

PERBANDINGAN METODE *DEEP LEARNING* DAN *MACHINE LEARNING* UNTUK KLASIFIKASI (UJI COBA PADA DATA PENYAKIT KANKER PAYUDARA)

Yudicy Amelia¹, Puspa Eosina^{2*}, Foni Agus Setiawan³

¹Jurusan Teknik informatika, Fakultas Teknik, Universitas Ibn Khaldun Bogor
Jl. KH Sholeh Ishkandar Km2 Kota Bogor Telp 0251 311564
Email: yudicy.amelia@gmail.com

²Jurusan Teknik Mesin, Fakultas Teknik, Universitas Ibn Khaldun Bogor
Jl. KH Sholeh Ishkandar Km2 Kota Bogor Telp 0251 311564
Email: puspa.eosina@ft.uika-bogor.ac.id

³Jurusan Teknik Mesin, Fakultas Teknik, Universitas Ibn Khaldun Bogor
Jl. KH Sholeh Ishkandar Km2 Kota Bogor Telp 0251 311564
Email: fonj.agus@ui.ac.id

ABSTRAK

Penyebab kematian utama di seluruh dunia merupakan penyakit kanker, salah satunya adalah kanker payudara. Faktor penentu penyakit tersebut termasuk kategori ganas atau jinak bisa dilihat dari sembilan faktor utama berdasarkan ciri kanker tersebut. Tujuan dari penelitian ini adalah mendapatkan perbandingan metode yang paling akurat terhadap pengklasifikasian penyakit kanker payudara dan melihat selisih nilai mean square error (MSE) dari metode deep learning dan machine learning serta pencocokan hasil klasifikasi kedua metode dengan pernyataan klasifikasi yang sudah ada sehingga didapatkan nilai perbandingan metode yang paling akurat. Data yang digunakan pada penelitian ini diambil dari UCI Edu, dengan data latih sebanyak 546 data dan data uji sebanyak 137 data. Dari proses pengklasifikasian menggunakan FFNN pada machine learning dan RBM pada deep learning, maka dapat dilihat ada empat (2.92%) data kanker payudara pada proses FFNN yang tidak memenuhi class klasifikasi, sedangkan data yang tidak memenuhi class klasifikasi pada proses RBM ada dua (1.46%) data. Dalam kasus klasifikasi penyakit kanker payudara, akurasi metode machine learning lebih kecil dibandingkan dengan akurasi deep learning dan hasil hipotesa menggunakan uji t menunjukkan bahwa nilai signifikansi (2-tailed) lebih kecil dari $\alpha = 0.05$ yaitu sebesar $8,68844E-16$ yang menandakan bahwa deep learning lebih baik dibandingkan machine learning. Dengan akurasi metode machine learning sebesar 97.0803% dan deep learning sebesar 98.5401%. Nilai MSE pada pengklasifikasian menggunakan FFNN adalah sebesar 0.0814, sedangkan pada RBM sebesar 0.0584.

Kata kunci: deep learning; FFNN; klasifikasi; machine learning; RBM;

PENDAHULUAN

Representasi buatan dan otak manusia yang selalu mencoba mensimulasikan proses pembelajaran pada otak manusia salah satunya adalah *neural network* atau jaringan syaraf tiruan. Istilah tiruan atau buatan digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran. Model *neural network* mengadaptasi metode *backpropagation* untuk pembelajarannya. Pada awalnya metode *backpropagation* dirancang untuk *neural network feedforward*[2].

