

Klasifikasi Kemampuan Perawatan Diri Anak dengan Disabilitas Menggunakan *Neural Network dan Greedy Stepwise* Sebagai Seleksi Fitur

Sari Susanti

Universitas Adhirajasa Reswara Sanjaya
Jl. Sekolah Internasional No 1-6 Antapani, Bandung, Indonesia

e-mail: sarisusanti@ars.ac.id

Informasi Artikel

Diterima: 14-09-2020

Direvisi: 26-11-2020

Disetujui: 27-11-2020

Abstrak

Disabilitas merupakan gangguan, keterbatasan aktivitas dan pembatasan partisipasi. Disabilitas disebut juga interaksi antara individu dengan kondisi kesehatan seperti (Cerebral palsy, sindrom Down dan depresi), faktor pribadi dan lingkungan seperti sikap negatif. Disabilitas dapat mengganggu perkembangan alami tubuh tergantung pada jenis kelamin, usia dan lingkungan. Penderita disabilitas merupakan kelompok minoritas terbesar di dunia, 80% penderita berasal dari negara-negara berkembang. Selain itu Anak-anak menempati menyandang disabilitas dengan jumlah sepertiga dari jumlah keseluruhan penyandang disabilitas di dunia. Pada penerapannya proses diagnosis dan klasifikasi dimensi disabilitas membutuhkan ahli terapis okupasi. Teknik *data mining* dapat digunakan untuk membantu proses diagnosis yang bertujuan untuk menghindari kesalahan dalam diagnosis. Tujuan penelitian ini adalah untuk mengklasifikasikan masalah kemampuan perawatan diri anak disabilitas menjadi 7 kelas. Penelitian ini menggunakan dataset yang merepresentasikan masalah kemampuan perawatan diri anak dengan disabilitas. Dataset yang akan digunakan memiliki permasalahan *multidimensional dataset* dimana *dataset* memiliki fitur yang lebih banyak dibandingkan dengan jumlah datanya. *Multidimensional dataset* dilihat dari Jumlah fitur yang dimiliki yaitu 205 fitur dan 1 label dengan jumlah data sebanyak 70. Metode yang diusulkan pada penelitian ini yaitu *greedy stepwise* sebagai metode untuk mengatasi masalah *multidimensional dataset* dengan menyeleksi fitur bertujuan memilih fitur yang paling relevan. Selain *greedy stepwise* diterapkan juga metode *neural network* yang digunakan sebagai algoritme klasifikasi pada datamining yang dibuat. Implikasi dengan diterapkannya neural network dan *greedy stepwise* mampu memberikan hasil akurasi klasifikasi yang lebih baik jika dibandingkan dengan tidak menerapkan algoritma tersebut. Hasil penelitian menunjukkan bahwa metode seleksi fitur *greedy stepwise* dengan penerapan *neural network* memperoleh nilai akurasi sebesar 84.2857% yang bisa disimpulkan hasil akurasinya baik.

Kata kunci: Masalah Perawatan Diri, *Neural Network*, *Greedy stepwise* sebagai Seleksi Fitur.

Abstract

Disorders, activity limitations and percentage. Disability is also called the interaction between individuals with health conditions such as (Cerebral palsy, Down syndrome and depression), personal and environmental factors such as negative attitudes. Disabilities can interfere with the body's natural development depending on gender, age and environment. Persons with disabilities are the largest minority group in the world, 80% of sufferers come from developing countries. In addition, children with disabilities account for one third of the total number of persons with disabilities in the world. In its application, the process of diagnosis and dimensional classification requires an occupational therapist. Data mining techniques can be used to assist the diagnosis process in order to avoid errors in diagnosis. The purpose of this study was to classify self-care problems for children with disabilities into 7 classes. This study uses a dataset that represents self-care problems for children with disabilities. The dataset to be used has a multidimensional dataset where the dataset has more features than the amount of data. The multidimensional dataset is seen from the number of features it has, namely 205 features and 1 label with a total data of 70. The method proposed in this study is greedy stepwise as a method to solve the problem of multidimensional datasets by selecting features aimed at selecting the most relevant features. In addition to greedy stepwise, the neural network method is also applied which is used as a classification algorithm for the datamining that is made. The implication is that the application of neural networks and greedy stepwise is able to provide better classification results when compared



to not implementing the algorithm. The results showed that the feature selection method which was greedy with the application of neural networks reached an accuracy value of 84.2857% which could show good accuracy results.

Keywords: Self Care Problems, Neural Network, Greedy stepwise as Feature Selection.

1. Pendahuluan

Disabilitas merupakan sebuah kondisi berupa gangguan, keterbatasan aktivitas atau pembatasan partisipasi seseorang. Disabilitas merupakan juga interaksi antara individu (WHO, 2018). Disabilitas dapat mengganggu perkembangan alami tubuh tergantung pada jenis kelamin, usia dan lingkungan (Zarchi, Bushehri, & Dehghanizadeh, 2018). Data dari Kementerian kesehatan menyebutkan Penderita disabilitas 80% berasal dari negara-negara berkembang. Selain itu Anak-anak menempati menyandang disabilitas dengan jumlah sepertiga dari jumlah keseluruhan penyandang disabilitas di dunia (Kemenkes, 2014).

Data dari WHO sekitar 15% populasi di dunia antara 110 juta (2,2%) dan 190 juta (3,8%) dengan usia 15 tahun atau lebih mengalami kesulitan fungsionalitas yang signifikan (WHO, 2018). Klasifikasi dan pengkodean dimensi disabilitas pada anak-anak dan remaja merupakan proses kompleks. Proses ini membutuhkan pertimbangan keterbatasan fungsi tubuh, struktur tubuh, aktivitas dan partisipasi dalam perkembangan fisik, sosial dan psikologis (WHO, 2007).

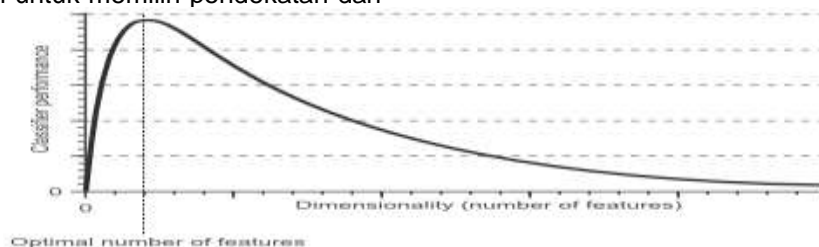
Proses diagnosis dan klasifikasi dimensi disabilitas pada implementasinya membutuhkan ahli terapis okupasi. Jumlah terapis okupasi yang terbatas mengakibatkan penanganan pasien disabilitas tertunda, akibatnya membuat pengobatan pasien menjadi lebih sulit dan mahal. Berdasarkan hal tersebut sistem klasifikasi masalah perawatan diri anak dengan disabilitas merupakan faktor penting dalam memilih pendekatan perawatan. Karena keragaman dan kompleksitas masalah perawatan diri dan kurangnya terapis okupasi, menggunakan sistem klasifikasi otomatis untuk masalah perawatan diri dapat membantu terapis okupasi untuk memilih pendekatan dan

pengobatan terbaik (Zarchi *et al.*, 2018) permasalahan tersebut yang menjadi alasan utama dilakukannya penelitian ini.

