Kumpulan jurnaL Ilmu Komputer (KLIK)
Volume 08, No 3 Oktober 2021
ISSN: 2406-7857

PERBANDINGAN *K-NEAREST NEIGHBOR* DAN *BACKPROPAGATION NEURAL NETWORK* DALAM PREDIKSI RESIKO DIABETES TAHAP AWAL

Jaka Permadi [1], Herfia Rhomadhona [2], Winda Aprianti [3]
[1, 2, 3] Politeknik Negeri Tanah Laut
Jalan A. Yani Km. 6 Desa Panggung, Kecamatan Pelaihari (0512) 2021065
Jakapermadi.88@politala.ac.id [1], herfia.rhomadhona@politala.ac.id [2], winda@politala.ac.id [3]

Abstract

Diabetes is one of the most dangerous and deadly diseases in Indonesia, after stroke and coronary heart disease. Early prediction of diabetes risk is needed for early treatment. In this study, a comparison of classification methods was carried out in predicting the risk of early-stage diabetes. The dataset used is an open database collected from a questionnaire to a sample of patients at Sylhet Hospital, Bangladesh. The classification methods compared are K-Nearest Neighbor (KNN) and Backpropagation Neural Network (BPNN), because both are often used in research for the classification of human diseases. Based on the results of the study, BPNN is a classification method that is better than KNN in predicting the risk of early stage diabetes. BPNN with learning rate = 0.3, = 0.4 or = 0.5 and the number of hidden nodes = 5 units, has an accuracy rate of 90%, precision is 90% and recall is 90%. Meanwhile, KNN with K = 5, K = 7 or K = 9 has an accuracy rate of 83.75%, precision of 85.5497% and recall of 83.75%.

Keywords*: Backpropagation Neural Network, K-Nearest Neighbor, diabetes prediction*

Abstrak

Diabetes adalah salah satu penyakit berbahaya dan mematikan di Indonesia, setelah stroke dan jantung koroner. Prediksi resiko diabetes sejak awal diperlukan untuk penanganan penyakit ini sejak dini. Pada penelitian ini dilakukan perbandingan metode klasifikasi dalam prediksi resiko diabetes tahap awal. Dataset yang digunakan merupakan open database yang dikumpulkan dari kuesioner terhadap sample pasien di rumah sakit Sylhet, Bangladesh. Metode klasifikasi yang dibandingkan adalah K-Nearest Neighbor (KNN) dan Backpropagation Neural Network (BPNN), karena keduanya sering digunakan dalam penelitian untuk klasifikasi penyakit manusia. Berdasarkan hasil penelitian, BPNN merupakan metode klasifikasi yang lebih baik dibandingkan dengan KNN dalam memprediksi resiko diabetes tahap awal. BPNN dengan learning rate α = 0.3, α = 0.4 atau α = 0.5 dan jumlah node hidden = 5 unit, memiliki tingkat akurasi sebesar 90%, presisi sebesar 90% dan recall sebesar 90%. Sementara KNN dengan K = 5, K = 7 atau K = 9 memiliki tingkat akurasi sebesar 83.75%, presisi sebesar 85.5497% dan recall sebesar 83.75%.

Kata kunci*: Backpropagation Neural Network, K-Nearest Neighbor, prediksi diabetes*

1. PENDAHULUAN

Diabetes adalah salah satu penyakit berbahaya dan mematikan di Indonesia, setelah stroke dan jantung koroner [1]. Indonesia berada pada peringkat ke-6 dunia dalam jumlah penderita diabetes terbanyak, yaitu sebanyak 10,3 juta jiwa pada tahun 2017. Prediksi resiko diabetes sejak awal diperlukan untuk penanganan penyakit ini sejak dini.

Pada penelitian ini dilakukan perbandingan metode klasifikasi dalam prediksi resiko diabetes tahap awal. Dataset yang digunakan merupakan *open database* yang dikumpulkan dari kuesioner terhadap *sample* pasien di rumah sakit Sylhet, Bangladesh [2]. Dataset tersebut memuat 520 data dengan rincian 320 data *positive* dan 200 data *negative*. Setiap data terdiri atas 16 atribut, yaitu *Age*, *Sex*, *Polyuria*, *Polydipsia*, *Sudden weight loss*, *Weakness*, *Polyphagia*, *Genital thrush¸ Visual blurring*, *Itching*, *Irritability*, *Delayed healing*, *Partial paresis*, *Muscle stiffness*, *Alopecia* dan *Obesity*. Metode klasifikasi yang dibandingkan adalah *K-Nearest Neighbor* (KNN) dan *Backpropagation Neural Network* (BPNN), karena keduanya sering digunakan dalam penelitian untuk klasifikasi penyakit manusia.