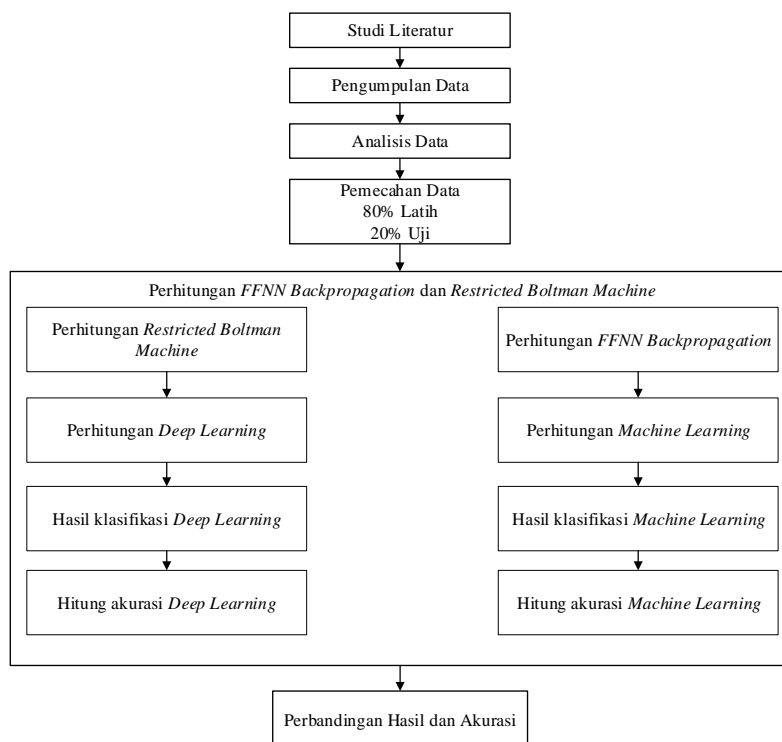
Deep learning melakukan model abstraksi pada data menggunakan algoritma *supervised* dan *unsupervised learning*, untuk belajar dari berbagai tingkat abstraksi. *Deep learning* menggunakan hirarki data untuk klasifikasi, karena peningkatan eksponensial data dalam aplikasi ini *deep learning* berguna untuk memprediksi keakuratan data. *Deep learning* banyak digunakan untuk mengambil keputusan, pencarian informasi, dan pengindeksian semantik. *Deep learning* merupakan representasi data yang mempunyai beberapa lapisan. Hal ini dapat mengefisienkan proses peningkatan pada data yang jumlahnya besar. Selain itu *deep learning* cocok digunakan untuk menganalisis data yang tidak terstruktur dan kumpulan data heterogen dari berbagai sumber[3].

Penyakit kanker merupakan salah satu penyebab kematian utama di seluruh dunia. Pada tahun 2012, kanker menjadi penyebab kematian sekitar 8,2 juta orang. Kanker paru, hati, perut, kolorektal, dan kanker payudara adalah penyebab terbesar kematian akibat kanker setiap tahunnya. Diketahui bahwa pada tahun 2012 terdapat 14.067.894 kasus baru kanker dan 8.201.575 kematian akibat kanker di seluruh dunia menurut data GLOBOCAN, *International Agency for Research on Cancer (IARC)* [6].

Dengan adanya repositori UCI yang telah menyediakan data penyakit kanker payudara yang dapat diakses secara umum, penulis memanfaatkan data tersebut untuk menentukan tingkat keganasan kanker. Keganasan kanker dimodelkan dengan menggunakan model klasifikasi, dimana penulis membandingkan metode *deep learning* dengan *restricted boltzmann machine* dan *machine learning* dengan *feedforward neural network* dalam hal mengklasifikasikan dan mengolah data.

Metode

Metode penelitian yang digunakan dalam penyusunan naskah ini meliputi dua bagian pokok yaitu metode pengumpulan data dan metode analisis. Metode penelitian dapat dilihat pada Gambar 1.



Gambar 1. Metode penelitian

6. Pengumpulan Data

Data diperoleh peneliti dari sumber yang sudah tersedia yaitu studi pustaka. Dalam tahap ini yang digunakan sebagai acuan untuk mengumpulkan data-data dan informasi yang dibutuhkan dalam menganalisa metode pada studi kasus adalah dengan menggunakan jurnal dan buku-buku DAFTAR PUSTAKA.

7. Analisis Kebutuhan Data

Analisis kebutuhan data terbagi ke dalam dua bentuk data. Terdapat 699 data penyakit kanker yang tercantum pada informasi UCI Edu, namun hanya 683 data yang dapat digunakan karena adanya atribut dalam data yang hilang. Data pertama merupakan data latih yang diambil 80% (546 data) dari total keseluruhan data dan data kedua merupakan data uji yang diambil 20% (137 data) dari total keseluruhan data. Data yang diambil sebagai data latih dan data uji merupakan data yang didapatkan secara acak. Terdapat dua *class* klasifikasi pada data, yaitu jinak dan ganas, jumlah data yang termasuk kedalam *class* ditunjukkan pada Tabel 1.