WHO memberikan solusi untuk mempermudah klasifikasi jenis perawatan untuk penderita disabilitas dengan kerangka kerja *The International Classification of Functioning, Disability and Health for Children and Youth (ICF-CY)* (Yeh, Hou, & Chang, 2012). *ICF-CY* merupakan kerangka konseptual klasifikasi multiguna yang banyak digunakan dan merupakan versi turunan pertama dari rujukannya *ICF* (WHO, 2007). Berdasarkan hal tersebut diperlukan suatu penelitian untuk mendapatkan model yang mampu mengklasifikasikan hasil (*ICF-CY*) dengan tingkat kesalahan klasifikasi yang minimal. Penelitian sebelumnya mengenai klasifikasi kemampuan perawatan diri anak disabilitas telah dilakukan (Zarchi *et al.*, 2018). Selain itu penelitian mengenai klasifikasi kemampuan perawatan diri anak dengan disabilitas yang fokus pada masalah ketidakseimbangan kelas (*imbalanced class*) sudah dilakukan (Susanti, 2019).

Masalah yang muncul pada penelitian sebelumnya (Zarchi *et al.*, 2018) yaitu, adanya *multidimensional dataset* dimana jumlah *record dataset* yang sedikit sedangkan jumlah fitur yang banyak, masalah *multidimensional dataset* dapat dilihat dari jumlah dataset yang berjumlah 70 data sedangkan fitur yang dimiliki yaitu sebanyak 205 fitur dengan 1 label.

Multidimensional dataset atau *high dimensionality dataset* merupakan bahasan yang menarik dalam bidang *machine learning* dikarenakan masalah tersebut dapat mempengaruhi kinerja algoritma *classifier* yang digunakan (Spruyt, 2014). Gambar 1 merupakan gambar dari *multidimensional dataset*.



Gambar 1. Multidimensional dataset

Pada gambar 1 Terlihat bahwa Seiring meningkatnya dimensi atau jumlah fitur, kinerja

algoritma *classifier* meningkat sampai jumlah fitur optimal tercapai. Lebih lanjut meningkatkan

dimensi tanpa meningkatkan jumlah sampel atau *record* dari *dataset training* menghasilkan penurunan kinerja pada algoritma *classifier*. Semakin kecil ukuran data *training*, semakin sedikit fitur yang harus digunakan. Jika pada *dataset* terdapat kondisi jumlah fitur lebih banyak dibandingkan dengan jumlah *record* sehingga harus dilakukan pemilihan fitur yang paling relevan untuk digunakan. Algoritma pemilihan fitur dan sering menggunakan heuristik (*greedy methods, best-first methods*, dan lain-lain) Untuk mencari jumlah dan kombinasi fitur yang optimal (Spruyt, 2014).

Seleksi fitur diterapkan pada dataset berdimensi tinggi dengan jumlah fitur yang banyak. Tujuan dari seleksi fitur adalah untuk memilih fitur terbaik atau yang paling relevan untuk digunakan sehingga dapat meningkatkan hasil akurasi dan performa pada model yang diusulkan (Mongkareng, Setiawan, & Permanasari, 2017). Seleksi fitur juga diterapkan dalam penelitian (Fitriyani & Wahono, 2015) untuk mengatasi data yang berdimensi tinggi atau fitur-fitur yang dianggap tidak memiliki kontribusi.

Selain penanganan masalah *multidimensional dataset* dengan menerapkan seleksi fitur, pada penelitian kali ini juga dilakukan klasifikasi dengan menggunakan metode *neural network (NN)*.

Metode klasifikasi dengan menggunakan metode *NN* sebelumnya telah digunakan oleh banyak peneliti untuk beberapa objek yang berbeda dari penelitian kali ini. Diantara beberapa penelitian yang menggunakan metode *NN* sebagai metode klasifikasinya adalah penelitian yang dilakukan oleh (Radhimeenakshi, 2018; Zarchi et al., 2018; Susanti, 2019).

Berdasarkan hal tersebut *Data Mining* menjadi solusi yang dapat digunakan untuk klasifikasi secara otomatis. *Data mining* merupakan alat bantu untuk mengoptimalkan data pada basis data yang berukuran besar, dengan spesifikasi tingkat kerumitan yang digunakan pada banyak aplikasi. *Data Mining* medis memiliki potensi besar untuk mempelajari pola-pola yang tersembunyi dalam kumpulan data medis. Pola atau data tersebut dapat digunakan untuk diagnosis klinis. Data dikumpulkan dan terintegrasi untuk membentuk sebuah sistem informasi rumah sakit. *Data Mining* menyediakan pendekatan *user-oriented* untuk penelitian pola-pola yang tersembunyi dalam data (Soni, Ujma, Dipesh, & Sunita, 2011).

Neural networks merupakan algoritma *Data Mining* yang dapat dimanfaatkan untuk membuat prediksi data medis. *neural network* mengadopsi kemampuan otak manusia yang

mampu memberikan stimulasi atau rangsangan (Suhartono, 2012). Alasan penggunaan *Neural network* adalah akurasi yang tinggi, toleransi terhadap *noise*, independensi dari asumsi prior, pola yang paralel, cocok untuk *dataset* yang memiliki label lebih dari dua kelas atau polinomial (Hermanto, 2010). Klasifikasi menggunakan *neural network* mendapatkan hasil akurasi tinggi untuk klasifikasi pada *dataset* yang memiliki label polinomial seperti memiliki tujuh kelas yang dilakukan pada penelitian (Ramdhani & Riana, 2017).

Tujuan dari penelitian ini adalah untuk membantu terapis okupasi dalam mengklasifikasikan kemampuan perawatan diri anak dengan kondisi disabilitas secara otomatis dan akurat.

Pada *dataset Scadi* setiap fitur adalah fitur *multidimensional* yang dibuat menggunakan kode berdasarkan kategori aktivitas perawatan diri. Pengkodean untuk setiap fitur seperti tercantum pada tabel 1 dan 2.

