**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# **IDENTITAS PENGUSUL**

Nama : **Azwar Rizal Alfarisi**

NRP : **5108 100 117**

Dosen Wali : **Imam Kuswardayan, S.Kom, M.Kom**

1. **JUDUL TUGAS AKHIR**

***“Implementasi Model Hibrida yang Menggabungkan Case-Based Reasoning dan Fuzzy Decision Tree untuk Klasifikasi Data Medis”***

1. **LATAR BELAKANG**

Saat ini penggunaan ilmu kecerdasan komputasional pada diagnosis medis menjadi suatu kecenderungan baru. Banyaknya aplikasi medis dalam jumlah besar adalah buktinya. Banyak prosedur dalam diagnosis medis bisa diimplementasikan dengan ilmu klasifikasi data. Para peneliti telah mengembangkan berbagai metode untuk meningkatkan akurasi dalam klasifikasi data medis. Semakin bagus akurasi klasifikasi maka informasi yang didapat untuk mengindentifikasi pasien juga semakin akurat. Algoritma-algoritma *meta-heuristic* seperti *genetic algorithms*, *particle swarm optimizations*, dan *tabu search* serta metode *data mining* seperti *neural network* dan *decision tree* telah digunakan dan menghasilkan hasil yang cukup signifikan[1].

Model *neural network* pernah digunakan dan didapatkan hasil dengan rata-rata akurasi yang tinggi. Namun, proses keputusan dalam model tersebut berupa *black box*, yaitu model sulit untuk diinterpretasikan secara manual[2]. Untuk mengatasi masalah *black box* tersebut digunakanlah algoritma *decision tree* ID3 (Iterative Dichotomizer 3). Tetapi, dalam penerapannya ID3 mempunyai kelemahan dalam menangani data yang bertipe kontinu. Oleh karena itu, semua data kontinu akan diubah menjadi data bertipe diskrit. Caranya adalah dengan menggunakan *fuzzy set theory*.

Selain model klasifikasi juga digunakan seleksi fitur. Seleksi fitur berfungsi untuk meringankan masalah dimensi yang besar sehingga mengoptimalkan waktu yang digunakan dan mengurangi kerja *memory* pada komputer. Walaupun begitu, hasil yang didapatkan tetap hasil yang optimal. Seleksi fitur yang digunakan adalah salah satu metode regresi yaitu *stepwise regression analysis* (SRA).

Berdasarkan kelebihan-kelebihan metode tersebut, maka dikembangkanlah suatu model hibrida yang menggabungkan metode-metode yang terdiri dari SRA, *clustering*, teori *fuzzy* dan ID3[3].

Tugas akhir ini akan mengimplementasikan model hibrida tersebut untuk melakukan klasifikasi pada beberapa data medis. Yaitu data gangguan hati dan kanker payudara. Implementasi ini kemudian diharapkan dapat bermanfaat di bidang kedokteran dalam diagnosis medis yang lain.

1. **TUJUAN TUGAS AKHIR**

Tugas akhir ini bertujuan mengimplementasikan sistem perangkat lunak yang menerapkan model hibrida gabungan antara SRA, *clustering*, teori *fuzzy* danID3untuk klasifikasi data medis.

1. **MANFAAT TUGAS AKHIR**

Tugas akhir ini dikerjakan dengan harapan dapat memberikan manfaat yang besar

pada peneliti atau dokter untuk membuat keputusan dalam diagnosis medis.

1. **RUMUSAN MASALAH**

Untuk mengimplementasikan metode kategorisasi data medisini, maka dirumuskan permasalahan sebagai berikut :

1. Bagaimana mengimplementasikan metode *stepwise regression* untuk memilih faktor-faktor paling penting dari kumpulan input.
2. Bagaimana melakukan pembagian dataset utama menjadi *case-case* dataset yang lebih kecil dimana tiap case berupa data-data yang homogen dengan metode *weighted clustering*.
3. Bagaimana mengimplementasikan teory *fuzzy* dalam diskritisasi tipe data dan membangun model *decision tree* ID3 agar memperoleh hasil klasifikasi yang maksimal.
4. Bagaimana mengevaluasi model yang telah dibuat
5. **BATASAN MASALAH**

Permasalahan yang dibahas dalam Tugas Akhir ini memiliki beberapa batasan, yaitu sebagai berikut :

1. Sistem perangkat lunak dibuat dengan menggunakan bahasa pemrograman Java SE dan perangkat lunak NetBeans IDE.
2. Menggunakan library Weka untuk melakukan perbandingan terhadap metode lain pada saat evaluasi model.
3. Dataset yang digunakan adalah *liver disorders* dan *Wisconsin Diagnostic Breast Cancer*. Dua dataset pertama dapat diperoleh dari UCI *Repository of Machine Learning Databases*[4].
4. **RINGKASAN TUGAS AKHIR**

Dalam implementasi tugas akhir ini dataset yang digunakan adalah dataset *liver disordes* yang terdiri dari 7 atributdan 345data. Pada *liver disorders dataset* 5 atribut pertama merupakan hasil tes darah yang muncul karena konsumsi alkohol yang berlebihan. Serta dataset *Wisconsin Diagnostic Breast Cancer* yang terdiri dari 32 atribut dan 569.

Dalam pengimplementasian Tugas Akhir ini akan mengikuti diagram alir sebagai berikut :

Gambar 8.1 Diagram alir secara umum

Berdasarkan diagram alir diatas, ada 6 tahap yang akan dilakukan pada implementasi tugas akhir ini, yaitu :

1. Pemilihan faktor-faktor input.
2. Menghitung nilai *weight* tiap data.
3. Membagi dataset ke beberapa *cluster* berdasarkan *weight.*
4. Diskritisasi data atribut yang bertipe kontinu dengan *fuzzy set theory* pada setiap *cluster.*
5. Membangun model *decision tree* pada tiap *cluster.*
6. Mengevaluasi model dengan membandingkan dengan model-model lain.

Tahap pertama dalam pengerjaan setelah didapatkan dataset adalah memilih faktor-faktor penting pada dataset. Faktor yang dimaksud adalah atribut. Metode yang digunakan adalah *stepwise regression analysis* (SRA)*.* Metode ini akan menentukan himpunan variabel independen yang paling mempengaruhi variabel dependen. Prosedur langkah per langkah pendekatan SRA dijelaskan sebagai berikut :

* Langkah 1 : Hitung koefisien korelasi (*r*) setiap variabel input (*X1, X2, ....., Xn*) dan data output (*Y*). Susun semua angka dalam matriks korelasi.
* Langkah 2 : Pilih jumlah pangkat terbesar (*r2*) dari matriks korelasi dan dapatkan model regresi = *f* (*Xi*), kemudian hitung korelasi *Y* dengan data input yang lain. Asumsikan *Xj* mempunyai *statistical significance α*, gunakan *α* untuk menghitung signifikansi variabel input yang lain.
* Langkah 3 : Hitung nilai parsial F data input yang lain seperti pada persamaan (1) dan (2), dan pilih koefesien korelasi terbesar diantara variabel-variabel tersebut (asumsikan *Xj*). Lalu, dapatkan model regresi = *f* (*Xi* ,*Xj*) yang lain lagi.

SSR =

SSE = (1)

= = , *i ϵ I* (2)

* Langkah 4 : Hitung nilai partial F data asli untuk variabel input *Xj*. Jika nilai lebih kecil dari *threshold* yang diberikan user, hapus variabel input dari model karena *Xj*tidak *statistically significant* untuk output.
* Langkah 5 : Ulangi langkah 3 dan langkah 4 sampai nilai partial F setiap variabel input lebih besar dari *threshold*. Ini berarti setiap nilai input punya pengaruh signifikan terhadap nilai output.

Tahap selanjutnya adalah menghitung nilai *weight* tiap data. Pertama asumsikan setiap data/kasus dengan {*e1,e2,...., N*} dimana *N* adalah jumlah total data/kasus di dalam dataset. Lalu kita gunakan kumpulan atribut {*Fj* (*j = 1, 2,...., n*)} dimana *n* adalah jumlah atribut. Kasus *ej* ke-*i* dapat direpresentasikan dalam *vector* dimensi *n + 1*, dimana *xj* sesuai dengan nilai atribut *Fj* (*1j n*) dan *yi*(*1i N*) adalah aksi yang akan diambil (atribut *class*). Misalkan setiap *j* (*1j n*) mempunyai *weight wj*(*wj ϵ* [0, 1]) telah didapatkan maka untuk setiap *ep*dan *eq weighted distance metric* dapat didenifisikan sebagai berikut :

= *d(w)*(*ep*, *eq*) = = (3)

dimana = . Dengan menggunakan *weighted distance* yang didefinisikan di persamaan (3), ukuran *similarity* antara 2 kasus, , dapat didefinisikan sebagai berikut :

= (4)

dimana α adalah parameter positif. Ketika semua *weight* mengambil nilai 1 maka ukuran *similarity* dinyatakan dengan . Setelah 2 langkah diatas selanjutnya adalah mencari nilai *weight* setiap data dan mendefinisikan fungsi evaluasi atribut. Semakin kecil nilai evaluasi, semakin baik hubungan antar atribut. Jadi, kita akan mencari *weight* agar fungsi evaluasi mencapai nilai minimum. Berikut *pseudocode* tahap ini

procedure weights\_calculate

parameter input : α dan *learning rate* η

inisialisasi *weight* tiap atribut *j* pada tiap data dengan nilai acak [0, 1];

Do

Hitung ∆ *wj* = -η;

/\*E(w) = \*/

Update *wj = wj + ∆ wj*

While not convergent;

Update *weight* baru tiap atribut *j;*

End

Tahap selanjutnya adalah membagi dataset menjadi beberapa *cluster* (*subcluster*). Dan data di tiap *subcluster* mempunyai pola yang hampir sama dalam *term* atribut dan kelas. Dalam pengimplementasiannya algoritma *clustering* yang digunakan adalah algoritma yang diperkenalkan oleh Shiu[6]. Algoritma ini menggunakan informasi *similarity* antar kasus. Pendekatan ini pertama mentransformasi matriks *similarity* ke matriks ekivalen lalu lihatlah apakah kasus saling ekivalen satu sama lain dalam satu *cluster*. Berikut *pseudocode* proses.

procedure divide\_class\_library\_int0\_clusters

tentukan *significant level* (*threshold*) *β ϵ* [0, 1]

Tentukan matriks *similarity* SM = dari persamaan (3) dan (4)

Do

Hitung SM1 = SM. SM = Spq dimana Spq= maxk(min())

Ganti SM dengan SM1

While not (SM1cSM);

Tentukan *cluster;*

//kasus *p* dan kasus q berada dalam satu *cluster* jika dan hanya jika

// Spq ≥β

End

Selanjutnya pada tiap cluster akan dilakukan diskritisasi data atribut yang kontinu menjadi diskrit dengan menerapkan *fuzzy set theory*. Dalam *fuzzy set theory*, fungsi *membership* adalah salah satu konsep dasar, dengan konsep ini kita akan dapat memproses data himpunan *fuzzy* kuantitatif dan menghilangkan kesan *fuzzy*. Cara untuk menemukan fungsi *membership* yang jitu adalah penting dalam *fuzzy set theory*. Tetapi, tidak ada aturan sempurna yang dapat diadopsi dalam berbagai jenis data himpunan *fuzzy*. Dalam implementasi disini akan mengadopsi *Triangle membership functions* sebagai fungsi *membership* utama.

Tahap selanjutnya adalah membangun model *decision tree* ID3. Model ini menghitung *information gain* G berdasarkan tiap atribut A, yang didefinisikan sebagai berikut

G(S, A) = entropy(S) – (5)

dimana S adalah total *space* input dan Sv adalah subset S yang mana mempunyai nilai v dari nilai tiap atribut A. Entropy(S) terhadap *class* didefinisikan , dimana *pi* adalah probabilitas *class* “*i.*” Atribut dengan *information gain* paling tinggi, misal B, dipilih sebagai node *root* dari *tree*. Kemudian, *decision tree* yang baru dibuat dengan secara rekursif terhadap tiap nilai B menggunakan *training subspace* S – {SB}. Node *leaf* atau node keputusan dibentuk ketika semua *instance* dalam *subspace training* yang tersedia berasal dari satu *class*.

1. **METODOLOGI**

Metodologi yang akan dilakukan dalam Tugas Akhir ini memiliki beberapa tahapan, diantaranya sebagai berikut:

1. Penyusunan Proposal Tugas Akhir

Tahap awal untuk memulai pengerjaan Tugas Akhir adalah penyusunan Proposal Tugas Akhir. Pada proposal ini, penulis mengajukan gagasan pembuatan sistem untuk memprediksi kebangkrutan bank atau perusahaan.

1. Studi Literatur

Pada tahap ini dilakukan pencarian, pengumpulan, penyaringan, pembelajaran dan pemahaman literatur yang berhubungan dengan algoritma metode *case-based reasoning* dan *fuzzy decision tree* serta pengumpulan dataset. Literatur yang digunakan dalam pengerjaan Tugas Akhir ini sebagian besar berasal dari internet berupa makalah ilmiah, tesis, artikel, materi kuliah, serta beberapa buku referensi.

1. Implementasi

Implementasi merupakan tahap untuk membangun sistem tersebut.

1. Pengujian dan Analisis

Pada tahap ini dilakukan uji coba terhadap sistem yang telah dibuat, mengamati kinerja sistem yang baru dibuat, serta mengidentifikasi kendala yang mungkin timbul.

5. Penyusunan Buku Tugas Akhir

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi perangkat lunak yang telah dibuat.

1. **JADWAL PEMBUATAN TUGAS AKHIR**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Kegiatan** | **Bulan** | | | | | | | | | | | | | | | |
| **Maret** | | | | **April** | | | | **Mei** | | | | **Juni** | | | |
| 1 | Penyusunan Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Studi Literatur |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Implementasi |  | | | | | | | | | | | | | | | |
| 1. Stepwise Regression 2. Weighting 3. Clustering 4. Fuzzyfication 5. ID3 Decision Tree 6. Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Uji Coba dan Analisis |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. **DAFTAR PUSTAKA**
2. C.-C. Bojarczuk, H.-S. Lopes, A.-A. Freitas, Genetic programming for knowledge discovery in chest-pain diagnosis, IEEE Engineering in Medicine and Biology Magazine 19 (4) (2000) 38-44.
3. R. Setiono, Extracting rules from pruned neural networks for breast cancer diagnosis, Artifical Intelligence in Medicine 8 (1) (1996) 37-51.
4. C.-Y. Fan, P.C. Chang, J.J. Lin, J.C. Hsieh. A hybrid model combining case-based reasoning and fuzzy decision tree for medical data classification, Applied Soft Computing 11 (2011) 632-644.
5. UCI Machine Learning Repository, University of California. <http://archive.ics.uci.edu/ml>.
6. S.C.K. Shiu, C.H. Sun, X.Z. Wang, D.S. Yeung, Maintaining case-based reasoning systems using fuzzy decision trees, in : Proceedings of the 5th European Workshop on Case-Based Reasoning-EWCBR2K 5, Trento, Italy, 2000, pp. 285-296