Penelitian yang dilakukan oleh Lestari menyatakan bahwa KNN dapat diterapkan untuk mendeteksi penyakit jantung dengan nilai akurasi sebesar 70% dan AUC sebesar 0.875 [3]. Lestari kemudian menyatakan bahwa KNN adalah *classifier* dengan kategori baik. Ariani dan Samsuryadi melakukan penelitian yang menerapkan KNN untuk klasifikasi penyakit ginjal kronis. Pada penelitiannya ini KNN mampu mengklasifikasikan penyakit ginjal kronis dengan akurasi mencapai 85,83 % [4]. Atthalla dkk melakukan penelitian tentang penerapan KNN untuk klasifikasi penyakit kanker payudara. Pada penelitiannya ini mereka menggunakan rumus jarak Minkowski untuk mengukur kedekatan data uji dengan dataset. Hasil penelitian yang dilakukan menyatakan bahwa KNN memiliki tingkat akurasi sebesar 93% [5]. Cahyanti dkk melakukan penelitian tentang analisis performa KNN pada dataset pasien pengidap kanker payudara. Pada penelitiannya ini Cahyanti dkk menerapkan *K-Fold Validation*. Hasil penelitian yang dilakukan menyatakan bahwa KNN memiliki tingkat akurasi tertinggi sebesar 93%, niai presisi sebesar 97%, *recall* sebesar 98% dan *F-measure* sebesar 94% dalam klasifikasi pada dataset pasien pengidap kanker payudara tersebut [6].

BPNN merupakan *classifier* yang performanya bergantung pada beberapa faktor, salah satunya nilai *learning rate* dan model arsitektur yang digunakan. Penelitian yang dilakukan oleh Lubis dan Parlina berfokus pada penentuan model arsitektur terbaik dari BPNN dalam memprediksi tingkat keberhasilan pelatih sepakbola [7]. Hasil penelitiannya tersebut menyatakan bahwa jumlah node *hidden* pada arsitektur BPNN berpengaruh terhadap kemampuan BPNN tersebut dalam melakukan klasifikasi.

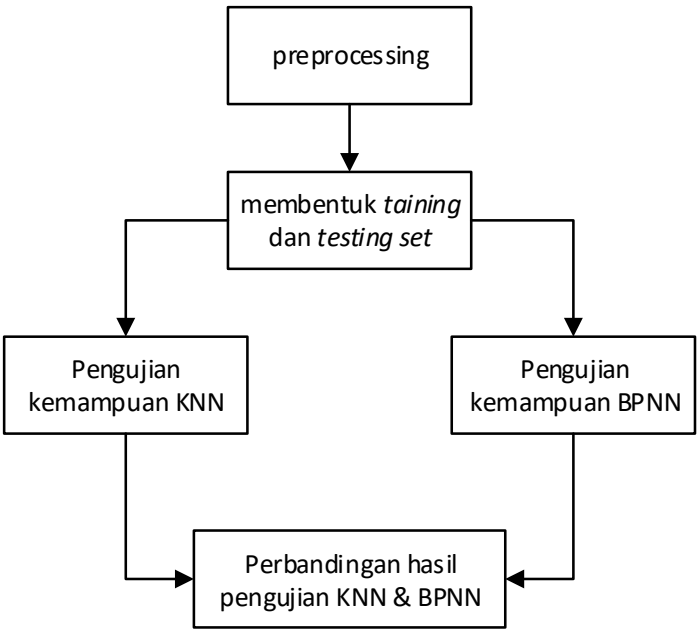
Penelitian yang menggunakan BPNN untuk prediksi penyakit manusia juga telah banyak dilakukan. Widodo dkk menerapkan BPNN untuk prediksi penyakit demam berdarah (DBD), dimana akurasi pada proses klasifikasinya sebesar 74% dan pada proses diagnosa mencapai 99% [8]. Ulum dan Laday melakukan penelitian tentang diagnosa penyakit kanker payudara menggunakan BPNN. Pada penelitiannya tersebut data latih dipilih sedemikian rupa sehingga akurasi yang

didapat mencapai 100% [9]. Setiawan dkk melakukan penelitian untuk membandingkan kemampuan BPNN dengan algoritma genetika dalam memprediksi penyakit autoimun. Hasil penelitiannya menyatakan bahwa kemampuan BPNN lebih baik daripada algoritma genetika dalam prediksi penyakit autoimun, dengan nilai akurasi pengujian sebesar 83%, *sensitifity* sebesar 97% dan presisi sebesar 75% [10].

2. METODOLOGI PENELITIAN

Penelitian yang dilakukan menggunakan dataset yang merupakan hasil kuesioner dari pasien-pasien di rumah sakit Sylhet, Bangladesh [2]. Pada dataset tersebut memuat 520 data dengan rincian 320 data untuk kelas *positive* dan 200 data untuk kelas *negative*. Setiap data memiliki 16 atribut, yaitu *Age*, *Sex*, *Polyuria*, *Polydipsia*, *Sudden weight loss*, *Weakness*, *Polyphagia*, *Genital thrush¸ Visual blurring*, *Itching*, *Irritability*, *Delayed healing*, *Partial paresis*, *Muscle stiffness*, *Alopecia* dan *Obesity*.

Tahapan penelitian yang dilakukan digambarkan sebagai berikut.