Tabel 1. Pengkodean Fitur

Code	Level-1	Level-2	Level-3	Level-4
ICF-CY Code =	Component Kode	10-99	0-9	0-9

Sumber Data: (Susanti, 2019)

Misalnya, kegiatan *toileting* seperti melakukan buang air kecil dengan tepat dikodekan dalam *ICF-CY* seperti berikut pada tabel 2

Tabel 2. Deskripsi Pengkodean Fitur

Code	Level-1	Level-2	Level-3	Level-4
d	d5	29	0	1
52901				

Sumber Data: (Susanti, 2019)

Tingkat pertama (level-1) mengklasifikasikan aktivitas umum. Tingkat ini termasuk empat kategori, yaitu Fungsi Tubuh, Struktur Tubuh, Aktivitas dan Partisipasi, dan Faktor Lingkungan. Kode Level-1 didefinisikan sebagai kombinasi angka dan huruf. Dalam *ICF-CY*, Perawatan mandiri adalah subbagian komponen Kegiatan dan Partisipasi dan dimulai dengan d5. Tingkat berikutnya (level-2, level-3, dan level-4) dikodekan oleh angka. Setiap level menjelaskan level sebelumnya dengan lebih jelas. Pada *dataset Scadi*, 29 aktivitas dipertimbangkan untuk perawatan diri berdasarkan kerangka *ICF-CY*. 7 kode dibuat untuk menggambarkan tingkat penurunan aktivitas dan komponen partisipasi. Tabel 3 menunjukkan kode batas kerusakan *Scadi* (Zarchi et al., 2018).

Masalah perawatan diri dikategorikan ke dalam tujuh kelompok oleh terapis okupasi seperti yang ditunjukkan pada Tabel 4. Kelas sasaran pada *dataset scadi* mengacu pada masalah perawatan diri yang ditentukan oleh para ahli. Setiap anak diklasifikasikan ke salah satu dari 7 kategori berdasarkan fitur perawatan diri dengan pemeriksaan terapis okupasi.

Tabel 3. Kode Gangguan *Dataset Scadi*

Code	Value	Description
xxx.0	0-4%	NO impairment
xxx.1	5-24%	MILD impairment
xxx.2	25-49%	MODERATE impairment
xxx.3	50-95%	SEVERE impairment
xxx.4	96-100%	COMPLETE impairment
xxx.8		not specified
xxx.9		not applicable

Sumber Data: (Susanti, 2019)

Dataset SCADI memiliki 7 kelas yang menjadi target atau label, untuk data 7 kelas akan dijelaskan pada tabel 4.

Tabel 4. Penjelasan 7 kelas

Kelas	Deskripsi
1	<i>Caring for body parts problem</i>
2	<i>Toileting problem</i>
3	<i>Dressing problem</i>
4	<i>Washing oneself and Caring for body parts and Dressing problem</i>
5	<i>Washing oneself, Caring for body parts, Toileting, and Dressing problem</i>
6	<i>Eating, Drinking, Washing oneself, Caring for body parts, toileting, Dressing, Looking after one's health and Looking after one's safety problem</i>
7	<i>No Problem</i>

Sumber Data: (Susanti, 2019)

2. Metode Penelitian

Penelitian ini mengusulkan metodologi berdasarkan paradigma *data mining*. Paradigma ini mengintegrasikan algoritma *greedy stepwise*, dengan algoritma klasifikasi *neural network*. Dataset yang digunakan pada penelitian ini sebanyak 70 dataset dengan memiliki 205 fitur.

Greedy stepwise

Algoritma *greedy* dapat disebut juga dengan *stepwise* dan weka menggunakan algoritma ini dengan sebutan *stepwise* (Fitriyani, 2015). Algoritma *greedy* merupakan algoritma *metaheuristic* untuk menyelesaikan permasalahan dengan pendekatan mengidentifikasi lokal optimum dan mengarah pada global optimum Algoritma *greedy* merupakan metode yang paling populer untuk memecahkan persoalan optimasi (*optimization problems*) atau dapat disebut mencari solusi optimum (Spruyt, 2014).

Algoritma *greedy* dapat diartikan sebagai rakus, tamak, dan lain-lain dengan menggunakan prinsip "*take what you can get now*". Algoritma ini membentuk solusi langkah per langkah (*step by step*). Pada setiap langkahnya terdapat banyak pilihan yang perlu dieksplorasi dan dari setiap langkah tersebut

harus dibuat keputusan yang terbaik dalam menentukan pilihan. Dalam setiap langkah, algoritma ini membuat pilihan optimum lokal untuk dapat mengarah pada solusi optimum global. Contoh pemecahan masalah penukaran uang menggunakan algoritma *greedy* (Fitriyani, 2015):

- Koin: 2, 3, 1, 4
- Uang yang ditukar: 10
- Solusi *greedy*: pada setiap langkah, memilih koin dengan nilai terbesar dari himpunan koin
- Langkah 1: 2+4+4 (total 3 koin)
- Langkah 2: 2+2+3+3 (total 4 koin)
- Langkah 3: 2+1+1+4+2 (total 5 koin)
Langkah 3 merupakan solusi yang digunakan oleh *greedy*.

Algoritma *Greedy stepwise* bekerja dengan menemukan fitur terbaik, fitur yang terbaik atau yang paling relevan mempunyai jumlah dimensi yang paling berkontribusi pada akurasi. Seleksi fitur untuk menentukan fitur terbaik dan terburuk (Han, Kamber, & Pei, 2012).

Algoritma *greedy* dengan seleksi subset atribut (Han et al., 2012) sebagai berikut untuk penjelasannya:

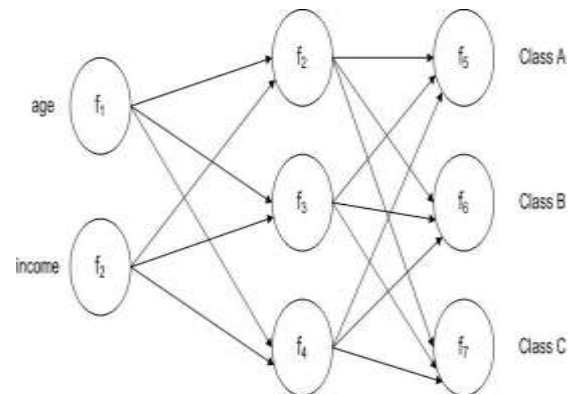
1. *Stepwise forward selection* Prosedur dimulai dengan himpunan kosong dari atribut sebagai set yang dikurangi, atribut yang terbaik dari atribut asli ditentukan dan ditambahkan pada set yang kurang. Pada setiap iterasi berikutnya yang terbaik dari atribut asli yang tersisa ditambahkan ke set.
2. *Stepwise backward elimination* Prosedur dimulai dengan *full set* atribut. Pada setiap langkah, teknik ini dapat menghilangkan atribut terburuk yang tersisa di *dataset*.
3. Kombinasi *forward selection* dan *backward elimination* *Stepwise forward selection* dan *backward elimination* dapat dikombinasikan. Pada setiap langkah, prosedur memilih atribut terbaik dan menghilangkan yang terburuk dari atribut yang tersisa.

