Implementation of the K-Nearest Neighbor Algorithm for Detecting Heart Attack Disease

Delima Sitanggang, Evta Indra, Juan Hardoni Gulo and Mardi Turnip

***Abstract*—Heart damage can be fatal to human health, and even some heart disorders can cause death. A sudden reduction in blood supply to the heart can occur when one of the coronary arteries is temporarily blocked due to a blood clot. The portion of the heart muscle normally supplied by the blocked pulse ceases to function properly as soon as the plasma subsides on its own, symptoms disappear completely, and the heart muscle functions normally again. Basically, heart attack disease can be analyzed as its emergence early on. However, due to a lack of knowledge, many people are late to realize it. Early detection of heart attack can be done by analysis. In this study, the analysis method used was Machine Learning using K-Nearest Neighbor algorithm. The algorithm looks for the best results so that the initial diagnosis error can be minimized. The K-Nearest Neighbor algorithm in this study provides the best accuracy of 90.16% in classifying 303 datasets of patients with heart attack detection.**

***Index Terms*— Heart Attack, K-Nearest Neighbor, Classification, Data Mining.**

I. INTRODUCTION

HEART attack is a serious problem in the health sector. In 2015, the World Health Organization (WHO) said more than 17 million people died from heart attacks and blood vessel damage. The number of deaths is equivalent to 31% of the global death toll. About 8.7 million people died of heart disease. The disease is common in many lower-middle-income developing countries [1]. A heart attack is a severe heart disorder. This disorder occurs when the heart muscle does not get good blood flow [2, 3]. A sudden reduction in the blood supply to the heart can occur when one of the coronary arteries is temporarily blocked due to a blood clot [4].

When the blood supply to the heart stops completely, the cells in question undergo permanent changes in just a few hours, and parts of the heart muscle are deformed or permanently damaged [5]. Heart attack disease can be analyzed for its emergence early on. But due to a lack of knowledge, many people find out too late [6, 7]. This situation can be fatal to health in the future if not detected early. One of the best solutions to treat heart disease is early detection [8].

The research conducted in [9] entitled "Stacking Algorithm for Classification of Heart Disease in Imbalanced Class Datasets" uses a stacking algorithm. Based on the tests carried out, it shows that the stacking algorithm is only able to produce an accuracy of 81%, and research conducted by [10] entitled "Attribute Reduction in the Heart Disease Dataset and Classification Using the C5.0 Algorithm" obtained an accuracy value of 82.92% on the C5.0 algorithm using the gaint ration and max_depth 20 criteria, so that higher accuracy is still needed in detecting heart attack with use another method. In this regard, the authors are interested in conducting a study of the classification of heart attack disease with the K-Nearest Neighbor algorithm.

II. METHOD

A. Types of Research

Some of the factors that cause heart disease are generally unhealthy living habits, smoking, staying up late, and poor diet [11]. The main causes of heart disease are tobacco use, physical inactivity, unhealthy diet and alcohol use, age, high blood pressure, and being overweight [12].

K-Nearest Neighbor (K-NN) is an instance-based learning group. The K-NN is done by looking for groups of k objects in the training data that are closest (similar) to the objects in the new data or test data [13]. Examples of cases include finding solutions to new patient problems using solutions from old patients [14]. To find solutions for new patients, proximity to old cases was used, and solutions from old cases that are close to new cases were used [15].

The best value of k for this algorithm depends on the data. In general, the value of k will reduce noise in the classification but the boundary between each classification [16]. A good value of k can be selected with optimization parameters, for example, by using cross-validation in some cases where the classification is predicted based on the closest training data (in other words, k=1) Nearest Neighbor algorithm [17].

Advantages of the KNN are Resilient to training data that has a lot of noise, and Effective when training data is significant. While its weakness is KNN needs to determine the value of the parameter k (sum of the nearest neighbors). Distance-based exercises are not clear about the distance that should be used: (i) Which attribute should be used to get the

Delima Sitanggang was with Faculty of Techonology and Computer Science, University of Prima Indonesia, Medan, Indonesia (e-mail: abdidharma@unprimdn.ac.id).

