

# Sistem Diagnosa Penyakit *Liver* Menggunakan Metode *Artificial Neural Network*: Studi Berdasarkan Dataset *Indian Liver Patient Dataset*

Nuril Afni Alviola<sup>1</sup>, Zaky Fathurrahman<sup>2</sup>, Rokhim Nur Rifai<sup>3</sup>, Ashri Shabrina Afrah<sup>4\*</sup>

<sup>1,2,3,4</sup>Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, UIN Maulana Malik Ibrahim, Malang

<sup>1,2,3,4</sup>Jl. Gajayana No.50, Dinoyo, Kec. Lowokwaru, Kota Malang, Jawa Timur 65144 Indonesia

email: <sup>1</sup>200605110141@student.uin-malang.ac.id, <sup>2</sup>200605110017@student.uin-malang.ac.id, <sup>3</sup>200605110171@student.uin-malang.ac.id, <sup>4\*</sup>ashri.shabrina@ti.uin-malang.com

**Abstract**—In humans, a condition known as liver disease attacks the organ's ability to regulate the body's levels of fat or cholesterol. The effects of liver illness vary according to their severity and how well they respond to treatment. Therefore, the development of a liver disease prediction system is pertinent and helpful in enabling medical professionals to act appropriately more immediately. The Artificial Neural Network (ANN) approach can be used to create this system. The aim of the classification is to reveal the accuracy of the ANN in classify the liver disease dataset. The dataset is divided into 3 steps using this method: data pretreatment, data processing, and data analysis. In order to create a new dataset, data preprocessing is done by separating and mending the original dataset. The determination of the hidden layer, the activation model, and the normalization of the model are carried out during data processing. There are accuracy values, error values, confusion matrices, and classification reports in the last stage, which is dataset evaluation. This model predicts a true negative score of 70, a true positive score of 14, a false negative score of 16, and a false positive score of 17. This model successfully classified datasets with an accuracy of 71.79%, showing that it is effective.

**Abstrak** – Penyakit Hati atau *liver* merupakan penyakit yang menyerang organ hati pada manusia dimana organ hati berfungsi dalam pengelolaan kolesterol atau lemak pada tubuh. Dampak yang diberikan oleh penyakit *liver* ini berbeda-beda tergantung pada tingkat keparahan dan respons pengobatan yang dilakukan oleh individu. Oleh karena itu, pengembangan sistem diagnosa penyakit *liver* menjadi relevan dan bermanfaat dalam membantu dokter dan tenaga medis untuk mengambil tindakan yang tepat secara lebih cepat. Sistem berbasis kecerdasan buatan ini dapat dibangun dengan menggunakan metode *Artificial Neural Network* (ANN). Tujuan dilakukan klasifikasi ini adalah untuk membantu mengetahui keakuratan model ANN dalam mengklasifikasi dataset penyakit *liver*. Menggunakan metode tersebut dataset dibagi menjadi 3 tahapan yaitu *preprocessing* data, pemrosesan data, dan evaluasi data. *Preprocessing* data dilakukan perbaikan terhadap dataset dan melakukan *split data* sehingga dihasilkan dataset baru. Pada pemrosesan data dilakukan penentuan *hidden layer*, model aktivasi, dan normalisasi pada model. Pada tahap terakhir yaitu evaluasi dataset, terdapat nilai akurasi, *confusion matrix*, dan *classification report*. Pada model ini didapatkan sebuah prediksi true negatif 70, true positif 14, false negatif 16, dan false positif 17. Dengan menggunakan model ini didapatkan hasil akurasi 71,79% yang menandakan bahwa model baik dalam melakukan klasifikasi pada dataset.

**\*) penulis korespondensi:** Ashri Shabrina Afrah  
Email: ashri.shabrina@ti.uin-malang.com

## I. PENDAHULUAN

Hati atau *liver* adalah organ tubuh yang terletak pada bagian atas rongga perut manusia yang beratnya berkisar antara 1200-1800 gram [1]. Fungsi organ ini berkaitan dengan pencernaan, metabolisme, pembuangan toksin, sistem imun, dan penyimpanan nutrisi di dalam tubuh. Organ hati juga memiliki peran penting dalam menjaga keseimbangan tubuh, produksi enzim, penyimpanan glikogen, dan produksi protein. Ketika terjadi kerusakan pada hati, berbagai masalah kesehatan dapat muncul, termasuk hepatitis, sirosis, dan kanker hati [1].