Neural network

Neural network atau jaringan saraf adalah satu set unit *input/output* yang terhubung, dimana tiap koneksinya memiliki bobot. Selama fase pembelajaran, *Neural network* menyesuaikan bobot sehingga dapat memprediksi *class* yang benar dari *tupple* (Han & Kamber, 2006). Algoritma *Neural network* yang paling populer adalah *Backpropagation*, algoritma *backpropagation* melakukan pembelajaran pada jaringan saraf *multi layer feed forward* yang terdiri dari tiga lapisan/*layer*, yaitu: lapisan *input*, lapisan tersembunyi, dan lapisan keluaran (Han & Kamber, 2006). Pada beberapa diagram *Neural network* dimungkinkan terdapat lebih dari satu lapisan tersembunyi, meskipun kebanyakan hanya mengandung satu lapisan tersembunyi yang dirasa cukup untuk berbagai tujuan (Larose, 2006).

Backpropagation merupakan algoritma *Neural network* untuk klasifikasi yang menggunakan *gradient descent*, *backpropagation* mencari satu set bobot yang dapat memodelkan data sehingga dapat meminimalkan jarak kuadrat rata-rata antara prediksi kelas jaringan dan label kelas yang sebenarnya dari *tuple* data (Han & Kamber, 2006).

Tiap observasi data *training* diproses melalui jaringan, nilai *output* dihasilkan dari node *input*. Nilai *output* ini kemudian dibandingkan dengan nilai aktual dari variabel target dan dihitung error yang dihasilkan (Larose, 2006). *Backpropagation* melakukan proses pembelajaran secara *iteratif* yang mencoba untuk meminimalkan *error* dari klasifikasi.



Gambar 2. Multi Layer Feed Forward Neural network (Han & Kamber, 2006)

Proses pembelajaran dalam *backpropagation* dilakukan dengan penyesuaian bobot-bobot *neuron* dengan arah mundur berdasarkan nilai *error* dalam proses pembelajaran (Kusrini & Luthfi, 2009). Untuk mendapatkan *error* maka tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu, pada saat perambatan maju maka *neuron-neuron* diaktifkan dengan menggunakan fungsi aktivasi yang dapat dideferensiasikan, seperti fungsi sigmoid (Kusumadewi, 2004).

Dalam setiap proses, bobot relasi dalam jaringan dimodifikasi untuk meminimalkan nilai *Mean Squared Error (MSE)* antara nilai prediksi dari jaringan dengan nilai sesungguhnya. Modifikasi relasi jaringan tersebut dilakukan dalam arah mundur (Kusrini & Luthfi, 2009).

Langkah pembelajaran dalam metode *backpropagation* adalah sebagai berikut (Myatt, 2007):

1. Inisialisasi semua bobot jaringan secara acak. (Biasanya antara -1.0 hingga 1.0).
2. Untuk setiap data *training*, hitung input untuk simpul berdasarkan nilai *input* dan bobot jaringan saat itu. Dengan menggunakan rumus :

$$I_j = \sum_i w_{ij} O_i + \theta_j \quad (2.3)$$

Keterangan :

O_i = *Output* simpul *i* dari layer sebelumnya

w_{ij} = bobot relasi dari simpul *i* pada *layer* sebelumnya ke simpul *j*

θ_j = bias (sebagai pembatas)

3. Berdasarkan *input* dari langkah kedua, selanjutnya membangkitkan *output* untuk simpul menggunakan fungsi aktifitas *sigmoid*

$$Output = \frac{1}{1 + \theta^{-input}} \quad (2.4)$$

4. Hitung nilai error antara nilai yang diprediksi dengan nilai yang sesungguhnya menggunakan rumus :

$$Error_i = output_i * (1 - Output_i) * (Target_i - Output_i) \quad (2.5)$$

Keterangan :

$Output_i$ = $Output$ actual dari simpul i

$Target_i$ = nilai target yang sudah diketahui pada data *training*

- Setelah nilai *error* dihitung, selanjutnya dibalik ke *layer* sebelumnya (*backpropagation*). Untuk menghitung nilai *error* pada *hidden layer*, menggunakan rumus.

$$Error_i = Output_i * (1 - Output_i) * \sum_{j=1}^n Error_j W_{ij} \quad (2.6)$$

Keterangan :

$Error_i$ = *Error* yang dihasilkan dari *hidden node*

$Output_i$ = Nilai *output* yang dari *hidden node*

$Error_j$ = *Error* yang dihasilkan dari simpul j yang terhubung ke *output*

W_{ij} = Bobot pada simpul i ke simpul j

- Nilai *error* yang dihasilkan dari langkah sebelumnya digunakan untuk memperbarui bobot relasi, dengan menggunakan rumus:

$$W_{ij} = W_{ij} + l * Error_j * Output_i \quad (2.7)$$

Keterangan :

W_{ij} = bobot relasi dari unit i pada *layer* sebelumnya ke unit j

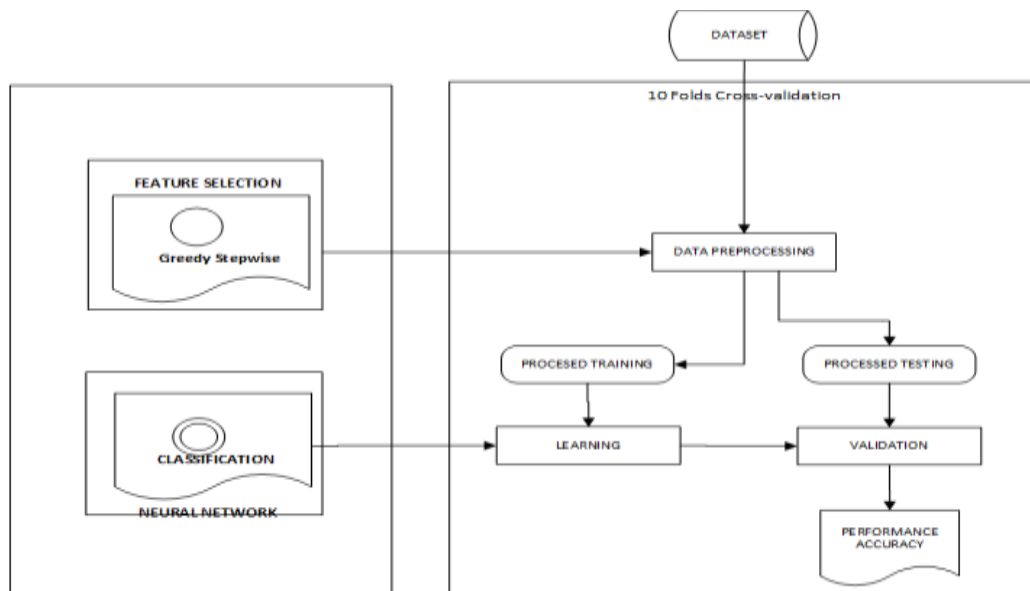
l = *learning rate* (konstantan, nilainya 0 sampai dengan 1)

$Error_j$ = *Error* pada *output layer* simpul j

$Output_i$ = *Output* dari simpul i .