Evta Indra was with Faculty of Techonology and Computer Science, University of Prima Indonesia, Medan, Indonesia (e-mail: sonianovel140620@gmail.com).

Juan Hardoni Gulo was with Faculty of Techonology and Computer Science, University of Prima Indonesia, Medan, Indonesia (e-mail: tanyenny012@gmail.com).

Mardi Turnip was with Faculty of Techonology and Computer Science, University of Prima Indonesia, Medan, Indonesia (email: marditurnip@unprimdn.ac.id).

best results and (ii) High computational costs because it requires the calculation of each sample query on the entire training sample.

B. Work Procedures

So that the research can run according to the topics presented and completed on time, the research is designed according to the flow diagram as shown in Fig. 1.

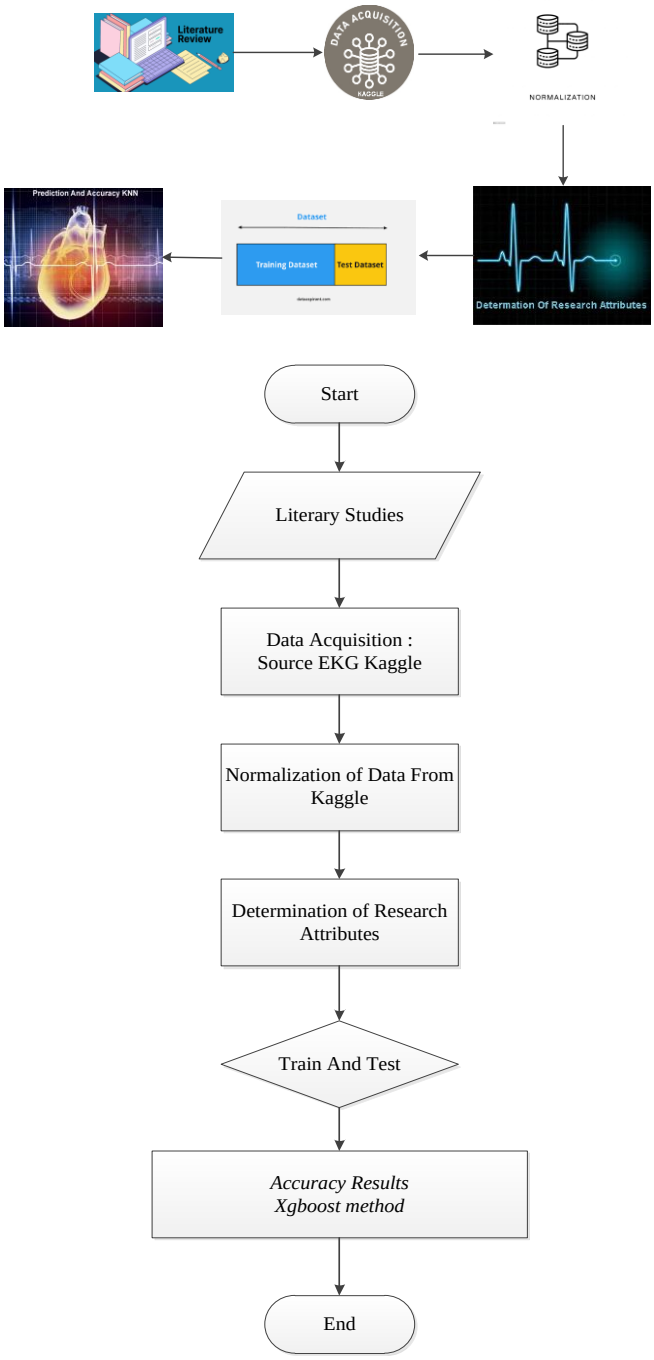

Fig. 1. Flowchart Research

A literature study is a research conducted by researchers by collecting many books and magazines related to the problem and research objectives. This technique is carried out to reveal various theories that follow the problem under study as reference material in discussing the results of the study. Data acquisition is a dataset retrieval process from Kaggle. This dataset was used to explore heart rate classification using a machine learning architecture and observe some of the transfer learning capabilities within it. Kaggle Data Normalization is carried out by eliminating empty fields or columns in the dataset and deleting data that are not needed in carrying out data analysis [18]. The attribute creation process consists of 11 types. Attributes describe and affect how the patient's body health condition, which will be processed data from attributes into machine learning to predict whether the patient has a heart attack or not. The process of training and testing the KNN algorithm to get accurate results in detecting abnormalities in the heart. To do the train and test in this study using the Python programming language with Google Colab. The final result of machine learning data processing uses the KNN algorithm to detect whether the patient has a heart attack.

