.2.10 Kfold Cross-Validation.....19
2.2.11 Confusion Matrix.....20
2.2.12 Python.....22
2.3 Integrasi Keilmuan.....23
BAB 3 METODOLOGI PENELITIAN.....25
3.1 Kerangka Penelitian.....25
3.1.1 Perumusan Masalah.....25
3.1.2 Studi Literatur.....26
3.1.3 Pengumpulan Data.....26
3.1.4 Pengolahan Data.....27
3.1.5 Pelatihan Model.....27
3.1.6 Pengujian Model.....33
3.1.7 Analisis Hasil.....34
BAB 4 HASIL DAN PEMBAHASAN.....35
4.1 Pengumpulan Data.....35
4.2 Pengolahan Data.....36
4.2.1 Data Cleaning.....37
4.2.2 Data Transfromation.....39
4.3 Klasifikasi Algoritma C4.5.....40
4.4 Klasifikasi Algoritma C4.5 + Bagging.....41
4.5 Klasifikasi Algoritma C4.5 + AdaBoost.....42
4.6 Uji Validasi dan Evaluasi.....43
4.6.1 Algortima C4.5.....43
4.6.2 Algoritma C4.5 + Bagging.....45
4.6.3 Algoritma C4.5 + Adaboost.....46
4.7 Analisis Hasil.....48

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

4.8 Pembahasan.....51

BAB 5 PENUTUP.....53

5.1 Kesimpulan.....53

5.2 Saran.....53

DAFTAR PUSTAKA.....55

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

DAFTAR GAMBAR

Gambar 2. 1 Langkah - Langkah proses data mining..... 12
Gambar 2. 2 Struktur Pohon Keputusan..... 14
Gambar 2. 3 Penggabungan beberapa model dari Bagging..... 17
Gambar 2. 4 K-*fold Cross Validation*..... 20
Gambar 3. 1 Kerangka Penelitian.....25
Gambar 3. 2 *Flowchart* Proses Klasifikasi..... 28
Gambar 3. 3 *Flowchart* Klasifikasi C4.5..... 30
Gambar 3. 4 *Flowchart* Klasifikasi C4.5 + *Bagging*..... 31
Gambar 3. 5 *Flowchart* klasifikasi C4.5 + adaboost.....32
Gambar 4. 1 Persentase Pasien Stroke.....36
Gambar 4. 2 Hasil boxplot atribut bmi sebelum dilakukan IQR.....38
Gambar 4. 3 Hasil boxplot atribut bmi setelah dilakukan IQR.....38

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu..... 6
Tabel 2. 2 *Confusion Matrix* 21
Tabel 3. 1 Deskripsi Atribut *Dataset Stroke Disease*.....26
Tabel 3. 2 Skenario Klasifikasi.....27
Tabel 4. 1 Data Mentah.....35
Tabel 4. 2 Sampel dataset setelah dilakukan proses data cleaning..... 37
Tabel 4. 3 Sampel dataset setelah dilakukan tahapan transformation data..... 39
Tabel 4. 4 Detil atribut kategorikal yang diubah menjadi numerik.....39
Tabel 4. 5 *Confusion Matrix* dari Algoritma C4.5..... 43
Tabel 4. 6 Hasil pengujian nilai akurasi menggunakan 10*fold cross-validation*... 44
Tabel 4. 7 *Confusion Matrix* dari Metode Bagging dan Algoritma C4.5.....45
Tabel 4. 8 Hasil pengujian nilai akurasi menggunakan 10*fold cross-validation*... 46
Tabel 4. 9 *Confusion Matrix* dari algoritma C4.5 + adaboost.....46
Tabel 4. 10 Hasil pengujian nilai akurasi menggunakan 10fold cross-validation. 47
Tabel 4. 11 Hasil pengujian skenario pertama..... 48
Tabel 4. 12 Hasil pengujian skenario kedua..... 49
Tabel 4. 13 Hasil pengujian skenario ketiga..... 49
Tabel 4. 14 Perbandingan hasil pengujian tiap skenario.....50

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Stroke adalah penyakit yang merusak bagian penting dari otak karena disebabkan oleh penyumbatan aliran darah yang berlangsung selama lebih dari 24jam. Penyakit ini tergolong ke dalam *cerebrovascular disease* (CVD) karena bersifat mendadak dan membutuhkan penanganan yang sangat cepat (Wardhani and Martini 2011). Secara global, penyakit stroke menjadi penyebab dari kematian dan kecacatan. Semua negara masih berjuang menghadapi masalah penyakit stroke dengan keterbatasan sumber daya dan ahli saraf stroke (Suwanwela, Poungvarin, and ASAP 2016). Dengan mempertimbangkan sistem klasifikasi data mining untuk melakukan prediksi stroke sangat bermanfaat bagi ahli saraf stroke dalam penanganan penyakit stroke. Beberapa penelitian yang dilakukan di masa lalu membuktikan bahwa sistem klasifikasi dapat menangani masalah di bidang kesehatan seperti penyakit stroke dengan cara melakukan penggabungan metode atau menggunakan metode klasifikasi tunggal (Nugraha and Dodu 2017). Berdasarkan pernyataan tersebut, penelitian ini mengimplementasikan metode klasifikasi data mining untuk melakukan prediksi stroke agar dapat mengurangi peningkatan kasus stroke baru dan mengurangi angka kematian yang tinggi.

Data mining menjadi hal yang sangat penting dalam melakukan prediksi dan klasifikasi terhadap penyakit. Beberapa penelitian yang telah menerapkan teknik data mining dan metode klasifikasi untuk memprediksi penyakit. Metode klasifikasi yang umum digunakan adalah SVM (*Support Vector Machine), Naïve Bayesian,* Pohon Keputusan*,* Jaringan Syaraf Tiruan*, k-NN* (Pristyanto 2019). Pemilihan metode klasifikasi yang tepat untuk prediksi stroke sangat berpengaruh pada hasil penelitian dengan memperhatikan karakteristik dan ciri – ciri data yang akan digunakan (Puspitawuri, Santoso, and Dewi 2019). Beberapa metode klasifikasi yang terkait dengan prediksi stroke yaitu *Fuzzy Tsukamoto, (*SVM)

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Support Vector Machine, *k-NN, Naïve Bayesian, Neural Network dan Decision Tree* (Adelina, Ratnawati, and Fauzi 2018), (S and Kumar 2016), (Tunjung 2016), (Rohmana and Arifudin 2014), (Amini et al. 2013). Dari beberapa penelitian tersebut, metode *Decision Tree* memiliki akurasi tertinggi sebesar 95%. Sehingga metode *decision tree* digunakan pada penelitian ini karena memiliki akurasi yang tinggi. Selain itu d*ecision tree* juga memiliki kelebihan seperti dapat memecahkan masalah overfitting, menangani nilai atribut yang hilang pada data latih dan dapat meningkatkan efisiensi komputasi (Bisri 2015).

Meskipun metode *Decision Tree* lebih unggul dari metode lain akan tetapi metode ini tidak dapat memprediksi penyakit stroke dengan tepat (Kansadub and Thammaboosadee 2015). Prediksi yang dilakukan oleh Kansadub tidak sesuai dengan harapan karena dalam penelitiannya masih memiliki hasil negatif palsu (FN) yang tinggi, sehingga pasien yang seharusnya stroke ternyata diprediksi non-stroke. Untuk mengatasi masalah tersebut beberapa penelitian telah menerapkan teknik *ensemble* yang mampu memperbaiki ketidaktepatan nilai prediksi dan meningkatkan akurasi algoritma C4.5 yang merupakan algoritma evolusi dari metode Decision Tree untuk mendapatkan hasil prediksi yang akurat (Byna and Basit 2020). Dari penelitian tersebut, teknik *ensemble* menjadi tujuan utama penelitian ini karena dapat menyelesaikan masalah ketidaktepatan prediksi.

Kelebihan yang dimiliki oleh teknik *ensemble* yaitu dapat membantu meningkatkan kinerja algoritma klasifikasi dengan membuat beberapa pengklasifikasi dari data latih dan menggunakan *aggregating* pengklasifikasi pada saat melakukan klasifikasi serta dapat menangani ketidakseimbangan data yang hasilnya lebih baik dari *resampling* (Muslim, Alimah, and Prasetiyo 2017).. Secara umum, teknik e*nsemble* terkenal dengan dua metode populernya yaitu *bagging* dan *boosting.* Metode *Adaboost* atau *Adaptive Boosting* dari metode *boosting* dipilih untuk penelitian ini. Kedua metode tersebut bersifat sederhana dan efektif (Pristyanto 2019).

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Metode *Bagging* atau *bootstrap aggregating* termasuk metode yang berhasil diterapkan pada algoritma klasifikasi. Metode *bagging* sangat cocok untuk algoritma yang bersifat *unstable learning algorithms* karena jika data trainingnya berubah maka modelnya juga ikut dirubah (Muslim, Alimah, and Prasetiyo 2017). *Bagging* dapat membuat pengklasifikasi yang lebih baik jika memiliki pengamatan data latih yang mengandung *noise* (Saifudin and Wahono 2015). Metode ini mampu meningkatkan akurasi dari model klasifikasi seperti pada paparan riset – riset sebelumnya (Wahono and Suryana 2013), (Tu, Shin, and Shin 2009b), (Nai-arun and Sittidech 2016) menunjukkan bahwa tingkat akurasi yang dihasilkan hingga 80%. Sehingga metode *bagging* terbukti dapat memberikan kinerja yang baik.

Metode *Adaptive Boosting* atau *Adaboost* adalah variasi dari metode *boosting* yang memiliki hasil prediksi akurat dan teori dasar yang kuat. Selain itu metode *adaboost* dapat dilatih secara efektif (Byna and Basit 2020). Secara umum, metode *adaboost* memang lebih unggul dari metode *bagging* (Wahono and Suryana 2013). Metode ini dapat menetapkan bobot yang berbeda untuk data latih pada setiap iterasi dengan menambahkan dan mengurangi bobot pada pengklasifi kasi sehingga dapat mengubah distribusi dari data latih secara efektif (Bisri 2015). Dari paparan tersebut metode ini juga terbukti dapat meningkatkan akurasi seperti pada riset – riset sebelumnya (Do et al. 2016), (Lestari and Alamsyah 2020), (Jusia 2018) yang menunjukkan tingkat akurasi hingga 90%.

Berdasarkan permasalahan diatas, penelitian ini membahas mengenai prediksi stroke menggunakan model klasifikasi *decision tree* dan teknik *ensemble*. Penelitian ini diharapkan dapat mengatasi masalah ketidaktepatan prediksi dengan menerapkan metode *bagging* dan *adaboost* pada algoritma C4.5 untuk melihat apakah kedua metode tersebut dapat memprediksi secara akurat atau tidak.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

1.2 Rumusan Masalah

1. Bagaimana penerapan metode Bagging dan Adaboost pada algoritma C4.5 untuk prediksi penyakit stroke?
2. Bagaimana komparasi dari metode Bagging dan Adaboost pada algoritma C4.5 dalam meningkatkan performa prediksi penyakit stroke?

1.3 Batasan Masalah

Batasan masalah merupakan upaya untuk mempersempit ruang lingkup masalah agar tidak terlalu luas dan dapat lebih fokus pada penelitian. Berikut batasan masalah yang dicantumkan pada penelitian yaitu :

1. Dataset yang digunakan adalah *Dataset Stroke Disease* dari situs web Kaggle 2021 version 1.
2. Algoritma yang digunakan untuk penelitian ini adalah algoritma classifier.
3. Batas iterasi maksimal untuk bagging dan adaboost adalah 10
4. Penelitian ini membandingkan hasil pengujian dari metode bagging dan adaboost untuk mengetahui metode mana yang memberikan performa terbaik.

1.4 Tujuan Penelitian

Setelah menentukan rumusan masalah, adapun tujuan penelitian yang disusun untuk penelitian ini adalah :

1. Mengetahui penerapan metode Bagging dan Adaboost pada algoritma C4.5 untuk prediksi penyakit stroke.
2. Mengetahui komparasi metode Bagging dan Adaboost pada algoritma C4.5 dalam meningkatkan performa prediksi penyakit stroke.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

1.5 Manfaat Penelitian

1. Secara Akademis
   a. Menambah pengetahuan terhadap pembaca mengenai metode *Decision Tree,* metode *Bagging* dan metode *Adaboost.*
   b. Membantu para peneliti di masa depan sebagai bahan referensi
2. Secara Praktis

   Membantu ahli medis dalam memprediksi penyakit stroke agar tidak terjadi peningkatan kasus stroke yang semakin tinggi

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB 2
TINJAUAN PUSTAKA

2.1 Tinjauan Penelitian Terdahulu

Beberapa penelitian terdahulu sudah banyak yang menerapkan metode atau topik yang sama dengan penelitian ini. Dengan mempelajari penelitian terdahulu dapat membantu penelitian ini dalam mengelola dan menganalisis data. Sehingga penelitian ini dapat dijadikan perbandingan dengan penelitian sebelumnya. Merujuk pada Tabel 2.1 terdapat beberapa penelitian terdahulu yang sesuai dengan penelitian ini.