Penyakit *liver* merupakan suatu masalah kesehatan yang menyerang organ hati dan dapat mempengaruhi kualitas hidup individu [2]. Penyakit *liver* ini menyerang organ hati pada manusia dimana organ hati berfungsi dalam pengelolaan kolesterol atau lemak pada tubuh [3]. Penyebab penyakit hati bisa mencakup infeksi, cedera, pengaruh obat-obatan atau zat berbahaya, dan keabnormalan genetik.

Dampak yang diberikan oleh penyakit *liver* ini berbeda-beda tergantung pada tingkat keparahan dan respons pengobatan yang dilakukan oleh individu [4]. Semakin parah penyakit *liver* maka akan memberikan dampak yang lebih serius bahkan bisa menyebabkan kematian. Respon pengobatan pada penyakit *liver* juga sangat berpengaruh pada akibat yang ditimbulkannya pada kesehatan pasien [5]. Semakin dini penyakit *liver* ditangani maka semakin rendah pula keadaan buruk yang bisa disebabkan oleh penyakit ini. Penyakit *liver* harus diatasi sedini mungkin untuk menghindari komplikasi penyakit lainnya yang bisa diakibatkan karena keadaan hati yang terganggu atau rusak [6]. Namun, diagnosa penyakit *liver* dapat menjadi tantangan, karena gejala-gejalanya sering kali tidak spesifik dan sulit untuk dideteksi secara dini [7]. Oleh karena itu, pengembangan sistem diagnosa penyakit *liver* menjadi relevan dan bermanfaat. Hasil dari sistem diagnosa yang dibuat dapat membantu dokter dan tenaga medis dalam pengambilan keputusan terkait penanganan medis terhadap pasien.

Dalam konteks ini, sistem klasifikasi penyakit *liver* menggunakan metode *Artificial Neural Network* (ANN) menjadi pilihan yang menarik. *Artificial Neural Network* (ANN) adalah sebuah model komputasi yang terinspirasi oleh kerja otak manusia dan mampu belajar dari data untuk mengenali pola-pola yang kompleks [8]. Dengan menerapkan metode ini, dapat dibangun sebuah model klasifikasi yang dapat mempelajari pola-pola pada dataset yang berisi informasi pasien yang didiagnosis dengan penyakit *liver*.

Untuk membangun dan melatih model klasifikasi, kita dapat menggunakan dataset *Indian Liver Patient Dataset*

(ILPD) yang tersedia. Dataset ini berisi informasi yang relevan seperti umur, jenis kelamin, tingkat bilirubin, enzim hati, protein total, dan rasio albumin dan globulin dari pasien. Dengan memanfaatkan dataset ini, kita dapat melatih model klasifikasi yang mampu memprediksi apakah seseorang menderita penyakit *liver* atau tidak.

Melalui sistem klasifikasi ini, diharapkan kita dapat mengenali pola-pola yang berkaitan dengan penyakit *liver* dan membangun model yang dapat memberikan prediksi yang akurat. Dengan adanya prediksi yang lebih awal, para tenaga medis dapat melakukan tindakan yang tepat dan segera untuk mencegah perkembangan penyakit yang lebih serius. Selain itu, pengembangan sistem prediksi ini juga dapat menjadi dasar untuk pengembangan sistem pendukung keputusan di bidang kesehatan yang dapat memberikan rekomendasi pengobatan yang lebih efektif dan tepat [9].