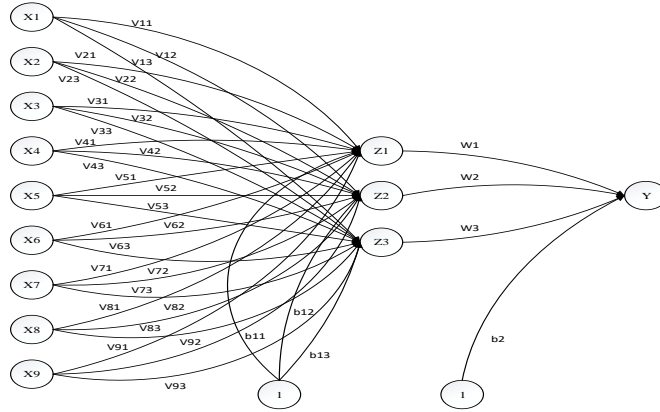
Tabel 1. *Class* Data Kanker Payudara

<i>Class</i>	
Jinak	Ganas
458 (65.5%)	241 (34.5%)



8. Mendefinisikan Jaringan Neural Network Machine learning dengan Feedforward Neural Network (FFNN)

Arsitektur *neural network* yang dibentuk mengacu pada arsitektur *neural network backpropagation* yang terdiri dari 3 pola. Pola pertama terdiri dari 9 unit input X , 3 unit tersembunyi Z , dan 1 unit output Y , serta 2 bias (untuk unit tersembunyi dan unit *output*). Gambar 2 menunjukkan pola 9-3-1.



Gambar 2. Arsitektur Jaringan dengan Pola 9-3-1

Algoritma pelatihan *Backpropagation* terdiri dari dua proses, yaitu proses *feedforward* dan *backpropagation* dari *error*-nya. Yaitu [12][15]:

Langkah 0: Inisialisasi bobot

Langkah 1: Ketika belum memenuhi kondisi berhenti, lakukan langkah 2-9

Langkah 2: Untuk setiap data pelatihan, lakukan langkah 3-8

Proses Feedforward

Langkah 3: Unit *input* ($x_i = 1, 2, 3, \dots, n$) menerima sinyal dan meneruskannya ke unit tersembunyi.

Langkah 4: Hitung semua *output* pada unit tersembunyi ($Z_j, i = 1, 2, \dots, p$).

$$Z_{netj} = V_{j0} + \sum_{i=1}^n X_i V_{ji} \quad (2.5)$$

Gunakan fungsi aktivasi untuk menghitung sinyal *output*-nya:

$$Z_j = f(Z_{netj}) \quad (2.6)$$

Dan kirimkan sinyal tersebut ke semua unit *output*. Lakukan langkah ini sebanyak jumlah lapisan tersembunyi.

Langkah 5: Hitung semua *output* jaringan di unit *output* ($Y_k, k = 1, 2, \dots, m$)

$$Y_{netk} = W_{k0} + \sum_{j=1}^p Z_j W_{kj} \quad (2.7)$$

Gunakan fungsi aktivasi untuk menghitung sinyal *output*-nya:

$$Y_k = f(y_{netk}) \quad (2.8)$$

Proses Backpropagation

Langkah 6: Hitung faktor δ unit *output* berdasarkan informasi *error* disetiap unit output ($Y_k, k = 1, 2, \dots, m$) dengan persamaan:

$$\delta_k = (t_k - y_k) f'(y_{netk}) \quad (2.9)$$

δ merupakan unit kesalahan yang akan dipakai dalam perubahan *layer* dibawahnya (Langkah 7).

Langkah selanjutnya hitung bobot koreksi (untuk memperbarui nilai W_{jk} sebelumnya) dengan laju percepatan α

$$\Delta W_{kj} = \alpha \cdot \delta_k \cdot z_j \quad (2.10)$$

Menghitung koreksi bias dengan menggunakan persamaan (untuk memperbarui nilai W_{k0} sebelumnya)

$$\Delta W_{k0} = \alpha \delta_k \quad (2.11)$$

dan mengirim δ_k ke unit pada *layer* sebelumnya.