Tabel 2. 1 Penelitian Terdahulu

No	Judul Penelitian	Metode	Kesimpulan	Korelasi Penelitian
1.	Mendefinisikan Program Perawatan Rehabilitasi Pasien Stroke dengan Menerapkan Model Neural Network dan Decision Tree (Prasertsakul, Kaimuk, and Charoensuk 2015)	Algoritma *Decision Tree ,Neural Network*	Hasil penelitiannya menunjukkan bahwa algoritma DT dan NN berhasil menentukan program perawatan mana yang tepat, namun algoritma DT memiliki kinerja lebih unguul dari algoritma NN	Menggunakan algoritma Decision Tree untuk prediksi penyakit stroke
2.	Classification of Ischemic Stroke (Adam and Bashir 2016)	Algoritma kNN dan *Decision Tree*	Dari hasil model klasifikasi stroke iskemik, algoritma *Decision Tree* menghasilkan kinerja yang lebih baik dari algoritma kNN. Karena algoritma *Decision Tree*	

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

			dapat menentukan jenis stroke iskemik secara langsung.	
3.	Prediksi stroke menggunakan data mining (Amini et al. 2013)	Algoritma C4.5 , Algortima k-NN	Dalam penelitiannya algoritmaC4.5 dan kNN menghasilkan akurasi yang sama tingginya. Namun, algoritma C4.5 lebih unggul yaitu 95.42%, sedangkan algoritma k-NN akurasinya sebesar 94,18%	
4.	Meningkatkan akurasi Algoritma C4.5 Menggunakan Model Split Feature Redaction dan Bagging Ensemble untuk Prediksi Risiko Kartu Kredit (Muslim, Nurzahputra, and Prasetiyo 2018)	Algoritma C4.5 , *Split Feature Redaction* Model, dan Metode Bagging	Dengan menerapkan reduksi fitur split pada algoritma C4.5 didapatkan hasil terbaik dari split 3 dengan akurasi 73,1% . Kemudian hasil akurasi terbaik didapatkan oleh *Split Feature Redaction* dan metode bagging yang diterapkan kedalam algoritma C4.5 menghasilkan nilai sebesar 75.1% .	Menggunakan algoritma C4.5 dan metode Bagging untuk meningkatkan akurasi pada prediksi stroke
5.	Increasing Accuracy of C4.5 Algorithm Using Adaboost for Classification of Chronic Kidney Disease (Lestari and Alamsyah 2020)	Algoritma C4.5 , Information Gain Ratio , dan Metode Adaboost	Hasil klasifikasi dari mengaplikasikan Information Gain Ratio dan adaboost pada algoritma C4.5 menghasilkan akurasi yang sangat baik secara berturut – turut yaitu sebesar 96,66% , 97,5% dan 98,33%	Menggunakan algoritma C4.5 dan metode adaboost untuk meningkatkan akurasi pada prediksi stroke
6.	Penerapan	Algoritma	Kedua algoritma yang	

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

	Adaboost untuk menyelesaikan Ketidakseimbangan Kelas Pada Penentuan Kelulusan Mahasiswa menggunakan Metode *Decision Tree* (Bisri 2015)	Decision Tree, dan metode Adaboost	diusulkan terbukti mampu menangani masalah kelas yang tidak seimbang serta terbukti meningkatkan nilai akurasi sebesar 90,45%	
7.	Penerapan metode adaboost untuk mengoptimasi penyakit stroke (Byna and Basit 2020)	Algoritma *Naive Bayes, Adaboost*	Pengujian Algoritma *Naive Bayes* tanpa *Adaboost* menghasilkan *accuracy* sebesar 0,976 . Sedangkan pengujian dengan *Adaboost* menghasilkan *accuracy* 0,981. Metode Adaboost terbukti dapat meningkatkan dan mengoptimasi algoritma Naïve Bayes dengan menghasilkan akurasi terbaik dan memiliki margin error yang kecil.	Menggunakan metode adaboost untuk meningkatkan akurasi pada prediksi stroke
8.	Penerapan teknik Bagging pada algoritma C4.5 dalam mendiagnosis penyakit diabetes (Muslim, Alimah, and Prasetiyo 2017)	Algoritma C4.5, Metode Bagging, dan metode *Discretization*	Hasil akhir dari algoritma C4.5 yang dikombinasikan dengan metode *discretization* dan Bagging ketika mengatasi atribut yang bersifat *continuous* mendapatkan *akurasi* sebesar 74.87%	Menggunakan algoritma C4.5 dan metode Bagging untuk meningkatkan akurasi pada prediksi stroke

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

9.	Optimasi algoritma C4.5 menggunakan Seleksi Fitur PSO dan Bagging Pada Diagnosis Penyakit Kanker Payudara (Saputra 2020)	Algoritma C4.5 , PSO dan Teknik *Bagging*	Hasil penelitian menunjukkan bahwa seleksi fitur PSO dan Teknik Bagging berhasil meningkatkan akurasi sebesar 98,54% pada algoritma C4.5. Namun PSO kurang efektif dalam melakukan seleksi fitur di penelitian ini.	

2.2 Teori Dasar yang Digunakan

2.2.1 Stroke

Stroke adalah gangguan otak yang disebabkan oleh hipertensi. Stroke dapat menyerang kapan saja, ketika adanya pendarahan karena tekanan intrakranial yang meninggi serta embolus terlepas dari pembuluh non otak dan apabila terjadi peningkatan maka dapat menyebabkan hipertensi. Stroke juga dapat disebabkan oleh perubahan tekanan darah yang akut dan adanya penyumbatan sirkulasi darah sehingga fungsi otak dapat mudah terserang atau terjadi kematian sel saraf pada otak. Tubuh manusia memiliki sistem yang dapat mengelola darah secara baik agar tidak terjadi gangguan pada pembuluh darah. Jika manusia menerapkan pola hidup sehat maka dapat mempertahankan stabilitas tekanan darah dalam jangka panjang (Permatasari 2020). Secara garis besar stroke memiliki dua jenis yaitu *ischemic stroke* atau *Non hemorrhagic* (penyumbatan) dan *hemorrhagic stroke* (pendarahan). Kedua tipe stroke tersebut dapat mengakibatkan kelumpuhan organ tubuh secara menyeluruh atau sebagian, tidak dapat berbicara dan berjalan hingga dapat menyebabkan kematian (Arifianto, Sarosa, and Setyawati 2014).

Sekitar 80% orang dinyatakan terkena kasus *Ischemic stroke* karena jenis stroke ini paling sering terjadi. *Ischemic stroke* merupakan kelumpuhan pada jaringan otak yang dikarenakan pasokan darah tidak mencukupi (Tunjung 2016). Sedangkan *hemorrhagic stroke* terjadi ketika pembuluh darah yang ada di otak

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

mengalami kerusakan akibat tekanan darah tinggi (Arifianto, Sarosa, and Setyawati 2014). Kasus *Ischemic stroke* atau Stroke *Non Hemoragik* kebanyakan terjadi kepada penderita diabetes melitus, hipertensi, dan seseorang yang memiliki riwayat kedua penyakit tersebut (Permatasari 2020). Jenis stroke *non hemorrhagic* atau *ischemic stroke* dapat dikenali dengan empat faktor risiko tertinggi seperti usia yang lebih dari 50 tahun, meningkatnya kolesterol, gula darah dan hipertensi yang mempengaruhi tekanan pada perfusi otak (Handayani and Dominica 2018).

2.2.2 Faktor dan Gejala Stroke

Beberapa faktor dari kehidupan sehari hari yang dapat menyebabkan terjadinya penyakit stroke :

1. Merokok

   Kebiasaan merokok dapat membahayakan kesehatan tubuh karena dapat mengakibatkan penyumbatan pembuluh darah sehingga hal tersebut dapat mengganggu aliran darah. Sebagian besar merokok menjadi kebiasaan yang dilakukan oleh laki - laki daripada perempuan, sehingga peluang terserang stroke pada laki laki lebih tinggi (Ramadhini and Angliadi 2011).

2. Hipertensi

   Kerusakan pada otak karena tekanan intra kranial meningkat dan embolus yang terlepas dari pembuluh non otak dapat menyebabkan hipertensi jika terjadi peningkatan yang tidak terkompensasi. Penyakit hipertensi dapat dicegah dengan melakukan pola hidup sehat dengan cara melakukan diet, konsumsi makanan rendah lemak, konsumsi buah dan sayuran, serta olahraga yang teratur. Apabila tidak dapat mengendalikan hipertensi maka akan terjadi peluang lebih besar terserang stroke (Permatasari 2020).

3. Diabetes Mellitus

   Meningkatnya kadar gula darah secara drastis di dalam tubuh yang disebabkan oleh makanan yang kandungan gula darahnya berlebihan atau tidak dapat dikontrol. Perubahan struktur darah di dalam tubuh dapat mempengaruhi tekanan darah yang secara tidak langsung dapat

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

mengakibatkan hipertensi. Hal yang harus diperhatikan oleh penderita diabetes agar dapat dikendalikan adalah pola hidup yang sehat, olahraga secara teratur, melakukan diet sehat dan seimbang serta mengurangi konsumsi makanan yang memiliki kandungan kadar gula tinggi (Tarigan, Rini, and Samsuryadi 2020).

4. Usia

   Stroke dapat menyerang pada semua usia, untuk usia muda sekitar 15 tahun dan usia dewasa sekitar lebih dari 60 tahun. Berdasarkan data kesehatan di Indonesia pada tahun 2007 - 2011 menunjukkan bahwa usia produktif terjadi sekitar 15 - 64 tahun, usia tersebut sangat berpotensi terserang stroke (Wardhani and Martini 2011).

5. Jenis Kelamin

   Mayoritas penderita stroke berjenis kelamin pria. Karena seorang pria memiliki pola gaya hidup merokok yang dapat menyebabkan penyumbatan pembuluh darah pada tubuh (Handayani and Dominica 2018).

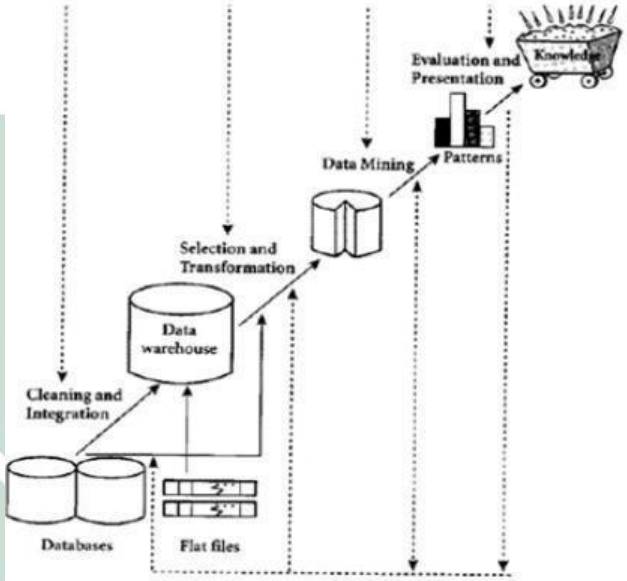
Selain faktor resiko pada penyakit stroke perlu diperhatikan juga gejala gejala yang terjadi ketika mengalami stroke seperti kesemutan tanpa sebab, sakit kepala, mendadak pelupa, sulit berbicara, dan lumpuh pada sebagian tubuh. Jika penderita stroke pada tahap awal mengalami kesembuhan dan kemudian hari mengalami serangan stroke kedua maka stroke tersebut dapat dikatakan berbahaya dan dapat mengakibatkan kematian. Pengobatan bagi penderita stroke harus dilakukan secara tepat dan cepat dimulai dengan pengenalan tanda – tanda gejala awal stroke yang terjadi. Hal tersebut dilakukan agar memberikan hasil yang optimal bagi penderita stroke pertama sehingga dapat mengurangi kenaikan kasus stroke yang berakhir kematian (Wardhani and Martini 2011).

2.2.3 Data Mining

Data mining telah menjadi ilmu yang sangat diminati oleh masyarakat di berbagai dunia, karena di dunia sistem informasi *data mining* berfungsi sebagai alat mengukur data yang memiliki jumlah besar dan dapat mengubah data menjadi

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

informasi yang berguna. *Data mining* juga dapat mengekstrak data menjadi pengetahuan. Hal ini biasa disebut dengan *knowledge discovery in database* (Septiani 2017). Di dalam *data mining* memiliki langkah - langkah penting yang harus dilakukan ketika proses mengukur data sebelum dijadikan sebagai pengetahuan. Berikut Gambar 2. yang menunjukkan urutan langkah - langkah proses data mining.



Gambar 2. 1 Langkah - Langkah proses data mining
(Abdillah 2015)

Berikut penjelasan dari langkah - langkah proses data mining (Carolina and Ade 2020) :

1. *Data Cleaning*. Proses pembersihan data agar tidak ada data *noise,* data yang tidak sesuai, dan data yang hilang di dalam dataset ketika dilakukan proses klasifikasi. Jika terdapat data yang memiliki atribut tidak konsisten maka harus dihilangkan agar tidak mengurangi nilai akurasi.
2. *Data Integration*. Mengidentifikasi data pada attribute - attribute untuk dilakukan integrasi data agar hasil yang diinginkan tidak menyimpang. Setelah melakukan integrasi data selesai, pembersihan dan transformasi data diulang kembali untuk menjaga data. Hal tersebut dilakukan agar data yang digunakan tetap konsisten.
3. *Data Selection*. Melakukan pemilihan data yang relevan agar tidak ada kesamaan data ketika mengolah data menggunakan metode data mining.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

4. *Data Transformation.* Melakukan konversi data menjadi data yang khusus sebelum dilanjutkan ke proses mining.
5. *Process Mining*. Tahap ini melakukan perhitungan menggunakan metode *data mining* populer yang digunakan untuk mengukur data yaitu *Estimation, Forecasting, Classification*, *Clustering*, dan A*ssociation*. Selain itu, beberapa metode *data mining* ini dapat dikelompokkan menjadi dua jenis metode pembelajaran yaitu *Supervised* dan *Unsupervised Learning*.
6. *Evaluation*. Melakukan identifikasi pola ke dalam *knowledge based*. Tahap ini merupakan evaluasi dari tahap *process mining*. Jika hasil olahan tidak sesuai, maka bisa menggunakan metode pembelajaran lain dalam *process mining*.
7. *Knowledge Presentation*. Teknik visualisasi dan representasi untuk menyajikan pengetahuan. Menjelaskan bagaimana hasil akhir keputusan atau analisis yang dihasilkan. Tahap ini merupakan akhir dari langkah - langkah data mining. Karena dalam mengolah data membutuhkan format data yang khusus agar dapat diproses.

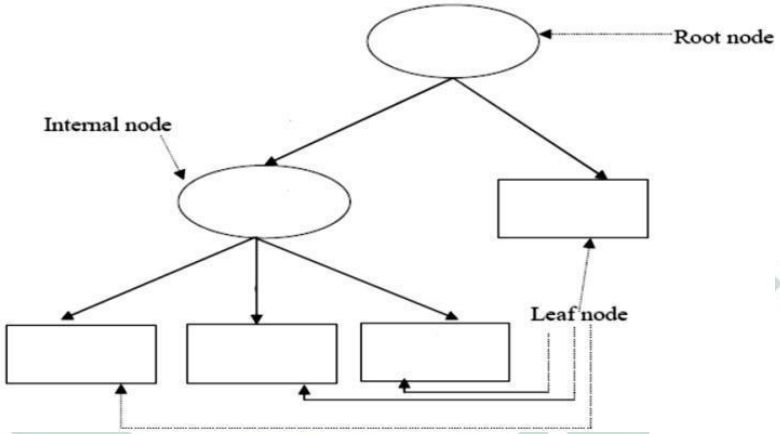
2.2.4 Klasifikasi

Klasifikasi adalah metode pengelompokan yang mudah dikenali jumlah dan nama kelompoknya (Setiawati, Wibowo, and Hermawan 2019). Metode klasifikasi merupakan teknik dari data mining yang sederhana dan populer sehingga telah banyak digunakan oleh masyarakat dunia. Di dalam metode klasifikasi terdapat dua tahap prosedur yaitu pengembangan model dan evaluasi model menggunakan pengujian. Beberapa algoritma dan teknik analisis dari metode klasifikasi yaitu Analisis Korelasi, Analisis Regresi, Model Bayesian, Algoritma berdasarkan jarak, Algoritma berdasarkan Pohon Keputusan (Decision Tree), dan Algoritma berdasarkan Jaringan Syaraf (Neural Network) (Jalota and Agrawal 2019).

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

2.2.5 Algoritma Decision Tree

Algoritma *Decision Tree* adalah suatu aturan yang menghasilkan pohon keputusan dan memiliki *node* yang dapat menunjukkan hasil yang disampaikan pada tiap *branch* dan kelas kelas yang ditunjukkan oleh *leaf node*. Algoritma *Decision Tree* dapat mencari data dan menemukan model yang dapat dijadikan sebuah target untuk dibagi menjadi kumpulan data dengan memperhatikan variabel tujuannya (Jusia 2018). Berikut Gambar 2.2 yang menunjukkan bentuk dari struktur dari pohon keputusan :



Gambar 2. 2 Struktur Pohon Keputusan
(Muzakir 2016)

Berdasarkan Gambar 2.2 diatas, Root node pada pohon keputusan merupakan akar dari pohon atau node yang diprioritaskan, sehingga tidak ada input tetapi memiliki output yang lebih dari satu. Setelah root node, terdapat internal node dan leaf node. Internal node merupakan node cabang yang memiliki satu input dan output yang lebih dari satu. Sedangkan leaf node adalah output yang dihasilkan oleh internal node atau root node jika tidak nilai yang diproses. .