III. Results and Discussions

A. Problem Analysis

In this study, data mining will be used to classify data from heart attack patients so that the output will be used for predicting whether the patient has a heart attack using the K-Nearest Neighbor algorithm.

B. Data Analysis

Early detection of heart attack can be done by analysis. In this case, Machine Learning can help make early detection easier and faster. To do this, Machine Learning requires a dataset that is used to train the network. The dataset used in this study is sourced from [19].

The dataset in this study has a total of 303 instances, with data in the form of numbers consisting of 14 attributes that are processed with the K-Nearest Neighbor algorithm. The following is a list of attributes in this study: Age: Patient's age; Sex: Patient's gender; cp: Chest pain typical of angina, atypical angina, non-angina pain, and asymptomatic; trtbps: high blood pressure (in mm Hg); chol : cholestoral in mg/dl taken via BMI sensor; fbs : (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false); Restecg: wave electrocardiography results: (i) 0: normal, (ii) Grade 1: have ST-T wave abnormalities (T wave inversion and/or ST elevation or depression > 0.05 mV) (iii) Score 2: indicates possible or definite left ventricular hypertrophy based on Estes criteria; thalach: maximum heart rate achieved; Exang: exercise induced angina (1 = yes; 0 = no); Oldpeak: ST depression caused by exercise relative to rest; Slope: the slope of the ST segment of the peak exercise: (i) Value 0 : uphill; (ii) Value 1: flat (iii) Value 2: downsloping; ca: number of main ships (0-3); Thalach: maximum heart rate reached; Output : Attribute prediction.

TABLE I
DATASET ATTRIBUTES

	Age	Sex	CP	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa	thall	output
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	0
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	0
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	0
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	0

```
cp_data= df['cp'].value_counts().reset_index()
cp_data['index'][3]= 'asymptomatic'
cp_data['index'][2]= 'non-anginal'
cp_data['index'][1]= 'Atyppical Anigma'
cp_data['index'][0]= 'Typical Anigma'
cp_data
```

	index	cp
0	Typical Anigma	143
1	Atyppical Anigma	87
2	non-anginal	50
3	asymptomatic	23

C. Electronic Image Files (Optional)

The stages of data preprocessing that will be carried out are data cleaning and discretization.

D. Data Cleaning

In this study, data clean or the process of calculating and deleting data that has no value or commonly referred to as null is carried out.

Source Code:

df.isnull().sum()

```
age         0
sex         0
cp          0
trtbps      0
chol        0
fbs         0
restecg     0
thalachh    0
exng        0
caa         0
output      0
dtype: int64
```

The second step is to delete the oldpeak, slp, thall columns with the drop command.

TABLE II
DROP DATASET COLUMN

```
df= df.drop(["oldpeak","slp",'thall'],axis=1)
df.head(400)
```

	Age	Sex	CP	trtbps	chol	fbs	restecg	thalachh	exng	caa	output
0	63	1	3	145	233	1	0	150	0	0	1
1	37	1	2	130	250	0	1	187	0	0	1
2	41	0	1	130	204	0	0	172	0	0	1
3	56	1	1	120	236	0	1	178	0	0	1
4	57	0	0	120	354	0	1	163	1	0	1
…	…	…	…	…	…	…	…	…	…	…	…
298	57	0	0	140	241	0	1	123	1	0	0
299	45	1	3	110	264	0	1	132	0	0	0
300	68	1	0	144	193	1	1	141	0	2	0
301	57	1	0	130	131	0	1	115	1	1	0
302	57	0	1	130	236	0	0	174	0	1	0