Langkah 7: Hitung faktor δ unit tersembunyi berdasarkan *error* di setiap unit tersembunyi ($Z_j, j = 1, 2, \dots, p$).

$$\delta_{netj} = \sum_{k=1}^m \delta_k \cdot W_{kj} \quad (2.12)$$

Mengalikan dengan fungsi aktivasi untuk menghitung informasi *error* dengan persamaan (2.13).

$$\delta_j = \delta_{netj} f'(z_{netj}) \quad (2.13)$$

Menghitung koreksi bobot (untuk mengupdate V_{ij} sebelumnya)

$$\Delta V_{ij} = \alpha \cdot \sigma_j \cdot x_i \quad (2.14)$$

Dan menghitung koreksi bias (untuk mengupdate V_{0j} sebelumnya)

$$\Delta V_{j0} = \alpha \cdot \delta_j \quad (2.15)$$

Proses memperbarui bobot dan bias

Langkah 8: Tiap unit *output* (Y_k , $k = 1, 2, \dots, m$) memperbarui bobot dan biasnya ($j = 0, 1, 2, \dots, p$) dengan persamaan:

$$W_{jk}(\text{baru}) = W_{jk}(\text{lama}) + \Delta W_{jk} \quad (2.16)$$

Setiap unit tersembunyi (Z_j , $j = 1, 2, \dots, p$) memperbarui bobot dan biasnya ($i = 0, \dots, n$) dengan persamaan:

$$V_{ji}(\text{baru}) = V_{ji}(\text{lama}) + \Delta V_{ji} \quad (2.17)$$

Langkah 9: Periksa kondisi pelatihan berhenti

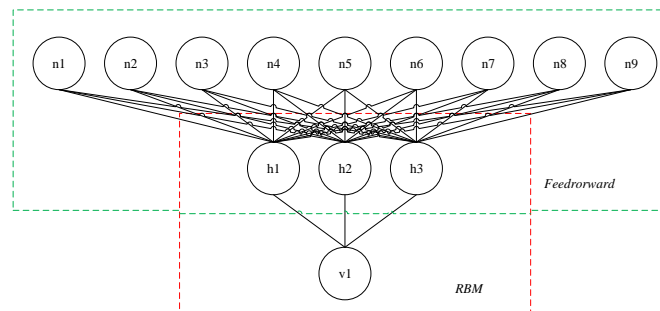
Model FFNN algoritma *backpropagation* secara sistematis dapat ditulis sebagai berikut:

$$y_k = \sum_{j=1}^p W_{kj} \cdot f \left[V_{j0} + \sum_{i=1}^n X_i V_{ji} \right] + W_{k0} \quad (2.18)$$

Pemilihan bobot akan sangat mempengaruhi ANN dalam mencapai global minimum (atau lokal saja) terhadap nilai *error*, dan cepat tidaknya proses pelatihan menuju kekonvergenan. Apabila nilai bobot awal terlalu besar, maka *input* ke lapisan tersembunyi atau lapisan *output* akan jatuh pada daerah dimana turunan fungsi *sigmoid*-nya akan sangat kecil. Sedangkan jika nilai bobot awal terlalu kecil, maka *input* ke setiap lapisan tersembunyi atau lapisan *output* akan sangat kecil.

Deep Learning dengan Restricted Boltzmann Machine (RBM)

Arsitektur jaringan RBM yang dibentuk mengacu pada arsitektur *neural network backpropagation* yang telah didefinisikan. Pola pertama terdiri dari 9 unit input, 3 unit tersembunyi, dan 1 unit output. Untuk unit input menuju unit tersembunyi dikerjakan dengan persamaan *feedforward* kemudian hasil dari persamaan yang berada pada unit tersembunyi dikerjakan menggunakan persamaan RBM.



Gambar 3. Arsitektur Jaringan RBM dengan Pola 9-3-1

Pada RBM, unit tersembunyi dapat dilihat sebagai sifat *detector* jaringan menetapkan sebuah probabilitas kesetiap pasangan dari unit *visible* dan unit tersembunyi neuron direkam untuk distribusi. Menjadi model berbasis energi berarti bahwa distribusi probabilitas atas variabel v dan h didefinisikan oleh fungsi entropi [16][17].

Arsitektur pada RBM hampir sama dengan arsitektur *Artificial Neural Networks* (ANNs) [16][5].

Summing function:

$$Sum = \sum w \cdot v + b \quad (2.19)$$

Dimana v mempresentasikan data *input*, w mempresentasikan bobot, dan b mempresentasikan bias.