2.2.6 Algoritma C4.5

Algoritma C4.5 merupakan evolusi algortima ID3 yang dapat menghasilkan cabang pohon. Proses tersebut adalah membuat aturan atau *rule* keputusan dari data menjadi pohon keputusan. Aturan tersebut berupa nilai dari atribut dan membentuk menjadi akar pohon hingga ke daun. Akar pohon merupakan simpul

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

atau node tanpa inputan dan simpul lain yang tidak termasuk akar pohon namun memiliki inputan disebut *node internal, s*edangkan daun sebagai simpul lain yang memiliki nilai sasaran yang benar dari satu kelas (Prasetio and Pratiwi 2015).

Tahapan dari algoritma C4.5 secara singkat dalam membentuk pohon keputusan yaitu (Muzakir 2016) :

1. Menentukan atribut sebagai akar
2. Membentuk cabang untuk tiap-tiap nilai
3. Membagi kasus untuk dimasukkan ke cabang
4. Mengulang proses yang sama pada setiap cabang hingga mendapatkan kelas yang sama

Langkah selanjutnya adalah membentuk pohon keputusan berdasarkan tahapan algoritma C4.5 sebagai berikut (Septiani 2017) :

1. Mempersiapkan atribut data yang sudah dikelompokkan sebelumnya ke dalam kelas tertentu untuk digunakan sebagai akar
2. Menghitung nilai gain atau nilai *index entropy*. Kemudian tentukan akar atau *node* pohon dengan nilai gain tertinggi dan nilai entropi terendah untuk setiap atribut. Hitung nilai entropi terlebih dahulu sebelum menghitung gain atribut. Rumus untuk nilai entropi dan nilai gain sebagai berikut :
   - Rumus nilai entropi :

$$\text{Entropy (S)} = \sum_{i=1}^{n} - pi\ log_2 pi \qquad (1)$$

Keterangan :

S = Himpunan Kasus

n = Jumlah Partisi pada *S*

pi = Proporsi dari Si terhadap S

- Rumus nilai *gain* :

$$\text{Gain (S,A)} = \text{entropy (S)} - \sum_{i=1}^{n} \frac{|Si|}{S} \text{ x Entropy (S)} \qquad (2)$$

Keterangan :

S = Himpunan Kasus

A = Atribut

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

n = Jumlah Partisi pada Atribut

$|si|$ = Jumlah kasus pada partisi ke i

$|s|$ = Jumlah kasus pada yang ada di dalam S

3. Mengulang – ulang proses tahap 2 sampai *record* dapat terisi secara merata.
4. Proses akan berhenti jika semua tupel *record* yang ada di dalam sampul *n* telah mendapatkan kelas yang sama

2.2.7 Teknik *Ensemble*

Teknik *ensemble* adalah strategi yang dapat meningkatkan akurasi pengklasifikasi. Meningkatkan akurasi dapat dilakukan dengan cara menggabungkan teknik *ensemble* dengan pengklasifikasi individu agar mendapatkan performa yang lebih baik (Latha and Jeeva 2019). Selain berfungsi sebagai meningkatkan akurasi, teknik *ensemble* juga dapat menangani *imbalance class* (Pristyanto 2019). Dua metode dari Teknik ensemble yang populer dan dapat membangun akurasi yang akurat adalah Bagging dan Boosting. Kedua metode tersebut menggunakan teknik resampling atau pengambilan data pelatihan secara acak untuk membuat set pelatihan yang berbeda pada setiap pengklasifikasi. (Tran et al. 2017a).

2.2.8 Metode Bagging

Bagging merupakan singkatan dari *bootstrap aggregating,* metode ini menjadi salah satu teknik ensemble yang paling sederhana dan efektif untuk diterapkan (Saifudin and Wahono 2015). Metode bagging dapat membantu algoritma klasifikasi yang tidak stabil seperti neural network dan decision tree sehingga perubahan kecil pada data pelatihan dapat menjadi perubahan yang besar dalam prediksi. Dengan demikian metode bagging selalu mencapai akurasi yang lebih baik daripada pengklasifikasian tunggal (Tran et al. 2017a). Pada gambar 2.3 merupakan struktur gabungan model dari bagging secara umum. Algoritma

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

bagging yang digunakan untuk hasil final klasifikasi menggunakan persamaan berikut (Bisri 2015):

Input :

Data Pelatihan

Algoritma Pengklasifikasi Dasar / *Base Learn* = L

Jumlah Iterasi = T

Proses :

Langkah pertama yaitu mengambil sampel pelatihan secara acak dari data pelatihan.

Kemudian latih *base learn*(L) h_t dari sampel pelatihan menggunakan distribusi D_t

$h_t = (L)D_t$

Keterangan :

h_t = Desain pengklasifikasi lemah

L = Algoritma pengklasifikasi lemah

D_t = Distribusi

t = Iterasi

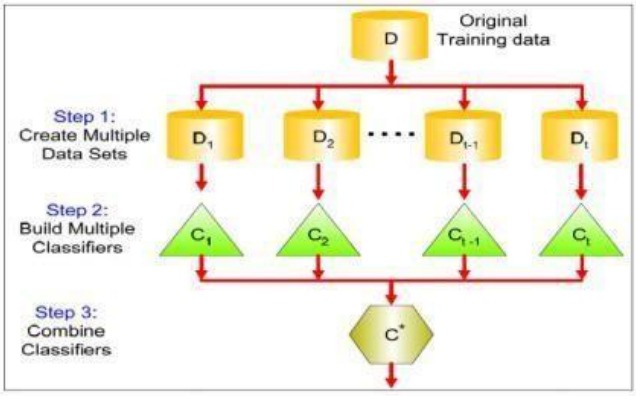
Selanjutnya mengambil nilai rata – rata dari semua output, setelah itu hitung output agregat.

Output hasil akhir menggunakan persamaan :

$$H(x) = \arg\max_y \sum_{t=1}^{T} 1\,(y = h_t(x)) \quad (3)$$

Keterangan :

H(x) = Final hipotesis



Gambar 2. 3 Penggabungan beberapa model dari Bagging
(Tu, Shin, and Shin 2009a)

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

2.2.9 Metode Adaboost

Metode Adaboost atau *Adaptive Boosting* adalah jenis algoritma *boosting* yang sering digunakan dalam klasifikasi. Peran *adaboost* adalah sebagai penyeimbang dan kombinasi algoritma *classifier* yang dapat meningkatkan performa dan meminimalkan *function error* pada klasifikasi (Jusia 2018). Namun, terkadang metode ini akan kurang akurat jika menangani data dengan noise pada klasifikasi yang substansial. Meskipun begitu metode adaboost dapat menunjukkan bahwa dengan sedikit atau tanpa noise klasifikasi bisa mencapai akurasi yang lebih baik (Tran et al. 2017b).

Kesalahan yang diukur dari distribusi D_t merupakan algoritma pembelajar lemah dan dapat menggunakan bobot D_t pada sampel pelatihan. Jika tidak memungkinkan maka sampel pelatihan dapat di resampling dan untuk hasil yang tidak berbobot akan dijadikan sebagai pelatihan algoritma pembelajar lemah (Saifudin and Wahono 2015). Algoritma *adaboost* untuk melakukan pembobotan pada setiap atribut menggunakan persamaan berikut (Bisri 2015):

Input :

Kumpulan Sampel Pelatihan D = $\{(x_1, y_1), \ldots, (x_m, y_m)\}$

Algoritma Pembelajar Dasar / Base Learn(L)

Jumlah Iterasi(T)

Proses :

Inisiasi nilai bobot pada sampel pelatihan dengan persamaan :

$$D_1(i) = \frac{1}{m} \text{ untuk } i = 1, \ldots, m \quad (4)$$

$$\text{Untuk } t = 1, 2, \ldots, T; \quad (5)$$

Keterangan :

t = Iterasi

T = Jumlah Iterasi

Latih *base learn* h_t dari sampel pelatihan menggunakan distribusi D_t

$h_t = L(D_t)$

Keterangan :

h_t = Desain pengklasifikasi lemah

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

L = Algoritma pengklasifikasi lemah

Dt = Distribusi

Hitung error sampel pelatihan menggunakan persamaan :

$€t = \sum Dt(i)$ (6)

Keterangan :

€ = Kesalahan bobot

Kemudian hitung nilai bobot sampel pelatihan menggunakan persamaan :

$\alpha t = \frac{1}{2} \text{In} \left(\frac{1 - €t}{€t}\right)$ (7)

Keterangan :

αt = Bobot hipotesis

Setelah itu update nilai bobot sampel pelatihan menggunakan persamaan :

$Dt+1(i) = \frac{Dt(i)}{Zt} x\{^{-\alpha t}$ atau $^{\alpha t}$ (8)

Keterangan :

$-\alpha t$ = Hasil klasifikasi benar

αt = Hasil klasifikasi salah

Zt = Faktor normalisasi yang mengubah Dt+1(i) menjadi distribusi

Output dari pengklasifikasi akhir :

$H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$ (9)

Keterangan :

H(x) = Final hipotesis

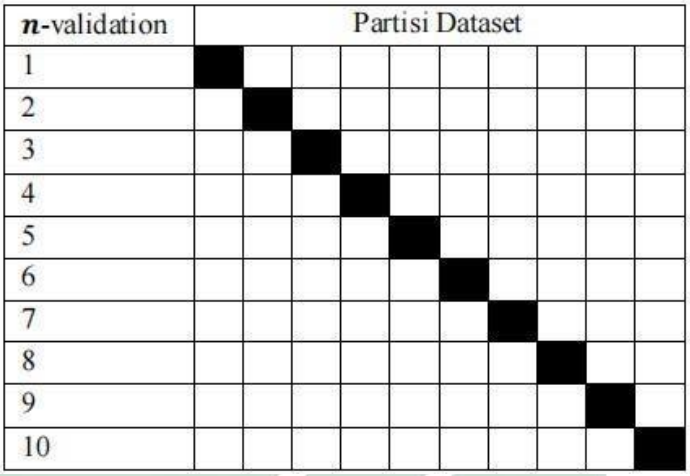
2.2.10 Kfold Cross-Validation

Kfold cross-validation adalah metode statistik analisi yang dapat menilai kinerja model yang digunakan dengan membagi data menjadi data latih dan data uji. Data latih dibentuk untuk melatih model klasifikasi dan data uji dibentuk untuk menguji model (Ramdhani 2016). *K-fold cross validation* melakukan k iterasi data, dimana k adalah jumlah sub yang akan dilatih. Untuk menentukan berapa nilai k yang digunakan, maka pilihlah k=5 atau k=10 dalam validasi silang k-fold. Pada umumnya nilai – nilai tersebut yang telah ditunjukkan secara empiris

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

menghasilkan estimasi tingkat kesalahan uji yang tidak terlalu tinggi (James et al. 2013). Oleh karena itu, dengan menentukan nilai k yang tepat dapat menghasilkan nilai pengujian yang terbaik.

Mekanisme dari validasi silang k-fold jika k=10, maka pengujian data yang akan diiterasi adalah sebanyak 10 kali dan posisinya dilakukan secara berurutan agar terdapat satu data yang tervalidasi. Untuk iterasi pertama maka pengujiannya berada di posisi pertama dan urutan lainnya merupakan data training. Lalu pada iterasi kedua berada di posisi kedua dan seterusnya hingga mencapai batas yang telah ditentukan. Pada Gambar 2.4 menunjukkan cara kerja kfold cross-validation. Setiap iterasi terdiri dari sembilan data pelatihan yaitu kotak yang berwarna putih dan satu data pengujian yaitu kotak yang berwarna hitam.



Gambar 2. 4 K-*fold Cross Validation*
(Issn and Ilham 2017)

2.2.11 Confusion Matrix

Confusion matrix di dalam data mining merupakan metode evaluasi yang menghitung nilai akurasi dari model klasifikasi. Metode ini memberikan hasil kinerja berdasarkan objek dengan benar atau salah yang diperoleh dari data pelatihan dan pengujian. Nilai akurasi pada confusion matrix memiliki dua kelas yang membentuk tabel matrix yaitu kelas positif dan kelas negative (Ramdhani 2016). Kedua kelas tersebut terdiri dari *True positive*, *False positive*, *True*

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Negative dan *False Negative*. Tabel 2.5 menunjukkan ilustrasi dari metode *confusion matrix.*

Tabel 2. 2 *Confusion Matrix* (Rais and Subekti 2019)

Confusion Matrix		**Class Prediksi**	
		Yes	No
Class Aktual	Yes	TP	FN
	No	FP	TN

1. TP (*True Positive*) atau Benar Positif

   Merupakan keadaan dimana data positif dari *class* aktual diprediksi benar. Contoh seorang pasien diprediksi terserang penyakit stroke
2. TN (*True Negative*) atau Positif Palsu

   Merupakan keadaan dimana data negative dari class aktual diprediksi benar. Contoh seorang pasien diprediksi tidak terserang penyakit stroke
3. FP (*False Positive*) atau Positif Salah

   Merupakan keadaan dimana data negative dari class aktual diprediksi sebagai data positif. Contoh seorang pasien tidak terserang penyakit stroke, tetapi diprediksi terserang penyakit stroke.
4. FN (*False Negative*) atau Negatif Salah

   Merupakan keadaan dimana data positif dari class aktual diprediksi sebagai data negatif. Contoh seorang pasien terserang penyakit stroke, tetapi diprediksi tidak terserang penyakit stroke.

Tingkat keberhasilan dari suatu model pada confusion matrix dapat dilihat dari nilai akurasi, sensivisitas, spesifisitas, presisi dan F1score. Semakin tinggi nilai akurasi, semakin baik metode yang digunakan. Penjelasan dari setiap nilai *confusion matrix* dan rumus yang digunakan untuk menghitung nilai tersebut dapat dilihat di bawah ini :

1. Akurasi adalah nilai evaluasi dari confusion matrix yang memiliki jumlah prediksi benar dari keseluruhan data. Karena tingkat keberhasilan dari akurasi dilihat dari hasil prediksi yang mendekati nilai aktual.

   $$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \quad (10)$$
2. Sensivisitas adalah nilai evaluasi dari confusion matrix yang mengukur suatu

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

kondisi seseorang yang memiliki suatu penyakit dengan benar. Sensivisitas merupakan nama lain dari Recall yang menghitung kelas positif.

$$\text{Sensivisitas} = \frac{TP}{TP + FN} \quad (11)$$

3. Spesifisitas adalah nilai evaluasi dari confusion matrix yang mengukur suatu kondisi seseorang tanpa penyakit dengan benar. Sensivisitas merupakan nama lain dari Recall yang menghitung kelas negatif.

$$\text{Spesifisitas} = \frac{TN}{TN + FP} \quad (12)$$

4. Presisi adalah nilai evaluasi dari confusion matrix yang memiliki rasio prediksi benar dari keseluruhan data yang diprediksi positif.

$$\text{Presisi} = \frac{TP}{TP + FP} \quad (13)$$

5. F1score adalah nilai evaluasi yang digunakan untuk mencari keseimbangan antara nilai presisi dan recall.

$$\text{F1Score} = 2 \text{ x } \frac{Presisi\ x\ Sensivisitas}{Presisi + Sensivisitas} \quad (14)$$

2.2.12 Python

Bahasa pemrograman Python merupakan lintas platform yang bersifat *open source* yang dapat diterapkan di semua jenis sistem operasi. Keunggulan dari Python ini tidak membutuhkan proses mendefinisikan variabel, sehingga penggunaanya lebih mudah, sangat fleksibel dan mudah untuk dipelajari. Selain itu, Python juga memiliki kelemahan, seperti variabel numerik yang berjalan di Python menjadi lebih lambat. Namun, Python tetap dapat bekerja dengan baik karena memiliki macam-macam fungsi yang sederhana sehingga dapat melakukan sesuatu yang kompleks. Saat ini Python telah memiliki banyak proses numerik dan matematis populer yang dapat diimpor ke Python seperti Numpy (*Numerical Python*) dan SciPy (*Scientific Python*) (Herho 2017).