E. Discretionary Data

The second stage after data cleaning is complete is the data discretization stage by categorizing the value of each attribute. The explanation of the dataset categories will be explained as follows:

Age

This attribute is categorized into 2 values: age under 54 years is categorized as 0, and age 54 years is categorized as 1.

```
plt.figure(figsize=(15,7))
plt.title("Sex of Patients,0=Female and 1=Male")

sns.countplot(x='sex',data=df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f75fab86790>
```

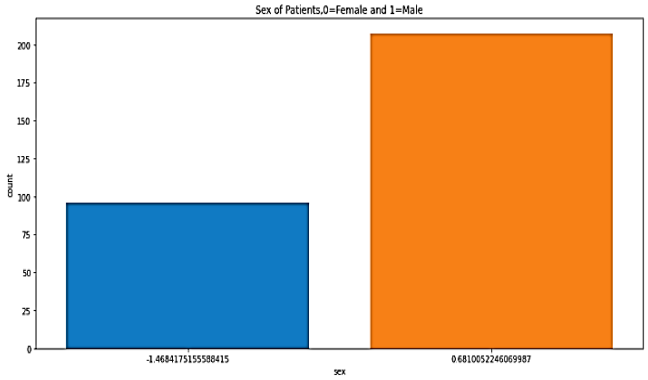

Fig. 2. Age Atrribute

Sex

This attribute has 2 categories, so it will only be converted to 2 symbols. A value of 1 is female and represents a male with a value of 0.

```
plt.figure(figsize=(15, 7))
plt.title("Chest Pain of Patients")

sns.barplot(x=cp_data['index'],y= cp_data['cp'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0ba22d8910>
```

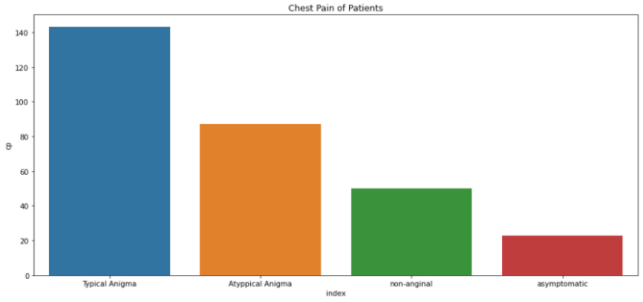

Fig. 3. Sex Atrribute

Cp

This attribute has 4 categories, and changed to 4 typical symbols Anigma with a value of 0, Atypical Anigma with a value of 1, Non Anginal with a value of 2, Asymptomatic with a value of 3.

```
cp_data= df['cp'].value_counts().reset_index()
cp_data['index'][3]= 'asymptomatic'
cp_data['index'][2]= 'non-anginal'
cp_data['index'][1]= 'Atyppical Anigma'
cp_data['index'][0]= 'Typical Anigma'
cp_data
```

	index	cp
0	Typical Anigma	143
1	Atyppical Anigma	87
2	non-anginal	50
3	asymptomatic	23

Fig. 4. CP Atrribute

Trtbps

This attribute has a range of values from 80 to 180 with units of mmHg.

```
plt.figure(figsize=(15 ,7))
plt.subplot(1,2,1)
sns.distplot(df['trtbps'], kde=True, color = 'magenta')
plt.xlabel("Resting Blood Pressure (mmHg)")
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.
  warnings.warn(msg, FutureWarning)
Text(0.5, 0, 'Resting Blood Pressure (mmHg)')
```

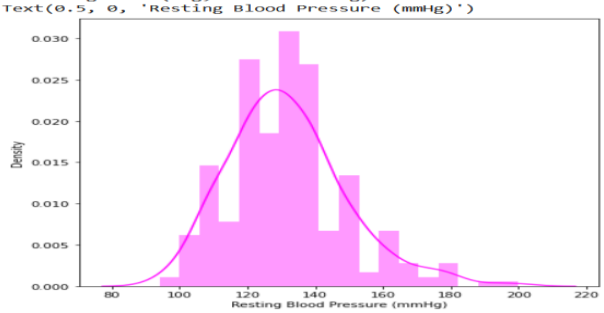

Fig.5. Trtbps Atrribute

Cholostrol

This attribute has a range of 0 to 600.

```
plt.figure(figsize=(5,5))
sns.distplot(df['chol'], kde=True, color = 'red')
plt.xlabel("Cholestrol")
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions
  warnings.warn(msg, FutureWarning)
Text(0.5, 0, 'Cholestrol')
```

