

Klasifikasi Kinerja Teknisi Pada PT. Telkom Menggunakan Metode *K-Nearest Neighbor*

Muhammad Amidhan, Rudi Heriansyah*, Indah Permatasari

Fakultas Ilmu Komputer dan Sains Universitas Indo Global Mandiri

Penulis Korespondensi: rudi@uigm.ac.id*

2020110029@students.uigm.ac.id, indah@uigm.ac.id

Abstract

Productivity is expected to support the company's goal of achieving high profits. Technician performance assessment still relies on subjective assessment from the team leader, causing difficulties in objective evaluation. This research aims to improve the appraisal of technician performance at PT Telkom by proposing using the k-nearest Neighbor (k-NN) method as a classification method. The k-NN method was chosen because of its robustness to training data and effectiveness on large training data. The classification process includes finding the distance between two points using the Euclidean equation, sorting the lowest value as the nearest neighbor, calculating the number of neighbors based on the 5 (five) nearest neighbors for the classification of "Dissatisfied", "Quite Satisfied", "Satisfied", and "Very Satisfied". Based on calculations carried out using a confusion matrix, this algorithm has an accuracy rate of 74%, precision of 80%, and recall of 80%, so it can be recommended to PT Telkom to know the actions to be taken, whether a technician can be retained or not in the company.

Keywords: Classification, k-NN, Technician

1. Pendahuluan

1.1 Latar Belakang

Produktifitas sangat berhubungan langsung dengan sumber daya manusia, maka hal ini sangat penting diperhatikan oleh pimpinan perusahaan (Silaen dkk., 2021). Jika produktifitas meningkat maka tujuan perusahaan untuk mendapatkan laba juga pasti meningkat (Silaen dkk., 2021). Peningkatan produktifitas ini sangat berkaitan dengan kinerja karyawan yang merupakan sumber daya manusia dalam perusahaan, sehingga sumber daya manusia merupakan asset yang sangat penting dalam perusahaan (Silaen dkk., 2021).

PT. Telkom memiliki karyawan yang berkualitas dan berdedikasi tinggi dalam bekerja dan melakukan pekerjaannya secara efektif dan efisien, untuk menunjukkan kinerja yang tinggi dalam usaha mewujudkan misi dan tujuan yang telah ditetapkan (Yapentra, A. 2021). Namun pihak PT. Telkom masih kesulitan dalam melakukan evaluasi kinerja para teknisi menggunakan beberapa standar atau kriteria penilaian yang diinginkan (Septiawan, I., & Topiq, S, 2021). Proses evaluasi

tersebut masih menggunakan penilaian berdasarkan persepsi subjektif dari team leader saja, hal ini dikarenakan belum adanya suatu metode yang tepat untuk diterapkan pada proses evaluasi kinerja teknisi (Septiawan, I., & Topiq, S, 2021).

Penelitian ini memiliki beberapa referensi, salah satu penelitian terdahulu yang memiliki tema yang terdekat adalah “Klasifikasi Kinerja Programmer pada Aktivitas Media Sosial dengan Metode Support Vector Machines” yang diterbitkan pada tahun 2020, dimana hasil dari penelitian tersebut didapatkan bahwa performa dari metode SVM dalam klasifikasi kinerja programmer pada aktivitas media sosial memiliki tingkat akurasi yang cukup baik. Algoritma SVM menghasilkan rata-rata tingkat akurasi dengan $k=10$ cross validation sebesar 81,3%. Akurasi tertinggi yang dimiliki oleh algoritma SVM sebesar 85,45% pada pengujian ke-1 dan nilai akurasi terendah pada pengujian ke-4 dengan nilai sebesar 77,7%. Kontribusi penelitian ini didasari pada penelitian sebelumnya, adanya keterbedaan objek penelitian yaitu pada PT. Telkom, metode, dan atribut yang digunakan.

Penelitian ini dilakukan untuk memberikan solusi dengan dilakukannya klasifikasi kinerja teknisi menggunakan metode k-NN. Pada penelitian ini metode yang digunakan adalah k-Nearest Neighbor dikarenakan belum diketahui tingkat akurasinya maka penelitian ini dilakukan uji coba pada data dan penelitian ini penting untuk dilakukan oleh karena itu dilakukan uji coba, dan metode k-Nearest Neighbor memiliki kelebihan yaitu bahwa algoritmanya tangguh terhadap training data dan efektif apabila data latihnya besar (Qiudandra, E., & Akram, R, 2022). Adapun atribut yang telah ditetapkan seperti jumlah perbaikan, waktu pengerjaan perbaikan, disiplin kerja, dan kerjasama tim.

