

Efektifitas Algoritma Data Mining dalam Menentukan Pendoror Darah Potensial

Asry Yuniarti^{1*}, Aldi Yasin², Yohanes Adi Nugroho³

^{1,2}Program Studi Pendidikan Sistem dan Teknologi Informasi, Universitas Pendidikan Indonesia
Jl. Veteran 08, Purwakarta
Email: *asryyuniarti@upi.edu

Abstrak. Darah merupakan salah satu cairan fungsional yang memiliki banyak manfaat sehingga sangat dibutuhkan oleh tubuh manusia agar dapat bertahan hidup. Persediaan darah di suatu negara setiap tahunnya harus terpenuhi sesuai dengan ketentuan dari WHO. Namun kebutuhan darah di Indonesia hingga saat ini masih belum terpenuhi. Hal ini mengakibatkan banyak orang sulit untuk mendapatkan darah dan harus mencari calon pendonor darah. Penelitian ini memuat solusi untuk mendapatkan calon pendonor darah potensial dengan memanfaatkan kemajuan teknologi yaitu teknik data mining. Penelitian ini bertujuan untuk mencari keefektifan dari 2 algoritma klasifikasi data mining yang diimplementasikan ke dalam Bahasa Pemrograman Python dengan *tools Jupyter Notebook* dan *software RapidMiner*. Hasil penelitian menghasilkan bahwa algoritma KNN mempunyai *accuracy* 78,00% dengan menggunakan *software RapidMiner*, dan 68,66% dengan menggunakan *tools Jupyter Notebook* sedangkan algoritma Naive Bayes menghasilkan nilai *accuracy* 71,33%, dan 73,33%. Dilihat dari nilai *Accuracy* menunjukkan bahwa algoritma KNN yang diimplementasikan menggunakan *software RapidMiner* memiliki nilai *accuracy* paling tinggi sehingga dalam penelitian ini algoritma KNN merupakan algoritma yang paling efektif dalam mencari calon pendonor darah

Kata kunci: Donor Darah, Data Mining, KNN, Naive Bayes

1 Pendahuluan

Darah merupakan cairan dalam tubuh yang berfungsi untuk mengangkut oksigen ke seluruh tubuh. *World Health Organization* (WHO) menetapkan standar kebutuhan darah di suatu negara adalah 2% dari jumlah penduduknya. Namun, dilansir dari *mediaindonesia.com* kebutuhan darah di Indonesia baru terpenuhi 92%.

Kurangnya unit darah yang tersedia mengakibatkan lambatnya proses pengobatan serta memperburuk kondisi penyakit pasien bahkan dapat menyebabkan kematian pada pasien yang mengalami kekurangan darah. Pernyataan di atas sejalan dengan berita yang dilansir dari *beritasatu.com* bahwa “28% ibu melahirkan meninggal akibat pendarahan dan tidak tertolong karena

tidak tersedianya pasokan darah setelah tindakan persalinan". Untuk mengisi kekurangan stok kantong darah di Indonesia biasanya didapatkan dengan donor darah. Donor darah merupakan proses pengambilan darah dari seseorang yang nantinya digunakan untuk transfusi darah. Tidak sembarang orang dapat melakukan donor darah, karena ada beberapa persyaratan yang harus dipenuhi ketika akan melakukan donor darah, seperti kondisi tubuh yang sehat, tekanan darah yang normal, dan beragam alasan kesehatan lainnya yang berkaitan dengan kualifikasi sebagai pendonor darah. Sehingga tidak mudah dalam mencari calon pendonor darah yang layak atau berpotensi.