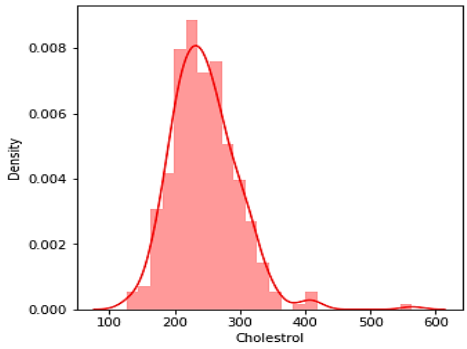

Fig. 6. Cholestrol Atrribute

Fbs

This attribute has 2 categories, so it will only be converted to 2 symbols. False is denoted by a value of 0 and True is represented by a value of 1.

```
plt.figure(figsize=(10,7))
plt.title("FBS ,0=False and 1=True")

sns.countplot(x='fbs',data=df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0ba1e8c1d0>
```

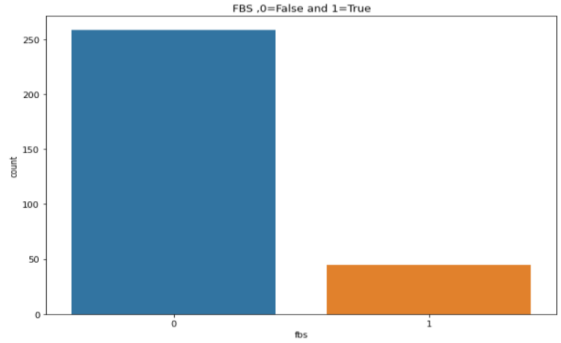

Fig. 7. Fbs Atrribute

Restecg

This attribute has 3 categories, and is changed to 3 symbols. Normal is denoted by a value of 0, having ST-T wave abnormality is symbolized by a value of 1, and showing probable or definite left ventricular hypertrophy by Estes' criteria is symbolized by a value of 2.

```
plt.figure(figsize=(15, 7))
plt.title("Chest Pain of Patients")

sns.barplot(x=cp_data['index'],y= cp_data['cp'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0ba22d8910>
```

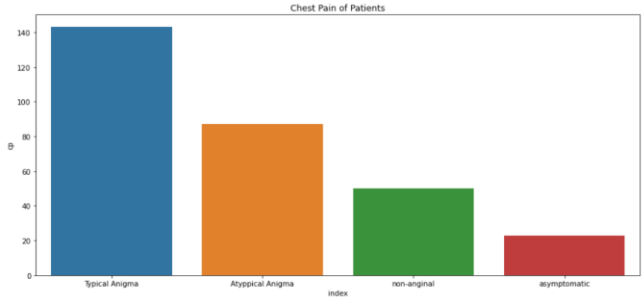

Fig. 8. Restecg Atrribute

Thalach

This attribute has a range of values between 70 to 200 in bpm units.

```
plt.figure(figsize=(20,5))
plt.subplot(1,2,2)
sns.distplot(df['thalachh'], kde=True, color = 'teal')
plt.xlabel("Maximum Heart Rate Achieved (bpm)")
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning:
  warnings.warn(msg, FutureWarning)
Text(0.5, 0, 'Maximum Heart Rate Achieved (bpm)')
```

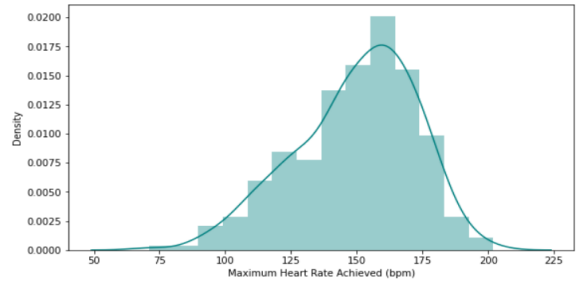

Fig. 9. Thalach Atrribute

Exang

This attribute has 2 categories, so it will only be converted to 2 symbols. False is denoted by a value of 0 and True is represented by a value of 1.

```
plt.figure(figsize=(10,7))
plt.title("Exang ,0=False and 1=True")

sns.countplot(x='exng',data=df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0ba22ced50>
```