Gambar 1. Tahapan penelitian

2.1. Preprocessing

Dataset yang digunakan memiliki jumlah data yang berbeda untuk kelas *positive* dan kelas *negative*. Pada tahap *preprocessing* ini dataset diatur sehingga memiliki data *positive* dan *negative* dengan jumlah yang sama, yaitu 200 data kelas *positive* dan 200 data kelas *negative*. Pengaturan ini dilakukan agar proses pelatihan dari BPNN dapat lebih optimal. Kemudian setiap data dilakukan transformasi. Nilai data pada atribut umur dengan menggunakan *min-max normalization* ditransformasikan sehingga berada pada *range* 0 – 1. Sementara atribut lainnya yang sebelumnya memiliki nilai kualitatif, dikuantitatifkan ke dalam *range* 0 sampai

1. Contohnya nilai "Male" dan "Female" pada atribut *Sex* dikuantitatifkan dengan merubah nilai "Male" menjadi 1 , dan nilai "female" menjadi 0. Nilai aktual (target) dari setiap data juga dikuantitatifkan, yang sebelumnya "Positive" menjadi 1 dan "Negative" menjadi 0.

2.2. *Training set* dan *Testing set*

Setelah melakukan *preprocessing*, selanjutnya *training set* dan *testing set* dibentuk berdasarkan rasio dan dipilih secara acak dari dataset. Rasio yang digunakan pada penelitian ini adalah 80 : 20, yaitu 80% data *training* dan 20% data *testing*. Sehingga jumlah data *training* adalah sebanyak 320 data, dan jumlah data *testing* sebanyak 80 data.

2.3. Menguji Kemampuan KNN

Pengujian KNN dilakukan dengan menghitung jarak antara data *testing* dengan data *training*. Kelas prediksi dari data *testing* adalah kelas aktual terbanyak dari K-data *training* yang memiliki jarak terkecil dengan data *testing* tersebut. Pengujian yang dilakukan menggunakan K yang beragam, yaitu K = {5, 7, 9, 11, 13, 15}. Pengukuran jarak yang dilakukan menggunakan *euclidean distance*.

Hasil pengujian direpresentasikan dalam nilai akurasi, presisi dan *recall*. Nilai-nilai tersebut didapatkan dengan terlebih dahulu menghitung nilai *True Positive* (TP), *False Positive* (FP), *True Negative* (TN) dan *False Negative* (FN). Nilai-nilai TP, FP, FN dan TN digambarkan menggunakan *confusion matrix*, yang ditampilkan pada Gambar 2 berikut.

		Kelas Aktual	
		1	0
Kelas Prediksi	1	TP	FP
	0	FN	TN

Gambar 2. *Confusion Matrix*

Selanjutnya nilai akurasi, presisi dan *recall* didapatkan dengan menggunakan rumus sebagai berikut:

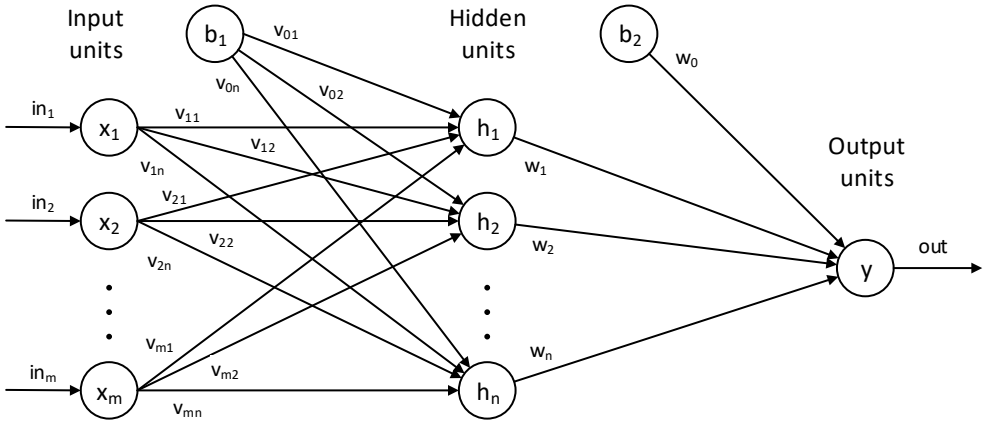
$$\text{Akurasi} = ((TP + TN)/(TP + FP + FN + TN)) \times 100\% \quad (1)$$

$$\text{Presisi} = (TP / (TP + FP)) \times 100\% \quad (2)$$

$$\text{Recall} = (TP / (TP + FN)) \times 100\% \quad (3)$$

2.4. Menguji Kemampuan BPNN

Pengujian menggunakan BPNN dilakukan dengan nilai *learning rate* (α) dan jumlah node *hidden* yang bervariasi. Arsitektur BPNN yang digunakan digambarkan pada Gambar 3 berikut.



Gambar 3. Arsitektur BPNN

Keterangan:
x_i : Unit *input* ke-i
h_j : Unit *hidden* ke-j
y : Unit *output*
b1 : Bias pada *input layer*
b2 : Bias pada *hidden layer*
v_{ij} : Bobot dari jaringan antara *input layer* dan *hidden layer*
w_j : Bobot dari jaringan antara *hidden layer* dan *output layer*

Jumlah node *input* adalah sebanyak 16 unit, yaitu atribut-atribut dari dataset. Keenambelas unit tersebut adalah nilai *Age*, *Sex*, *Polyuria*, *Polydipsia*, *Sudden weight loss*, *Weakness*, *Polyphagia*, *Genital thrush*, *Visual blurring*, *Itching*, *Irritability*, *Delayed healing*, *Partial paresis*, *Muscle stiffness*, *Alopecia* dan *Obesity* dari setiap data.