*Artificial Neural Network* (ANN) / Jaringan Syaraf Tiruan (JST) adalah model *non-linear* yang kompleks, dibangun dari komponen yang secara individu berperilaku mirip dengan model regresi [10]. Metode ini menggunakan perhitungan *non-linear* dasar yang disebut *neuron* dan saling berhubungan sehingga menyerupai jaringan syaraf manusia [11]. Salah satu pengembangan dari ANN adalah Algoritma *Backpropagation*. *Backpropagation* merupakan algoritma pelatihan jenis terawasi (*supervised*) yang mempunyai banyak lapisan. *Backpropagation* akan mengadakan proses pembelajaran dengan pengaturan bobot hingga output yang dihasilkan jaringan sesuai dengan yang diharapkan [12]. Metode ini banyak diterapkan pada sistem peramalan dan klasifikasi.

Dengan demikian, pengembangan sistem klasifikasi penyakit *liver* menggunakan metode *Artificial Neural Network* merupakan langkah penting dalam upaya meningkatkan deteksi dini dan penanganan penyakit *liver*. Diharapkan bahwa sistem ini dapat memberikan kontribusi dalam meningkatkan kualitas hidup pasien dan meminimalkan risiko komplikasi yang disebabkan oleh penyakit *liver*.

## II. PENELITIAN YANG TERKAIT

Penelitian yang terkait dalam penelitian ini adalah “*An Artificial Neural Network Approach for Classification of Liver Diseases in Indian Patients*” oleh S. A. Bhosale, S. P. Patil [13]. Dalam penelitian ini menggunakan metode *Artificial Neural Network* (ANN) untuk melakukan klasifikasi penyakit *liver* pada pasien India menggunakan dataset *Indian Liver Patient Dataset*. Penelitian ini melibatkan beberapa tahap, seperti pra-pemrosesan data, pemilihan fitur, dan pelatihan model ANN. Parameter klinis yang relevan digunakan sebagai fitur untuk mengklasifikasikan penyakit *liver*. Hasil penelitian menunjukkan bahwa ANN dapat memberikan tingkat akurasi yang baik, dengan keakuratan prediksi sekitar 80-90%.

Penelitian terkait yang kedua adalah penelitian “Implementasi *Artificial Neural Network* dalam Mendeteksi Penyakit Hati (*Liver*)” oleh Irmawati, dkk [14]. Penelitian membahas tentang klasifikasi penyakit hati menggunakan metode ANN. Penelitian ini menggunakan dataset yang berasal dari Kaggle serta digunakan python untuk mengklasifikasi dataset. Pada penelitian ini didapatkan hasil akurasi sebesar 74% untuk memprediksi penyakit hati.

Penelitian lain yang membahas tentang pemanfaatan *Artificial Neural Network* (ANN) pada prediksi penyakit *liver*

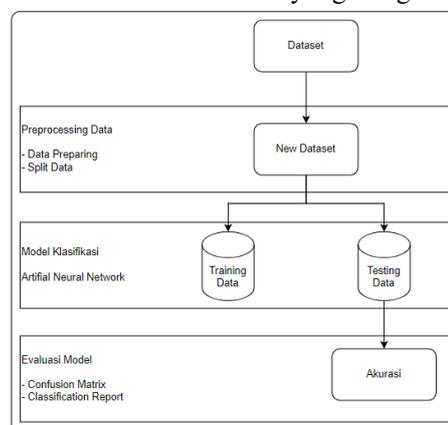
dilakukan oleh Rakshit et al. [15]. Penelitian yang berjudul “*Liver Disease Prediction System using Machine Learning Techniques*” ini membahas perbandingan berbagai metode *Machine Learning* untuk memprediksi resiko terjadinya penyakit *liver*. Pada penelitian ini, diperoleh bahwa metode *Artificial Neural Network* (ANN) menghasilkan akurasi yang lebih tinggi dibanding metode *Machine Learning* lain, seperti *Naïve Bayes* dan *Support Vector Machine* (SVM).

Penelitian sebelumnya dan penelitian yang kami lakukan saat ini memiliki persamaan pada topik yang diangkat dan metode yang digunakan, yaitu *Artificial Neural Network*. Akan tetapi, kami menggunakan dataset yang tersedia secara *open-source* pada UCL Machine Learning Repository, yaitu *Indian Liver Patient Dataset* (ILPD). Selain itu, pengujian akurasi hasil diagnosa dilakukan dengan *confusion matrix* dan *classification report*.