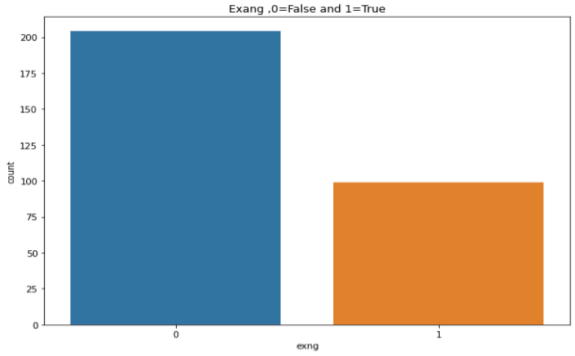

Fig. 10. Exang Atrribute

Caa

This attribute is categorized into 3 values, namely value 0, value 1, value 2, value 3, value 4.

```
ecg_data= df['caa'].value_counts().reset_index()
ecg_data['index'][0]= '0'
ecg_data['index'][1]= '1'
ecg_data['index'][2]= '2'
ecg_data['index'][2]= '3'
ecg_data['index'][2]= '4'

ecg_data
```

	index	caa
0	0	175
1	1	65
2	4	38
3	3	20
4	4	5

Fig. 11. Caa Atrribute

F. Final Result

Sample Data

The application of the K-NN algorithm at this stage is the determination of training data and test data, the application of the K-NN algorithm to the calculation of performance or accuracy. The following 11 samples of data that will be used for data processing in Table 1 show 11 data samples

TABLE III
DATA SAMPLE

```
df=pd.DataFrame(df,columns=['age', 'sex', 'cp', 'trtbps', 'chol', 'fbs', 'restecg', 'thalachh', 'exng', 'caa', 'output'])
df
```

	Age	Sex	CP	trtbps	chol	fbs	restecg	thalachh	exng	caa	output
0	63	1	3	145	233	1	0	150	0	0	1
1	37	1	2	130	250	0	1	187	0	0	1
2	41	0	1	130	204	0	0	172	0	0	1
3	56	1	1	120	236	0	1	178	0	0	1
4	57	0	0	120	354	0	1	163	1	0	1
…	…	…	…	…	…	…	…	…	…	…	…
298	57	0	0	140	241	0	1	123	1	0	0
299	45	1	3	110	264	0	1	132	0	0	0
300	68	1	0	144	193	1	1	141	0	2	0
301	57	1	0	130	131	0	1	115	1	1	0
302	57	0	1	130	236	0	0	174	0	1	0

Ttain and Test

Train and Testcarried out on the sample data using the K-Nearst Neighbor method with the following formula:
KNN formula:

$$d_i = \sqrt{\sum_{i=1}^{p} \left(x_{2i} - x_{1i}\right)^2} \quad (1)$$

where x_1 is Sample Data, x_2 is Test Data / Testing, I is data Variable, d is distance, p is Data Dimension.

After classifying all the data, the results showed that from 303 patient data there were 165 people with a small risk of heart attack and 138 people with a greater chance of having a heart attack.