Nur, Nahya. dkk (2021) menjelaskan bahwa berdasarkan hasil penelitiannya dengan menggunakan bahasa pemrograman PHP, dilihat dari nilai akurasi algoritma *K-Nearest Neighbor* (K-NN) lebih baik untuk melakukan klasifikasi calon pendonor darah dibandingkan dengan algoritma Naive Bayes [11]. Sedangkan dalam penelitian Hutrim dan Nurahman (2020) menjelaskan bahwa data pendonor dari PMI Kabupaten Kotawaringin Timur yang diuji dengan algoritma Naive Bayes menggunakan *software* WEKA memiliki nilai akurasi sebesar 77,00% dan dikategorikan cukup baik dalam menentukan calon pendonor darah [12]. Pada penelitian Lestandy, dkk (2020) klasifikasi pendonor darah potensial dilakukan menggunakan *software* RapidMiner dengan algoritma KNN, naïve Bayes dan *neural network*. Hasil penelitiannya menunjukkan bahwa algoritma *neural network* mempunyai nilai akurasi yang paling tinggi yaitu sebesar 84,3%, jika dibandingkan dengan algoritma Naive Bayes yang memiliki nilai akurasi sebesar 84,167% dan KNN yang memiliki nilai akurasi yang paling rendah yaitu sebesar 75% [13].

Berdasarkan pembahasan sebelumnya, penelitian ini bertujuan untuk mempermudah masyarakat dalam mencari pendonor darah dengan penerapan klasifikasi data mining menggunakan algoritma KNN dan Naive Bayes dengan menggunakan *software* RapidMiner dan Bahasa Pemrograman Python dalam menentukan seseorang apakah layak untuk mendonorkan darahnya berdasarkan kriteria yang telah ditentukan atau tidak. Algoritma *K-Nearest Neighbor* (KNN) merupakan suatu metode klasifikasi yang sederhana dan efektif dan melakukan klasifikasi sesuai dengan banyaknya kedekatan jarak yang mayoritas terhadap data yang diklasifikasi (Santoso, 2007)[14]. Algoritma Naive Bayes merupakan klasifikasi dengan probabilitas yang sederhana. Namun memiliki nilai hasil akurasi yang bagus dan mempunyai kinerja yang baik meskipun diimplementasikan untuk data dengan jumlah yang besar[15] Sehingga kedua algoritma KNN dan Naive Bayes ini menarik untuk dibandingkan.

Penelitian ini menggunakan pendekatan kuantitatif dengan teknik pengumpulan data studi literatur dan pengolahan datanya menggunakan proses data mining. Melalui penelitian ini penulis akan membandingkan algoritma yang paling efektif antara KNN dan Naive Bayes dalam mencari pendonor darah potensial dengan menggunakan dataset yang diperoleh dari UCI Data Repository.

2 Metode Penelitian

Penelitian ini merupakan penelitian eksperimen dengan menggunakan pendekatan kuantitatif. Menurut Sugiyono (2015:14) bahwa pendekatan kuantitatif merupakan penelitian yang berlandaskan pada filsafat positivisme untuk meneliti populasi atau sampel tertentu dan pengambilan sampel secara random dengan pengumpulan data menggunakan instrumen, analisis data bersifat statistik[16]. Pengumpulan data yang digunakan adalah studi literatur dari artikel penelitian terdahulu yang relevan dengan topik yang diangkat. Menurut J. Supranto seperti yang dikutip Ruslan dalam bukunya metode Penelitian *Public Relations* dan Komunikasi, bahwa studi kepustakaan adalah dilakukan mencari data atau informasi riset melalui membaca jurnal ilmiah, buku-buku referensi dan bahan-bahan publikasi yang tersedia di perpustakaan (Ruslan, 2008:31)[17]. Sedangkan menurut Danial dan Warsiah (2009:80), Studi Literatur merupakan penelitian yang dilakukan oleh peneliti dengan mengumpulkan sejumlah buku buku, majalah yang berkaitan dengan masalah dan tujuan penelitian[18]. Sumber data yang digunakan berupa data sekunder yang diperoleh dari website UCI Data *Repository* dan teknik pengolahan data yang digunakan adalah proses data mining. Untuk melahirkan sebuah pengetahuan atau *knowledge*, terdapat beberapa proses yang harus dilewati. Tahapan proses data mining ini terdiri dari :

- a. Pembersihan Data (*Data Cleaning*)

Pembersihan data merupakan proses untuk menghilangkan noise data yang tidak relevan dan data yang tidak konsisten agar tidak menghambat laju ketika data diproses. Pada umumnya data yang diperoleh masih memiliki isian yang tidak sempurna dan tidak kompleks.
- b. Integrasi Data (*Data Integration*)

Integrasi data merupakan penggabungan dari beberapa data kedalam satu database. Integrasi data sangat diperlukan secara cermat, karena kesalahan pada integritas data bisa menghasilkan hasil menyimpang bahkan menyesatkan.
- c. Seleksi Data (*Data Selection*)

Seleksi data bertujuan untuk mengambil data yang diperlukan dalam proses mining. Seleksi data adalah data yang ada pada *database* dan umumnya tidak semuanya terpakai, hanya data yang konsisten dan valid yang akan diambil dari *database* untuk dianalisis.
- d. Transformasi Data (*Data Transformation*)

Transformasi data merupakan data yang diubah atau digabungkan ke dalam format yang sesuai untuk diproses dalam data mining.
- e. Proses Mining

Proses mining adalah proses utama saat metode akan diterapkan atau diimplementasikan untuk menemukan sebuah informasi dan pengetahuan yang bersumber dari database.

f. Evaluasi, Interpretasi Dan Visualisasi

Proses ini merupakan proses untuk melihat data yang sudah diolah, apakah data tersebut terdapat ketertarikan ataupun menarik untuk dilakukan identifikasi ke dalam *Knowledge Based*.

Dalam melakukan pengujian implementasi algoritma ada beberapa proses yang harus dilakukan yaitu :

a. Pengumpulan data

Pengumpulan data berdasarkan sumber terbagi menjadi dua yaitu data primer dan data sekunder. Data yang dipakai oleh penulis dalam mengklasifikasikan donor darah potensial adalah data sekunder yang didapat dari pihak ketiga yaitu *UCI Machine Learning Repository* dengan atribut RFMTC yang merupakan pengembangan dari metode *Recency, Frequency, Monetary* (RFM).

b. Transformasi Data

Penulis mengubah format data dari format .data ke format .csv dan mengubah kolom *label Classification* dari numerik (1, 0) menjadi kategorik karena untuk algoritma Naive Bayes tidak mendukung untuk labels numerik.

c. Proses Mining

Dalam proses mining ini penulis mengimplementasikan metode *Naive Bayes* dan K-NN dengan menggunakan *software* RapidMiner dan Bahasa Pemrograman Python (Jupyter Notebook).

d. Evaluasi, Interpretasi, dan Visualisasi

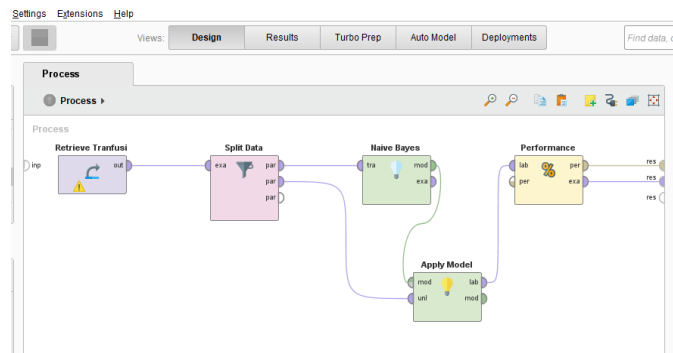
Pada tahap ini digunakan untuk mengevaluasi dan menganalisis bagaimana hasil dari proses mining yang telah dilakukan pada RapidMiner dan pada Bahasa Pemrograman Python. Sehingga dari hasil evaluasi dan analisis yang telah dilakukan penulis dapat memberikan interpretasi atau pendapat mengenai metode yang paling tepat dalam mengklasifikasikan donor darah potensial yang divisualisasikan dengan sebuah gambar dari proses mining.

3 Hasil dan Pembahasan

Klasifikasi kelayakan pendonor darah diimplementasikan dengan algoritma *Naive Bayes* dan K-NN. Pengujian algoritma KNN dan Naive Bayes dilakukan dengan menggunakan *software* RapidMiner dan Bahasa Pemrograman Python. Dengan jumlah dataset 748 yang memiliki 4 atribut dan 1 label. Pengujian dilakukan sebanyak 1 (satu) kali dengan komposisi yang digunakan pada data training dan data test adalah 80% dan 20%.

Ketika melakukan pengujian dengan menggunakan *software* RapidMiner dan bahasa pemrograman python ada beberapa langkah yang dilakukan :

1. Input data yang akan digunakan
2. Split data untuk membagi antara data training dan data test dari data yang telah diinputkan.
3. Implementasi algoritma. Pilih sesuai dengan algoritma yang akan digunakan
4. *Apply* algoritma
5. *Performance*
6. Perhatikan apakah ada garis yang *error* atau tidak. Jika tidak terdapat maka dapat melihat langsung hasil pengujian yang telah dilakukan



Gambar 1. Langkah Pengujian Pada Software RapidMiner

Sedangkan untuk langkah-langkah pengujian dengan bahasa pemrograman python adalah sebagai berikut :

1. Definisikan *library* standar untuk *classification*. Disini penulis menggunakan :
 - a. `numpy as np` untuk memanipulasi atau mempermudah dalam pembentukan *array* atau matriks
 - b. `pandas as pd` untuk memanipulasi data
2. Definisikan *library* yang akan digunakan di dalam machine learning dalam penelitian ini penulis menggunakan :
 - a. `sklearn.naive_bayes` dengan *library* `MultinomialNB` dan `sklearn.neighbors` dengan *library* `kNeighborsClassifier` sebagai algoritma yang akan digunakan dalam penelitian ini
 - b. `sklearn.model_selection` dengan *library* `train_test_split` untuk membuat *splitting* atau memisahkan antara data *training* dan data *test* yang akan digunakan.
 - c. `sklearn.utils.multiclass` dengan *library* `unique_labels` untuk melihat keunikan label

- d. sklearn.preprocessing dengan *library* LabelEncoder untuk mengubah setiap nilai yang ada pada kolom menjadi angka yang berurutan
 - e. jcopml.plot dengan *library plot missing value* untuk melihat kelengkapan data yang dimiliki apakah ada data yang kosong atau tidak.
3. Definisikan *library* untuk memvisualisasikan data atau hasil pengujian yang dilakukan, penulis menggunakan *library* matplotlib.pyplot as plt dan sklearn.metrics classification_report, accuracy_score, confusion_matrix

```
# Definisikan Library yang digunakan

In [1]: #Library Standar
import numpy as np
import pandas as pd

#Library machine learning
from luwji.knn import illustration, demo
from jcopml.plot import plot_missing_value
from sklearn.preprocessing import LabelEncoder
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from sklearn.utils.multiclass import unique_labels

#Library visualisasi hasil
import matplotlib.pyplot as plt
from sklearn.metrics import classification_report, accuracy_score, confusion_matrix

# Import Data
```

Gambar 2. Mendefinisikan Library

4. Melakukan pengujian
- a. Import data

```
# Import Data

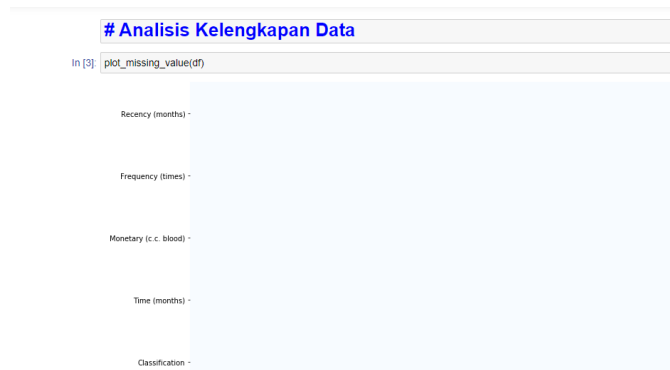
In [2]: df = pd.read_csv("data/Tranfusi.csv")
df.head()

Out[2]:
```

	Recency (months)	Frequency (times)	Monetary (c.c. blood)	Time (months)	Classification
0	2	50	12500	98	Potensial
1	0	13	3250	28	Potensial
2	1	16	4000	35	Potensial
3	2	20	5000	45	Potensial
4	1	24	6000	77	Non - Potensial