Metode yang diusulkan

Metode yang diusulkan pada penelitian ini adalah *Greedy stepwise* untuk menangani masalah *multidimensional dataset* atau penerapan seleksi fitur dengan metode klasifikasi *neural network*. Tujuan seleksi fitur adalah untuk pengurangan fitur, untuk menghilangkan dari dataset subset dari variabel yang tidak dianggap relevan untuk tujuan dari kegiatan Data Mining. Seleksi fitur masuk ke tahap *Preprocessing*. Tahap selanjutnya *Dataset* yang sudah diseleksi fitur kemudian diklasifikasikan menggunakan algoritma *neural network* dengan menggunakan *10-cross validation* menggunakan *Software WEKA v.3.8*. Kemudian hasil evaluasi model akan dibandingkan dengan *confusion matrix* sebelum dilakukan seleksi fitur. Metode yang diusulkan dapat dilihat pada gambar 3.



Gambar 3. Kerangka Pemikiran/Model Penelitian

Analisis desain pada penelitian ini dijelaskan pada Gambar 3. Dimana terdaat beberapa tahapan dimulai dengan melakukan tahap *preprocessing* pada dataset dengan menggunakan *greedy stepwise* untuk menyeleksi fitur yang paling relevan. Setelah fitur di seleksi dengan *greedy stepwise* langkah selanjutnya adalah diklasifikasi dengan menggunakan *10 cross-validation* (iterasi 10 kali). Dimana dataset otomatis dibagi menjadi 90% data training dan 10% testing. Selajutnya diterapkan algoritma *Neural network* untuk

mengklasifikasikan dataset menjadi kelas 1-7. Hasil kinerja pada model ini akan diukur dengan menggunakan akurasi. Proses tahapan penelitian dilakukan dengan menggunakan *software Weka*.

3. Hasil dan Pembahasan

Dalam penelitian ini dilakukan eksperimen terhadap pengklasifikasian *data mining* menggunakan algoritma *Neural network*. Pemodelan akan dilakukan dengan menggunakan *software WEKA*.

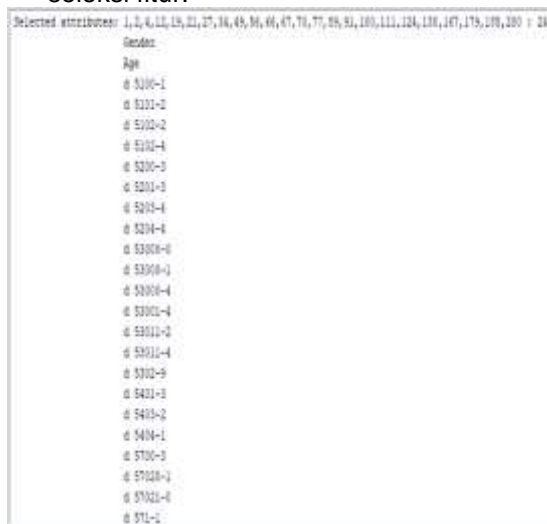
Tahap *modeling* dengan menggunakan *software* WEKA dapat dilakukan dengan cara sebagai berikut:

1. Menyiapkan *dataset* yang akan diproses.
2. Buka *software* Weka, Pilih menu explorer seperti terlihat di gambar 4.



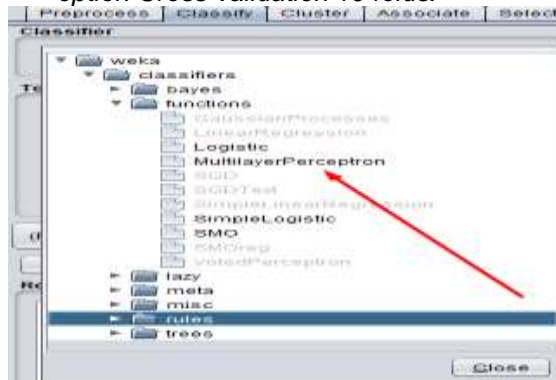
Gambar 4. Memilih menu *Explorer*

3. Pilih *button open file* untuk mengambil *dataset* yang akan dilakukan pemodelan data mining
4. Tahap selanjutnya pilih *filter* di menu *preprocess*. Step1 pilih *supervised*, step 2 pilih *instance*.
5. Setelah itu pilih *start* untuk melakukan seleksi fitur.



Gambar 5. fitur Hasil Seleksi *Greedy Stepwise*

6. Tahap selanjutnya lakukan klasifikasi dengan memilih menu *classify*.
7. Pilih Algoritma klasifikasi *neural network*, kemudian klik *start* untuk menampilkan model dan hasil akurasi dari *neural network*. Dengan menggunakan *test option Cross-validation 10 folds*.



Gambar 6. Memilih Algoritma Klasifikasi

Hasil pada penelitian ini didapatkan dengan melakukan eksperimen terhadap algoritma klasifikasi *data mining* yaitu *neural network* dan dua algoritma pembandingan yaitu *decision tree*, dan *logistic regression* dengan menggunakan *software weka*. Eksperimen dilakukan dengan menggunakan *Scadi datasets* yang berisi data kemampuan perawatan diri anak dengan disabilitas.

3.1. Hasil Eksperimen Menggunakan *Neural Network*

Eksperimen pada *dataset scadi* menggunakan algoritma *neural network*. Eksperimen dilakukan terhadap 70 *dataset* yang memiliki 205 fitur dan 1 label. Tahap selanjutnya dilakukan pemisahan terhadap *dataset training* dan *testing* menggunakan metode *10 folds cross-validation*.

Tabel 5.
Confusion matrix klasifikasi menggunakan *NN*

	True Class1	True Class2	True Class3	True Class4	True Class5	True Class6	True Class7	Class Precision
Prediction Class 1	0	0	0	1	0	0	1	0.00%
Prediction Class 2	0	6	0	1	0	0	0	85.7%
Prediction Class 3	0	0	0	0	0	0	1	0.00%
Prediction Class 4	1	0	0	10	1	1	2	66.7%
Prediction Class 5	0	0	0	0	0	0	0	0.00%
Prediction Class 6	0	0	0	0	2	28	0	93.3%
Prediction Class 7	1	1	1	0	0	0	12	80.0%
Class Recall	0.00%	85.7%	0.00%	83.3%	0.00%	96.6%	75%	

$$Akurasi = \frac{T1 + T2 + T3 + T4 + T5 + T6 + T7}{T1 + T2 + T3 + T4 + T5 + T6 + T7 + F1 + F2 + F3 + F4 + F5 + F6 + F7}$$

$$Akurasi = \frac{0 + 6 + 0 + 10 + 0 + 28 + 12}{0 + 6 + 0 + 10 + 0 + 28 + 12 + 2 + 1 + 1 + 2 + 3 + 1 + 4}$$

$$Akurasi = \frac{56}{70} = 0.8 * 100 = 80\%$$

Keterangan:

T= Kelas prediksi benar

F= Kelas prediksi salah

Tabel 5. menampilkan *confusion matrix* hasil klasifikasi *neural network*. Akurasi yang didapatkan sebesar **80.00%** untuk klasifikasi 7 kelas. Hasil untuk kelas 1, kelas 3 dan kelas 5 memiliki nilai *recall* dan *precision* paling rendah 0% dikarenakan jumlah data pada kelas tersebut sangat sedikit dibandingkan kelas lainnya.