Accuracy

After all the process stages are complete, the researcher divides the dataset into two parts, namely as training data and testing data. The percentage of distribution made to the dataset is 80% for the training data, while the 20% model data with random state 42 and the Google Colab default scaler is used for validation of training data or testing data. In this study, the algorithm used is the K-NN algorithm.

```
plt.figure(figsize=(15, 7))
plt.title("Heart Attack Prediction")

sns.barplot(x=output_data['index'],y= output_data['output'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fea85b2fed0>
```

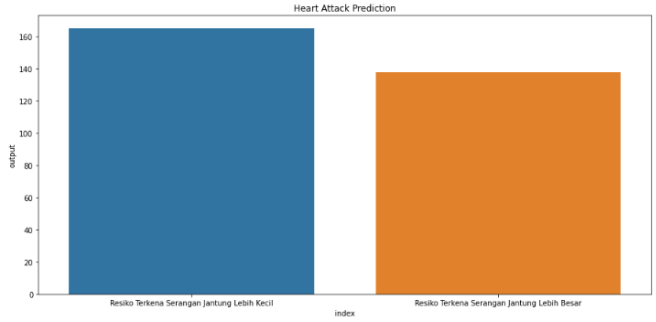

Fig. 12. Result Prediction

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20, random_state=42)
```

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.preprocessing import LabelEncoder
lbl= LabelEncoder()
encoded_y= lbl.fit_transform(y_train)
encoded_ytest= lbl.fit_transform(y_test)
```

Fig. 13. Split Test and Train Data

Checking the correct K value is carried out 40 times to get the K value that has the best truth to get maximum accuracy. From the best examination value. K1 to K40 using the python library, the best K13 truth value is obtained with a truth value of 0.10.

```
plt.figure(figsize=(10,5))
plt.plot(range(1,40),error_rate,color='blue', linestyle='dashed', marker='o',
         markerfacecolor='red', markersize=10)
plt.xlabel('K Vlaue')
plt.ylabel('Error rate')
plt.title('To check the correct value of K')
plt.show()
```

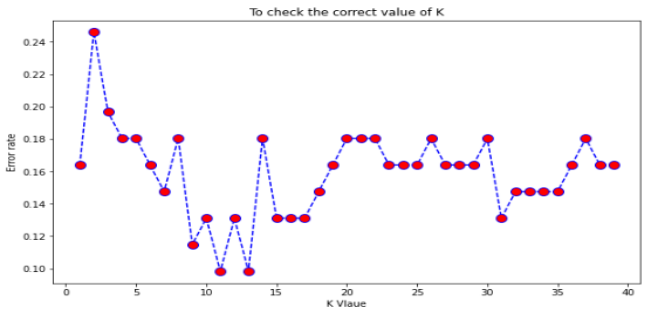

Fig. 14. K 13 Truth Value Graph

After getting the best correct value of K13 on the train data and test data, the accuracy is calculated and obtained an accuracy value of 90.16%.

```
knn= KNeighborsClassifier(n_neighbors=13)
knn.fit(x_train,encoded_y)
ypred4= knn.predict(x_test)
knn_conf_matrix = confusion_matrix(encoded_ytest,ypred4 )
knn_acc_score = accuracy_score(encoded_ytest, ypred4)
knn_conf_matrix
hasil=(knn_acc_score*100)

print("Confussion Matrix :")
print(knn_conf_matrix)
print("The Result of K-13 Accuracy Value is :",hasil,"%")
```

```
Confussion Matrix :
[[26  3]
 [ 3 29]]
The Result of K-13 Accuracy Value is : 90.1639344262295 %
```

Fig. 15. Value Accuracy

IV. CONCLUSION

In this study, predictions of heart attack disease were made to 303 patients and it was found that there were 165 people with a lower risk of heart attack and 138 people with a greater chance of having a heart attack. Prediction is done with the python program using the KNN method and obtains a fairly accurate accuracy value with a value of 90.16%.

ACKNOWLEDGMENT

In this study, predictions of heart attack disease were made to 303 patients and it was found that there were 165 people with a lower risk of heart attack and 138 people with a greater chance of having a heart attack. Prediction was done with the python program using the KNN method and obtained a fairly accurate accuracy value with a value of 90.16%.

REFERENCES

[1] Cardiovascular diseases (CVDs), Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds).

[2] N. Yudistira, and A.F. Putra, "Algoritma Decision Tree Dan Smote Untuk Klasifikasi Serangan Jantung Miokarditis Yang Imbalance," *Jurnal Litbang Edusaintech*, vol. 2, no. 2, pp.112-122, 2021.