Penelitian ini dilakukan untuk membantu PT. Telkom dalam mengklasifikasikan kinerja teknisi. Melalui penelitian ini perusahaan dapat melihat dan mengambil keputusan dari hasil penelitian apakah teknisi tersebut dapat dipertahankan atau tidak didalam perusahaan.

1.2 Rumusan Masalah

Adapun rumusan masalah dari penelitian ini, yaitu, bagaimana melakukan klasifikasi kinerja teknisi menggunakan metode k-Nearest Neighbor?

2. Tinjauan Literatur

2.1 Definisi k-NN (k-Nearest Neighbor)

k-NN adalah teknik yang populer untuk pembelajaran mesin dan sebagian besar diterapkan untuk klasifikasi pada set data benchmark (Wazery dkk., 2021). Ini algoritma yang sederhana dan mudah digunakan ini memberikan hasil yang sangat baik dalam banyak domain, bahkan dalam perbandingan dengan pendekatan pembelajaran pendekatan pembelajaran mesin yang paling canggih (Wazery dkk., 2021). k-Nearest Neighbour (k-NN) adalah nonparametrik yang sangat efisien skema untuk membedakan antara fitur yang berbeda (Harrou dkk., 2020).

k-Nearest Neighbor (k-NN) adalah sebuah metode klasifikasi terhadap sekumpulan data berdasarkan pembelajaran data yang sudah terklasifikasikan sebelumnya. k-NN termasuk dalam golongan supervised learning, dimana hasil query instance yang baru diklasifikasikan berdasarkan mayoritas kedekatan jarak dari kategori yang ada dalam k-NN (Amilia, R., & Prasetyo, E, 2021).

Nantinya kelas yang baru dari suatu data akan dipilih berdasarkan grup kelas yang paling dekat jarak vektornya (Amilia, R., & Prasetyo, E, 2021). Tujuan dari algoritma ini adalah mengklasifikasikan objek baru berdasarkan atribut dan training sample (Amilia, R., & Prasetyo, E, 2021). Classifier tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan pada memori (Amilia, R., & Prasetyo, E, 2021). Diberikan titik query, akan ditemukan sejumlah k objek atau (titik training) yang paling dekat dengan titik query (Amilia, R., & Prasetyo, E, 2021). Klasifikasi menggunakan voting Algoritma k-Nearest Neighbor menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari query instance yang baru (Amilia, R., & Prasetyo, E, 2021).

2.2 Algoritma k-NN (K-Nearest Neighbor)

Algoritma KNN merupakan algoritma generalisasi terdekat aturan tetangga (Xing, W., & Bei, Y, 2019). Offset induktifnya adalah label kelas dari k-sample dengan label kelas yang akan diuji paling mirip dengan yang terdekat (Xing, W., & Bei, Y, 2019). Berbeda dengan tetangga terdekatnya dalam hal ini memperluas tetangga terdekat ke k dalam fase pengambilan keputusan (Xing, W., & Bei, Y, 2019). Ekstensi ini memungkinkan algoritma KNN untuk memperoleh dan memanfaatkan informasi lebih lanjut (Xing, W., & Bei, Y, 2019). Algoritma k-NN adalah salah satu algoritma klasifikasi yang paling sederhana dan salah satu algoritma pembelajaran yang paling banyak digunakan karena kesederhanaannya, kemudahan implementasi dan prinsip non-parametrik (Onyezewe dkk., 2021).

Langkah-langkah untuk menghitung metode k-Nearest Neighbor antara lain:

1. Menentukan parameter K.
2. Menghitung jarak antara data training dan data testing

Perhitungan jarak yang paling umum dipakai pada perhitungan pada metode k-NN adalah menggunakan perhitungan jarak Euclidean. Rumusannya adalah sebagai berikut:

$$euc = \sqrt{(\sum_{i=1}^n (p_i - q_i)^2)}$$

dimana : euc = euclidean. pi =
sample data / data training. qi =
data uji / data testing. i =
variabel data. n = dimensi data.

3. Mengurutkan jarak yang terbentuk.
4. Menentukan jarak terdekat sampai urutan k.
5. Memasangkan kelas yang bersesuaian.
6. Mencari jumlah kelas dari tetangga yang terdekat dan tetapkan kelas tersebut sebagai kelas data yang akan dievaluasi.

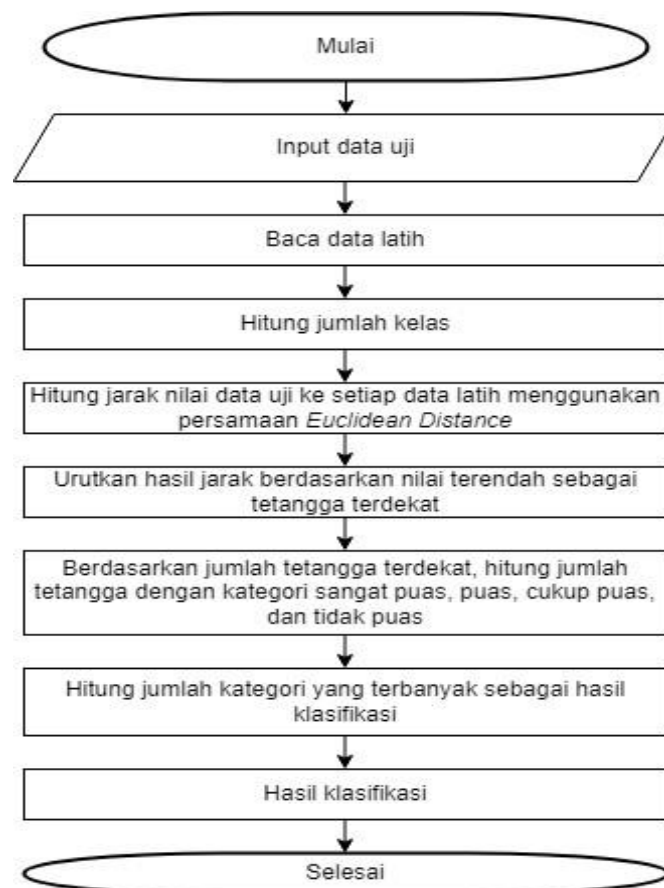
3. Metode Penelitian

3.1 Tahapan Penelitian

Tahapan penelitian yang dilakukan meliputi tahapan data mining dalam menerapkan metode klasifikasi k-NN. Istilah “data mining” ini digunakan cukup luas di industri Teknologi Informasi (TI). Metode ini sering diterapkan pada berbagai kegiatan pemrosesan data skala besar seperti pengumpulan, ekstraksi, pergudangan, dan analisis data.

3.2 Penerapan Metode k-NN

Langkah yang dilakukan pada tahapan ini adalah melakukan perhitungan dari data yang telah didapatkan sebelumnya lewat studi literatur menggunakan algoritma k-NN. Berdasarkan data yang akan didapatkan akan dibangun sebuah sistem untuk menentukan kinerja teknisi dengan menghitung mean setiap variabel yang bernilai kontinu, dilanjutkan dengan menghitung kedekatan setiap kategori apabila diberikan input tertentu dengan menggunakan metode k-NN untuk keperluan tersebut, dan dilanjutkan dengan tahap pengujian. Berikut Flowchart Algoritma k-NN.



4. Hasil dan Pembahasan

4.1 Menerapkan Metode k-NN

Langkah yang dilakukan pada tahapan ini adalah melakukan perhitungan dari data yang telah didapatkan pada Tabel 4.1 menggunakan algoritma k-NN. Berdasarkan data yang didapatkan akan dibangun sebuah sistem untuk menentukan kinerja teknisi dengan mencari 5 tetangga terdekat yang didapat dengan menghitung jarak dua titik setiap atribut apabila diberikan input tertentu. Kode program dari penerapan metode k-NN dapat dilihat sebagai berikut

```
from sklearn.neighbors import KNeighborsClassifier

# Membuat dan melatih model KNN
classifier = KNeighborsClassifier(n_neighbors=5, metric='euclidean', p=2)
classifier.fit(x_train, y_train)

# Menampilkan hasil
for i in range(len(x_test)):
    print(f"\nData uji ke-{i+1} (kelas {y_test[i]}):")
    print(f" - Tetangga terdekat:")

    # Menampilkan informasi tetangga terdekat
    neighbors = classifier.kneighbors([x_test[i]])
    for j in range(len(neighbors[1][0])):
        neighbor_distance = neighbors[0][0][j]
        neighbor_class = y_train[neighbors[1][0][j]]
        print(f" - Tetangga {j+1}: Jarak Euclidean: {neighbor_distance:.4f}, Kelas: {neighbor_class}")

# Hasil Klasifikasi
prediction = classifier.predict([x_test[i]])
print(f" - Hasil Klasifikasi: {prediction[0]}")
```