## III. METODE PENELITIAN

### A. Kerangka Penelitian

Proses penelitian dilakukan dengan melalui beberapa tahap. Tahapan yang harus dilewati adalah persiapan dataset, pemrosesan dataset menjadi dataset baru, klasifikasi dataset, dan langkah terakhir adalah evaluasi dari dataset tersebut. Kerangka penelitian harus dilakukan secara urut agar mendapatkan hasil akurasi sesuai yang diinginkan.



Gbr. 1 Kerangka Penelitian yang Dilakukan

Gbr. 1 menjelaskan tentang kerangka atau tahapan penelitian yang dilakukan. Pada tahap pertama, dilakukan persiapan dataset yang akan diklasifikasikan. Dataset yang digunakan merupakan dataset *Indian Liver Patient Dataset* (ILPD) yang diambil dari UCI *Machine Learning Repository*. Setelah dataset dibaca oleh sistem, selanjutnya data diproses pada tahap *preprocessing*, antara lain menghapus atau menangani missing values, normalisasi atau standarisasi atribut, melakukan pemilihan fitur jika diperlukan, serta pembagian data ke dalam data *training* dan data *testing*. Selanjutnya, dilakukan proses *training* atau proses latih pada model yang dibuat. Model yang sudah melalui proses latih selanjutnya diuji dengan proses *testing*. Pada proses *testing*, diperoleh hasil peramalan yang diuji akurasinya dengan *confusion matrix* dan *classification report*.

### B. Dataset

Dataset yang digunakan merupakan dataset *Indian Liver Patient Dataset* (ILPD) yang diambil dari UCI *Machine Learning Repository*. Dataset ILPD diambil dari rumah sakit pemerintah yang terletak di Bikaner, Rajasthan, India. Dataset ini terdiri dari 11 kolom, yaitu *age*, *gender*, *total bilirubin*, *direct bilirubin*, *alkaline phosphatase*, *alanine aminotransferase*, *aspartate aminotransferase*, *total protein*, *albumin*, *rasio albumin dan globulin* dan kolom penentu penyakit hati atau tidak.

```
# Load dataset
data = pd.read_csv('ILPD.csv', header=None)
```

Gbr. 2 Code read file dataset

Gbr.2 merupakan coding python yang digunakan untuk membaca file dataset berupa file CSV. Pada proses diatas digunakan *Library* Pandas untuk membaca file dataset. Proses tersebut menggunakan fungsi *read\_csv()* dari Pandas. File CSV harus berada pada direktori yang sama dengan script Python yang dieksekusi atau dengan memberikan path lengkap jika file CSV berada di lokasi yang berbeda.

```
In [3]: # Mengecek jumlah kolom
jumlah_kolom = dataset.shape[1]

print("Jumlah kolom:", jumlah_kolom)

Jumlah kolom: 11
```

Gbr. 3 Code Pengecekan kolom

Gbr. 3 menunjukkan kode program untuk melakukan pengecekan pada dataset untuk memastikan bahwa dataset tidak mengalami masalah atau *error*. Pada dataset ini dideteksi memiliki 11 kolom pada dataset, dimana kolom terakhir adalah kolom target dari prediksi. Tidak terdapat *error* pada dataset tersebut.

### C. Data Preprocessing

Proses ini dilakukan untuk memastikan kualitas dan kecocokan dataset. Ini termasuk menghapus atau menangani missing values, normalisasi atau standarisasi atribut, serta melakukan pemilihan fitur jika diperlukan. Gbr. 4 menunjukkan kode program untuk tahap *data preprocessing*.

```
# Pisahkan fitur dan label
X = data.iloc[:, :-1].values
y = data.iloc[:, -1].values

# Preprocessing data
for i in range(X.shape[1]):
    if isinstance(X[0, i], str):
        X[:, i] = pd.factorize(X[:, i])[0]
# factorize merubah kategorikal menjadi numerik

# Mengganti nilai NaN dengan nilai rata-rata kolom
from sklearn.impute import SimpleImputer
imputer = SimpleImputer(strategy='mean')
X = imputer.fit_transform(X)

# Normalisasi fitur
scaler = StandardScaler()
X = scaler.fit_transform(X)
```