Tabel 6 Hasil Eksperimen klasifikasi

NO	Algoritma	Nilai Akurasi
1	<i>Neural network</i>	80,00%
2	<i>Decision Tree (J48)</i>	80,00%
3	<i>Logistic Regression</i>	80,00%

Tabel 6. Merupakan hasil akurasi klasifikasi kemampuan perawatan diri anak dengan disabilitas. Dimana hasil akurasi untuk ketiga algoritma yaitu *neural network*, *decision tree* dan *logistic regression* sebesar 80% yang menyimpulkan tidak ada perbedaan hasil akurasi antara satu algoritma dengan yang lainnya.

3.2. Hasil Eksperimen Menggunakan *Neural Network* dan *Greedy stepwise*

Seleksi fitur diterapkan untuk mengatasi masalah dimensional *dataset* dimana jumlah fitur yang dimiliki lebih banyak dibandingkan dengan jumlah *record*. Bagian ini menampilkan hasil eksperimen terhadap *dataset* yang sudah dilakukan seleksi fitur menggunakan algoritma *greedy stepwise*. Jumlah fitur sebelum diseleksi sebanyak 205 fitur sedangkan yang telah terseleksi sebanyak 19 fitur seperti pada tabel 7.

Tabel 7. Hasil Seleksi Fitur Awal

d 5100-1	d 5102-2	d 5200-3	d 5201-2	d 5201-3
d 5203-4	d 5204-4	d 53000-0	d 53000-4	d 53001-4
d 53011-4	d 5400-4	d 5403-2	d 5501-1	d 5602-2
d 5701-3	d 57020-0	d 571-0	d 571-1	Label

Setelah dilakukan seleksi fitur selanjutnya dilakukan klasifikasi menggunakan algoritma *neural network*. Model *neural network* yang dihasilkan setelah diseleksi fitur

terdiri dari 19 *node input*, 13 *node hidden layer* dengan 1 bias dan 7 *node output* dengan 1 *threshold*.

Tabel 8. *Confusion matrix* klasifikasi menggunakan *NN+GS*

	True Class1	True Class2	True Class3	True Class4	True Class5	True Class6	True Class7	Class Precision
Prediction Class 1	0	0	0	1	0	0	0	0.00%
Prediction Class 2	0	7	0	1	0	0	0	87.5%
Prediction Class 3	0	0	0	0	0	0	1	0.00%
Prediction Class 4	1	0	0	9	1	0	1	75%
Prediction Class 5	0	0	0	0	0	0	0	0.00%
Prediction Class 6	0	0	0	0	2	29	0	93.5%
Prediction Class 7	1	0	1	1	0	0	14	82.4%
Class Recall	0.00%	100%	0.00%	75%	0.00%	100%	87.5%	

$$Akurasi = \frac{T1 + T2 + T3 + T4 + T5 + T6 + T7}{T1 + T2 + T3 + T4 + T5 + T6 + T7 + F1 + F2 + F3 + F4 + F5 + F6 + F7}$$

$$Akurasi = \frac{0 + 7 + 0 + 9 + 0 + 29 + 14}{0 + 7 + 0 + 9 + 0 + 29 + 14 + 2 + 0 + 1 + 3 + 3 + 0 + 2}$$

$$Akurasi = \frac{59}{70} = 0.84285714 * 100 = 84.285714\%$$

Tabel 8. menampilkan *confusion matrix* hasil klasifikasi *neural network* dengan menggunakan seleksi fitur pada *dataset* awal. Akurasi yang didapatkan sebesar **84.2857%** untuk klasifikasi 7 kelas. Berdasarkan hal tersebut pada tahap selanjutnya akan dilakukan seleksi fitur pada *dataset* yang sudah dilakukan *resampling*.

Tabel 9. Hasil Eksperimen klasifikasi dengan optimasi *greedy stepwise*

NO	Algoritma	Tanpa Optimasi	Optimasi GS
1	<i>Neural network</i>	80,00%	84.2857 %
2	<i>Decision Tree (J48)</i>	80,00%	82.8571 %
3	<i>Logistic Regression</i>	80,00%	78.5714 %

Tabel 9. merupakan hasil akurasi klasifikasi dengan menggunakan algoritma klasifikasi *neural network*, *decision tree* dan *logistic regression* dengan menambahkan optimasi yaitu *greedy stepwise* sebagai seleksi fitur, dimana hasil akurasi mengalami peningkatan untuk semua algoritma yang diujikan. Adapun *neural network* memiliki hasil akurasi yang paling tinggi dibandingkan algoritma *decision tree* dan *logistic regression* sebesar **84.2857%**.

Berdasarkan hasil percobaan yang tertera pada Tabel 9 dilakukan analisa uji statistik uji beda *Paired Sample t-Test*, hasil yang diperoleh dicantumkan pada Tabel 10.

Tabel 10. Hasil Uji T-test Algoritma klasifikasi Tanpa Optimasi dengan Optimasi GS
t-Test: Paired Two Sample for Means

	Tanpa Optimasi	Optimasi GS
<i>Mean</i>	80	81.90473333
<i>Variance</i>	0	8.843557823
<i>Observations</i>	3	3
<i>t Stat</i>	-1.109382471	
<i>P(T<=t) one-tail</i>	0.191396386	
<i>t Critical one-tail</i>	2.91998558	
<i>P(T<=t) two-tail</i>	0.382792772	
<i>t Critical two-tail</i>	4.30265273	

Berdasarkan hasil uji t dua sampel berpasangan pada Tabel 4.22 dapat diambil hipotesa berdasarkan perbandingan t hitung dan t tabel, juga berdasarkan nilai probabilitas. Nilai t hitung diwakili oleh t stat sebesar -1.109382471 dan nilai t tabel diwakili oleh t *critical two-tail* sebesar 4.30265273, maka dapat dipastikan nilai t hitung < t tabel dengan arah *negative* yang artinya H₁ ditolak dan H₀ diterima, sedangkan diketahui nilai probabilitas sebesar 0.382792772, maka dapat dipastikan bahwa nilai probabilitas > 0,05 yang artinya tidak terdapat perbedaan yang signifikan dari rata-rata akurasi metode *neural network* dengan *neural network* yang dikombinasikan *greedy stepwise*.