Fungsi aktivasi yang digunakan adalah fungsi *sigmoid* seperti rumus (2.3), dimana x adalah hasil dari *summing function*.

Energi:

$$E(v, h) = - \sum_{i \in \text{visible}} a_i v_i - \sum_{j \in \text{tersembunyi}} b_j h_j - \sum_{i,j} v_i h_j w_{ij} \quad (2.20)$$

Dimana a dan b mempresentasikan bias, v mempresentasikan unit *visible*, h mempresentasikan unit tersembunyi, dan w mempresentasikan bobot.

Kemudian, langkah berikutnya yaitu melakukan proses *fine-tuning* dengan menggunakan metode *backpropagation* dengan tahapan hampir sama dengan *summing function*.

Update bobot:



$$\Delta W_{ij} = \epsilon (\langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model}) \quad (2.21)$$

Dimana ϵ mempresentasikan *learning rate*.

Langkah lain RBM yang menjadi pembeda yaitu perhitungan *probability*.

$$p(h_j = 1|v) = \text{sigm}(b_j + \sum_{i=1}^m v_i w_{ij}) \quad (2.22)$$

$$p(v_i = 1|h) = \text{sigm}(a_i + \sum_{j=1}^n h_j w_{ij}) \quad (2.23)$$

Dimana h mempresentasikan unit tersembunyi, v mempresentasikan unit *visible*, *sigm* mempresentasikan fungsi *sigmoid*, a dan b mempresentasikan bias, dan w mempresentasikan bobot.

Pelatihan RBM terdiri dalam meminimalkan kemungkinan log negatif yang diberikan dalam rumus (2.22).

$$\Delta W_{ij} = \epsilon \frac{\partial \log p(v)}{\partial w_{ij}} = \epsilon \langle v_i h_j \rangle_d - \langle v_i h_j \rangle_m \quad (2.24)$$

Dimana ϵ adalah *learning rate* $\langle \cdot \rangle_d$ dan $\langle \cdot \rangle_m$ digunakan untuk mewakili nilai data dan model yang diharapkan dari model.

Nilai yang diharapkan dapat diperoleh dari persamaan (2.22) dan (2.23) untuk data biner. Ekspektasi dari model lebih sulit untuk diperoleh. Bagaimanapun ini merupakan sebuah perkiraan stokastik yang bagus yang dikenal sebagai algoritma *contradictive divergence* (CD). Algoritma ini menggantikan model ekspektasi untuk estimasi menggunakan *Gibbs Sampling* dengan sedikit pengulangan. Walaupun hanya menggunakan satu pengulangan dari *Gibbs Sampling*, algoritma CD tetap mampu memberikan hasil yang bagus.

Algoritma RBM sebagai berikut[7][17]:

Proses Inisialisasi

Langkah 1: Inisialisasi bobot awal dan bias dengan nilai *random* yang kecil.

Langkah 2: Tetapkan maksimum *epoch* dan *learning rate* (α).

Langkah 3: Lakukan langkah di bawah ini selama (*epoch* < maksimum *epoch*).

Langkah 4: Lakukan langkah di bawah ini selama (data sampel < maksimum data sampel).

Proses positive phase (mengambil data dan sampel dari hidden unit)

Langkah 5: Hitunglah energi aktivasi menggunakan persamaan (2.20), probabilitas dan *state* dari unit *hidden* (i) dengan menggunakan persamaan (2.22).

Langkah 6: Hitung positif_assosiatif

Positif assosiatif diperoleh dari perkalian matriks data sampel yang ditranspose dari *visible* neuron dengan probabilitas yang dihasilkan dari langkah 5 dengan persamaan (2.21).

Proses negative phase (rekonstruksi visible unit dan data sampel dari hidden unit).

Langkah 7: Hitung energi aktivasi dan probabilitas dari unit *visible* (j) dengan menggunakan persamaan (2.23).

Langkah 8: Lakukan langkah 5 untuk *update hidden* unit

Langkah 9: Hitung negatif_assosiatif

Negatif assosiatif diperoleh dari perkalian matriks data (probabilitas dari unit *visible* yang diperoleh dari langkah 7) yang ditranspose dengan probabilitas dari unit *hidden* yang dihasilkan dari langkah 8 menggunakan persamaan (2.21).

Proses memperbarui parameter

Langkah 10: Memperbarui bobot

Update bobot dengan menggunakan persamaan (2.24).