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

2.3 Integrasi Keilmuan

Kesehatan merupakan sesuatu yang sangat disayangi semua orang. Dengan menjaga kesehatan dapat terhindar dari berbagai macam penyakit. Sudah banyak penyakit yang tidak bisa disembuhkan dan berujung pada kematian. Salah satunya adalah penyakit stroke. Penyakit ini sangat berbahaya bagi seseorang karena dapat menyerang secara mendadak pada otak dan dapat menyebabkan lumpuh pada sebagian tubuh. Berdasarkan hasil wawancara dengan ibu Luluk Ernawati,S.Ag., M.A. selaku guru Pendidikan Agama Islam di SMA Nahdlatul Ulama 1 Gresik, beliau mengatakan bahwa memang ada hubungan antara penelitian ini dengan ayat Al - Qur'an. Karena penyakit stroke berkaitan dengan hipertensi, diabetes dan jantung. Maka semua penyakit tersebut bermula dari pola makan dan pola pikir manusia yang kurang baik.

Pola makan setiap manusia berbeda – beda. Manusia sebagai umat Islam telah dianjurkan untuk selalu mengonsumsi makanan yang halal, bersih dan sehat. Tidak semua makanan dapat dikonsumsi oleh umat Islam seperti daging hewan yang mati secara tidak wajar, daging babi, daging hewan buas bertaring dan mimiliki kuku yang tajam. Semua makanan tersebut dianggap haram dan tidak layak untuk dikonsumsi manusia, karena pada dasarnya makanan yang baik adalah makanan halal yang sehat yang baik untuk tubuh manusia. Di dalam surat Al – Qur'an terdapat ayat yang membahas mengenai pola makan yang baik pada surah Al – Maidah ayat 88 yang berbunyi :

وَكُلُوا مِمَّا رَزَقَكُمُ اللَّهُ حَلَالًا طَيِّبًا ۚ وَاتَّقُوا اللَّهَ الَّذِي أَنْتُمْ بِهِ مُؤْمِنُونَ

Arti ayat 88 : Dan makanlah dari apa yang telah diberikan Allah kepadamu sebagai rezeki yang halal dan baik, dan bertakwalah kepada Allah yang kamu beriman kepadaNya.

Selain menjaga pola makan yang baik, manusia juga perlu menjaga pola pikir agar memiliki pikiran dan prasangka yang baik untuk diri sendiri, orang lain dan kepada Allah SWT. Islam mengajarkan manusia untuk selalu berbuat baik dan memperbanyak husnudzon bukan su'udzon kepada sesama umat Islam. Sikap

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

su'udzon sering datang ketika seseorang sedang merasa kesal dan tidak bisa menahan amarah karena mengalami cobaan atau ujian hidup yang berat. Beberapa orang berasumsi bahwa seseorang yang tidak bisa menahan amarahnya dapat mengalami tekanan darah tinggi yang berpengaruh buruk pada kesehatan tubuh manusia. Sehingga seseorang harus bisa menahan amarah dan tetap sabar dalam menghadapai cobaan dari Allah SWT. Di dalam Al Qur'an telah dijelaskan pada surat Ali Imran (3) Ayat 134, tentang pola pikir yang baik. Ayat tersebut berbunyi :

الَّذِينَ يُنْفِقُونَ فِي السَّرَّاءِ وَالضَّرَّاءِ وَالْكَاظِمِينَ الْغَيْظَ وَالْعَافِينَ عَنِ النَّاسِ
وَاللَّهُ يُحِبُّ الْمُحْسِنِينَ

Arti ayat 134 : (yaitu) orang-orang yang menafkahkan (hartanya), baik di waktu lapang maupun sempit, dan orang-orang yang menahan amarahnya dan memaafkan (kesalahan) orang. Allah menyukai orang-orang yang berbuat kebajikan.

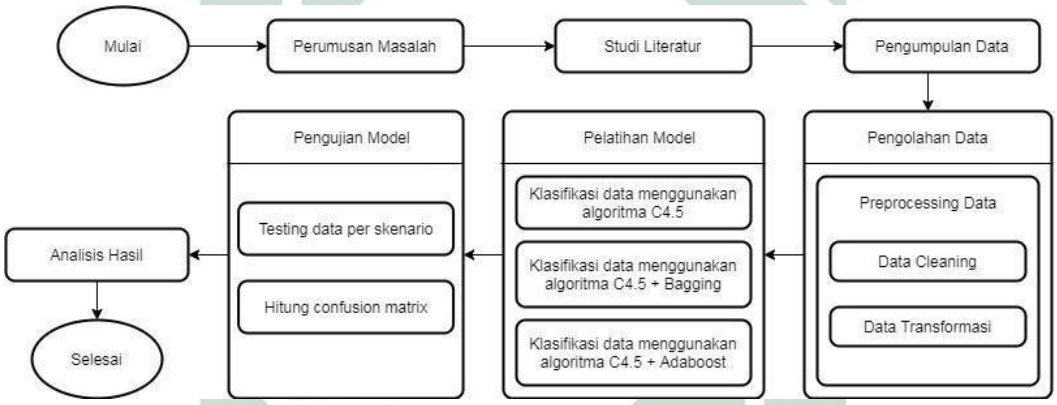
Dari penjelasan diatas pada ayat pertama yaitu surat Al - Maidah 5 : ayat 88 menjelaskan bahwa Allah SWT berpesan kepada setiap umatnya untuk selalu menjaga pola makan yang baik seperti makan makanan yang sehat dan halal serta menghindari makanan yang dilarang oleh Allah SWT. Sedangkan pada ayat kedua yaitu surat Ali - Imran 3 : ayat 134 menjelaskan bahwa umat Islam harus memiliki pola pikir atau mental yang baik dalam memperbanyak husnudzon pada diri sendiri atau orang lain. Sehingga pelajaran yang dapat diambil dari kedua surat tersebut yakni selalu bersyukur dan taat kepada Allah SWT, berprasangka baik dalam menghadapi cobaan serta saling memaafkan sesama umat manusia.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB 3
METODOLOGI PENELITIAN

3.1 Kerangka Penelitian

Penelitian dapat bekerja dengan baik jika langkah - langkah yang diperlukan harus konsisten dan mengikuti alur yang telah ditetapkan, tahapan penelitian ini dimulai dari perumusan masalah hingga pengujian model. Kerangka penelitian ini berfungsi sebagai pedoman untuk melakukan penelitian yang terstruktur dan dapat diselesaikan tepat pada waktunya. Gambar 3.1 berikut menunjukkan alur dari penelitian ini.



Gambar 3. 1 Kerangka Penelitian

3.1.1 Perumusan Masalah

Tahap ini merupakan tahap awal sebelum melakukan penelitian yaitu rumusan masalah. Dimana pada tahap ini melakukan pemahaman terhadap masalah penelitian yang telah dipaparkan di latar belakang penelitian mengenai prediksi penyakit stroke. Metode yang diterapkan untuk penelitian ini yaitu algoritma C4.5 dengan teknik *ensemble* yaitu metode bagging dan metode adaboost untuk prediksi penyakit stroke. Dengan menerapkan metode tersebut, dapat mengetahui apakah pasien mengalami stroke atau tidak.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

3.1.2 Studi Literatur

Tahap kedua yaitu mencari *paper* dari berbagai sumber yang relevan dengan penelitian untuk dijadikan bahan referensi. Sumber referensi yang digunakan bisa berupa buku, jurnal, dan e-book yang sesuai dengan topik penelitian atau metode dari penelitian. Tahapan studi literatur ini dilakukan guna mempelajari penelitian terdahulu yang berkaitan dengan stroke dan metode penelitian.

3.1.3 Pengumpulan Data

Dataset Stroke Disease merupakan catatan data medis pasien dari Madrid, Spanyol yang berjumlah 5110 data mentah. Dataset yang digunakan adalah dataset *public* yang bersumber dari *Kaggle* (Senthilkumar and Ramasubramanian 2020). Dataset ini memiliki 11 atribut dan satu atribut *label* yaitu stroke. Pada atribut stroke meliputi data pasien yang pernah dan belum pernah terkena stroke. Pasien yang tidak terkena stroke sebanyak 95.1% dan yang terkena stroke hanya 4.5%. Dari pernyataan tersebut, data ini termasuk ke dalam data yang tidak seimbang karena proporsi data pasien yang tidak terkena stroke lebih banyak daripada pasien yang terkena stroke. Tabel 3.1 di bawah ini menunjukkan deskripsi atribut dari *Dataset Stroke Disease* .

Tabel 3. 1 Deskripsi Atribut *Dataset Stroke Disease*

No	Atribut	Deskripsi	Tipe Data
1.	Id	Pengenal Unik	Numerik
2.	Gender	Perempuan, Laki - Laki	Kategorikal
3.	Age	Umur Pasien (Tahun)	Numerik
4.	Hypertension	0 : Tidak, 1 : Ya	Numerik
5.	Heart Disease	0 : Tidak, 1: Ya	Numerik
6.	Ever Married	Ya atau Tidak	Kategorikal
7.	Work_Type	Pekerjaan Pemerintah, Pribadi, Wiraswasta, Anak - anak, Tidak Bekerja	Kategorikal
8.	Residence	Pedesaan, Perkotaan	Kategorikal
9.	Avg_glucose_level	Rata - rata tingkat glukosa darah	Numerik

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

10.	BMI	Indeks Massa Tubuh	Numerik
11.	Smoking _Status	Sebelumnya merokok, Tidak pernah merokok,Merokok atau Tidak diketahui	Kategorikal
12.	Stroke	0 : Tidak, 1: Ya	Numerik

3.1.4 Pengolahan Data

Tahap ketiga yaitu pengolahan data. Tugas utama dari tahap ini yaitu melakukan proses preprocessing data untuk mengubah data mentah menjadi data yang dapat dengan mudah diolah menggunakan proses data mining. Beberapa data masih ada yang tidak konsisten seperti "N/A" dan terdapat data outlier serta tipe datanya juga memiliki perbedaan seperti numerik dan kategorik. Pada tipe data kategorik akan diubah menjadi angka yang dikodekan menjadi 0 atau 1 agar mudah melakukan pengklasifikasi. Berdasarakan data tersebut akan dilakukan pembersihan data atau data cleaning untuk menghapus data yang tidak sesuai dan transformasi data akan dilakukan untuk mengubah format pada tipe data yang berbeda.

3.1.5 Pelatihan Model

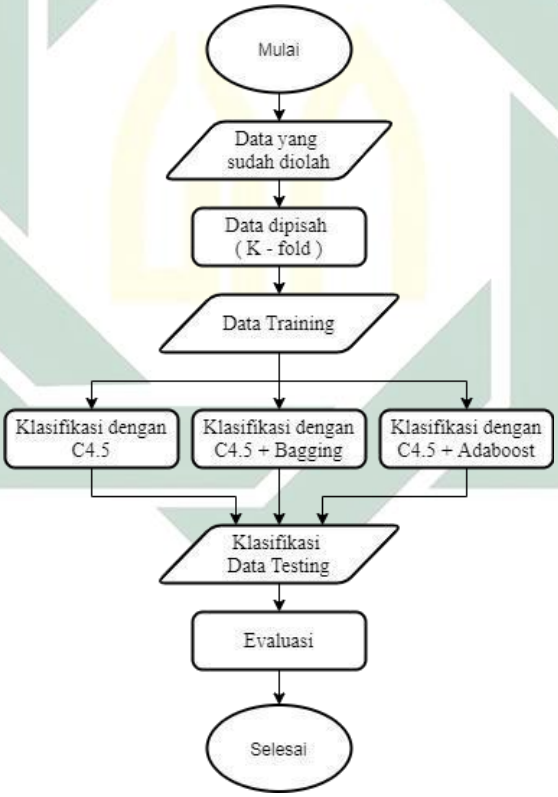
Setelah tahap preprocessing selesai, dilanjutkan dengan tahap pelatihan model. Tahap keempat ini akan melakukan klasifikasi menggunakan algoritma C4.5 dengan metode bagging dan metode adaboost. Hal ini dilakukan dengan membagi proses klasifikasi menjadi tiga skenario. Tabel 3.2 di bawah ini menunjukkan proses skenario klasifikasi.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Tabel 3. 2 Skenario Klasifikasi

Algoritma C4.5	**Algoritma C4.5 + Bagging**	**Algoritma C4.5 + Adaboost**
Pada skenario ini akan dilakukan klasifikasi menggunakan algoritma C4.5 saja tanpa menerapkan metode *Bagging* atau *Adaboost*	Pada skenario ini akan dilakukan klasifikasi menggunakan algoritma C4.5 dan metode bagging untuk mendapatkan hasil klasifikasi yang akurat	Pada skenario ini akan dilakukan klasifikasi menggunakan algoritma C4.5 dan metode adaboost untuk mendapatkan hasil klasifikasi yang akurat

Berdasarkan skenario sebelumnya, langkah -langkah dalam proses klasifikasi dapat diilustrasikan ke dalam flowchart seperti Gambar 3.2 di bawah ini.



Gambar 3. 2 *Flowchart* Proses Klasifikasi

Langkah – langkah *flowchart* akan dijelaskan di bawah ini :

1. Data yang sudah diolah di tahap preprocessing.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

2. Data tersebut dipisah menjadi 2 data, yaitu data latih dan data uji. Pemisah data ini memakai metode kfold cross-validation agar hasil pengujian menjadi lebih optimal.
3. Setelah tahap preprocessing akan dilakukan proses klasifikasi menggunakan algortima C4.5, klasifikasi menggunakan C4.5 + Bagging dan klasifikasi menggunakan C4.5 + Adaboost
4. Tahap pengujian klasifikasi menggunakan data testing
5. Tahap evaluasi untuk menguji proses klasifikasi dengan melakukan perhitungan berdasarkan persamaan 10, 11, 12 dan 13 untuk menghasilkan nilai akurasi, spesifisitas, sensivisitas dan presisi.

Pada proses klasifikasi terdapat tiga proses yang dilakukan yaitu klasifikasi menggunakan C4.5, klasifikasi menggunakan C4.5 + Bagging dan klasifikasi menggunakan C4.5 + Adaboost. Masing – masing proses klasifikasi tersebut dijelaskan dengan diagram *flowchart* seperti di bawah ini :

a. Klasifikasi algoritma C4.5

Proses klasifikasi pertama adalah algoritma C4.5, algoritma pengembangan dari metode pohon keputusan. Gambar 3.3 menunjukkan bentuk *flowchart* dari klasifikasi algortima C4.5 beserta langkah – langkahnya.