Gbr. 4. Kode Program *Data Preprocessing*

### D. Splitting Data

Splitting data merujuk pada sebuah proses yang membagi dataset menjadi data pelatihan (*training data*) dan data uji (*test data*). Tujuan dari *splitting data* adalah untuk menguji performa model pada data yang belum pernah dilihat sebelumnya. Dataset awal dibagi menjadi dua bagian, yaitu data pelatihan dan data uji, dengan proporsi tertentu. Proporsi yang umum digunakan adalah 70-80% data untuk pelatihan dan 20-30% data untuk pengujian. Namun, pada penelitian yang dilakukan menggunakan proporsi 80% data training dan 20% data testing. Pemilihan dataset dengan proporsi 80:20 dilakukan untuk menghindari *overfitting* pada sistem. Selain itu, proporsi 20% data pengujian memberikan sampel yang cukup besar untuk menguji kinerja model secara objektif. Namun, proporsi ini tidak bisa dijadikan patokan, karena dapat digunakan proporsi berapapun menyesuaikan kondisi dataset.

```
# Split data menjadi training set dan test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Gbr. 5 Code Split data

Gbr. 5 menunjukkan kode program pada proses *splitting data*. Variabel X didefinisikan sebagai semua fitur yang ada pada dataset. Sedangkan variabel Y diidentifikasi sebagai kolom target. Kemudian, fungsi *train\_test\_split* digunakan untuk membagi data ke dalam data pelatihan dan data uji. Argumen *test\_size=0.2* menentukan proporsi data uji, dalam penelitian ini 20% dari seluruh data atau setara dengan 117 data akan menjadi data uji. Argumen *random\_state=42* digunakan untuk mengatur *state* yang memastikan hasil pembagian data akan sama setiap kali kode dijalankan. Angka 42 yang digunakan dalam kode tersebut merupakan nilai yang tidak memiliki makna khusus secara intrinsik. Angka tersebut dipilih semata-mata sebagai contoh atau nilai yang dianggap acak. Hasil dari *train\_test\_split* adalah empat variabel, yaitu *X\_train*, *X\_test*, *y\_train*, dan *y\_test*, yang masing-masing berisi data pelatihan dan data uji untuk fitur dan target.

### E. Klasifikasi

Pada model klasifikasi yang kami lakukan, kami menggunakan metode *Artificial Neural Network*. Pada tahapan ini akan dilakukan pemanggilan library neural network dan pelatihan data latih untuk mendapatkan pola pada dataset.

```
# Membangun model ANN
# Aktivasi ReLU (Rectified Linear Unit)
model = MLPClassifier(hidden_layer_sizes=(10, 10), activation='relu', solvers='adam', max_iter=1500, random_state=42)

# Melatih model dengan training set
model.fit(X_train, y_train)

# Prediksi Label untuk test set
y_pred = model.predict(X_test)

# Hitung matriks konfusi
confusion_mat = confusion_matrix(y_test, y_pred)
```

Gbr. 6 Coding model ANN

Untuk mempersiapkan algoritma ANN yang akan digunakan maka digunakan fungsi *MLPClassifier()* dari modul *sklearn.neural\_network*. Metode ini akan menggunakan 10 hidden layer dengan masing-masing layer memiliki 10 unit sel. Jenis aktivasi yang akan digunakan adalah *Rectified Linear Unit* (ReLU). Jenis aktivasi ini merupakan jenis yang paling umum digunakan pada metode ANN. Sedangkan untuk *solver*

yang digunakan adalah Adam. Jenis *solver* yang digunakan tergantung berapa banyak jumlah dataset, namun pada umumnya *solver* yang banyak digunakan adalah adam. *Max\_iter* atau maksimum perulangan yang digunakan adalah 1500. Jumlah *max\_iter* bisa ditentukan bebas sesuai kebutuhan. Jika tidak menggunakan *max\_iter* secara otomatis sistem akan memasang *max\_iter* 200. Algoritma *random state* digunakan untuk menentukan bobot awal pada metode ini.

F. Evaluasi Model

Evaluasi model pada penelitian ini dimaksudkan untuk mengetahui nilai akurasi pada model dan juga *classification report* dataset. Tahap ini akan memberikan hasil dari tahap klasifikasi dataset yang dilakukan. Hasil yang dihasilkan pada tahap ini akan akan dianalisis dan dibahas pada bab selanjutnya.

IV. HASIL DAN PEMBAHASAN

Evaluasi model menggunakan metode ANN akan menghasilkan nilai akurasi untuk mengukur sejauh mana model dapat memprediksi penyakit *liver* dengan benar berdasarkan dataset yang ada.