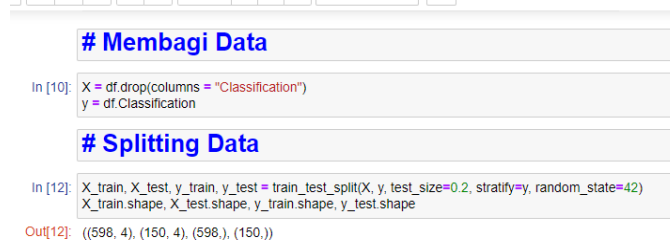
Gambar 3. Import Data

- b. Analisis kelengkapan data



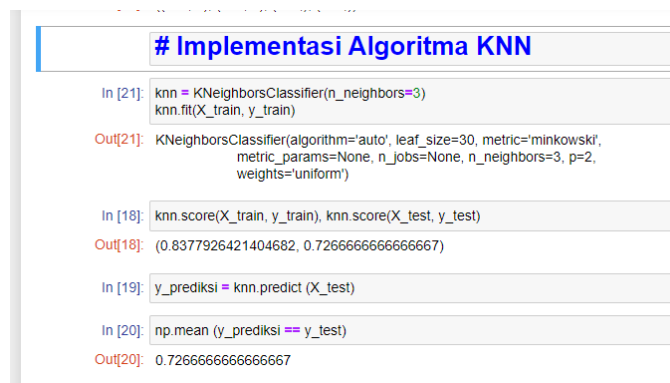
Gambar 4. Analisis Kelengkapan Data

- c. Memecah data menjadi 2 dengan pembagian X sebagai *feature* atau *input* dan y sebagai target atau *labels*
- d. Lakukan *splitting* data menjadi data *training* dan data test



Gambar 5. Membagi Data & Splitting Data

- e. Implementasi algoritma



Gambar 6. Implementasi Algoritma KNN

```

# Implementasi Algoritma NaiveBayes

In [31]: NaiveBayes = MultinomialNB().fit(X_train, np.ravel(y_train, order = 'C'))
print(NaiveBayes)
MultinomialNB(alpha=1.0, class_prior=None, fit_prior=True)

In [32]: prediction = NaiveBayes.predict(X_test)
accuracies = accuracy_score(y_test, prediction)

In [34]: print(accuracies)
0.7333333333333333

In [35]: print(confusion_matrix(y_test, prediction))
[[89 24]
 [16 21]]

In [33]: print(prediction)
['Non - Potensial' 'Non - Potensial' 'Potensial' 'Non - Potensial'
'Non - Potensial' 'Non - Potensial' 'Non - Potensial' 'Non - Potensial'
'Potensial' 'Non - Potensial' 'Potensial' 'Non - Potensial'
'Non - Potensial' 'Non - Potensial' 'Non - Potensial' 'Non - Potensial'
'Potensial' 'Non - Potensial' 'Non - Potensial' 'Potensial'
'Non - Potensial' 'Non - Potensial' 'Potensial' 'Non - Potensial']
    
```

Gambar 7. Implementasi Algoritma Naive Bayes

accuracy: 78.00%

	true Potensial	true Non - Potensial	class precision
pred. Potensial	13	10	56.52%
pred. Non - Potensial	23	104	81.89%
class recall	36.11%	91.23%	

Gambar 8. Klasifikasi Pendonor Darah dengan Algoritma KNN pada RapidMiner

accuracy: 71.33%

	true Potensial	true Non - Potensial	class precision
pred. Potensial	4	11	26.67%
pred. Non - Potensial	32	103	76.30%
class recall	11.11%	90.35%	

Gambar 9. Klasifikasi Pendonor Darah dengan Algoritma Naive Bayes Pada RapidMiner

Tabel 1. Hasil Klasifikasi pada Pemrograman Bahasa Python

No	Objek 3D VLE	Fungsi / Aktivitas
1	Algoritma KNN (Jumlah tetangga = 1)	64.66%
2	Algoritma KNN (Jumlah tetangga = 2)	73.33%
3	Algoritma KNN (Jumlah tetangga = 3)	76.67%
4	Algoritma Naive Bayes	73.33%

Dengan komposisi data *training* dan data set yang sama. Hasil klasifikasi pendonor darah pada RapidMiner dengan algoritma KNN menghasilkan nilai akurasi sebesar 78% sedangkan algoritma Naive Bayes menghasilkan akurasi 71,33%. Untuk hasil klasifikasi dengan menggunakan Bahasa Pemrograman Python menghasilkan nilai akurasi pada algoritma KNN berbeda-beda tergantung dengan jumlah tetangga yang ditentukan yaitu 64.66%, 73.33%, 72.67% sedangkan pada algoritma Naive Bayes adalah 73,33%.

Dari pengujian yang telah dilakukan dengan menggunakan RapidMiner dan Bahasa Pemrograman Python dapat kita lihat bahwa terdapat perbedaan hasil akurasi dari masing-masing algoritma. Algoritma KNN menghasilkan nilai akurasi yang lebih baik dengan menggunakan *software* RapidMiner namun algoritma Naive Bayes menghasilkan nilai akurasi lebih baik dengan menggunakan Bahasa Pemrograman Python. Untuk algoritma KNN yang diimplementasikan dengan menggunakan bahasa pemrograman python hasil akurasinya bergantung dengan jumlah tetangga yang diinputkan, namun tidak menjamin semakin banyak jumlah tetangga yang diinputkan akan semakin baik nilai akurasinya.

4 Kesimpulan

Klasifikasi pendonor darah potensial dapat dilakukan menggunakan metode KNN dan Naive Bayes pada *software* RapidMiner dan Bahasa Pemrograman Python. Dari 748 dataset UCI, klasifikasi calon pendonor darah potensial menggunakan algoritma KNN pada RapidMiner menghasilkan nilai akurasi sebesar 78% sedangkan algoritma Naive Bayes menghasilkan akurasi 71,33%. Untuk hasil klasifikasi dengan menggunakan Bahasa Pemrograman Python menghasilkan nilai akurasi pada algoritma KNN berbeda-beda tergantung dengan jumlah tetangga yang diinputkan atau ditentukan yaitu 64.66%, 73.33%, 72.67% namun tidak menjamin semakin banyak jumlah tetangga yang diinputkan akan semakin baik nilai akurasinya. Sedangkan pada algoritma Naive Bayes dengan pemrograman bahasa *python* menghasilkan nilai akurasi sebesar 73,33% lebih baik dari hasil pengujian dengan menggunakan *software* RapidMiner.

Namun, penulis memberikan beberapa saran untuk dilaksanakan pada penelitian selanjutnya yaitu :

1. Menggunakan algoritma yang lain agar bisa mendapatkan variasi hasil yang dapat dijadikan perbandingan.
2. Menambahkan jenis variabel baru untuk memprediksi calon pendonor darah potensial agar memiliki nilai akurasi yang lebih tinggi.
3. Pengujian dilakukan tidak hanya satu kali dengan komposisi data *test* dan data *training* yang berbeda beda

5 Referensi

- [1] Astuti, Indriyani. (2019). Indonesia Kekurangan 1 Juta Kantong Darah Tiap Tahun. Media Indonesia. <https://mediaindonesia.com/humaniora/169772/indonesia-kekurangan-1-juta-kantong-darah-tiap-tahun>
- [2] Astuti, Indriyani. (2019). Kebutuhan Darah di Indonesia Belum Terpenuhi. Media Indonesia. <https://mediaindonesia.com/humaniora/242935/kebutuhan-darah-di-indonesia-belum-terpenuhi>
- [3] YUD. (2017). 28 Persen Kematian Ibu Melahirkan Akibat Pendarahan. Berita Satu. <https://www.beritasatu.com/nasional/441051/28-persen-kematian-ibu-melahirkan-akibat-endarahan>
- [4] Lestandy, Merinda. (2020). Tutorial Sederhana Naive Bayes dan KNN Menggunakan Rapidminer - volume 2. <https://www.youtube.com/watch?v=BJL-gTytzRk>
- [5] Nursikuw, Agus. (2020). Belajar Classification Learning Naive Bayes dengan Python. https://www.youtube.com/watch?v=nUGphlVF_9s
- [6] Indonesia, JCop untuk. (2020). 2-3 | Pengantar Scikit-learn. <https://www.youtube.com/watch?v=XQX9hr5rcSM&list=PLGn1wRmlR3MvipYvFEnmaTwd59XjSyrG&index=5>
- [7] Indonesia, JCop untuk. (2020). 2-4 | Kenapa Dataset Splitting itu penting. <https://www.youtube.com/watch?v=nHIMdHKISDc&list=PLGn1wRmlR3MvipYvFEnmaTwd59XjSyrG&index=4>
- [8] Adha, A. C., Yuhandri, Y., & Nurcahyo, G. W. (2021). Prediksi Potensi Relawan Pendonor Darah Menjadi Pendonor Darah Tetap dengan Penerapan Metode Klasifikasi Decision Tree. *Jurnal Informasi dan Teknologi*, 233-238.
- [9] Atmaja, K. J., Anandita, I. B. G., & Dewi, N. K. C. (2017). Penerapan Data Mining Untuk Memprediksi Potensi Pendonor Darah Menjadi Pendonor Tetap Menggunakan Metode Decision Tree C. 45. *S@ CIES*, 7(2), 101-108.
- [10] Andriska Candra P, "Prediksi Kepuasan Mahasiswa Terhadap Tingkat Pelayanan Menggunakan Algoritma C4.5 (Decision Tree) (Studi Kasus : Fakultas Teknik Universitas Hamzanwadi)," *Infotek J. Inform. dan Teknol.*, vol. 1, no. 1, pp. 32–39, 2018.
- [11] Nur, N., & Syahra, N. (2021). Perbandingan Metode k-NN dan Naïve Bayes dalam Klasifikasi Penentuan Calon Pendonor Darah. *Journal of Computer and Information System (J-CIS)*, 1(1), 21-28.
- [12] Hutrim, H. P. (2020). Penerapan Data Mining untuk Penentuan Kelayakan Calon Pendonor Darah Menggunakan Metode Naïve Bayes (Studi Kasus Pmi Kabupaten Kotawaringin Timur). *Juisi*, 06(02).
- [13] [B] Lestandy, M., Syafaah, L., & Faruq, A. (2020). Klasifikasi pendonor darah potensial menggunakan pendekatan algoritme pembelajaran mesin. *Jurnal Teknologi dan Sistem Komputer*, 8(3), 217-221.
- [14] Setiawan, M. R., Sari, Y. A., & Adikara, P. P. (2019). Klasifikasi Citra Makanan Menggunakan K-Nearest Neighbor dengan Fitur Bentuk Simple Morphological Shape Descriptors dan Fitur Warna Grayscale Histogram. 3(3), 2726–2731.
- [15] Cokrowibowo, S., Nur, N., & Irmayanti. (2020). Web Page Extraction and Classification Using JSOUP and Naïve Bayes. *IOP Conference Series: Materials*

- Science and Engineering, 875(1). <https://doi.org/10.1088/1757-899X/875/1/012089>
- [16] Purhanta. (2010). PENGUMPULAN DATA DAN INSTRUMEN PENELITIAN Penelitian. Http://Etheses.Uin-Malang.Ac.Id/1670/7/11510004_Bab_3.Pdf, 1–7.
- [17] Permana, I. (2019). Strategi Komunikasi Pemasaran Divisi Community Tokocrypto Dalam Kegiatan Cryptour Untuk Pembentukan Brand Awareness Di Kota Bandung (Doctoral dissertation, Universitas Komputer Indonesia).
- [18] Hidayah, N., Sulfahmi, S., Zairani, I., Yusuf, M., & Sufiati, S. (2020). COMBINE ASSURANCE DALAM KONTEKS PENGENDALIAN. *Equilibrium: Jurnal Ilmiah Ekonomi, Manajemen dan Akuntansi*, 8(2).