Langkah – langkah diagram klasifikasi algoritma *C4.5* yaitu :

1. Menyiapkan data latih yang sudah dilakukan tahap *preprocessing*
2. Menentukan atribut dari data latih untuk dijadikan sebagai kelas atau node
3. Setelah kelas ditentukan, menghitung nilai *entropy* dan nilai *gain* untuk setiap kelas dengan rumus dari persamaan 1 dan 2
4. Mencari nilai *gain* yang tertinggi dari setiap kelas dan setelah itu membuat pohon keputusan dari kelas yang sudah terpilih
5. Mengecek kelas apakah semua sudah terisi, jika masih belum terisi maka akan dilakukan perhitungan ulang pada nilai entropy dan nilai gain hingga semua pohon terisi
6. Setelah semua kelas terisi maka pohon keputusan dapat dibentuk

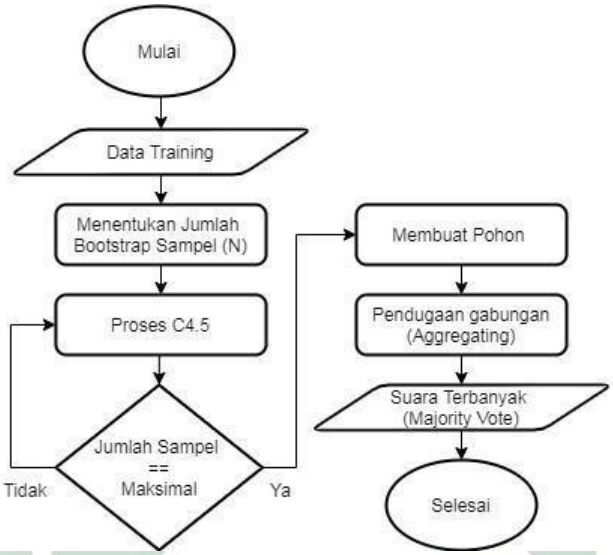
digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Gambar 3. 3 *Flowchart* Klasifikasi C4.5
(Muslim, Alimah, and Prasetiyo 2017)

b. Klasifikasi algoritma C4.5 + Bagging

Proses klasifikasi kedua yaitu algoritma C4.5 dan Bagging. Kombinasi dari algortima C4.5 dan metode Bagging untuk menghasilkan akurasi yang lebih baik. Gambar 3.4 menunjukkan bentuk *flowchart* dari klasifikasi algortima C4.5 + Bagging beserta langkah – langkahnya.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id



Gambar 3. 4 *Flowchart* Klasifikasi C4.5 + *Bagging*

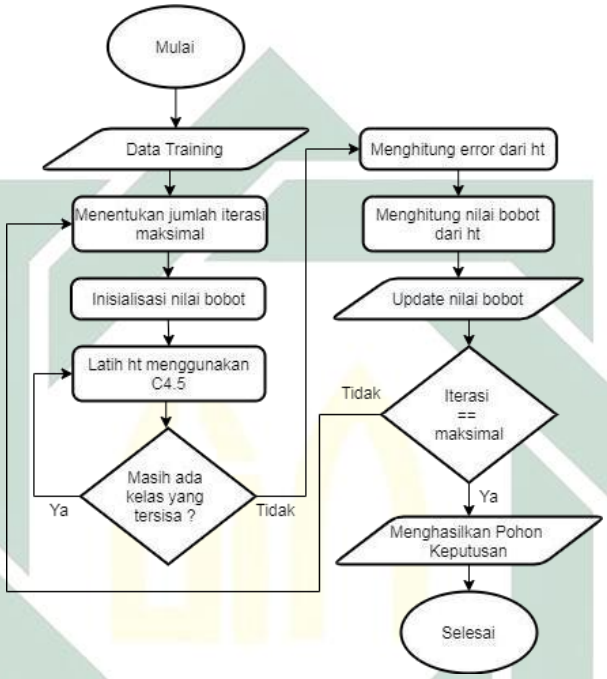
Langkah – langkah dari klasifikasi menggunakan algoritma *C4.5 + Adaboost* akan dijelaskan di bawah ini :

1. Menyiapkan data latih yang sudah dilakukan tahap *pre-processing*.
2. Mengambil sampel data dari data training secara acak sebanyak N sampel
3. Melakukan proses C4.5 yaitu diawali dengan menetukan atribut untuk dijadikan node lalu dilakukan perhitungan nilai *entropy* dan nilai *gain*. Node yang memiliki nilai gain tertinggi akan dipilih untuk diprioritaskan menjadi node pertama dalam membuat pohon keputusan.
4. Melakukan pengecekan dari hasil pohon keputusan apakah sudah memenuhi ukuran sample (N) atau tidak. Jika iya maka akan menghasilkan beberapa model pohon. Jika tidak maka akan kembali pada tahap 2. Pembentukan pohon didapat dari hasil sampel acak yang telah ditentukan.
5. Membuat pendugaan gabungan berdasarkan jumlah pohon yang terbentuk. Pendugaan gabungan adalah proses estimasi yang menggunakan banyak nilai estimasi menjadi satu estimasi untuk dianalisis.
6. Hasil pohon keputusan yang memiliki suara terbanyak akan dipilih berdasarkan nilai estimasi beberapa pohon menjadi satu estimasi akhir.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

c. Klasifikasi algoritma C4.5 + Adaboost

Proses klasifikasi ketiga yaitu algoritma C4.5 dan adaboost. Kombinasi algoritma C4.5 dan metode adaboost untuk menghasilkan akurasi yang lebih baik. Gambar 3.5 menunjukkan bentuk *flowchart* dari klasifikasi algortima C4.5 + Adaboost beserta langkah – langkahnya.



Gambar 3. 5 *Flowchart* klasifikasi C4.5 + adaboost

Langkah – langkah diagram klasifikasi menggunakan algoritma *C4.5* + adaboost akan dijelaskan di bawah ini :

1. Menyiapkan data training yang sudah dilakukan tahap *preprocessing*
2. Menentukan jumlah iterasi maksimal sebanyak N iterasi
3. Melakukan perhitungan inisialisasi bobot dengan menggunakan persamaan 4 dan 5
4. Menggunakan algoritma C4.5 sebagai pengklasifikasi lemah untuk melatih data dengan menetukan atribut untuk dijadikan node lalu dilakukan perhitungan nilai *entropy* dan nilai *gain*. Node yang memiliki nilai gain tertinggi akan dipilih untuk diprioritaskan menjadi node pertama dalam membuat pohon keputusan.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

5. Mengecek apakah terdapat kelas yang tersisa, jika iya maka akan mengulang ke tahap nomor 3 hingga tidak ada kelas yang tersisa. Namun jika tidak maka akan menuju pada tahap selanjutnya
6. Menghitung error digunakan untuk mendapatkan tingkat error minimum pada pengklasifikasian dengan menggunakan persamaan 7. Apabila semakin kecil kesalahannya, maka semakin tinggi bobotnya dan semakin banyak kesalaannya maka semakin sedikit bobotnya.
7. Menghitung nilai bobot dan update nilai bobot menggunakan persamaan 8 dan 9.
8. Mengecek jumlah iterasi apakah sudah maksimal. Jika iya maka proses perhitungan selesai, jika tidak maka akan kembali pada tahap keempat sampai iterasi mencapai jumlah maksimal.
9. Membentuk pohon keputusan akhir

3.1.6 Pengujian Model

Tahap ini merupakan tahapan terakhir dari penelitian yaitu pengujian model untuk tiga skenario klasifikasi, dimana data yang digunakan sudah memiliki label. Sebelum dilakukan pengujian model, data harus diolah terlebih dahulu agar hasil pengujiannya menjadi lebih maksimal. Pengolahan data dilakukan melalui tahap *preprocessing*. Selanjutnya data akan diproses dengan model atau algoritma klasifikasi untuk mengevaluasi kinerja model.

Metode untuk pengujian model menggunakan metode *confusion matrix*. Sebelum dilakukan pengujian, data dipisah terlebih dahulu menjadi data latih atau dan data uji. Data tersebut akan diuji oleh k*fold cross-validation* dengan mengambil 10 sampel data secara acak untuk mengevaluasi model. Dari beberapa sampel data tersebut akan dipilih model yang memiliki nilai optimal dengan cara mencari rata – rata dari 10 sampel tersebut untuk menghasilkan nilai akurasi, spesifisitas dan sensivisitas. Perhitungan nilai akurasi, spesifisitas, sensivisitas, presisi dan skor f1 dilakukan menggunakan persamaan 10, 11, 12, 13 dan 14.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

3.1.7 Analisis Hasil

Tahapan analisis membahas mengenai perbandingan hasil dari setiap skenario yang sudah dilakukan tahap pengujian sebelumnya. Tiga skenario tersebut terdiri dari algoritma C4.5, algoritma C4.5 + bagging, algoritma C4.5 + adaboost. Pengujian dari tiga skenario akan menghasilkan nilai presisi, spesifisitas, sensitifitas, skor f1 dan nilai akurasi. Dari nilai yang dihasilkan tersebut dapat dibandingkan dan diketahui metode mana yang memberikan pengaruh dalam meningkatnya performa klasifikasi. Selain itu tahap analisis juga membandingan hasil penelitian ini dengan penelitian lain yang menggunakan metode yang sama.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB 4
HASIL DAN PEMBAHASAN

Setelah membahas metode penelitian pada bab 3, selanjutnya yaitu bab 4 yang akan membahas mengenai tahap pengumpulan data, tahap pengolahan data, tahap pelatihan model dan tahap pengujian model yang sudah dijelaskan pada bab 3 sebelumnya. Tahapan pertama yaitu pengumpulan data. Tahapan kedua yaitu pengolahan data yang membahas mengenai proses *preprocessing* data. Tahapan ketiga yaitu pelatihan model, dimana pada tahap ini akan dilakukan pelatihan model untuk tiga skenario klasifikasi yang sudah ditentukan. Selanjutnya tahapan yang terakhir yaitu pengujian model, dimana pada tahap ini dilakukan testing dan evaluasi model untuk setiap skenario klasifikasi. Seluruh tahapan penelitian ini diproses menggunakan bahasa pemrograman Python dengan mengimplementasikan l*ibrary* yang digunakan yaitu *scikit learning* yang menyediakan berbagai fungsi matematika untuk membangun model *machine learning* dan menyediakan *pandas library* yang dapat mendeteksi file berupaExcel, CSV, dan lain - lain.

4.1 Pengumpulan Data

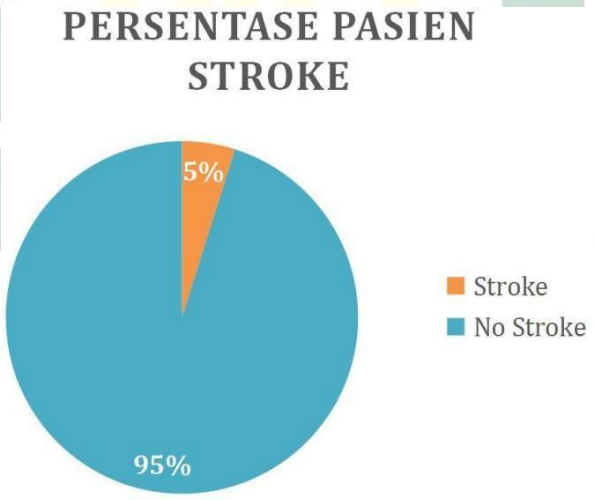
Data awal atau data mentah sebelum dilakukan proses *preprocessing* memiliki 5110 data. Dataset ini merupakan kumpulan data dari pasien stroke yang berasal dari Madrid, Spanyol. Dataset memiliki 12 atribut yang terdiri dari id, gender, age, hypertension, heart disease, ever married, work type, residence, avg glucose level, body massa index, smoking status dan stroke. Tabel 4.1 di bawah ini menunjukkan detil data mentah sebelum diolah.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Tabel 4. 1 Data Mentah

Id	Gender	Age	Hyper-tension	…	Avg glucose level	BMI	Smoking Status	Stroke
9046	Male	67.0	0	…	228.69	36.6	formerly smoked	1
51676	Female	61.0	0	…	202.21	N/A	never smoked	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
37544	Male	51.0	0	…	166.29	25.6	formerly smoked	0
44679	Female	44.0	0	…	85.28	26.2	Unknown	0

Berdasarkan Tabel 4.1 diatas, atribut stroke memiliki jumlah data pasien non stroke lebih banyak daripada pasien stroke. Jumlah data pasien hanya 249 dan jumlah pasien tidak stroke sebanyak 4861. Dapat disimpulkan bahwa atribut stroke merupakan kelas data yang tidak seimbang. Gambar 4.1 berikut menunjukkan ilustrasi perbandingan jumlah data stroke dan non stroke. Pasien yang tidak terkena stroke sebanyak 95% dan yang terkena stroke hanya 5%.



Gambar 4. 1 Persentase Pasien Stroke

4.2 Pengolahan Data

Berdasarkan dataset stroke disease yang memiliki jumlah data sebanyak 5110 data, didalamnya terdapat beberapa data yang tidak relevan. Sebelum melakukan tahap klasifikasi sebaiknya melakukan pengolahan data terlebih

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

dahulu, agar data yang digunakan menjadi valid dan proses klasifikasi dapat dilakukan dengan baik. Tahap *prepropcessing* data adalah tahap untuk melakukan pembersihan data dan transformasi data. Beberapa data pada atribut memiliki nilai yang kosong atau tidak konsisten, sehingga diperlukan pembersihan data agar terbebas dari data *noise*. Data *noise* adalah data yang rusak akibat adanya duplikat data, data tidak konsisten, data hilang, dan data yang tidak beraturan sehingga dapat mempengaruhi nilai akurasi pada proses klasifikasi. Sedangkan transformasi data yaitu mengubah format tipe data pada suatu atribut untuk memudahkan proses klasifikasi.

4.2.1 Data Cleaning

Data cleaning merupakan tahapan dimana pembersihan dan pemeriksaan data dilakukan pada data *noise* sebelum melakukan proses klasifikasi. Masalah pada data akan diperbaiki dengan menggunakan library python yaitu pandas yang berfokus pada analisis data seperti pembersihan data. Proses pembersihan data ini dapat dilakukan apabila terdapat *missing value* dan data *outlier* dari *Dataset Stroke Disease*.

1. Missing Value

Pada atribut bmi terdapat 201 nilai NaN, sehingga atribut tersebut harus diperbaiki. Di dalam Python nila NaN dianggap menjadi nilai yang hilang atau *missing value*. Untuk memperbaiki *missing value* pada atribut bmi, nilai NaN akan diganti dengan nilai median dari seluruh nilai yang ada di dalam atribut bmi. Mengganti nilai NaN lebih baik daripada menghapus nilai NaN. Tabel 4.2 di bawah ini menunjukkan hasil dari atribut bmi yang sudah dilakukan proses data cleaning.