[3] I. Prayogo, R. Alfita, and KA Wibisono, "Heart Rate and Body Temperature Monitoring System as an IOT (Internet of Thing)-Based Patient Health Level Indicator Using the Fuzzy Logic Method Using Android," *J. Tech. Electrical and Computing*, TRIAC, vol. 4, no. 2, 2017.

[4] M. Sofie and A. Rizal, "Classification of Electrocardiogram Signal Recording Images Using Texture Analysis Methods, K-Nn And Multilayer Perceptron," *Symmetrical J. Tek. Mechanical, Electrical and Computer Science*, vol. 7, no. 1, p. 233, 2016.

[5] A. Bataineh, W. Batayneh, T. Harahseh, K. Hijazi, A. Alrayes, M. Olimat, and A. Bataineh, "Early Detection of Cardiac Diseases from Electrocardiogram Using Artificial Intelligence Techniques," *International Review on Modelling and Simulations*, vol. 14, no.2, 2021

[6] A. Rohman, "Comparison of Data Mining Classification Methods for Predicting Heart Disease," *Neo Tech*, vol. 2, no. 2, pp. 21–28, 2017.

[7] E. B. Setiawan and M. Y. Tahir, "Electrocardiogram (ECG) Application for Checking Heart Conditions Based on Android," *IJNMT (International Journal of New Media Technology)*, vol. 6, no.1, June, 2019.

[8] A. Riani, Y. Susianto, and N. Rahman, "Implementation of Data Mining to Predict Heart Disease Using the Naive Bayes Method," *J. Innov inf. Technol. app*, vol. 2, no. 01, pp. 25–34, 2019.

[9] A. Nurmasani and Y. Pristyanto, "Algoritme Stacking Untuk Klasifikasi Penyakit Jantung Pada Dataset Imbalanced Class," *Pseudocode*. vol. 8, no. 1, pp. 21–26, 2021.

[10] D. P. Utomo, P. Sirait, and R. Yunis, "Reduksi Atribut Pada Dataset Penyakit Jantung dan Klasifikasi Menggunakan Algoritma C5. 0," *J. Media Inform. Budidarma*, vol. 4, no. 4, pp. 994–1006, 2020.

[11] G. Purnamasari, "Analisis Sinyal Ekg Aritmia Untuk Deteksi Risiko Jantung Koroner Menggunakan Adaptive Neuro Fuzzy Inference (ANFIS)," *Jurnal Kesehatan*, vol. 1, no. 1, pp. 40–44, 2019,

[12] E. Indra et al., "A Comparison of Heart Abnormalities Detection on ECG using KNN and Decision Tree," *Internetworking Indonesia Journal*, vol. 12, no. 2, pp. 19–23, 2020.

[13] H. M. Nawawi, J. J. Purnama, and A. B. Hikmah, "Komparasi Algoritma Neural Network Dan Naïve Bayes Untuk Memprediksi Penyakit Jantung," *J. Pilar Nusa Mandiri*, vol. 15, no. 2, pp. 189–194, 2019.

[14] R. Annisa, "Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Penderita Penyakit Jantung," *JTIK (Jurnal Teknik Informatika Kaputama)*, vol. 3, no. 1, pp. 22-28, 2019.

[15] H. Azis, P. Purnawansyah, F. Fattah, and I. P. Putri, "Performa Klasifikasi K-NN dan Cross Validation Pada Data Pasien Pengidap Penyakit Jantung," *ILKOM Jurnal Ilmiah*, vol. 12, no. 2, pp. 81–86, 2020.

[16] D. P. Utomo and M. Mesran, "Comparative Analysis of Data Mining Classification Methods and Attribute Reduction in Heart Disease Data Sets," *J. Media Inform. Budidharma*, vol. 4, no. 2, p. 437, 2020.

[17] M. E. I. Lestari, "Penerapan algoritma klasifikasi Nearest Neighbor (K-NN) untuk mendeteksi penyakit jantung," *Faktor Exacta*, vol. 7, no. 4, pp. 366-371, 2015.

[18] A. Septiani, "Classification of heart disease using the recurrent neural network (RNN) method," 2020.

[19] "Heart Attack Analysis and Prediction Dataset Kaggles," https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset