ANALISIS TEKNIK DATA MINING "ALGORITMA C4.5 DAN K-NEAREST NEIGHBOR" UNTUK MENDIAGNOSA PENYAKIT DIABETES MELLITUS

Giat Karyono
Teknik Informatika STMIK AMIKOM Purwokerto
Jl. Let Jend Pol Sumarto Watumas Purwanegara Purwokerto, Jawa Tengah 53123, Indonesia
e-mail: giant_mercy@yahoo.co.id

Abstrak – Penyakit diabetes mellitus (DM) merupakan masalah kesehatan yang serius baik di Indonesia maupun di dunia. Teknik data mining telah banyak dilakukan untuk membantu diagnosa penyakit diabetes mellitus. Makalah ini berisi perbandingan algortima C4.5 dan K-Nearest Neighbor (KNN) yang digunakan untuk mendiagnosa penyakit diabetes mellitus. Data set yang digunakan indian pima. Hasil penelitian menunjukan 76.105% dengan nilai precision 0.755%, recall 0.761%, F-measure 0.755% pada algoritma C4.5 dan 79.1436% pada algoritma KNN dengan precision 0.788%, recall 0.791%, F-measure 0.789%

Kata kunci – Diabetes Mellitus, Algoritma C4.5, K-Nearest Neighbor (KNN)

I. PENDAHULUAN

Diabetes mellitus merupakan penyakit metabolik dengan karakteristik hiperglikemia yang terjadi karena kelainan sekresi insulin, kerja insulin atau keduanya. Jika telah terkena kronik diabetes, maka akan terjadi kerusakan jangka panjang, disfungsi atau kegagalan beberapa organ tubuh terutama mata ginjal, mata, saraf, jantung, dan pembuluh darah [1]. Penyakit diabetes mellitus (DM) merupakan masalah kesehatan yang serius baik di Indonesia maupun di dunia. Menurut survei yang dilakukan WHO tahun 2005, Indonesia sebagai negara lower-middle income menempati urutan ke-4 dengan jumlah penderita diabetes mellitus terbesar di dunia setelah India, Cina, dan Amerika Serikat. Berdasarkan Profil Kesehatan Indonesia tahun 2008, diabetes mellitus merupakan penyebab kematian peringkat enam untuk semua umur di Indonesia dengan proporsi kematian 5,7%, di bawah stroke, TB, hipertensi, cedera, dan perinatal [2].

Data mining adalah suatu cara yang bertujuan dalam penemuan pola secara otomatis atau semi otomatis dari data yang sudah ada di dalam database atau sumber data lain yang dimanfaatkan untuk menyelesaikan suatu masalah melalui berbagai aturan proses [3]. Beberapa teknik data mining penelitian yang terkait diagnosa penyakit diabetes mellitus diantaranya Homogeneit y-Based Algoritma[4], Genetic Programming[5], Genetic Algoritm[6], MLP[7], ID3[8]. Beberapa peneliti lain juga telah melakukan penelitian yang membandingkan antar metode untuk memgetahui tingkat akurasi yang lebih baik diantaranya Artificial Neural Network dan Genetic Algoritma[8], Decision Tree dan Incremental Learning[9], ID3, C4.5, Decision Tree[10], Multilayer Perceptron, J48 and Naïve Bayes Classifier[11], EM, KNN, K-Means, amalgam KNN dan ANFIS Algoritma [12], Bayesian Network, Decision Tree[13], Naïve Bayes, MLP, Random Tree, REP Tree, RAD, Randome Forest, Jr8, Modified J48 Classifier [14]. Metode-metode tersebut memiliki tingkat akurasi yang berbeda-beda berdasarkan tools dan data set yang digunakan.

Berdasarkan hasil penelitian sebelumnya yang telah dilakukan oleh peneliti lain, penulis akan melakukan perbandingan metode algortima C4.5 dan KNN. Algoritma C4.5 merupakan algoritma yang nilai akurasinya tinggi terlihat pada penelitian terkait [15] dan algoritma KNN merupakan algoritma yang tangguh terhadap training data yang memiliki banyak noise [16].