Langkah 11: Hitung *error*

Hitung *error* menggunakan persamaan (2.25).

HASIL DAN PEMBAHASAN

7. Klasifikasi dengan FFNN Backpropagation pada Machine Learning menggunakan matlab

Hasil klasifikasi penyakit kanker payudara yang dikerjakan dengan FFNN *backpropagation* pada *machine learning* menggunakan matlab dapat dilihat pada Tabel 2. Tabel 2 hanya menampilkan beberapa data kanker payudara yang diklasifikasi.

Tabel 2. Hasil Klasifikasi Penyakit Kanker dengan FFNN Backpropagation

No	Klasifikasi FFNN	Pembulatan hasil klasifikasi FFNN	Target Uji	No	Klasifikasi FFNN	Pembulatan hasil klasifikasi FFNN	Target Uji
1	3,45416	3	4	11	2,33592	2	2
2	3,58153	4	4	12	3,58522	4	4

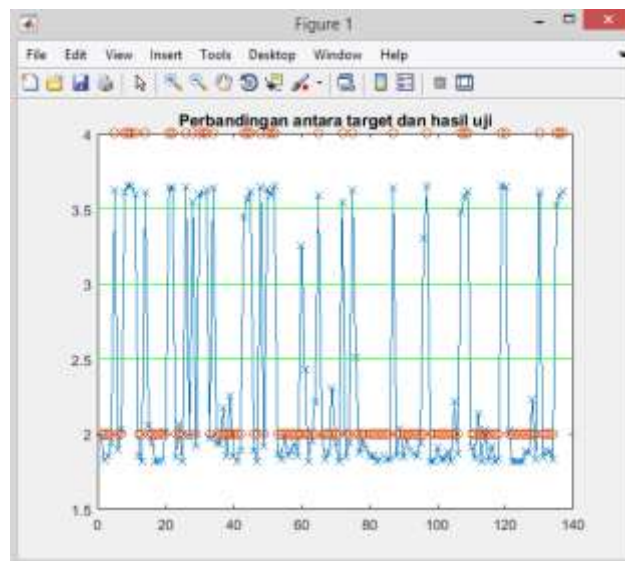
3	3,59619	4	4	13	1,97966	2	2
4	2,02496	2	2	14	3,3811	3	2
5	3,58633	4	4	15	3,59599	4	4
6	1,97966	2	2	16	1,8168	2	2
7	3,15833	3	2	17	1,81832	2	2
8	2,73244	3	2	18	1,91046	2	2
9	1,8168	2	2	19	1,82933	2	2
10	1,99064	2	2	20	1,82933	2	2

Dari proses pengklasifikasian tersebut terlihat bahwa ada data yang tidak terklasifikasi, data tersebut tidak memenuhi *class* klasifikasi, sehingga menghasilkan nilai diluar *class* klasifikasi. Ada 4 (2.92%) data kanker payudara yang tidak terklasifikasi, yaitu data pada nomor 30, 60, 61 dan 96. Hasil tersebut menunjukkan adanya kesalahan pada proses pengklasifikasian dengan menggunakan FFNN *backpropagation*, nilai MSE (*Mean Square Error*) pada pengklasifikasian data tersebut adalah sebesar 0.0814.

Dengan menggunakan persamaan (2.26), maka didapat nilai akurasi untuk klasifikasi kanker payudara menggunakan FFNN sebagai berikut:

$$\text{Akurasi} = (133 / 137) * 100\% = 97.0803\%.$$

Diagram perbandingan antara target klasifikasi (lingkaran merah) dan hasil uji (grafik biru) dapat dilihat pada Gambar 4.



Gambar 4. Perbandingan Target Klasifikasi dan Hasil Uji

8. Klasifikasi dengan RBM pada Deep Learning

Hasil klasifikasi penyakit kanker payudara yang dikerjakan dengan RBM pada *deep learning* menggunakan matlab dapat dilihat pada Tabel 3. Tabel 3 hanya menampilkan beberapa data kanker payudara yang diklasifikasi.