Proses pelatihan harus dilakukan terlebih dulu untuk mempersiapkan bobot-bobot pada BPNN sehingga arsitektur BPNN dapat digunakan untuk melakukan prediksi dengan kemampuan yang baik. Tahapan pelatihan dari BPNN adalah sebagai berikut.

1. Persiapkan nilai *learning rate* α, nilai maksimum *epoch* dan nilai toleransi *error*. Nilai toleransi *error* dan maksimum *epoch* akan digunakan sebagai acuan dalam menghentikan proses pelatihan.
2. Bangkitkan bobot acak untuk v_{ij} dan w_j.
3. Mulai tahapan ini terhitung sebagai *epoch* pertama. Hitung jaringan net_j dengan menjumlahkan semua unit *input* yang sudah dikali dengan bobotnya, menggunakan persamaan:

$$net_j = \sum_{i=1}^{m} x_i v_{ij} + v_{0j} \quad (4)$$

4. Hitung setiap unit pada *hidden layer* dengan menggunakan persamaan sebagai berikut:

$$h_j = f(net_j) = \frac{1}{1+e^{-net_j}} \quad (5)$$

5. Hitung setiap jaringan yang menuju *output layer* net_out dengan menjumlahkan semua *hidden unit* yang sudah dikalikan dengan bobotnya, menggunakan persamaan:
$$net_out = \sum_{j=1}^{n} h_j w_j + w_0 \quad (6)$$

6. Nilai prediksi y dihitung dengan menggunakan persamaan berikut:
$$y = f(net_out) = \frac{1}{1+e^{-net_out}} \quad (7)$$

7. Nilai prediksi y dibandingkan dengan nilai aktual d, dan untuk seluruh data *training* dihitung nilai *error*nya (pada penelitian ini menggunakan MSE). Kemudian proses pelatihan dapat dihentikan jika:
   a. Nilai *error* lebih kecil daripada toleransi *error*, atau
   b. Nilai *epoch* lebih besar daripada maksimum *epoch*
8. Poin 3 sampai 6 di atas disebut dengan *feedforward*. Lakukan *feedforward* untuk setiap data *training*. Setiap *feedforward* yang dilakukan langsung dilanjutkan dengan update bobot (*backpropagation*):
   a. Hitung perbaikan nilai bobot w_j menggunakan persamaan berikut:
   $$\Delta w_j = \alpha \cdot \delta_out \cdot h_j \quad (8)$$
   $$\Delta w_0 = \alpha \cdot \delta_out \quad (9)$$
   Dimana $\delta_out = y'(d-y) = y(1-y)(d-y)$

   b. Hitung perbaikan nilai bobot v_{ij} menggunakan persamaan berikut:
   $$\Delta v_{ij} = \alpha \delta_j x_i \quad (10)$$
   $$\Delta v_{0j} = \alpha \delta_j \quad (11)$$
   Dimana $\delta_j = h_j(1-h_j) \cdot \delta_out \cdot w_j$

   c. Lakukan *update* bobot menggunakan persamaan berikut:
   $$w_j(baru) = w_j(lama) + \Delta w_j \quad (12)$$
   $$v_{ij}(baru) = v_{ij}(lama) + \Delta v_{ij} \quad (13)$$

9. Kembali ke poin nomor 3 (*epoch* selanjutnya) dengan menggunakan bobot ter*update*.

Setelah tahapan pelatihan dilakukan, selanjutnya masuk ke tahap pengujian. Pada tahap pengujian data yang digunakan adalah data *testing*. Hitung nilai prediksi *y* dari setiap data *testing* menggunakan *feedforward*. Kemudian dapatkan nilai *True Positive* (TP), *False Positive* (FP), *False Negatif* (FN) dan *True Negative* (TN). Setiap nilai tersebut kemudian digunakan untuk mengitung akurasi, presisi dan *recall* dari BPNN.

3. HASIL DAN PEMBAHASAN

Pengujian dilakukan untuk mengetahui kemampuan KNN dan BPNN dalam memprediksi resiko diabetes tahap awal. Resiko diabetes tahap awal memiliki dua kelas, yaitu kelas *positive* dan kelas *negative*. Sehingga nilai TP, FP, FN dan TN dihitung untuk dua kelas tersebut. Nilai akurasi, presisi dan *recall* kemudian dihitung untuk setiap kelas, dan kemudian di rata-ratakan untuk mendapatkan akurasi, presisi dan *recall* dari *classifier* yang digunakan.

3.1. Hasil Pengujian Menggunakan KNN

Pengujian menggunakan KNN dilakukan dengan nilai K yang beragam, yaitu K = 5, K = 7, K = 9, K = 11, K = 13 dan K = 15. Hasil pengujian pada kelas *positive* digambarkan pada Tabel 1 berikut.

Tabel 1. Hasil Pengujian Menggunakan KNN untuk Kelas Positive

K	TP (positive)	FP (positive)	FN (positive)	TN (positive)	Akurasi	Presisi	Recall
5	29	2	11	38	83.75	93.5484	72.5
7	29	2	11	38	83.75	93.5484	72.5
9	29	2	11	38	83.75	93.5484	72.5
11	26	2	14	38	80	92.8571	65
13	25	2	15	38	78.75	92.5926	62.5
15	25	2	15	38	78.75	92.5926	62.5

Hasil pengujian pada kelas *negative* digambarkan pada Tabel 2 berikut.

Tabel 2. Hasil Pengujian Menggunakan KNN untuk Kelas Negative

K	TP (negative)	FP (negative)	FN (negative)	TN (negative)	Akurasi	Presisi	Recall
5	38	11	2	29	83.75	77.551	95
7	38	11	2	29	83.75	77.551	95
9	38	11	2	29	83.75	77.551	95
11	38	14	2	26	80	73.0769	95
13	38	15	2	25	78.75	71.6981	95
15	38	15	2	25	78.75	71.6981	95

Selanjutnya nilai akurasi, presisi dan *recall* di setiap kelas tersebut dirata-ratakan, hasilnya disajikan pada Tabel 3 berikut.

Tabel 3. Hasil Pengujian (Rata-Rata) Menggunakan KNN

K	Akurasi (%)	Presisi (%)	Recall (%)
5	83.75	85.5497	83.75
7	83.75	85.5497	83.75
9	83.75	85.5497	83.75
11	80	82.967	80
13	78.75	82.1454	78.75
15	78.75	82.1454	78.75

Berdasarkan pada Tabel 3 tersebut, nilai akurasi, presisi dan *recall* tertinggi ada pada K = 5, K = 7 dan K = 9, yaitu akurasi = 83,75%; presisi = 85,5497% dan *recall* = 83,75%. Dengan demikian kemampuan terbaik KNN dalam memprediksi resiko diabetes tahap awal adalah pada K = 5, K = 7 dan K = 9.

3.2. Hasil Pengujian Menggunakan BPNN

Pelatihan yang dilakukan menggunakan nilai maksimum *epoch* = 100.000 dan toleransi *error* = 0,001. Ada dua tahapan yang dilakukan penulis pada pengujian menggunakan BPNN. Tahapan pertama adalah dilakukan pengujian dengan arsitektur BPNN menggunakan 5 node *hidden* dengan nilai *learning rate* untuk pelatihan yang beragam, yaitu α = 0,2; α = 0,3; α = 0,4; α = 0,5; α = 0,6; α = 0,7; dan α = 0,8. Tujuannya adalah untuk menemukan nilai *learning rate* terbaik yang dapat menyebabkan pelatihan menjadi lebih optimal. Tahapan berikutnya adalah melakukan pengujian dengan arsitektur BPNN menggunakan *learning rate* untuk pelatihan yang diperoleh pada tahapan pertama, dengan jumlah node *hidden* yang beragam, yaitu 5 – 10. Tujuannya adalah untuk menemukan model arsitektur yang terbaik dalam memprediksi resiko diabetes tahap awal.

3.2.1. Hasil pengujian menggunakan BPNN (node *hidden* = 5)

Pada pengujian ini arsitektur BPNN yang digunakan memiliki node *hidden* sebanyak 5 unit. Pelatihan dilakukan dengan menggunakan *learning rate* yang beragam. Hasil pengujian pada kelas *positive* disajikan pada Tabel 4.

Tabel 4. Hasil Pengujian Menggunakan BPNN dengan 5 Node *Hidden* untuk Kelas *Positive*

α	Node hidden	TP (positive)	FP (positive)	FN (positive)	TN (positive)	Akurasi	Presisi	Recall
0.2	5	36	5	4	35	88.75	87.8049	90
0.3	5	36	4	4	36	90	90	90
0.4	5	36	4	4	36	90	90	90
0.5	5	36	4	4	36	90	90	90
0.6	5	34	6	6	34	85	85	85
0.7	5	34	6	6	34	85	85	85
0.8	5	34	6	6	34	85	85	85

Selanjutnya hasil pengujian pada kelas *negative* disajikan pada Tabel 5.

Tabel 5. Hasil Pengujian Menggunakan BPNN dengan 5 Node *Hidden* untuk Kelas *Negative*

α	Node hidden	TP (negative)	FP (negative)	FN (negative)	TN (negative)	Akurasi	Presisi	Recall
0.2	5	35	4	5	36	88.75	89.7436	87.5
0.3	5	36	4	4	36	90	90	90
0.4	5	36	4	4	36	90	90	90
0.5	5	36	4	4	36	90	90	90
0.6	5	34	6	6	34	85	85	85
0.7	5	34	6	6	34	85	85	85
0.8	5	34	6	6	34	85	85	85

Selanjutnya nilai akurasi, presisi dan *recall* di setiap kelas tersebut dirata-ratakan, hasilnya disajikan pada Tabel 6 berikut.

Tabel 6. Hasil Pengujian (Rata-Rata) Menggunakan BPNN dengan 5 Node *Hidden*

α	Node Hidden	Akurasi (%)	Presisi (%)	Recall (%)
0.2	5	88.75	88.7742	88.75
0.3	5	90	90	90
0.4	5	90	90	90
0.5	5	90	90	90
0.6	5	85	85	85
0.7	5	85	85	85
0.8	5	85	85	85

Berdasarkan pada hasil pengujian yang disajikan pada Tabel 6, dapat dilihat bahwa BPNN yang dilatih menggunakan *learning rate* 0.3, 0.4 dan 0.5 memiliki tingkat akurasi, presisi dan *recall* yang paling tinggi. Oleh karena itu di tahapan berikutnya akan dilakukan pengujian pada arsitektur BPNN dengan *learning rate* 0.3, 0.4 dan 0.6 dengan jumlah node *hidden* yang beragam.

3.2.2. Hasil pengujian menggunakan BPNN (α = 0.3)

Pengujian dilakukan menggunakan arsitektur BPNN yang telah dilatih dengan nilai α = 0.3 dan jumlah node *hidden* yang beragam. Hasil pengujian pada kelas *positive* disajikan pada Tabel 7.

Tabel 7. Hasil Pengujian Menggunakan BPNN dengan α = 0.3 untuk Kelas *Positive*

α	Node hidden	TP (positive)	FP (positive)	FN (positive)	TN (positive)	Akurasi	Presisi	Recall
0.3	5	36	4	4	36	90	90	90
0.3	6	36	8	4	32	85	81.8182	90
0.3	7	37	6	3	34	88.75	86.0465	92.5
0.3	8	37	8	3	32	86.25	82.2222	92.5
0.3	9	35	6	5	34	86.25	85.3659	87.5
0.3	10	35	7	5	33	85	83.3333	87.5

Selanjutnya hasil pengujian pada kelas *negative* disajikan pada Tabel 8.

Tabel 8. Hasil Pengujian Menggunakan BPNN dengan α = 0.3 untuk Kelas *Negative*

α	Node hidden	TP (negative)	FP (negative)	FN (negative)	TN (negative)	Akurasi	Presisi	Recall
0.3	5	36	4	4	36	90	90	90
0.3	6	32	4	8	36	85	88.8889	80

0.3	7	34	3	6	37	88.75	91.8919	85
0.3	8	32	3	8	37	86.25	91.4286	80
0.3	9	34	5	6	35	86.25	87.1795	85
0.3	10	33	5	7	35	85	86.8421	82.5

Selanjutnya nilai akurasi, presisi dan *recall* di setiap kelas tersebut dirata-ratakan, hasilnya disajikan pada Tabel 9 berikut.

Tabel 9. Hasil Pengujian (Rata-Rata) Menggunakan BPNN dengan α = 0.3

α	Node Hidden	Akurasi (%)	Presisi (%)	Recall (%)
0.3	5	90	90	90
0.3	6	85	85.3535	85
0.3	7	88.75	88.9692	88.75
0.3	8	86.25	86.8254	86.25
0.3	9	86.25	86.2727	86.25
0.3	10	85	85.0877	85

Bedasarkan pada Tabel 9 tersebut, jumlah node *hidden* terbaik dari arsitektur BPNN yang dilatih menggunakan α = 0.3 adalah arsitektur yang memiliki jumlah node *hidden* = 5 unit. Hasil pengujian dengan arsitektur tersebut menghasilkan tingkat akurasi, presisi dan *recall* yang tertinggi, yaitu akurasi = 90%, presisi = 90% dan *recall* = 90%.

3.2.3. Hasil pengujian menggunakan BPNN (α = 0.4)

Pengujian dilakukan menggunakan arsitektur BPNN yang telah dilatih dengan nilai α = 0.4 dan jumlah node *hidden* yang beragam. Hasil pengujian pada kelas *positive* disajikan pada Tabel 10.

Tabel 10. Hasil Pengujian Menggunakan BPNN dengan α = 0.4 untuk Kelas *Positive*

α	Node hidden	TP (positive)	FP (positive)	FN (positive)	TN (positive)	Akurasi	Presisi	Recall
0.4	5	36	4	4	36	90	90	90
0.4	6	36	8	4	32	85	81.8182	90
0.4	7	37	6	3	34	88.75	86.0465	92.5
0.4	8	37	7	3	33	87.5	84.0909	92.5
0.4	9	36	6	4	34	87.5	85.7143	90
0.4	10	36	6	4	34	87.5	85.7143	90

Selanjutnya hasil pengujian pada kelas *negative* disajikan pada Tabel 11.

Tabel 11. Hasil Pengujian Menggunakan BPNN dengan α = 0.4 untuk Kelas *Negative*

α	Node hidden	TP (negative)	FP (negative)	FN (negative)	TN (negative)	Akurasi	Presisi	Recall
0.4	5	36	4	4	36	90	90	90
0.4	6	32	4	8	36	85	88.8889	80
0.4	7	34	3	6	37	88.75	91.8919	85
0.4	8	33	3	7	37	87.5	91.6667	82.5

0.4	9	34	4	6	36	87.5	89.4737	85
0.4	10	34	4	6	36	87.5	89.4737	85

Selanjutnya nilai akurasi, presisi dan *recall* di setiap kelas tersebut dirata-ratakan, hasilnya disajikan pada Tabel 12 berikut.