```
# Hitung akurasi
# Model belum sepenuhnya konvergen saat mencapai batas iterasi maksimum
# Rasio jumlah prediksi yang benar (prediksi yang sesuai dengan label yang sebenarnya) terhadap total jumlah sampel
accuracy = accuracy_score(y_test, y_pred)
print("Akurasi: {:.2f}%".format(accuracy * 100))

Akurasi: 71.79%

# Menghitung MAPE
def mean_absolute_percentage_error(y_true, y_pred):
    return np.mean(np.abs((y_true - y_pred) / y_true)) * 100

mape = mean_absolute_percentage_error(y_test, y_pred)
print("MAPE: {:.2f}%".format(mape))

MAPE: 21.37%
```

Gbr. 7 Akurasi skor dan Nilai MAPE

Kode program pada Gbr. 7 di atas merupakan perhitungan nilai akurasi model terhadap dataset. Yang dimaksud dengan akurasi adalah rasio jumlah prediksi yang benar atau prediksi yang sesuai dengan label aktual terhadap total jumlah sampel yang digunakan. Nilai akurasi akan merujuk pada kesesuaian tabel aktual dengan tabel prediksi. Nilai akurasi yang didapatkan pada penelitian ini adalah 71,79%. Nilai akurasi ini cukup terbilang baik dalam mengklasifikasi dataset yang tersedia. Semakin tinggi nilai akurasi maka semakin baik model dalam mengklasifikasi dataset.

Pada perhitungan diatas juga terdapat perhitungan *mean absolute percentage error* (MAPE). MAPE merupakan salah satu metode yang dapat digunakan untuk menghitung nilai error pada metode. Nilai MAPE yang didapatkan adalah 21.37%. Nilai tersebut masih dikatakan cukup baik dalam model yang digunakan. Semakin kecil nilai dari MAPE maka akan semakin kecil kesalahan atau *error* yang diprediksi dan menandakan bahwa model semakin bagus dalam memprediksi suatu dataset. Akan tetapi, perhitungan MAPE tidak tepat digunakan untuk kasus klasifikasi karena MAPE digunakan untuk menentukan rentang nilai dari sebuah prediksi. Akan tetapi, MAPE bisa dijadikan salah satu pertimbangan akurasi pada kasus selain klasifikasi.



Gbr. 8 Confusion Matrix

Gbr 8 merupakan *confusion matrix* yang didapatkan pada klasifikasi dataset ILPD. Target dari klasifikasi ini adalah mengidap penyakit *liver* atau tidak *liver*. Dari hasil yang didapatkan kita bisa menentukan true positif, true negatif, false negatif dan false positif pada matrik tersebut. Dari hasil *confusion matrix* diatas dapat disimpulkan sebagai berikut:

1. True Positif: terdapat 14 sampel yang benar-benar diklasifikasikan sebagai "Liver" dan diprediksi dengan benar
2. True Negatif: terdapat 70 sampel yang benar-benar diklasifikasikan sebagai "Tidak Liver" dan diprediksi dengan benar
3. False Positif: terdapat 17 sampel yang sebenarnya "Tidak Liver" tetapi salah diklasifikasikan sebagai "Liver"
4. False Negatif: terdapat 16 sampel yang sebenarnya "Liver" tetapi salah diklasifikasikan sebagai "Tidak Liver"

```
# Classification Report
classification_rep = classification_report(y_test, y_pred)
print("Classification Report:")
print(classification_rep)
```

	precision	recall	f1-score	support
1	0.81	0.80	0.81	87
2	0.45	0.47	0.46	30
accuracy			0.72	117
macro avg	0.63	0.64	0.63	117
weighted avg	0.72	0.72	0.72	117

Gbr. 9 Classification Report

*Classification report* pada Gbr. 9 menampilkan presisi, *recall*, f1-score dan *support* dari hasil klasifikasi. Presisi untuk kelas 1 adalah 0.81, yang berarti 81% dari prediksi "Tidak Liver" (kelas 1) dan 0,45 atau 45% untuk prediksi "Liver" (kelas 2) adalah benar. *Recall* digunakan untuk mengukur seberapa baik model dalam menemukan kembali sampel-sampel positif. *Recall* untuk kelas 1 adalah 0.80, yang berarti 80% dari sampel-sampel yang sebenarnya "Tidak Liver" ditemukan dengan benar. Sedangkan *recall* untuk kelas "Liver" 47% ditemukan dengan benar. F1-score merupakan nilai rata-rata antara presisi dan *recall*. Nilai F1-score tertinggi dipegang oleh kelas "Tidak Liver". Support adalah jumlah sampel yang digunakan dalam setiap kelas. Pada kelas 1 digunakan 87 sampel dan kelas 2 menggunakan 30 sampel. Dari *classification report* ini dapat diperoleh jumlah true positif, true negatif, false negatif dan false positif seperti *confusion matrix* pada gbr 8. Nilai akurasi pada *classification report*

adalah 72 yang merupakan bilangan bulat dari hasil akurasi 71,79%. Pada *report* tersebut juga diberikan nilai rata-rata yang menggunakan bobot dan tidak menggunakan bobot. Nilai rata-rata akurasi yang menggunakan nilai bobot adalah 0,72 dan 0,63 untuk yang tidak menggunakan nilai bobot.

## V. KESIMPULAN DAN SARAN

Klasifikasi dataset *ILPD* dilakukan dengan beberapa tahap seperti *preprocessing* data, pemrosesan data, dan evaluasi data. Pada saat pemrosesan data digunakan model *Artificial Neural Network* (ANN) untuk mengklasifikasi dataset *ILPD*. Hasil dari klasifikasi dihasilkan nilai akurasi 71,79% yang menandakan model berjalan dengan baik dalam melakukan klasifikasi pada dataset. Selain itu, klasifikasi juga menghasilkan 21,37% nilai error yang menandakan bahwa tidak terlalu banyak error atau yang terjadi pada saat proses klasifikasi. Pada evaluasi dataset ditentukan nilai true negatif, true positif, false negatif dan false positif pada dataset. Hasil yang didapatkan adalah TN 70, TP 14, FN 16, FP 17. Hasil *classification report* juga menunjukkan nilai presisi, *recall*, *f1-score*, dan *support* pada kelas “Tidak *Liver*” dan “*Liver*”. Hasil nilai pada kelas “Tidak *Liver*” lebih besar daripada kelas “*Liver*” karena sampel yang digunakan lebih banyak pada kelas “Tidak *Liver*”. Metode ANN dikatakan berjalan dengan baik pada dataset *ILPD* karena hasil nilai akurasi dan MAPE yang didapat terbilang cukup tinggi dan rendah untuk *error*.

Sistem yang dibuat pada penelitian ini telah menghasilkan keakurasian yang cukup baik untuk memprediksi penyakit *liver* berdasarkan dataset yang digunakan. Akan tetapi, dibutuhkan pengembangan lebih lanjut dalam penentuan nilai bobot awal dan *learning rate* pada *Algoritma Backpropagation* yang digunakan. Metode yang dapat diterapkan, misalnya dengan memanfaatkan teknologi *Machine Learning*.

## DAFTAR PUSTAKA

- [1] E. Drtisas and M. Trigka, “Supervised Machine Learning Models for Liver Disease Risk Prediction”, *Machine and Deep Learning in the Health Domain (Special Issue)*, MDPI, 2023.
- [2] A. P. Ayudhitama and U. Pujianto, “Analisa 4 Algoritma Dalam Klasifikasi Liver Menggunakan Rapidminer”, *J. Inform. Polinema*, vol. 6, no. 2, pp. 1–9, 2020, doi: 10.33795/jip.v6i2.274.
- [3] M. R. F. Rizki, “Perbandingan Algoritma Klasifikasi Untuk Prediksi Penyakit Liver,” *Reputasi J. Rekayasa Perangkat Lunak*, vol. 1, no. 2, pp. 82–88, 2020, doi: 10.31294/reputasi.v1i2.109.
- [4] R. H. Bawafi, “Sistem Prediksi Diagnosa Penyakit Hepatitis Menggunakan Metode Artificial Neural Network (Ann) Single Layer Perceptron Studi Kasus Pada Puskesmas Tambak,” *Inform. Comput. Intell. J.*, vol. 4, no. 2, p. 85, 2022, doi: 10.30587/indexia.v4i2.3526.
- [5] W. Fadri, “Klasifikasi Penyakit Hati dengan Menggunakan Metode Naive Bayes,” *J. Inf. dan Teknol.*, vol. 5, no. 1, pp. 32–36, 2023, doi: 10.37034/jidt.v5i1.230.
- [6] I. Setiawati, A. P. Wibowo, and A. Hermawan, “Pendahuluan Tinjauan Pustaka Penelitian Sebelumnya Klasifikasi,” *J. Inf. Syst. Manag.*, vol. 1, no. 1, pp. 13–17, 2019.
- [7] A. Desiani, “Perbandingan Implementasi Algoritma Naïve Bayes dan K-Nearest Neighbor Pada Klasifikasi Penyakit Hati,” *J. Sist. Inf. dan Sist. Komput.*, vol. 7, no. 2, pp. 104–110, 2022, doi: 10.51717/simkom.v7i2.96.
- [8] H. Hananti and K. Sari, “Perbandingan Metode Support Vector Machine (SVM) dan Artificial Neural Network (ANN) pada Klasifikasi Gizi Balita,” *Semin. Nas. Off. Stat.*, vol. 2021, no. 1, pp. 1036–1043, 2021, doi: 10.34123/semnasoffstat.v2021i1.1014.
- [9] D. Pradana, M. Luthfi Alghifari, M. Farhan Juna, and D. Palaguna, “Klasifikasi Penyakit Jantung Menggunakan Metode Artificial Neural Network,” *Indones. J. Data Sci.*, vol. 3, no. 2, pp. 55–60, 2022, doi: 10.56705/ijodas.v3i2.35.
- [10] M. D. Yalidhan and M. F. Amin, “Implementasi Algoritma *Backpropagation* untuk Memprediksi Kelulusan Mahasiswa”, *Kumpulan Jurnal Ilmu Komputer (KLIK)*, Volume 5, No. 02, 2018
- [11] F. A. Hizham, Y. Nurdiansyah, and D. M. Firmansyah, “Implementasi Metode *Backpropagation* Neural Network (BNN) dalam Sistem Klasifikasi Ketepatan Waktu Kelulusan Mahasiswa”, *Berkala Sainstek*, Vol. VI, No. 2, 2018
- [12] Syaharuddin, E. Pujiana, I. P. Sari, V. M. Mardika, and M. Putri, “Analisis Algoritma *Backpropagation* dalam Prediksi Angka Kemiskinan di Indonesia”, *Jurnal Pendidikan Berkarakter*, Vol. 3, No. 1, 2020
- [13] Y. Yordanov, G. Tsenov and V. Mladenov, “Humanoid robot control with EEG brainwaves,” 2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Bucharest, Romania, 2017, pp. 238–242, doi: 10.1109/IDAACS.2017.8095083.
- [14] I. Irmawati, K. Widiyanto, F. Aziz, A. Rifai, and A. Rahmawati, “Implementasi artificial neural network dalam mendeteksi penyakit hati (liver),” *J. Inf. Syst. Applied, Manag. Account. Res.*, vol. 6, no. 1, pp. 193–198, 2022, doi: 10.52362/jisamar.v6i1.694.
- [15] D. B. Rakshith, M. Srivastava, A. Kumar, and S.P. Gururaj, “Liver Disease Prediction System using Machine Learning Techniques”, *International Journal Of Engineering Research & Technology (IJERT)*, Vol. 10, Issue 6, 2021