II. METODE PENELITIAN

2.1. Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian ini adalah studi pustaka dengan mengambil data sekunder.

1. Studi Pustaka

   Studi pustaka merupakan metode pengumpulan data yang diarahkan kepada pencarian data dan informasi melalui dokumen-dokumen, baik dokumen tertulis, gambar, maupun dokumen elektronik yang dapat mendukung dalam proses penulisan. Misalnya dengan membaca dan mempelajari dari buku-buku, jurnal/paper yang terkait dengan penelitian yang dilakukan.

2. Data Sekunder

   Data sekunder merupakan sumber data penelitian yang diperoleh peneliti secara tidak langsung melalui media perantara (diperoleh, dicatat atau telah diteliti pihak lain). Data sekunder umumnya berupa bukti, catatan atau laporan historis yang telah tersusun dalam arsip yang dipublikasikan dan tidak dipublikasikan. Data ini digunakan karena sumber data diambil dari repositiri UCI Indian Pima.

2.2. Alur Penelitian

1. Pengumpulan Data

Dalam penelitian ini, data sekunder yang digunakan diambil dari repository *database UCI* Indian Pima. *Dataset* Indian Pima terdiri dari 768 data klinis. Semua pasien berjenis kelamin perempuan dan berumur sekurang-kurang 21 tahun yang tinggal di phoenix, Arizona, USA. Data set ini berisi dua kelas yang dipresentasikan dalam variabel binary yang bernilai '0' atau '1'. Bilangan "1" dari hasil tes menunjukan positif diabetes dan "0" menunjukan negative diabetes. Dataset berisi 768 pasien dengan 9 variabel numerik. Terdapat 268 (34,9%) kasus positif diabetes dengan ditandai kelas "1" dan 500(65,1%) kasus di kelas"0". Dataset tersebut tidak terdapat missing values. Lima pasien memiliki glucose "0", 11 pasien mempunyai BMI "0", 28 pasien memiliki blood pressure "0", 192 pasien mempunyai skin fold thickness "0", 140 mempunyai serum insulin level "0". Atribut dari dataset pima indian ditunjukan dalam tabel 1.

Tabel 1 Atribut Datase Diabetes Mellitus di Indian Pima

No Atribut		Deskripsi	Tipe	Unit
A1	PREGNANT	Number of Times pregnant	Numeric	-
A2	GTT	2-hour OGTT plasma glucose	Numeric	mg/dl
A3	BP	Diastolic blood Preseure	Numeric	mmHg
A4	SKIN	Triceps skin fold thickness	Numeric	mm
A5	INSULIN	2-hour serum insulin	Numeric	mm U/ml
A6	BMI	Body mass index (kg/m)	Numeric	Kg/m^2
A7	DPF	Diabetes pedigree function	Numeric	-
A8	AGE	Age of patient (years)	Numeric	-
Class	DIABETES	Diabetes onset within 5 years (0,1)	Numeric	-

2. Tahap *Pre-processing* dan sampling

Tahap ini dilakukan untuk mendapatkan data bersih dan siap untuk digunakan. Tahap *pre-processing* data meliputi identifikasi dan pemilihan atribut (*attribute identification and selection*), penanganan nilai atribut yang tidak lengkap (*handling missing values*), dan proses diskritisasi nilai. Analisis statistic Indian Pima diabetes dataset ditunjukan dalam tabel 2 dan tabel 3 dibawah ini. Nilai rata-rata sebelum normalisasi pada tabel 2. Selanjutnya data tersebut dinormalisasi menggunakan 'weka.filtes Discretize' untuk menormalisasikan data. Hasil normalisasi pada tabel 3.

Tabel 2 Sebelum Normalisasi

No Atribut	Mean	Standard deviation
Atr_1	3.84	3.37
Atr_2	120.89	31.97
Atr_3	69.1	19.35
Atr_4	20.53	16.0
Atr_5	79.79	115.24
Atr_6	31.99	7.88
Atr_7	0.47	0.33
Atr_8	33.24	11.76

Tabel 3 Sesudah Normalisasi

No Atribut	Mean	Standard deviation
Atr_1	0.226	0.19
Atr_2	0.608	0.16
Atr_3	0.566	0.15
Atr_4	0.207	0.16
Atr_5	0.094	0.13
Atr_6	0.477	0.11
Atr_7	0.168	0.14
Atr_8	0.204	0.19

3. Penggunaan Metode Klasifikasi

3.1. Algoritma C4.5

Algoritma pohon keputusan C4.5 memiliki prinsip kerja mengubah fakta yang besar menjadi pohon keputusan yang merepresentasikan aturan. Secara singkat logika algoritma C4.5 yang digunakan adalah sebagai berikut:

a. Pilih atribut sebagai akar.
b. Buat cabang untuk masing-masing nilai.
c. Bagi kasus dalam cabang.
d. Ulangi proses untuk masing-masing cabang sampai semua kasus pada cabang memiliki kelas yang sama.

Untuk memilih atribut sebagai akar, didasarkan pada nilai *gain* tertinggi dari atribut-atribut yang ada. Untuk menghitung *gain* digunakan rumus seperti tertera dalam Rumus 1

$$Gain(S,A) = Entropy(S) - \sum_{i=1}^{n} \frac{|Si|}{|S|} * Entropy(Si) \quad (1)$$

Keterangan :
S : Himpunan kasus
A : Atribut
n : jumlah partisi atribut A

|Si| : Jumlah kasus pada partisi ke-i
|S| : Jumlah kasus dalam S

Sedangkan penghitungan nilai *entropy* dapat dilihat pada rumus 2 berikut:

$$Entropy(S) = Entropy(S) - \sum_{i=1}^{n} - p_i * \log_2(p_i) \quad (2)$$

Keterangan:
S : Himpunan kasus
n : jumlah partisi S
pi : Proporsi S_i terhadap S

Langkah-langkah yang dilakukan dalam algoritma C4.5 yaitu:

a. *Dataset* Indian Pima diklasifikasikan menggunakan algoritma C4.5 (J48) dalam weka.
b. Lakukan pelatihan dan pengujian dengan menggunakan metode *10-crosss validation*.
c. Didapatkan hasil *classifier* dan menghasilkan *confusion matrix*.
d. Dapat melihat hasil *rule* dengan disajikan pohon keputusan.
e. Dari hasil *confusion matrix* dapat dihitung nilai *precision, recall,* dan *F-measure* dengan menjabarkan *confusion matrix* menjadi *of confusion*.

3.2. Algoritma K-Nearest Neighbor

K-Nearest Neighbor merupakan salah satu algoritma yang paling sering digunakan dalam klasifikasi atau prediksi data baru. Tujuan algoritma KNN adalah mengklasifikasikan obyek baru berdasarkan atribut dan *training sample. Clasifier* tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan pada memori. Diberikan titik *query*, akan ditemukan sejumlah *k* obyek atau (titik *training*) yang paling dekat dengan titik *query*. Klasifikasi menggunakan *voting* terbanyak diantara klasifikasi dari *k* obyek. Algoritma KNN menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari *query instance* yang baru. Algoritma metode KNN sangatlah sederhana, bekerja berdasarkan jarak terpendek dari *query instance* ke *training sample* untuk menentukan KNN-nya. Kelebihan *K-Nearest Neighbor*:

a. Tangguh terhadap *training* data yang memiliki banyak *noise*.
b. Efektif apabila *training* datanya besar.

Langkah-langkah untuk menghitung metode Algoritma *K-Nearest Neighbor*:

a. Menentukan Parameter K (Jumlah tetangga paling dekat).
b. Menghitung kuadrat jarak *Euclid* (*queri instance*) masing-masing objek terhadap data sampel yang diberikan. Rumus *Euclidean:*

$$D(x,y) = \sqrt{\sum_{k-1}^{n}(x_k - y_k)^2} \quad (3)$$

c. Kemudian mengurutkan objek-objek tersebut ke dalam kelompok yang mempunyai jarak *Euclid* terkecil.
d. Mengumpulkan kategori Y (Klasifikasi *Nearest Neighbor*)
e. Dengan menggunakan kategori *Nearest Neighbor* yang paling mayoritas maka dapat diprediksi nilai *queri instance* yang telah dihitung.

Langkah-langkah yang dilakukan dalam algoritma KNN adalah:

a. *Dataset* Indian Pima di klasifikasikan menggunakan algoritma KNN (IBk) dalam weka.
b. Lakukan pelatihan dan pengujian dengan menggunakan nilai k yang berbeda-beda.
c. Didapatkan hasil *classifier* dan menghasilkan *confusion matrix*.
d. Dari hasil *confusion matrix* dapat dihitung nilai *precision, recall,* dan *F-measure* dengan menjabarkan *confusion matrix* menjadi *of confusion*.

3.3. Weka

Weka adalah aplikasi *data mining open source* berbasis Java. Aplikasi ini dikembangkan pertama kali oleh Universitas Waikato di Selandia Baru sebelum menjadi bagian dari Pentaho. Weka terdiri dari koleksi algoritma *machine learning* yang dapat digunakan untuk melakukan generalisasi/formulasi dari sekumpulan data sampling

3.4. Akurasi

Akurasi adalah nilai derajat kedekatan dari pengukuran kuantitas untuk nilai sebenarnya (*true*). Berikut ini rumus dari nilai *accuracy*:
Rumus *accuracy*:

$$Accuracy = \frac{\text{number of TP+number of TN}}{\text{numbers of TP+FP+FN+TN}} \quad (4)$$

Dimana:
TP: *True Positive* FP: *False Positive*
TN: *True Negative* FN: *False Negative*

4. Validasi dan Evaluasi

Dalam tahap ini dilakukan validasi dan pengukuran keakuratan hasil yang dicapai oleh model menggunakan teknik yang terdapat dalam aplikasi weka yaitu *confusion matrix* dan *cross-validation*.

5. Penarikan Kesimpulan

Tahapan selanjutnya yaitu menyimpulkan hasil yang diperoleh dari penelitian. Algoritma C4.5 atau KNN yang memberikan hasil akurasi terbaik untuk mendiagnosis penyakit diabetes mellitus berdasarkan nilai *precision, recall, F-measure* dari masing-masing algoritma.

III. HASIL DAN PEMBAHASAN

1. Pengumpulan Data

Dalam penelitian ini data yang digunakan yaitu mengambil dari repositori *database UCI* Indian Pima yang terdiri dari 768 data klinis yang semuanya berasal dari jenis kelamin wanita dengan umur sekurang-kurangnya 21 tahun. Berikut adalah tabel 4 *dataset* Indian Pima diabetes.

Tabel 4 *Dataset* Indian Pima

No	*Pregnant*	*Glucose*	DBP	TSFT	Insulin	BMI	DPF	*Age*	*Class*
1	6	148	72	35	0	33.6	0.627	50	positive
2	1	85	66	29	0	26.6	0.351	31	negative
…	…	…	…	…	…	…	…	…	…
768	1	93	70	31	0	30.4	0.315	23	negative

2. Tahap *Pre-processing*

Setelah melakukan analisis terhadap *dataset* Indian Pima, diketahui bahwa tidak semua atribut memiliki nilai yang lengkap, dimana kelengkapan nilai atribut sangat mempengaruhi hasil klasifikasi. Atribut yang memiliki jumlah data tidak lengkap yaitu *pregnant* sebanyak 111, atribut *glucose* sebanyak 5, atribut DBP sebanyak 35, atribut TSFT sebanyak 227, atribut INS sebanyak 374, dan atribut BMI sebanyak 11. Sedangkan atribut *age dan class* memiliki nilai yang lengkap. Untuk menangani *missing value* dilakukan:

a. Nilai nol pada atribut *pregnant* dapat diasumsikan bahwa nilai tersebut menyatakan pasien belum pernah melahirkan, sehingga hal ini dimungkinkan sesuai dengan kondisi sebenarnya.
b. Data dengan nilai nol pada atribut *glucose*, DBP, dan BMI dapat dihilangkan karena jumlahnya tidak terlalu banyak sehingga tidak begitu mempengaruhi hasil klasifikasi.
c. Karena atribut TSFT dan INS memiliki jumlah nilai yang tidak ada sangat besar, maka kedua atribut ini tidak mungkin dihilangkan dan tidak mungkin dipakai dalam pengklasifikasian. Oleh karena itu, dalam penelitian ini atribut TSFT dan INS tidak digunakan.

Setelah proses penanganan *missing value* dilakukan, maka didapatkan 724 data (249 *class* positif dan 475 *class* negatif) dari 768 data aslinya dan siap diolah lebih lanjut dengan pilihan atribut *pregnant, glucose,* DBP, BMI, DPF*, age,* dan *class.*

Tetapi, terlebih dahulu dilakukan proses diskritisasi atribut. Tujuannya untuk mempermudah pengelompokan nilai berdasarkan kriteria yang telah ditetapkan. Hal ini juga bertujuan untuk menyederhanakan permasalahan dan meningkatkan akurasi dalam proses pembelajaran (Lesmana, 2012). Atribut *glucose* dibagi menjadi tiga, yaitu *low*, *medium*, dan *high*. Atribut DBP dibagi menjadi tiga, yaitu normal, *normal-to-high*, dan *high* (Jianchao, dkk, 2008). Sedangkan atribut BMI dikelompokkan menjadi empat, yaitu *low*, normal, *obese*, dan *severely-obese* (Patil, dkk, 2010). Atribut DPF terbagi menjadi dua kelompok, yaitu *low* dan *high*. Atribut *class* dibagi menjadi dua kelompok, yaitu positif diabetes dan negatif diabetes.

3. Penggunaan Metode Klasifikasi

Dari hasil perhitungan dan uji coba menggunakan aplikasi weka dengan algoritma C4.5 menghasilkan nilai akurasi sebesar 76.105%. Nilai akurasi tersebut didapatkan dari hasil perhitungan dari *precison*, *recall*, dan *F-measure.*
Hasil perhitungan nilai akurasi berdasarkan *confusion matrix* disajikan pada tabel 5:

Tabel 5 Nilai Akurasi Berdasarkan *Confusion Matrix*

Class	*Precision*	*Recall*	*F-measure*
Tested_negative	0.793%	0.861%	0.825%
Tested_positive	0.683%	0.570%	0.621%
Weighted Avg	0.755%	0.761%	0.755%

Sedangakan apabila pengklasifikasian menggunakan metode KNN dengan 5 kali percobaan menggunakan k yang berbeda-beda pada setiap kali percobaannya, k yang digunakan yaitu 9, 10, 11, 12, dan 13. Hasil akurasi yang diperoleh ditunjukkan pada tabel 4.3:

Tabel 6 Perbandingan Hasil Akurasi Yang Diperoleh Dengan K Yang Berbeda-beda.

Nilai k yang digunakan	Hasil akurasi
k-9	79.1436 %
k-10	78.5912%
k-11	78.8674%
k-12	78.3149%
k-13	78.3149%

Berdasarkan uji coba yang telah dilakukan pada *dataset* Indian Pima dapat diketahui bahwa nilai k sangat berpengaruh terhadap hasil akurasi. Namun, semakin tinggi nilai k yang dimasukkan, maka semakin rendah hasil akurasi yang didapatkan. Hal ini dikarenakan semakin besar nilai k-nya maka semakin banyak tetangga yang digunakan untuk proses klasifikasi dan kemungkinan untuk terjadinya *noise* juga semakin besar. Dari percobaan yang dilakukan sebanyak 5 kali dengan nilai k yang berbeda-beda, hasil akurasi yang diperoleh pada tabel 4 dengan *dataset* Indian Pima yang diuji menggunakan algoritma KNN mendapatkan hasil akurasi terbaik pada nilai k-9 yaitu sebesar 79.1436%. Hasil perhitungan nilai akurasi berdasarkan *confusion matrix* disajikan pada tabel 7 :

Tabel 7 Nilai Akurasi Berdasarkan *Confusion Matrix*

Class	*Precision*	*Recall*	*F-measure*
Tested_negative	0.824%	0.867%	0.845%
Tested_positive	0.719%	0.647%	0.681%
Weighted Avg	0.788%	0.791%	0.789%

Berikut perbedaan hasil akurasi yang diperoleh menggunaka algoritma C.45 dan KNN pada tabel 8.

Tabel 8 Perbandingan Hasil Akurasi C.45 dan KNN

Algoritma	Hasil akurasi	*Precision*	*Recall*	*F-measure*	Waktu
C4.5	76.105 %	0.755%	0.761 %	0.755 %	0.07 *s*

KNN	79.143 %	0.788%	0.791 %	0.789 %	0.1 9 *s*

Perbedaan akurasi yang diperoleh dengan menggunakan algoritma C.45 dan KNN sebesar 3.0386%. Waktu yang digunakan untuk *running dataset* dalam weka juga berbeda.

4. Validasi dan Evaluasi

Tabel 9 dan 10 merupakan tabel hasil *confusion matrix* dari pengujian *dataset* menggunakan algoritma C4.5 dan KNN dengan 10-*fold cross validation.*

Tabel 9 *Confusion Matrix* C4.5

	Positif diabetes	Negatif diabetes
Positif diabetes	409	66
Negatif diabetes	107	142
724	516	208

Tabel 10 *Confusion Matrix* KNN

	Positif diabetes	Negatif diabetes
Positif diabetes	412	63
Negatif diabetes	88	161
724	500	224

Tabel 9 menunjukan bahwa jumlah data hasil bentukan *rule* yang terkena diabetes yang sama dengan data *testing* yang juga terkena diabetes sebanyak 409. Kemudian, jumlah data hasil bentuka *rule* yang tidak terkena diabetes dengan data *testing* yang terkena diabetes sebanyak 66. Lalu, jumlah data hasil bentukan *rule* yang terkena diabetes dan data *testing* yang tidak terkena diabetes sebanyak 107. Yang terakhir, jumlah data hasil bentukan *rule* yang tidak terkena diabetes yang sama dengan data *testing* yang juga tidak terkena diabetes sebanyak 142.

Tabel10 menunjukan bahwa jumlah data hasil bentukan *rule* yang terkena diabetes yang sama dengan data *testing* yang juga terkena diabetes sebanyak 412. Kemudian, jumlah data hasil bentukan *rule* yang tidak terkena diabetes dengan data *testing* yang terkena diabetes sebanyak 63. Lalu, jumlah data hasil bentukan *rule* yang terkena diabetes dan data *testing* yang tidak terkena diabetes sebanyak 88. Yang terakhir, jumlah data hasil bentukan *rule* yang tidak terkena diabetes yang sama dengan data *testing* yang juga tidak terkena diabetes sebanyak 161.

IV. KESIMPULAN

Hasil akurasi dari masing-masing algoritma yaitu 76.105% dengan nilai *precision* 0.755%, *recall* 0.761%, *F-measure* 0.755% pada algoritma C4.5 dan 79.1436% pada algoritma KNN dengan *precision* 0.788%, *recall* 0.791%, *F-measure* 0.789%. Dengan demikian untuk menentukan diagnosa penyakit diabetes mellitus lebih baik menggunakan algoritma KNN karena tingkat akurasinya lebih tinggi dibandingkan dengan C4.5.

V. SARAN

Hasil analisis yang dilakukan untuk mendiagnosa penyakit diabetes mellitus didapatkan hasil akurasi yang lebih tinggi pada algoritma K-Nearest Neighbor. Untuk mendapatkan akurasi yang lebih baik saran untuk penelitian selanjutnta antara lain adalah:

1. Dilakukan penanganan nilai yang hilang pada setiap atribut.
2. Mencoba menggunakan algoritma yang lain seperti *Naïve Bayes*, ID3, CART dengan melihat tingkat akurasi yang lebih tinggi.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada STMIK AMIKOM Purwokerto yang telah memberi dukungan finansial terhadap penelitian ini.

REFERENSI

[1] Setiawan Meddy, “Buku Ajar Endokrin, Malang, FK:UMM

[2] Depkes RI, “Profil Keseheatan Indonesia”, Jakarta: Depkes RI, 2009.

[3] Witten, I. H, Frank, E., & Hall, M. A, “Data Mining Practical Machine”, Burlington:Elsevie, 2011.

[4] Huy Nguyen Anh Pham and Evangelos Triantaphyllous, “Predicition of Diabetes by Employing New Data Mining Approach Which Balances Fitting and Generalization”, Springer, 2008.

[5] Muhammad Waqar Aslam dan Asoke Kumar Nandi, “Detection of Diabetes Using Genetic Programming”, Euorpean Signal Processing Conference (EUSIPCO-2010), ISSN 2076-1465.

[6] S. Sapna, Dr. A. Tamilarasi dan M Pravin Kumar, “Implementation of Genetic Algoritm Predicting Diabetes”, International Journal of Computer Science, Vol 9 Issue 1, No.3 January 2012.

[7] Arwa Al-Rofiyee, Maram Al-Nowiser, Nasebih Al Mufad, Dr. Mohammed Abdullah Al-Hagery, “Using Prediction Methods in Data Mining for Diabetes Diagnosis”, http://www.psu.edu.sa/megdam/sdma/Downloads/Posters/Poster%2003.pdf, diakses tanggal 16 April 2016

[8] Jayanto, Yusnita Asri, “Sistem Pendukung Keputusan Pendeteksian Penyakit Diabetes dengan Metode Decision Tree Menggunakan Algoritma Iterative Dichotomiser 3(ID3)”, Universitas Pembangunan Nasional “Veteran”, 2011

[9] Aswinkumar.U.M and Dr. Anandakumar K.R, “Predicting Eearly Detection of Cardiac and Diabetes Symptoms Using Data Mining Techniques”, International Conference on Computer Design and Engineering, Vol. 49, 2012

[10] Rupa Bagdi, Prof. Pramod Patil, “Diagnosis of Diabetes Using OLAP and Data Mining Integration”, International Journal of Computer Science & Communication Networks, Vol 2(3), pp. 314-322, 2012

[11] Murat Koklu and Yauz Unal, “Analysis of a Population of Diabetic Patients Databases with Classifiers”, International Journal of Medical, Health, Pharmaceutical and Biomedical Engineering”, Vol. 7 No. 8, 2013.

[12] Veena vijayan, Aswathy Ravikumar, “Study of Data mining Algoritms for Prediction and Diagnosis of Diabetes Mellitus”, International Journal of Computer Applications (0975-8887), vol. 95-No.17, June 2014

[13] Mohtaram Mohammadi, Mitra Hosseini, Hamid Tabatabaee, “Using Bayesian Network for The Prediction and Diagnosis of Diabetes”, Bull. Env. Pharmacol. Life Sci., Vol 4[9]August 2015

[14] P. Radha, Dr. B. Srinivasan, "Predicting Diabetes by Consequencing the Various Data Mining Classification Technique", International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 6, August 2014, pp. 334-339.

[15] Gorunescu, F, "Data Mining Concepts, Models and Techniques", Verlah Berlon Heidelberg:Spinger, 2011.

[16] Lestari, Mei, "Penerapan Algoritma Klasifikasi Nearest Neighbor (K-NN) untuk mendeteksi penyakit jantung", Faktor Exacta 7(4): 366-371, ISSN:1979-276X, 2014

[17] Lesmana, I Putu Dody, "Perbandingan Kinerja Decision Tree J48 dan ID3 dalam Pengklasifikasian Diagnosis Penyakit Diabetes Mellitus", Jurnal Teknologi dan Informatika, Vol. 2 No.2 Mei 2012.