Tabel 3. Hasil Klasifikasi Penyakit Kanker dengan RBM

No	Klasifikasi RBM	Target Uji	No	Klasifikasi RBM	Target Uji
1	4	4	11	2	2
2	4	4	12	4	4



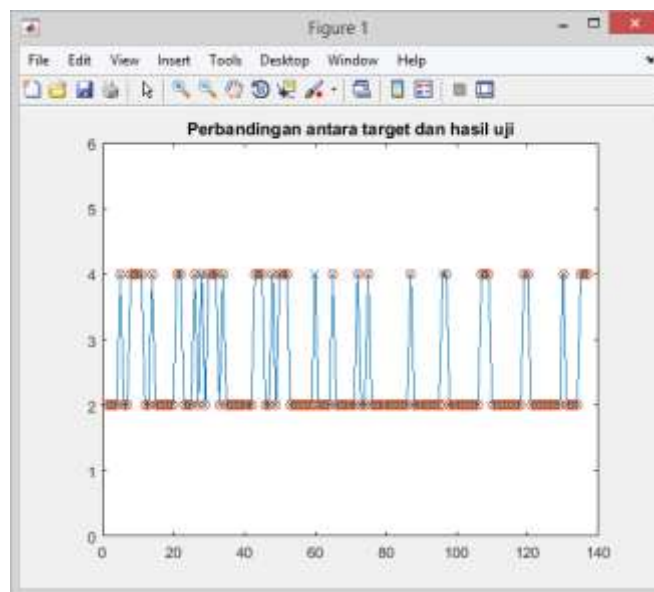
3	4	4	13	2	2
4	2	2	14	4	2
5	4	4	15	4	4
6	2	2	16	2	2
7	4	2	17	2	2
8	2	2	18	2	2
9	2	2	19	2	2
10	2	2	20	2	2

Dari proses pengklasifikasian tersebut terlihat bahwa ada data yang tidak terklasifikasi, data tersebut tidak memenuhi *class* klasifikasi, sehingga menghasilkan nilai yang tidak sesuai dengan target uji seharusnya. Ada 2 (1.46%) data kanker payudara yang tidak terklasifikasi, yaitu data pada nomor 60 dan 96. Hasil tersebut menunjukkan adanya kesalahan pada proses pengklasifikasian dengan menggunakan RBM, nilai MSE pada pengklasifikasian data tersebut adalah sebesar 0.0584.

Dengan menggunakan Persamaan (2.26), maka didapat nilai akurasi untuk klasifikasi kanker payudara menggunakan RBM sebagai berikut:

$$\text{Akurasi} = (135 / 137) * 100\% = 98.5401\%.$$

Diagram perbandingan antara target klasifikasi (lingkaran merah) dan hasil uji (grafik biru) dapat dilihat pada Gambar 5.



Gambar 5. Perbandingan Target Klasifikasi dan Hasil Uji

KESIMPULAN

KESIMPULAN yang didapat setelah melakukan analisis terhadap metode *machine learning* dan *deep learning* untuk klasifikasi penyakit kanker payudara adalah sebagai berikut:

1. Hasil klasifikasi dengan FFNN pada *machine learning* terdapat 4 data yang tidak terklasifikasi dengan benar, sedangkan dengan RBM pada *deep learning* terdapat 2 data yang tidak terklasifikasi dengan benar. Hasil tersebut didapat dari perbandingan hasil klasifikasi dengan data uji yang berjumlah 137 data kanker payudara.
2. Dalam kasus klasifikasi penyakit kanker payudara, akurasi metode *machine learning* lebih kecil dibandingkan dengan akurasi *deep learning* dan hasil hipotesa menggunakan uji t menunjukkan bahwa nilai signifikansi (*2-tailed*) lebih kecil dari $\alpha = 0.05$ yaitu sebesar $8,68844E-16$ yang menandakan bahwa *deep learning* lebih baik dibandingkan *machine learning*. Dengan akurasi metode *machine learning* sebesar 97.0803% dan *deep learning* sebesar 98.5401%. Nilai MSE pada pengklasifikasian menggunakan FFNN adalah sebesar 0.0814, sedangkan pada RBM sebesar 0.0584.