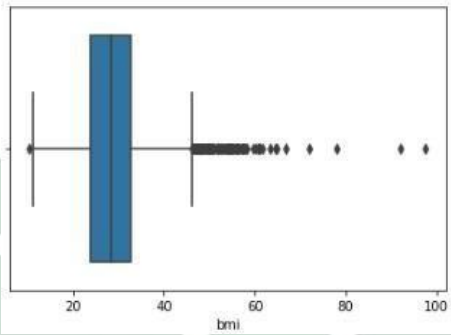
digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Tabel 4. 2 Sampel dataset setelah dilakukan proses data cleaning

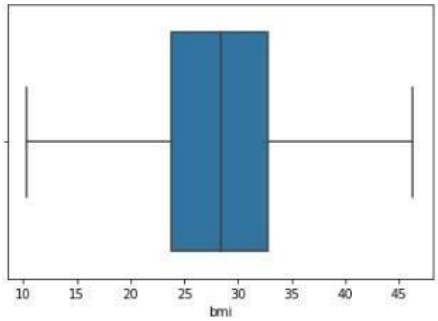
Gender	Age	Hyper-tension	Heart_ Disease	…	Avg_ glucose _level	BMI	Smoking Status	Stroke
Male	67.0	0	1	…	228.69	36.6	formerly smoked	1
Female	61.0	0	0	…	202.21	28.8	never smoked	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Male	51.0	0	0	…	166.29	25.6	formerly smoked	0
Female	44.0	0	0	…	85.28	26.2	Unknown	0

2. Data Oultier

Selain memiliki data yang hilang, atribut bmi memiliki data yang tidak sesuai dengan karakteristik atau berada jauh dari data yang sebenarnya. Nilai untuk bmi maksimal adalah 50. Adapun bukti Gambar 4.2 di bawah ini yang menunjukkan adanya data *outlier* pada atribut bmi.



Gambar 4. 2 Hasil boxplot atribut bmi sebelum dilakukan IQR



Gambar 4. 3 Hasil boxplot atribut bmi setelah dilakukan IQR

Berdasarkan gambar 4.2 diatas, data *outlier* yang pada atribut bmi berada di atas kuartil 3 (Q3), sehingga *outlier* dapat diperbaiki menggunakan metode *Interquartile Range* (IQR) yang merupakan pembagian data dari kuartil

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

pertama(Q1), median(Q2) dan kuartil ketiga(Q3). Kemudian pada gambar 4.3, nilai data *outlier* pada atribut bmi telah diubah ke nilai maximum sebesar 46.3 dan nilai minimumnya sebesar 10.3 dari *Interquartile Range.*

4.2.2 Data Transfromation

Data transformation merupakan proses mengubah format tipe data pada dengan menyesuaikan tipe data yang lain. Tipe data pada setiap atribut terbagi menjadi dua yaitu numerik dan kategorikal. Atribut dengan tipe data kategorikal akan dikonversi menjadi angka 0, 1, dst. Beberapa atribut yang memiliki tipe data kategorikal yaitu atribut gender, ever married, work type, residence type, dan smoking status.

Mengubah tipe data kategorikal menjadi data numerik dapat menggunakan *LabelEncoder()* dari sklearn.preprocessing yang ada didalam *library* Python guna memecahkan kode label tertentu agar dapat mempermudah proses klasifikasi. Karena python tidak dapat mengkonversi data string menjadi float, sehingga dilakukan konversi data terlebih dahulu dengan mengubahnya data menjadi numerik. Tabel 4.3 di bawah ini menunjukkan dataset yang berhasil dikonversi ke data baru.

Tabel 4. 3 Sampel dataset setelah dilakukan tahapan transformation data

Gender	Age	Hyper-tension	Heart Disease	Ever Married	…	BMI	Smoking Status	Stroke
1	67.0	0	1	1	…	36.0	1	1
0	61.0	0	0	1	…	28.0	2	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	51.0	0	0	1	...	25.0	1	0
0	44.0	0	0	1	…	26.0	0	0

Berdasarkan tabel 4.3 atribut yang memiliki tipe data kategorikal telah diubah menjadi tipe data numerik. Untuk mempermudah mengetahui detil perubahan dari setiap data. Tabel 4.4 berikut menunjukkan detil atribut kategorikal yang sudah dilakukan perubahan.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Tabel 4. 4 Detil atribut kategorikal yang diubah menjadi numerik

Atribut Kategorikal	Sebelum dikonversi	Sesudah dikonversi
Gender	Female Male	Female = 0 Male = 1
Ever Married	Tidak Ya	Tidak = 0 Ya = 1
Work Type	Govt_Jov Private Self_employed Children Never_worked	Govt_Jov = 0 Never worked = 1 Private = 2 Self employed = 3 Children = 4
Resindence Type	Urban Rural	Urban = 0 Rural = 1
Smoking Status	Unknown Formely_smoked Never_smoked Smokes	Unknown = 0 Formely_smoked = 1 Never_smoked = 2 Smokes = 3

Setelah dilakukan proses cleaning dan transformation data, terdapat atribut yang dihapus yaitu Id. Karena atribut tersebut tidak memiliki pengaruh terhadap kemungkinan terjadinya stroke. Maka dari itu, untuk proses klasifikasi data yang digunakan hanya 11 atribut dengan jumlah data sebanyak 5010 data.

4.3 Klasifikasi Algoritma C4.5

Dataset Stroke Disease telah melalui tahap pengolahan data, untuk tahap selanjutnya yaitu melakukan tahap pelatihan model. Berdasarkan skenario klasifikasi pada Tabel 3.2, pelatihan model yang pertama yaitu menggunakan algoritma C4.5. Proses klasifikasi diolah ke dalam bahasa pemrograman Python menggunakan *library* s*cikit learn.* Ada dua tahapan yang dilakukan ketika proses klasifikasi yaitu *training* dan *testing.* Tahapan tersebut dilakukan untuk menemukan hasil *output* terbaik dari klasifikasi algoritma C4.5. Berikut merupakan *peseudocode* dari algoritma C4.5.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

```
Algoritma C4.5
Input values:
D = Dataset; D* = Sub-datasets from D based on a*
T = Tree;
a = Attribute; a* = best_attribute
Process :
 Check for dataset
 For all attribute a € D
 Calculate entropy and gain criteria,
 Choose a with the highest gain criteria,
 Create decision node with a* in the root,
 For all D*
 Add to decision node for corresponding branch of T
End for
 Return T
```

4.4 Klasifikasi Algoritma C4.5 + Bagging

Pelatihan model yang kedua yaitu menerapkan metode bagging pada algoritma C4.5. Berbeda dengan proses klasifikasi sebelumnya, pada proses ini algortima C4.5 akan dikombinasikan dengan metode Bagging guna meningkatkan kinerja dari algoritma C4.5. Proses klasifikasi untuk *training* akan dilakukan secara berulang sampai memenuhi batas maksimal yang sudah ditentukan. Setelah itu hasil dari pengulangan tersebut akan dipilih melalui pemungutan suara terbanyak untuk menemukan *output* yang terbaik. Proses klasifikasi untuk tahap ini masih sama dengan sebelumnya yaitu diolah ke dalam bahasa pemrograman Python menggunakan *library scikit learn.*Berikut adalah *peseudocode* dari metode Bagging.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

```
Peseudocode Algoritma  Bagging

Input :
Training Sample(S); Classifier(L); Iterations(T)
Output : H(x)
Process :
for t = 1 to T
Dt = boostrap sample from D
ht = train a classifier on Dt via L
end for
H(x) = arg maxy(∑ =1 1 (y = ht(x))
```

4.5 Klasifikasi Algoritma C4.5 + AdaBoost

Pada pelatihan model yang ketiga yaitu menggunakan algoritma C4.5 dan metode adaboost. Pada tahap ini tidak ada bedanya dengan tahap sebelumnya, hanya saja pada tahap ini algoritma C4.5 akan dikombinasikan dengan metode adaboost. Berikut merupakan *peseudocode* dari metode adaboost.

```
Pseudocode Algoritma  Adaboost :

Input :
Training sample(D); Base Classifier(L); Iterations(i)
Output : H(x)Process :
Normalize the weights and make the total weight Dt(i)= 1

  Dt = Sample from D according to the distribution
  ht = Train a classifier on Dt via L
  ei = ∑Dt(i)
  αt = 1/2 In((1−ϵ)/ϵ)
  Dt+1= ()/ x {e^-αt atau e^αt
  End For
H(x)= sign (∑ =1 t ht (x))
```

Kombinasi algoritma C4.5 dengan metode adaboost dapat membantu meningkatkan perfroma algoritma C4.5 menjadi lebih baik. Proses klasifikasi untuk tahap ini masih sama dengan sebelumnya yaitu diolah ke dalam bahasa pemrograman Python menggunakan *library scikit learn.* Kemudian tahap

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

selanjutnya adalah perhitungan *confusion matrix* dan *cross validation* untuk menemukan hasil *output* yang terbaik dari beberapa skenario.

4.6 Uji Validasi dan Evaluasi

Pada tahapan ini dilakukan pengujian model pada *Dataset Stroke Disease* tanpa dilakukan penyeimbangan data. Pengujian model klasifikasi dataset stroke dilakukan melalui dua tahapan yaitu tahap *training* dan *testing*. Data *training* atau *train set* merupakan tahapan untuk melatih algoritma sehingga dapat membentuk sebuah model prediksi, sedangkan data *testing* atau *test set* merupakan tahapan untuk mengetahui hasil performa dari algortima klasifikasi. Sebelum melakukan dua tahap tersebut, sebaiknya menentukan ukuran parameter untuk *test set* yaitu sebesar 30% dan 70% untuk *train set.*

4.6.1 Algortima C4.5

Pengujian model ini menerapkan metode confusion matrix, untuk mengetahui nilai dari kelas yang sebenarnya dan nilai kelas prediksi dari *Dataset Stroke Disease*. Tabel 4.5 menunjukkan hasil pengujian model confusion matrix menggunakan algoritma C4.5.

Tabel 4. 5 *Confusion Matrix* dari Algoritma C4.5

Confusion Matrix		**Kelas Sebenarnya**		**Class Presisi**
		Stroke	No Stroke	
Kelas Prediksi	Stroke	10	66	0.27
	No Stroke	27	1430	0.96
Class Recall		0.13	0.98	
Class F1Score		0.18	0.97	

Diketahui :

TP (Benar Positif) = 10

TN (Benar Negatif) = 1430

FP (Salah Positif) = 27

FN (Salah Negatif) = 66

Dilihat dari Tabel 4.5 dapat disimpulkan bahwa pasien yang dinyatakan terkena stroke sebanyak 10, pasien dinyatakan tidak terkena stroke atau non stroke sebanyak 1430, pasien stroke yang diidentifikasi menjadi non stroke sebanyak 27 termasuk ke dalam "Type I Error", sedangkan pasien non stroke yang diidentifikasi menjadi stroke sebanyak 66 termasuk ke dalam "Type II Error".

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Setelah diketahui hasil dari nilai confusion matrix, maka selanjutnya adalah ,,menghitung nilai dari presisi, spesifisitas, sensivisitas dan f1score yang sudah dijelaskan di persamaan 11, 12, 13 dan 14.

Presisi = TP/(TP+FP) = 10/(10+27) = 0.27

Spesifisitas = TN/(TN+FP) = 1430/(1430+27) = 0.98

Sensivisitas = TP/(TP+FN) = 10/(10+66) = 0.13

F1Score = 2 * (Presisi*Sensivisitas/ Presisi+Sensivisitas) = 0.17

Selain melakukan perhitungan pada nilai - nilai *confusion matrix.* Adapun nilai akurasi yang dihitung menggunakan *k-fold cross validation* untuk menentukan akurasi mana yang memiliki nilai terbaik. Tabel 4.6 di bawah ini menunjukkan hasil akhir dari perhitungan akurasi.

Tabel 4. 6 Hasil pengujian nilai akurasi menggunakan 10*fold cross-validation*

Eksperimen	Akurasi Algoritma C4.5	Eksperimen	Akurasi Algoritma C4.5
K1	0.9217	**K6**	0.9022
K2	0.9413	**K7**	0.9329
K3	0.9106	**K8**	0.9327
K4	0.9385	**K9**	0.9299
K5	0.9329	**K10**	0.9439
Rata - Rata			**0.9287**

Berdasarkan Tabel 4.6, kesimpulan yang didapat adalah bahwa akurasi yang dihasilkan memiliki 10 nilai akurasi yang baik. Namun dari 10 nilai tersebut akan dilakukan penyeleksian hasil dengan cara menghitung rata – rata dari keseluruhan nilai guna mendapatkan nilai akurasi yang terbaik. Nilai akurasi yang diperoleh dari perhitungan rata – rata 10-*fold cross validation* adalah sebesar 92.87%.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

4.6.2 Algoritma C4.5 + Bagging

Pengujian model kedua yaitu kombinasi algoritma C4.5 dan Bagging yang menerapkan metode *confusion matrix* dan *cross validation* untuk mengetahui nilai dari kelas yang sebenarnya dan nilai kelas prediksi dari *Dataset Stroke Disease*. Tabel 4.7 menunjukkan hasil pengujian model dari *confusion matrix* menggunakan algoritma C4.5 dan bagging.

Tabel 4. 7 *Confusion Matrix* dari Metode Bagging dan Algoritma C4.5

Confusion Matrix		**Kelas Prediksi**		**Class Presisi**
		Stroke	No Stroke	
Kelas Sebenarnya	Stroke	2	74	0.50
	No Stroke	2	1455	0.95
Class Recall		0.03	1.00	
Class F1Score		0.05	0.97	

Diketahui :

TP (Benar Positif) = 2

TN (Benar Negatif) = 1455

FP (Salah Positif) = 2

FN (Salah Negatif) = 74

Dilihat dari Tabel 4.7 dapat disimpulkan bahwa pasien yang dinyatakan terkena stroke sebanyak 2, pasien yang dinyatakan tidak terkena stroke atau non stroke sebanyak 1455, pasien stroke yang diidentifikasi menjadi non stroke sebanyak 2 termasuk ke dalam "Type I Error", sedangkan pasien non stroke yang diidentifikasi menjadi stroke sebanyak 74 termasuk ke dalam "Type II Error". Setelah diketahui hasil dari presisi, spesifisitas, sensivisitas dan F1Score yang sudah dijelaskan di persamaan 11, 12, 13 dan 14.

Presisi = TP/(TP+FP) = 2/(2+2) = 0.50

Spesifisitas = TN/(TN+FP) = 1455/(1455+2) = 1.00

Sensivisitas = TP/(TP+FN) = 2/(2+74) = 0.03

F1Score = 2 * (Presisi*Sensivisitas/ Presisi+Sensivisitas) = 0.05

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Selain melakukan perhitungan pada nilai - nilai *confusion matrix.* Adapun nilai akurasi yang dihitung menggunakan *k-fold cross validation* untuk menentukan akurasi mana yang memiliki nilai terbaik. Tabel 4.8 di bawah ini menunjukkan hasil akhir dari perhitungan akurasi.