3.3. Alur Perhitungan Manual Neural Network

Perhitungan manual *neural network* dilakukan dengan langkah-langkah sebagai berikut:

1. Tentukan nilai bobot untuk setiap simpul. Pada penelitian ini nilai bobot setiap simpul diambil dari model klasifikasi dengan menggunakan *software* weka.

2. Langkah selanjutnya dilakukan normalisasi data dengan *range* nilai 0-1 untuk semua atribut.

$$\text{Normalisasi} = ((\text{data} - \text{min}) / (\text{max} - \text{min}))$$

Berikut dilakukan normalisasi untuk salah satu atribut.

$$\text{Normalisasi} = ((2 - 1) / (2 - 1))$$

$$\text{Normalisasi} = 1$$

Setelah semua nilai dilakukan normalisasi langkah selanjutnya adalah melakukan inialisasi input, bobot dan bias awal.

3. Hitung total bobot untuk masing-masing simpul *hidden layer*.

$$\begin{aligned} \Sigma w_{i1} &= (n_1 * w_{i1.1}) + (n_2 * w_{i1.2}) + (n_3 * w_{i1.3}) + (n_4 * w_{i1.4}) + (n_5 * w_{i1.5}) + \\ &\quad (n_5 * w_{i1.6}) + (n_5 * w_{i1.7}) + (n_5 * w_{i1.8}) + (n_5 * w_{i1.9}) + (n_5 * w_{i1.10}) + \\ &\quad (n_5 * w_{i1.11}) + (n_5 * w_{i1.12}) + (n_5 * w_{i1.13}) + (n_5 * w_{i1.14}) + (n_5 * w_{i1.15}) + \\ &\quad (n_5 * w_{i1.16}) + (n_5 * w_{i1.17}) + (n_5 * w_{i1.18}) + (n_5 * w_{i1.19}) + (n_5 * w_{i1.20}) + \\ &\quad (n_5 * w_{i1.21}) + (n_5 * w_{i1.22}) + (n_5 * w_{i1.23}) + (n_5 * w_{i1.24}) + \theta_j \\ \Sigma w_{i1} &= (0 * 0.55) + (0.75 * 0.54) \\ &\quad + (0 * -0.10) + (0 * -0.89) \\ &\quad + (0 * -0.13) + \end{aligned}$$

$$\begin{aligned} & (1 * -0.13) + (0 * 1.39) + (0 * 1.41) + \\ & (1 * -0.10) + (1 * -0.13) + (0 * -0.72) + \\ & (1 * 0.58) + (0 * -0.14) + (1 * \\ & -0.17) + (0 * 0.20) + (1 * -0.18) + (1 * \\ & -2.00) + (0 * 1.45) + (0 * 0.08) + (1 * \\ & -1.97) + (0 * 0.25) + (1 * -0.79) + (0 * \\ & 0.28) + (0 * 0.30) + (0.03) \\ & \Sigma wi1 = -4.453006534 \end{aligned}$$

Keterangan:

$\Sigma wi1$ merupakan total nilai bobot input untuk simpul *hidden layer* pertama

θ_j merupakan bobot bias

Lakukan perhitungan tersebut untuk setiap simpul *hidden layer* sampai dengan simpul ke 15.

4. Aktifasi fungsi *sigmoid* untuk masing-masing simpul *hidden layer* dengan menggunakan rumus berikut:

$$\text{Sigmoid} = \frac{1}{1 + \exp(-1 * \Sigma wi1)}$$

$$\text{Sigmoid} = \frac{1}{1 + \exp(-1 * -4.453006534)}$$

$$\text{Sigmoid} = 0.011509497$$

Keterangan : exp adalah nilai eksponensial

Lakukan aktifasi fungsi sigmoid untuk setiap simpul *hidden layer*.

5. Hitung nilai bobot simpul *output*

$$\begin{aligned} \Sigma wo1 &= ((\Sigma wi1 * wo1.1) + (\Sigma wi2 * wo1.2) + \\ & (\Sigma wi3 * wo1.3) + (\Sigma wi4 * wo1.4) + \\ & (\Sigma wi5 * wo1.5) + (\Sigma wi6 * wo1.6) + \\ & (\Sigma wi7 * wo1.7) + (\Sigma wi8 * wo1.8) + \\ & (\Sigma wi9 * wo1.9) + (\Sigma wi10 * wo1.10) + \\ & (\Sigma wi11 * wo1.11) + (\Sigma wi12 * wo1.12) + \\ & (\Sigma wi13 * wo1.13) + (\Sigma wi14 * \\ & wo1.14) + (\Sigma wi15 * wo1.15) + (\theta_j)) \\ \Sigma wo1 &= ((0.011509497 * -0.90) + \\ & (0.614785228 * -1.82) + (0.668276994 * \\ & 0.37) + (0.010172982 * -1.37) + \\ & (0.914862496 * 1.67) + (0.255376538 * \\ & -3.51) + (0.085784572 * -1.46) + \\ & (0.987601869 * 2.03) + (0.817451557 * \\ & 0.71) + (0.983181773 * 1.54) + \\ & (0.831222786 * 0.65) + (0.999319007 * \\ & 0.51) + (0.3472808 * -2.68) + \end{aligned}$$

$$\begin{aligned} & (0.007502722 * -1.84) + (0.577496927 * \\ & 0.60) + (-0.82) \\ & \Sigma wo1 = 3.334927628 \end{aligned}$$

Keterangan:

$\Sigma wo1$ merupakan nilai bobot *output* keseluruhan untuk simpul *output* pertama.

Lakukan perhitungan nilai bobot untuk setiap simpul *output*.

6. Aktifasi fungsi *sigmoid* untuk masing-masing *output layer* dengan menggunakan rumus berikut:

$$\text{Sigmoid} = \frac{1}{1 + \exp(-1 * \Sigma wo1)}$$

$$\text{Sigmoid} = \frac{1}{1 + \exp(-1 * 3.334927628)}$$

$$\text{Sigmoid} = 0.965607789$$

Keterangan : exp adalah nilai eksponensial

Lakukan aktifasi fungsi *sigmoid* untuk setiap simpul *output layer*.