DAFTAR PUSTAKA

- [1] Widodo P. *Rule-Based Classifier untuk Mendeteksi Penyakit Liver*. Bianglala Informatika. 2014. Vol II. No 1.
- [2] Rudianto. *Penentuan Penyakit Peradangan Hati Dengan Menggunakan Neural Network Backpropagation*. Indonesian Journal on Computer Information Technology. 2016. Vol 1. No 1.
- [3] Hirak K dkk. *Big Data Analytics In Bio Informatics: Machine Learning Perspective*. Jurnal of iatex class files. 2014. Vol 13. No 9.
- [4] Putra BRD, Kusuma WA, dan Kustiyo A. *Klasifikasi khasiat Formla Jamu dengan Metode Deep Belief Network*. Makalah Kolokium Program S1 Ilmu Komputer Alih Jenis, Departemen Ilmu Komputer, FMIPA-IPB. 2011.
- [5] Ersti Renette, Wisesty Untari Novia, Jonri. *Klasifikasi Sinyal EEG Menggunakan Deep Neural Network*. Teknik Informatika, Fakultas Teknik, Universitas Telkom. ____.
- [6] Pusat data dan informasi. *Situasi Penyakit Kanker*. Buletin Jendela Pusat Data dan Informasi. 2015. ISSN 2088 - 270X.
- [7] Susilawati. *Algoritma Restricted Boltzmann Machine (RBM) untuk Pengenalan Tulisan Tangan Angka*. Seminar Nasional Teknologi Informatika. Universitas Medan Area. 2017. ISBN:978-602-50006-0-7.
- [8] Seonwoo M, Byunghan L, dan Sungroh Y. *Deep Larning in Bioinformatics*. Department of Electrical and Computer Engineering, Seoul National University, Seoul 08826, Korea. ____.
- [9] Feriante J. *Massiveli Multitask Deep Learning for Drug Discovery*. Master of science in computer science, university of wisconsin-Madison. 2015.
- [10] Diwandari S dan Setiawan NA. *Perbandingan Algoritme J48 dan NBTree untuk Klasifikasi Diaognosa Penyakit pada Soybean*. Seminar Nasional Teknologi Informasi dan Komunikasi (SENTIKA). Yogyakarta. 2015. ISSN 2089 – 9815.
- [11] Saputra RA. *Komparasi Algoritma Klasifikasi Data Minning untuk Memprediksi Penyakit Tuberculosis (TB): Studi Kasus Puskesmas Karawang Sukabumi*. Seminar Nasional Inovasi dan Tren (SNIT), Proceeding SNIT. 2014. Hal 3.
- [12] Fernanda JW. *Boosting Neural Network dan Boosting Cart Pada Klasifikasi Diabetes Militus Tipe II*. Jurnal Matematika. 2012. Vol 2. No 2.
- [13] Sinta R, Gernowo R, dan Suryono. *Rancang Bangun Sistem Peramalan Konsumsi Daya Listrik dengan Artificial Neural Network Backpropagation*. Jurnal Sistem Informasi Bisnis. 2013.
- [14] Beale MH, Hagan MT, dan Demuth HB. *MATLAB Neural Network Toolbox™ User's Guide*. The MathWorks, Inc. https://www.mathworks.com/help/pdf_doc/nnet/nnet_ug.pdf. Diakses 17 Februari 2018.
- [15] Kusumadewi F. *Peramalan Harga Emas Menggunakan Feedforward Neural Network Dengan Algoritma Backpropagation*. Skripsi. FMIPA, Matematika, Universitas Negeri Yogyakarta. 2014.
- [16] Hrasko R, Pacheco AGC, dan Krohlong RA. *Time Series Prediction using Restricted Boltzmann Machine and Backpropagation*. Procedia Computer Science. Information Technology and Quantitative Management (ITQM). 2015. Hal.55 ISSN 990 – 999
- [17] Ghazali G dan Jondri. *Peramalan Saham Menggunakan DBM (Deep Belief Networ)*. Fakultas Informatika, Universitas Telkom, Bandung.
- [18] Hansun Seng. *Penerapan WEMA dalam Peramalan Data IHSG*. Ultimatics. 2013. Vol V. No 2. ISSN 2085 – 4552.
- [19] Solikin R, Jusak, dan Sutomo E. *Sistem Pakar Diagnosis Penyakit pada Ayam Petelur Menggunakan Metode Certainty Factor*. Jurnal Sistem Informatika. 2014. JSIKA. Vol 3. No 2.
- [20] Laxmi GF. *Optimasi Pemilihan Threshold dan Operator Fuzzy Local Binary Pattern Menggunakan Multi Objective Genetic Algorithm*. Tesis, Institut Pertanian Bogor. 2012.