Tabel 4. 8 Hasil pengujian nilai akurasi menggunakan 10*fold cross-validation*

Eksperimen	Akurasi Algoritma C4.5 + Bagging	Eksperimen	Akurasi Algoritma C4.5 + Bagging
K1	0.9497	**K6**	0.9525
K2	0.9553	**K7**	0.9553
K3	0.9469	**K8**	0.9495
K4	0.9469	**K9**	0.9467
K5	0.9441	**K10**	0.9551
Rata - Rata			**0.9502**

Pada Tabel 4.8 dapat disimpulkan bahwa akurasi yang dihasilkan memiliki 10 nilai akurasi yang sangat baik dan beberapa diantaranya adalah 94 dan 95%. Namun dari 10 nilai tersebut akan dilakukan penyeleksian hasil dengan cara menghitung rata – rata dari keseluruhan nilai guna mendapatkan nilai akurasi yang terbaik. Nilai akurasi yang diperoleh dari perthitungan rata – rata 10-*fold cross validation* adalah sebesar 95.02%.

4.6.3 Algoritma C4.5 + Adaboost

Pengujian model ini menerapkan metode *confusion matrix*, sehingga dapat diketahui nilai dari kelas yang sebenarnya dan nilai kelas prediksi dari *Dataset Stroke Disease*. Tabel 4.9 menunjukkan hasil pengujian model dari *confusion matrix* menggunakan algoritma C4.5 dan adaboost.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Tabel 4. 9 *Confusion Matrix* dari algoritma C4.5 + adaboost

Confusion Matrix		**Kelas Prediksi**		**Class Presisi**
		Stroke	No Stroke	
Kelas Sebenarnya	Stroke	6	70	0.46
	No Stroke	7	1450	0.95
Class Recall		0.08	1.00	
Class F1Score		0.13	0.97	

Diketahui :

TP (Benar Positif) = 6

TN (Benar Negatif) = 1450

FP (Salah Positif) = 7

FN (Salah Negatif) = 70

Dilihat dari Tabel 4.9 dapat disimpulkan bahwa pasien yang dinyatakan terkena stroke sebanyak 6, pasien dinyatakan tidak terkena stroke atau non stroke sebanyak 1450, pasien stroke yang diidentifikasi menjadi non stroke sebanyak 7 termasuk ke dalam "Type I Error", sedangkan pasien non stroke yang diidentifikasi menjadi stroke sebanyak 70 termasuk ke dalam "Type II Error". Setelah diketahui hasil dari presisi, spesifisitas, sensivisitas dan F1Score yang sudah dijelaskan di persamaan 11, 12, 13 dan 14.

Presisi = TP/(TP+FP) = 6/(6+7) = 0.46

Spesifisitas = TN/(TN+FP) = 1450/(1450+7) = 1.00

Sensivisitas = TP/(TP+FN) = 6/(6+70) = 0.08

F1Score = 2 * (Presisi*Sensivisitas / Presisi+Sensivisitas) = 0.13

Selain melakukan perhitungan pada nilai - nilai *confusion matrix*. Adapun nilai akurasi yang dihitung menggunakan *k-fold cross validation* untuk menentukan akurasi mana yang memiliki nilai terbaik. Tabel 4.10 di bawah ini menunjukkan hasil akhir dari perhitungan akurasi.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Tabel 4. 10 Hasil pengujian nilai akurasi menggunakan 10fold cross-validation

Eksperimen	Akurasi Algoritma C4.5 + Adaboost	Eksperimen	Akurasi Algoritma C4.5 + Adaboost
K1	0.9413	**K6**	0.9329
K2	0.9553	**K7**	0.9469
K3	0.9441	**K8**	0.9467
K4	0.9469	**K9**	0.9523
K5	0.9469	**K10**	0.9495
Rata - Rata			**0.9463**

Kesimpulan dari Tabel 4.10 bahwa nilai akurasi yang terdiri dari 10 *fold* memiliki hasil yang sangat baik. Beberapa nilai akurasi yang dihasilkan antara 94% dan 95%. Namun dari 10 *fold* tersebut akan dilakukan penyeleksian hasil dengan cara menghitung rata – rata dari keseluruhan nilai guna mendapatkan nilai akurasi yang terbaik. Nilai akurasi yang diperoleh dari perthitungan rata – rata 10-*fold cross validation* adalah sebesar 94.63%.

4.7 Analisis Hasil

Setelah membahas hasil evaluasi dari setiap skenario pada *Dataset Stroke Disease,* maka proses selanjutnya adalah melakukan analisis hasil evaluasi dengan cara membandingkan performa klasifikasi. Pada sub bab ini akan membahas mengenai nilai presisi, spesifisitas, sensivisitas, f1score dan akurasi yang dihasilkan oleh setiap skenario. Kemudian nilai - nilai tersebut akan dibandingkan guna menemukan hasil yang terbaik. Perbandingan hasil evaluasi dapat membantu menjawab rumusan masalah yang telah dipaparkan sebelumnya.

Skenario pertama yaitu klasifikasi *Dataset Stroke Disease* menggunakan algoritma C4.5. Model algoritma C4.5 ketika diterapkan ke dalam Python menggunakan tingkat kedalaman pohon atau *max_depth* sebanyak 8 tingkat guna menghindari *overfiting* dan *underfiting*. Evaluasi model algortima C4.5 untuk *class* stroke menghasilkan nilai presisi sebesar 27 %, spesifisitas sebesar 98%, sensivisitas sebesar 13%, f1score sebesar 18% dan nilai akurasi sebesar 92,87%. Sedangkan untuk nilai presisi, sensivisitas dan f1score pada class no stroke memiliki nilai yang sangat tinggi dibandingan dengan nilai - nilai pada class stroke. Tabel 4.11 di bawah ini menunjukkan detil hasil evaluasi dari skenario pertama

.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Tabel 4. 11 Hasil pengujian skenario pertama

UJI VALIDASI	ALGORITMA C4.5
Presisi	0.27
Spesifisitas	0.98
Sensivisitas	0.13
F1Score	0.18
Akurasi	0.9287

Selanjutnya skenario kedua yaitu klasifikasi menggunakan algoritma C45. dan Bagging. Model skenario kedua merupakan model kombinasi dari algoritma C4.5 dan metode Bagging. Kombinasi model tersebut bertujuan untuk meningkatkan performa dari pengklasifikasi lemah yaitu algoritma C4.5. Nilai performa yang dihasilkan dari kombinasi algoritma C4.5 dan metode Bagging yaitu nilai presisi sebesar 50%, spesifisitas sebesar 100%, sensivisitas sebesar 3%, f1score sebesar 5% dan nilai akurasi sebesar 95,02%. Nilai sensivisitas dan f1score yang dihasilkan oleh skenario kedua lebih rendah daripada skenario pertama, karena berdasarkan dari tabel 4.6 pada *class false negatif* yang dihasilkan terbilang masih tinggi yaitu sebesar 74 data. Sehingga nilai sensivisitasnya mendapatkan nilai yang sangat rendah. Tabel 4.12 di bawah ini menunjukkan detil hasil evaluasi dari skenario kedua.

Tabel 4. 12 Hasil pengujian skenario kedua

UJI VALIDASI	ALGORITMA C4.5 + BAGGING
Presisi	0.50
Spesifisitas	1.00
Sensivisitas	0.03
F1Score	0.05
Akurasi	0.9502

Skenario yang terakhir atau yang ketiga yaitu klasifikasi menggunakan algoritma C4.5 dan Adaboost. Model skenario ini merupakan model kombinasi dari algoritma C4.5 dan metode Adaboost yang memiliki tujuan yang sama dengan skenario sebelumnya. Evaluasi model algortima C4.5 yang dikombinasikan dengan metode Adaboost menghasilkan nilai presisi sebesar 46%, spesitifitas sebesar 100%, sensitifitas sebesar 8%, f1score sebesar 13% dan nilai

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

akurasi sebesar 94,63%. Tabel 4.13 di bawah ini menunjukkan detil hasil evaluasi dari skenario ketiga.

Tabel 4. 13 Hasil pengujian skenario ketiga

UJI VALIDASI	ALGORITMA C4.5 + ADABOOST
Presisi	0.46
Spesifisitas	1.00
Sensivisitas	0.08
F1Score	0.13
Akurasi	0.9463

Berdasarkan Tabel 4.12 dan 4.13 penerapan metode bagging dan adaboost memiliki pengaruh dalam meningkatnya nilai akurasi dari algoritma C4.5. Akurasi yang dihasilkan oleh kombinasi algoritma dengan metode bagging dan adaboost untuk klasifikasi *Dataset Stroke Disease* dapat dikatakan lebih baik dibandingkan hanya menerapkan satu metode saja. Tabel 4.14 menunjukkan detil perbandingan hasil pengujian untuk keseluruhan skenario klasifikasi.

Tabel 4. 14 Perbandingan hasil pengujian tiap skenario

Uji Validasi	Algoritma C4.5	Algoritma C4.5 + Bagging	Algoritma C4.5 + Adaboost
Presisi	27%	50%	46%%
Spesitifitas	98%	100%	100%
Sensivisitas	13%	3%	8%
F1Score	18%	5%	13%
Akurasi	92,87%	95,02%	94,63%

Dari Tabel 4.14 klasifikasi *Dataset Stroke Disease* yang menggunakan algoritma C4.5 saja memberikan nilai yang kecil untuk nilai presisi, sensivisitas dan f1score pada class stroke. Namun berbeda dengan nilai yang hasilkan oleh spesifisitas yang menghitung class no stroke dan akurasi keduanya memiliki nilai yang sangat tinggi. Kemudian jika dibandingkan dengan skenario kedua yaitu algortima C4.5 dan metode Bagging, nilai presisi yang dihasilkan mengalami peningkatakan sebesar 23%. Meskipun peningkatan yang dihasilkan oleh metode bagging masih kurang signifikan, tetapi metode ini berhasil meningkatkan nilai presisi pada algortima C4.5. Namun nilai sensivisitas dan f1score pada skenario kedua ini menghasilkan nilai yang sangat rendah untuk class stroke dan jika

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

dibandingkan dengan skenario pertama yang menggunakan algoritma C4.5 nilai mengalami selisih sebesar 10% dan 13%.

Selain itu, adapun skenario ketiga yaitu algoritma C4.5 yang dikombinasikan dengan metode adaboost memiliki nilai presisi untuk class stroke sebesar 46%, dimana nilai ini mengalami peningkatan sebesar 19% dari nilai presisi algoritma C4.5 saja. Namun pada nilai sensivisitas yang dihasilkan oleh algoritma C4.5 dan adaboost mengalami penurunan dan nilai f1score yang dihasilkan oleh kombinasi metode ini memiliki nilai yang sama dengan algoritma C4.5 saja. Sedangkan untuk nilai akurasi setelah dikombinasikan dengan metode bagging mengalami peningkatan sebanyak 3% dan ketika dikombinasikan dengan metode adaboost mengalami pengingkatan sebanyak 2%. Karena akurasi awal dari algoritma C4.5 adalah 92% dan meningkat menjadi 95% oleh metode bagging dan 94% oleh metode adaboost pada algortima C4.5. Dari hasil tersebut dapat disimpulkan bahwa metode bagging dan adaboost terbukti dapat meningkatkan dan memperbaiki kinerja dari algoritma C4.5 seperti pada nilai presisi, spesifisitas dan akurasi. Namun karena peningkatan setiap nilai tidak terlalu tinggi sehingga dapat dikatakan bahwa nilai - nilai tersebut kurang signifkan. Dilihat dari ketiga nilai tersebut penerapan metode bagging pada algortima C4.5 terbukti lebih unggul dibandingkan metode adaboost.

4.8 Pembahasan

Hasil penelitian yang telah dipaparkan sebelumnya akan dibahas pada sub bab ini untuk mengetahui beberapa permasalahan dari penelitian yang sedang diteliti. Dari permasalahan yang ditemukan dapat membantu penelitian selanjutnya apabila menggunakan dataset atau metode yang sama dengan penelitian ini.

Dataset Stroke Disease memiliki 12 atribut yang terdiri dari id, gender, age, hypertension, heart disease, ever married, work type, residence, avg glucose level, bmi, smooking status dan stroke. Metode pertama yang digunakan pada klasifikasi stroke adalah algoritma C4.5. Algortima tersebut menghasilkan akurasi sebesar

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

92%, akurasi tersebut terlihat sangat baik karena berada di atas 90%. Hasil tersebut selaras dengan penelitian (Ashokan et al. 2020) yang menyatakan bahwa algoritma C4.5 mampu memberikan akurasi yang baik terhadap prediksi stroke.

Metode selanjutnya yang digunakan adalah kombinasi algoritma C4.5 dengan metode Bagging dan kombinasi algoritma C4.5 dengan metode Adaboost. Proses klasifikasi yang menggunakan teknik ensemble tersebut memiliki peran dalam meningkatkan akurasi dan dapat menangani ketidakseimbangan data atau *imbalance class* pada Dataset Stroke Disease yang sudah dijelaskan pada bab pengolahan data. Penelitian ini sesuai dengan (Saifudin and Wahono 2015) dan (Arrahimi, Ihsan, and Kartini 2019) yang menyatakan bahwa teknik ensemble mampu menangani masalah *imbalance class* serta mampu meningkatkan nilai performa pada algoritma pengklasifikasi lemah.

Metode Bagging dan Adaboost memang terbukti dapat meningkatkan nilai presisi, spesifisitas dan akurasi dari algoritma C4.5. Tetapi pada penelitian ini hanya meningkatkan beberapa persen saja sehingga hasilnya menjadi kurang signifikan. Berbeda dengan penelitian yang dilakukan oleh (Yulianti et al. 2020) dan (Jusia 2018) yang menyatakan bahwa kombinasi algoritma C4.5 dan metode bagging dapat meningkat lebih banyak jika dilakukan optimasi dengan metode *feature selection,* sehingga nilai performa yang dihasilkan menjadi lebih baik.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

BAB 5
PENUTUP

5.1 Kesimpulan

1. Penerapan metode Bagging dan AdaBoost pada algoritma C4.5 digunakan untuk mengatasi masalah ketidakseimbangan kelas dan meningkatkan performa klasifikasi. Metode Bagging dan AdaBoost sangat berpengaruh dalam meningkatkan nilai akurasi, presisi, dan spesitifitas pada algoritma C4.5.
2. Berdasarkan hasil perbandingan anlisa, algoritma C4.5 menghasilkan nilai akurasi sebesar 92,87%, nilai presisi 27 %, spesifisitas 98%, sensivisitas 13%, dan f1score sebesar 18%. Sedangkan akurasi dari algoritma C4.5 setelah menerapkan metode bagging adalah 95,02%, nilai presisi 50%, spesifisitas 100%, sensivisitas 3%, dan f1score sebesar 5% serta akurasi dari algoritma C4.5 setelah menerapkan metode adaboost adalah sebesar 94,63% dengan nilai presisi 46%, spesitifitas 100%, sensivisitas 8%, dan f1score sebesar 13%. Komparasi metode Bagging dan Adaboost pada algoritma C4.5 terbukti dapat meningkatkan dan memperbaiki kinerja klasifikasi. Nilai akurasi dari algoritma C4.5 setelah dikombinasikan dengan metode bagging dan adaboost meningkat sebanyak 3% dan 2%. Peningkatan nilai akurasi diikuti oleh nilai presisi yang meningkat sebanyak 23% dan 19%. Terlihat dari hasil evaluasi tersebut dapat disimpulkan bahwa kedua metode tersebut kurang signifikan dalam meningkatkan nilai akurasi dan presisi dari algoritma C4.5. Selain itu, metode bagging dan adaboost juga tidak dapat meningkatkan nilai sensivisitas dan f1score dari algortima C4.5, karena nilai *class false negative* yang dihasilkan terbilang masih tinggi.