Tabel 12. Hasil Pengujian (Rata-Rata) Menggunakan BPNN dengan α = 0.4

α	Node Hidden	Akurasi (%)	Presisi (%)	Recall (%)
0.4	5	90	90	90
0.4	6	85	85.3535	85
0.4	7	88.75	88.9692	88.75
0.4	8	87.5	87.8788	87.5
0.4	9	87.5	87.594	87.5
0.4	10	87.5	87.594	87.5

Bedasarkan pada Tabel 12 tersebut, jumlah node *hidden* terbaik dari arsitektur BPNN yang dilatih menggunakan α = 0.4 adalah arsitektur yang memiliki jumlah node *hidden* = 5 unit. Hasil pengujian dengan arsitektur tersebut menghasilkan tingkat akurasi, presisi dan *recall* yang tertinggi, yaitu akurasi = 90%, presisi = 90% dan *recall* = 90%.

3.2.4. Hasil pengujian menggunakan BPNN (α = 0.5)

Pengujian dilakukan menggunakan arsitektur BPNN yang telah dilatih dengan nilai α = 0.5 dan jumlah node *hidden* yang beragam. Hasil pengujian pada kelas *positive* disajikan pada Tabel 13.

Tabel 13. Hasil Pengujian Menggunakan BPNN dengan α = 0.5 untuk Kelas *Positive*

α	Node hidden	TP (positive)	FP (positive)	FN (positive)	TN (positive)	Akurasi	Presisi	Recall
0.5	5	36	4	4	36	90	90	90
0.5	6	36	9	4	31	83.75	80	90
0.5	7	35	10	5	30	81.25	77.7778	87.5
0.5	8	36	10	4	30	82.5	78.2609	90
0.5	9	30	4	10	36	82.5	88.2353	75
0.5	10	30	4	10	36	82.5	88.2353	75

Selanjutnya hasil pengujian pada kelas *negative* disajikan pada Tabel 14.

Tabel 14. Hasil Pengujian Menggunakan BPNN dengan α = 0.5 untuk Kelas *Negative*

α	Node hidden	TP (negative)	FP (negative)	FN (negative)	TN (negative)	Akurasi	Presisi	Recall
0.5	5	36	4	4	36	90	90	90
0.5	6	31	4	9	36	83.75	88.5714	77.5
0.5	7	30	5	10	35	81.25	85.7143	75
0.5	8	30	4	10	36	82.5	88.2353	75
0.5	9	36	10	4	30	82.5	78.2609	90
0.5	10	36	10	4	30	82.5	78.2609	90

Selanjutnya nilai akurasi, presisi dan *recall* di setiap kelas tersebut dirata-ratakan, hasilnya disajikan pada Tabel 15 berikut.

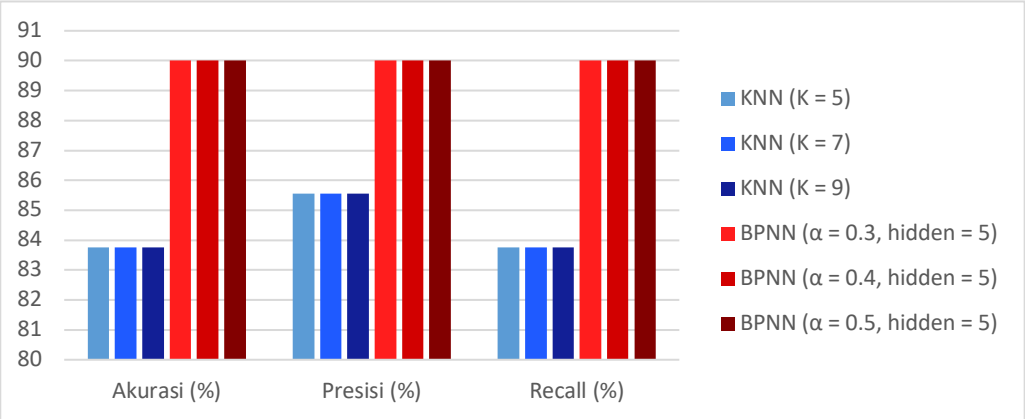
Tabel 15. Hasil Pengujian (Rata-Rata) Menggunakan BPNN dengan α = 0.5

α	Node Hidden	Akurasi (%)	Presisi (%)	Recall (%)
0.5	5	90	90	90
0.5	6	83.75	84.2857	83.75
0.5	7	81.25	81.746	81.25
0.5	8	82.5	83.2481	82.5
0.5	9	82.5	83.2481	82.5
0.5	10	82.5	83.2481	82.5

Bedasarkan pada Tabel 15 tersebut, jumlah node *hidden* terbaik dari arsitektur BPNN yang dilatih menggunakan α = 0.5 adalah arsitektur yang memiliki jumlah node *hidden* = 5 unit. Hasil pengujian dengan arsitektur tersebut menghasilkan tingkat akurasi, presisi dan *recall* yang tertinggi, yaitu akurasi = 90%, presisi = 90% dan *recall* = 90%.

3.3. Perbandingan Performa KNN dan BPNN

Berdasarkan hasil pengujian yang didapatkan, dapat disimpulkan bahwa kemampuan KNN yang terbaik dalam memprediksi resiko diabetes tahap awal adalah dengan pengaturan K = 5, K = 7 dan K = 9. Pada pengaturan tersebut kemampuan KNN dalam memprediksi memiliki tingkat akurasi sebesar 83.75%, presisi sebesar 85.5497% dan *recall* sebesar 83.75%. Sementara kemampuan BPNN yang terbaik dalam memprediksi resiko diabetes tahap awal adalah dengan arsitektur yang memiliki jumlah node *hidden* sebanyak 5 unit dan nilai *learning rate* untuk pelatihan yang digunakan adalah 0.3, 0.4 dan 0.5. Pada pengaturan tersebut BPNN memiliki tingkat akurasi sebesar 90%, presisi sebesar 90% dan *recall* sebesar 90%. Perbandingan performa KNN dan BPNN tersebut dapat dilihat pada Gambar 4.



Gambar 4. Grafik Perbedaan Tingkat Akurasi, Presisi dan Recall dari KNN dan BPNN

Pada Gambar 4, kelompok KNN diberi warna biru dan kelompok BPNN diberi warna merah. Terlihat bahwa baik tingkat akurasi, presisi dan *recall*, BPNN memiliki nilai yang lebih tinggi dibandingkan dengan KNN. Sehingga dapat disimpulkan bahwa BPNN memiliki kemampuan yang lebih baik dalam memprediksi resiko diabetes tahap awal daripada KNN, dengan nilai *learning rate* α = 0.3 atau α = 0.4 atau α = 0.5, dan jumlah node hidden sebanyak 5 node.

4. SIMPULAN

KNN memiliki kemampuan dalam memprediksi resiko diabetes tahap awal dengan tingkat akurasi sebesar 83.75%, presisi sebesar 85.5497% dan *recall* sebesar 83.75%, dengan pengaturan K = 5, K = 7 atau K = 9. BPNN memiliki kemampuan dalam memprediksi resiko diabetes tahap awal dengan tingkat akurasi sebesar 90%, presisi sebesar 90% dan *recall* sebesar 90%, dengan nilai *learning rate* α = 0.3, α = 0.4 atau α = 0.5 dan jumlah node *hidden* sebanyak 5 unit. Dengan demikian BPNN memiliki kemampuan yang lebih baik daripada KNN dalam memprediksi resiko diabetes tahap awal.

UCAPAN TERIMA KASIH

Ucapan terima kasih kami sampaikan kepada Politeknik Negeri Tanah Laut atas pendanaan Penelitian Dosen Dana Dipa (PD3) tahun pelaksanaan 2021.

DAFTAR PUSTAKA

[1] P2PTM Kemenkes RI, "Lindungi Keluarga Dari Diabetes," *p2ptm.kemkes.go.id*, 2018. http://p2ptm.kemkes.go.id/post/lindungi-keluarga-dari-diabetes (accessed Oct. 02, 2021).

[2] I. Dutta, "Early Stage Diabetes Risk Prediction Dataset," *kaggle*, 2020. https://www.kaggle.com/ishandutta/early-stage-diabetes-risk-prediction-dataset (accessed Oct. 02, 2021).

[3] M. Lestari, "Penerapan Algoritma Klasifikasi Nearest Neighbor (K-NN) untuk Mendeteksi Penyakit Jantung," *Fakt. Exacta*, vol. 7, no. 4, pp. 366–371, 2014.

[4] A. Ariani and S. Samsuryadi, "Klasifikasi Penyakit Ginjal Kronis Menggunakan K-Nearest Neighbor," in *Annual Research Seminar (ARS)*, 2019, pp. 148–151.

[5] I. N. Atthalla, A. Jovandy, and H. Habibie, "Klasifikasi Penyakit Kanker Payudara Menggunakan Metode K Nearest Neighbor (KNN)," in *Annual Research Seminar (ARS)*, 2018, pp. 148–151.

[6] D. Cahyanti, A. Rahmayani, and S. A. Husniar, "Analisis Performa Metode KNN pada Dataset Pasien Pengidap Kanker Payudara," *Indones. J. Data Sci.*, vol. 1, no. 2, pp. 39–43, 2020, doi: 10.33096/ijodas.v1i2.13.

[7] M. R. Lubis and I. Parlina, "Analisis Algoritma Backpropagation dalam Memprediksi Tingkat Keberhasilan Pelatih Sepakbola Sebagai Salah Satu

Kemajuan Olahraga di Indonesia," *Kumpul. J. Ilmu Komput.*, vol. 6, no. 3, pp. 264–274, 2019.

[8] W. Widodo, A. Rachman, and R. Amelia, "Jaringan Syaraf Tiruan Prediksi Penyakit Demam Berdarah dengan Menggunakan Metode Backpropagation," *J. IPTEK*, vol. 18, no. 1, pp. 64–70, 2014.

[9] M. B. Ulum and R. K. Laday, "Implementasi Metode Backpropagation Neural Network Untuk Mendiagnosis Penyakit Kanker Payudara," in *Seminar Nasional APTIKOM (SEMNASTIKOM)*, 2016, pp. 581–583.

[10] D. Setiawan, R. N. Putri, and R. Suryanita, "Perbandingan Algoritma Genetika dan Backpropagation pada Aplikasi Prediksi Penyakit Autoimun," *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 5, no. 1, pp. 21–27, 2019, doi: 10.23917/khif.v5i1.7173.