7. Hasil Klasifikasi

$IF(\max(\text{Sigmoid}1:\text{Sigmoid}7)=0.965607789, \text{Class}6,$

$IF(\max(\text{Sigmoid}1:\text{Sigmoid}7)=0.009806447, \text{Class}2,$

$IF(\max(\text{Sigmoid}1:\text{Sigmoid}7)=0.000252629, \text{Class}4,$

$IF(\max(\text{Sigmoid}1:\text{Sigmoid}7)=0.000667521, \text{Class}7,$

$IF(\max(\text{Sigmoid}1:\text{Sigmoid}7)=6.99431E05, \text{Class}1,$

$IF(\max(\text{Sigmoid}1:\text{Sigmoid}7)=0.012093192, \text{Class}5,$

$IF(\max(\text{Sigmoid}1:\text{Sigmoid}7)=0.000228766, \text{Class}3))))))$

Hasil Klasifikasi = Class 6

3.4. Perbandingan Nilai akurasi klasifikasi sebelum dan sesudah diterapkan seleksi fitur



Gambar 7. Grafik Perbandingan Performa NN dengan (NN+Greedy Stepwise)

Pada gambar 7. terlihat perubahan nilai nilai akurasi klasifikasi tanpa menerapkan seleksi fitur dengan ditambahkan seleksi fitur. bahwa menerapkan seleksi fitur dapat memperbaiki performa kinerja algoritma klasifikasi dengan meningkatkan nilai akurasi. Berdasarkan gambar 7 terlihat bahwa algoritma klasifikasi NN memiliki nilai yang paling besar 80,2857% dibandingkan dengan algoritma klasifikasi lainnya.

4. Kesimpulan

Penelitian ini mengkombinasikan algoritma *greedy stepwise* pada algoritma *classifier neural network* untuk mengatasi masalah *multidimensional dataset* pada *dataset Scadi*. Berdasarkan hasil eksperimen pada penelitian ini, maka dapat ditarik kesimpulan dengan penerapan seleksi fitur *greedy stepwise* dapat memperbaiki kinerja algoritma *neural network* pada *dataset Scadi*, untuk mengklasifikasikan tujuh kelas dengan nilai akurasi sebesar **84.2857%** jika dibandingkan sebelum menggunakan *greedy stepwise* dengan nilai akurasi **80%**.

Implikasi yang dapat diberikan dari penelitian ini adalah kedepannya dapat membantu terapis okupasi dalam mengklasifikasikan kemampuan perawatan diri anak dengan kondisi disabilitas secara otomatis. Sehingga proses perawatan yang diberikan dapat disesuaikan dengan kondisi setiap anak. Untuk mencapai hal tersebut selanjutnya akan dibuatkan aplikasi, sebagai luaran lanjutan dari penelitian ini.

Saran

Penerapan seleksi fitur menggunakan *greedy stepwise* telah terbukti dapat meningkatkan kinerja *Neural network* pada *dataset Scadi* yang memiliki masalah *multidimensional dataset*, namun beberapa hal dapat diterapkan untuk menyempurnakan penelitian ini, diantaranya:

1. Penelitian selanjutnya dapat menerapkan optimasi seleksi fitur untuk dengan menggunakan algoritma lainnya seperti PSO atau GA untuk mengatasi masalah *multidimensional dataset*.
2. Penelitian selanjutnya dapat menggunakan pendekatan level algoritma untuk menangani masalah ketidakseimbangan kelas (imbalanced class).
3. Penelitian selanjutnya dapat menggunakan algoritma klasifikasi *Naive bayes*, *Random Forest*, *k-NN* untuk mendapatkan hasil akurasi yang lebih baik.

Referensi

Fitriyani. (2015). *Integrasi Bagging Dan Greedy*

Forward Selection Pada Prediksi Cacat Software Menggunakan Naive Bayes. STMIK Nusa Mandiri.

- Fitriyani, & Wahono, R. S. (2015). Integrasi Bagging dan Greedy Forward Selection pada Prediksi Cacat Software dengan Menggunakan Naive Bayes. *Journal of Software Engineering*, 1(2).
- Han, J., & Kamber, M. (2006). *Data Mining Concepts and Techniques Second Edition*. San Francisco: Diane Cerra.
- Han, J., Kamber, M., & Pei, J. (2012). *Data Mining Concepts and Techniques*. San Fransisco: Morgan Kauffman.
- Hermanto, R. R. (2010). *Neural Network dan Implementasinya Dalam Data Mining*.
- Kemenkes. (2014). *Penyandang Disabilitas Pada Anak*. Pusat Data dan Informasi Kementrian Kesehatan RI.
- Kusrini, & Luthfi, E. T. (2009). *Algoritma Data Mining*. Yogyakarta: Penerbit Andi.
- Kusumadewi, S. (2004). *Membangun Jaringan Syaraf Tiruan Menggunakan Matlab & Excel Link*. Yogyakarta: Graha Ilmu.
- Larose, D. T. (2006). *Discovering Knowledge in Data: An Introduction to Data Mining*. New Jersey: John Wiley & Sons, Inc.
- Mongkareng, D., Setiawan, N. A., & Permanasari, A. E. (2017). Implementasi Data Mining dengan Seleksi Fitur untuk Klasifikasi Serangan pada Intrusion Detection System (IDS). *Citee*, (gambar 2), 314–321.
- Myatt, G. J. (2007). *Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining*. New Jersey: John Wiley & Sons, Inc.
- Ramdhani, Y., & Riana, D. (2017). Hierarchical Decision Approach based on Neural Network and Genetic Algorithm method for single image classification of Pap smear. In *In Informatics and Computing (ICIC), 2017 Second International Conference on* (pp. 1–6). IEEE.
- Soni, J., Ujma, A., Dipesh, S., & Sunita, S. (2011). Predictive Data Mining for Medical Diagnosis: An Overview of Heart Disease Prediction. *International Journal of Computer Applications*, 17(8), 43–48. Retrieved from <https://pdfs.semanticscholar.org/fbd6/5a18f6653b56138cd5196d20e2f39de189e3.pdf>
- Spruyt, V. (2014). The Curse of Dimensionality in classification. Retrieved April 1, 2018, from <http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/>
- Suhartono, D. (2012). *Dasar Pemahaman*

- Neural Network. Retrieved July 1, 2018, from <http://socs.binus.ac.id/2012/07/26/konsep-neural-network/>
- Susanti, S. (2019). Klasifikasi Kemampuan Perawatan Diri Anak dengan Disabilitas Menggunakan SMOTE Berbasis Neural Network. *Jurnal Informatika*, 6(2), 175–184. <https://doi.org/10.31311/ji.v6i2.5798>
- WHO. (2007). *ICF-CY (International Classification of Functioning, Disability and Health for Children and Youth)*. Switzerland: World Health Organization.
- WHO. (2018). Disabilities. Retrieved May 4, 2018, from <http://www.who.int/news-room/facts-in-pictures/detail/disabilities>
- Yeh, Y., Hou, T., & Chang, W. (2012). Expert Systems with Applications An intelligent model for the classification of children ' s occupational therapy problems. *Expert Systems With Applications*, 39(5), 5233–5242. <https://doi.org/10.1016/j.eswa.2011.11.016>
- Zarchi, M. S., Bushehri, F., & Dehghanizadeh, M. (2018). SCADI: *International Journal of Medical Informatics*, 114(2), 1–28. <https://doi.org/10.1016/j.ijmedinf.2018.03.003>