5.2 Saran

Berdasarkan penelitian klasifikasi pada Dataset Stroke Disease yang menggunakan algoritma C4.5 dengan menerapkan metode Bagging dan AdaBoost yang telah dilakukan tentunya masih banyak kekurangan. Untuk penelitian selanjutnya yang ingin mengembangkan penelitian dengan

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

menggunakan dataset yang sama. Adapun saran dari penulis sebagai rekomendasi dalam memperbaiki penelitian berikutnya.

1. Mengembangkan penelitian yang serupa dengan menggunakan algoritma klasifikasi lain agar hasil prediksi menjadi lebih baik.
2. Membandingkan metode Bagging dan Adaboost dengan metode *ensemble learning* lainnya seperti Random Forest atau Gradient Boosting untuk mengoptimalkan kinerja klasifikasi.
3. Melakukan penyeimbangan data pada *Dataset Stroke Disease* menggunakan metode *Oversampling* agar performa yang dihasilkan menjadi lebih baik dan akurat. Metode yang disarankan adalah SMOTE.
4. Melakukan penyeleksian atribut dengan menerapkan *Feature Selection* seperti *Particle Swarm Optimization* atau *Chi Square* agar dapat mengetahui atribut mana yang memiliki pengaruh secara signifikan terhadap stroke.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

DAFTAR PUSTAKA

Abdillah, Sigit. 2015. "Penerapan Algoritma Decision Tree C4.5 Untuk Diagnosa Penyakit Stroke Dengan Klasifikasi Data Mining Pada Rumah Sakit Santa Maria Pemalang." *Skripsi, Fakultas Ilmu Komputer*. http://eprints.dinus.ac.id/id/eprint/16477.

Adam, Selma Yahiya, and Mohammed Bakri Bashir. 2016. "Classification of Ischemic Stroke Using Machine Learning Algorithms." *International Journal of Computer Applications* 149 (10): 26–31.

Adelina, Vina, Dian Eka Ratnawati, and M. Ali Fauzi. 2018. "Klasifikasi Tingkat Risiko Penyakit Stroke Menggunakan Metode GA-Fuzzy." *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer* 2 (September): 3015–21. https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/2513.

Amini, Leila, Reza Azarpazhouh, Mohammad Taghi Farzadfar, Sayed Ali Mousavi, and Farahnaz Jazaieri. 2013. "Prediction and Control of Stroke by Data Mining." *International Journal of Preventive Medicine* 4 (May): 245–49.

Arifianto, Aji Seto, Moechammad Sarosa, and Onny Setyawati. 2014. "Klasifikasi Stroke Berdasarkan Kelainan Patologis Dengan Learning Vector Quantization." *Jurnal EECCIS* 8 (2): 117–22.

Arrahimi, Ahmad Rusadi, Muhammad Khairi Ihsan, and Dwi Kartini. 2019. "Teknik Bagging Dan Boosting Pada Algoritma CART Untuk Klasifikasi Masa Studi Mahasiswa" 5: 21–30.

Ashokan, Soodamani, Suriya G S Narayanan, S Mandresh, Vidhyasagar Bs, and Paavai Anand G. 2020. "An Effective Stroke Prediction System Using Predictive Models," 3979–85.

Bisri, Achmad. 2015. "Penerapan Adaboost Untuk Penyelesaian Ketidakseimbangan Kelas Pada Penentuan Kelulusan Mahasiswa Dengan Metode Decision Tree." *Journal of Inteligent System* 1 (1): 27–32.

Byna, Agus, and Muhammad Basit. 2020. "Penerapan Metode Adaboost Untuk

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Mengoptimasi Prediksi Penyakit Stroke Dengan Algoritma Naïve Bayes." *Jurnal Sisfokom (Sistem Informasi Dan Komputer)* 09 (November): 407–11.

Carolina, Anastasia, and Kemal Ade. 2020. "Penerapan Data Mining Dengan Menggunakan Provinsi Di Indonesia Pendahuluan." *Jurnal Ilmiah Kommputer* 19: 27–38.

Do, Merve, Pelin Sarı, Niyazi Kılıç, and Aydın Akan. 2016. "Detection of Chronic Kidney Disease by Using Adaboost Ensemble Learning Approach." *2016 24th Signal Processing and Communication Application Conference (SIU)*, 1–4.

Handayani, Dian, and Dwi Dominica. 2018. "Gambaran Drug Related Problems (DRP ' s) Pada Penatalaksanaan Pasien Stroke Hemoragik Dan Stroke Non Hemoragik Di RSUD Dr M Yunus Bengkulu." *Jurnal Farmasi Dan Ilmu Kefarmasian Indonesia* 5 (1). http://dx.doi.org/10.20473/jfiki.v5i12018.36-44.

Herho, Sandy H.S. 2017. *Tutorial Pemrograman Python 2 Untuk Pemula*. Weather and Climate Prediction Laboratory Press, Bandung.

Issn, P, and Ahmad Ilham. 2017. "Komparasi Algortima Klasifikasi Dengan Pendekatan Level Data Untuk Menangani Data Kelas Tidak Seimbang." *Jurnal Ilmiah Ilmu Komputer* 3 (1): 9–14.

Jalota, Chitra, and Rashmi Agrawal. 2019. "Analysis of Educational Data Mining Using Classification." *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, 243–47.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An Introduction to Statistical Learning with Applications in R*. New York: Springer Science+Business Media New York.

Jusia, Pareza Alam. 2018. "Analisis Komparasi Pemodelan Algoritma Decision Tree Menggunakan Metode Particle Swarm Optimization Dan Metode Adaboost Untuk Prediksi Awal Penyakit Jantung." *Seminar Nasional Sistem Informasi*, 1048–56.

Kansadub, Teerapat, and Sotarat Thammaboosadee. 2015. "Stroke Risk Prediction Model Based on Demographic Data." *2015 8th Biomedical Engineering*

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

International Conference (BMEiCON), 3–5.

Latha, C Beulah Christalin, and S Carolin Jeeva. 2019. "Improving the Accuracy of Prediction of Heart Disease Risk Based on Ensemble Classification Techniques." *Informatics in Medicine Unlocked* 16 (June): 100203. https://doi.org/10.1016/j.imu.2019.100203.

Lestari, Aprilia, and Alamsyah. 2020. "Increasing Accuracy of C4 . 5 Algorithm Using Information Gain Ratio and Adaboost for Classification of Chronic Kidney Disease." *JOSCEX* 1 (1): 32–38.

Muslim, Much Aziz, Siti Alimah, and Budi Prasetiyo. 2017. "Penerapan Dizcretization Dan Teknik Bagging Untuk Meningkatkan Akurasi Klasifikasi Berbasis Ensemble Pada Algoritma C4 . 5 Dalam Mendiagnosa Diabetes." *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi* 8 (2): 135–43. https://doi.org/10.24843/LKJITI.2017.v08.i02.p07.

Muslim, Much Aziz, Aldi Nurzahputra, and Budi Prasetiyo. 2018. "Improving Accuracy of C4 . 5 Algorithm Using Split Feature Reduction Model and Bagging Ensemble for Credit Card Risk Prediction." *2018 International Conference on Information and Communications Technology (ICOIACT)*, no. 1996: 141–45.

Muzakir, Ari. 2016. "Model Data Mining Sebagai Prediksi Penyakit Hipertensi Kehamilan Dengan Teknik Decision Tree Model Data Mining Sebagai Prediksi Penyakit Hipertensi Kehamilan Dengan Teknik Decision Tree." *Jurnal Ilmiah Informatika* 3 (1): 19–26. https://doi.org/10.15294/sji.v3i1.4610.

Nai-arun, Nongyao, and Punnee Sittidech. 2016. "Ensemble Learning Model for Diabetes Classification." *Advanced Materials Research* 931–932 (May 2014): 1427–31. https://doi.org/10.4028/www.scientific.net/AMR.931-932.1427.

Nugraha, Deny Wiria, and A Y Erwin Dodu. 2017. "Klasifikasi Penyakit Stroke Menggunakan Metode Naive Bayes Classifier (Studi Kasus Pada Rumah Sakit Umum Daerah Undata Palu)." *SEMANTIK* 3 (2): 13–22.

Permatasari, Nia. 2020. "Perbandingan Stroke Non Hemoragik Dengan Gangguan Motorik Pasien Memiliki Faktor Resiko Diabetes Melitus Dan Hipertensi

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Pendahuluan." *Jurnal Ilmiah Kesehatan Sandi Husada* 11 (1): 298–304. https://doi.org/10.35816/jiskh.v10i2.273.

Prasertsakul, Thunyanoot, Panya Kaimuk, and Warakorn Charoensuk. 2015. "Defining the Rehabilitation Treatment Programs for Stroke Patients by Applying Neural Network and Decision Trees Models." *Biomedical Engineering International Conference*.

Prasetio, Rizki Tri, and Pratiwi Pratiwi. 2015. "Penerapan Teknik Bagging Pada Algoritma Klasifikasi Untuk Mengatasi Ketidakseimbangan Kelas Dataset Medis." *Jurnal Informatika* II (2): 395–403.

Pristyanto, Yoga. 2019. "Penerapan Metode Ensemble Untuk Meningkatkan Kinerja Algoritme Klasifikasi Pada Imbalanced Dataset." *Jurnal TEKNOINFO* 13 (1): 11–16.

Puspitawuri, Annisa, Edy Santoso, and Candra Dewi. 2019. "Diagnosis Tingkat Risiko Penyakit Stroke Menggunakan Metode K-Nearest Neighbor Dan Naïve Bayes." *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer* 3 (4): 3319–24. https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/4916.

Rais, Amin Nur, and Agus Subekti. 2019. "Integrasi SMOTE Dan Ensemble AdaBoost Untuk Mengatasi Imbalance Class Pada Data Bank Direct Marketing." *Jurnal Informatika* 6 (2): 278–85.

Ramadhini, Ariesta Zubiah, and L. S. Angliadi. 2011. "Gambaran Angka Kejadian Stroke Akibat Hipertensi Di Instalasi Rehabilitasi Medik Blu Rsup Prof . Dr . R . D . Kandou Manado Periode Januari – Desember 2011." *Jurnal E-CLINIC (ECL)* 1 (2).

Ramdhani, Lis Saumi. 2016. "Penerapan Particle Swarm Optmization (PSO) Untuk Seleksi Atribut Dalam Meningkatkan Akurasi Prediksi Diagnosis Penyakit Hepatitis Dengan Metode Algoritma C4 . 5." *Jurnal Ilmu Komputer, Manajemen Dan Sosial (SWABUMI)* IV (1): 1–15.

Rohmana, I, and R Arifudin. 2014. "Perbandingan Jaringan Syaraf Tiruan Dan Naive Bayes Dalam Deteksi Seseorang Terkena Penyakit Stroke." *Jurnal MIPA* 37 (1): 92–104.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

S, Jeena R, and Dr.Sukesh Kumar. 2016. " Stroke Prediction Using SVM." *2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, 600–602.

Saifudin, Aries, and Romi Satria Wahono. 2015. "Penerapan Teknik Ensemble Untuk Menangani Ketidakseimbangan Kelas Pada Prediksi Cacat Software." *Journal of Software Engineering* 1 (1).

Saputra, Raka Hendra. 2020. "Optimasi Algoritma C4.5 Menggunakan Seleksi Fitur Particle Swarm Optimization (PSO) Dan Teknik Bagging Pada Diagnosis Ppenyakit Kanker Payudara Skripsi."

Senthilkumar, Kailash, and Kanthimathinathan Ramasubramanian. 2020. "Stroke Predictions Using Healthcare Dataset." *International Journal Engineering Science and Computing* 10 (10): 27427–30.

Septiani, Wisti Dwi. 2017. "Komparasi Metode Klasifikasi Data Mining Algoritma C4.5 Dan Naive Bayes Untuk Prediksi Penyakit Hepatitis." *Journal of Computing and Information System* 13 (1): 76–84.

Setiawati, Intan, Adityo Permana Wibowo, and Arief Hermawan. 2019. "Implementasi Decision Tree Untuk Mendiagnosis Penyakit Liver." *Journal of Information System Management* 1 (1): 13–17. https://doi.org/10.24076/JOISM.2019v1i1.17.

Suwanwela, Nijasri C, Niphon Poungvarin, and ASAP. 2016. "Stroke Burden and Stroke Care System in Asia." *Publication of The Neurology Society of India* 64 (7): 46–51.

Tarigan, D. M., Dian Palupi Rini, and Samsuryadi. 2020. "Seleksi Fitur Pada Klasifikasi Penyakit Gula Darah Menggunakan Particle Swarm Optimization (PSO) Pada Algoritma C4.5." *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)* 4 (3): 569–75. https://doi.org/10.29207/resti.v4i3.1881.

Tran, Cao Truong, Mengjie Zhang, Peter Andreae, and Bing Xue. 2017a. "Bagging and Feature Selection for Classification with Incomplete Data." *European Conference on the Applications of Evolutionary* 10199 (April): 471–86.

———. 2017b. "Bagging and Feature Selection for Classification with

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id

Incomplete Data." *European Conference on the Applications of Evolutionary* 10199 (April): 471–86. https://doi.org/10.1007/978-3-319-55849-3.

Tu, My Chau, Dongil Shin, and Dongkyoo Shin. 2009a. "Effective Diagnosis of Heart Disease through Bagging Approach." *2009 2nd International Conference on Biomedical Engineering and Informatics*.

Tu, My Chau, Dongil Shin, and DongKyoo Shin. 2009b. "A Comparative Study of Medical Data Classification Methods Based on Decision Tree and Bagging Algorithms." *2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing*, 183–87. https://doi.org/10.1109/DASC.2009.40.

Tunjung, I Wayan. 2016. "Penerapan Case Based Reasoning (Cbr) Untuk Mendiagnosis Penyakit Stroke Menggunakan Algoritma K-Nearest Neighbor." *Conference on Information Technology, Information System and Electrical Engineering*, 23–24.

Wahono, Romi Satria, and N Suryana. 2013. "Combining Particle Swarm Optimization Based Feature Selection and Bagging Technique for Software Defect Prediction Combining Particle Swarm Optimization Based Feature Selection and Bagging Technique for Software Defect Prediction." *International Journal of Software Engineering and Knowledge* 7 (5): 153–66. https://doi.org/10.14257/ijseia.2013.7.5.16.

Wardhani, Novida Rizky, and Santi Martini. 2011. "Faktor Yang Berhubungan Dengan Pengetahuan Tentang Stroke Pada Pekerja Institusi Pendidikan Tinggi." *Jurnal Berkala Epidemiologi* 2 (1): 13–23.

Yulianti, Ita, Rizal Amegia Saputra, Muhammad Sukrisno Mardiyanto, Ami Rahmawati, Fakultas Teknologi Informasi, Universitas Bina, Sarana Informatika, Fakultas Teknologi Informasi, and Penyakit Ginjal Kronis. 2020. "Optimasi Akurasi Algoritma C4 . 5 Berbasis Particle Swarm Optimization Dengan Teknik Bagging Pada Prediksi Penyakit Ginjal Kronis" 19 (4): 411–21.

digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id digilib.uinsby.ac.id