Gambar 4.1 Code Program Metode k-NN

Pada kode program KNeighborsClassifier terdapat `n_neighbors = 5` yang berfungsi untuk menentukan jumlah dari tetangga terdekat, `metric = „euclidean”` dan `p = 2` menunjukkan penggunaan persamaan (2.1) yaitu Euclidean Distance, lalu mencari iterasi pada setiap data uji dan menampilkan informasi tentang 5 tetangga terdekat untuk setiap data uji, lalu menampilkan jarak Euclidean, dan klasifikasi (kelas) dari tetangga.

```
Data uji ke-1 (kelas SP):
- Tetangga terdekat:
  - Tetangga 1: Jarak Euclidean: 2.0000, Kelas: SP
  - Tetangga 2: Jarak Euclidean: 2.0000, Kelas: SP
  - Tetangga 3: Jarak Euclidean: 2.0000, Kelas: SP
  - Tetangga 4: Jarak Euclidean: 2.4495, Kelas: CP
  - Tetangga 5: Jarak Euclidean: 3.4641, Kelas: P
- Hasil Klasifikasi: SP

Data uji ke-2 (kelas TP):
- Tetangga terdekat:
  - Tetangga 1: Jarak Euclidean: 1.4142, Kelas: TP
  - Tetangga 2: Jarak Euclidean: 3.7417, Kelas: CP
  - Tetangga 3: Jarak Euclidean: 5.9161, Kelas: CP
  - Tetangga 4: Jarak Euclidean: 6.7823, Kelas: CP
  - Tetangga 5: Jarak Euclidean: 8.3066, Kelas: SP
- Hasil Klasifikasi: CP

Data uji ke-3 (kelas P):
- Tetangga terdekat:
  - Tetangga 1: Jarak Euclidean: 2.4495, Kelas: CP
  - Tetangga 2: Jarak Euclidean: 4.7958, Kelas: TP
  - Tetangga 3: Jarak Euclidean: 5.1962, Kelas: CP
  - Tetangga 4: Jarak Euclidean: 5.3852, Kelas: P
  - Tetangga 5: Jarak Euclidean: 6.8557, Kelas: CP
- Hasil Klasifikasi: CP

Data uji ke-4 (kelas P):
- Tetangga terdekat:
  - Tetangga 1: Jarak Euclidean: 1.7321, Kelas: CP
  - Tetangga 2: Jarak Euclidean: 3.1623, Kelas: CP
  - Tetangga 3: Jarak Euclidean: 4.1231, Kelas: P
  - Tetangga 4: Jarak Euclidean: 4.4721, Kelas: P
  - Tetangga 5: Jarak Euclidean: 5.3852, Kelas: SP
- Hasil Klasifikasi: CP

Data uji ke-5 (kelas CP):
- Tetangga terdekat:
  - Tetangga 1: Jarak Euclidean: 1.7321, Kelas: SP
  - Tetangga 2: Jarak Euclidean: 2.4495, Kelas: CP
  - Tetangga 3: Jarak Euclidean: 4.5826, Kelas: P
  - Tetangga 4: Jarak Euclidean: 5.4772, Kelas: CP
  - Tetangga 5: Jarak Euclidean: 6.1644, Kelas: SP
- Hasil Klasifikasi: CP

Data uji ke-6 (kelas TP):
- Tetangga terdekat:
  - Tetangga 1: Jarak Euclidean: 4.6904, Kelas: TP
  - Tetangga 2: Jarak Euclidean: 6.7082, Kelas: TP
  - Tetangga 3: Jarak Euclidean: 7.1414, Kelas: TP
  - Tetangga 4: Jarak Euclidean: 8.8318, Kelas: CP
  - Tetangga 5: Jarak Euclidean: 8.9443, Kelas: TP
- Hasil Klasifikasi: TP

Data uji ke-7 (kelas TP):
- Tetangga terdekat:
  - Tetangga 1: Jarak Euclidean: 5.4772, Kelas: TP
  - Tetangga 2: Jarak Euclidean: 5.5678, Kelas: CP
  - Tetangga 3: Jarak Euclidean: 5.9161, Kelas: TP
  - Tetangga 4: Jarak Euclidean: 6.0000, Kelas: TP
  - Tetangga 5: Jarak Euclidean: 6.4807, Kelas: CP
- Hasil Klasifikasi: TP
```

Gambar 4.2 Output Metode *k*-NN

4.2.2 Melakukan Klasifikasi

Proses klasifikasi dengan *k*-NN dilakukan dengan tahapan yang telah ditentukan yaitu melakukan input data uji, baca data latih, hitung jumlah kelas, hitung jarak nilai data uji ke setiap data latih menggunakan persamaan *Euclidean Distance*, Urutkan hasil jarak berdasarkan nilai terendah sebagai tetangga terdekat, berdasarkan jumlah tetangga terdekat hitung jumlah tetangga dengan

kategori “Sangat Puas”, “Puas”, “Cukup Puas”, “Tidak Puas”, lalu menghitung jumlah kategori yang terbanyak sebagai hasil klasifikasi, hasil klasifikasi. Tahapan yang ada akan dilakukan oleh mesin komputer berdasarkan program yang telah dibuat pada Gambar 4.1 dan hasil klasifikasi dapat dilihat pada Gambar 4.2

4.2.3 Hasil Confusion Matrix

Untuk mencari hasil dari *confusion matrix* adalah dengan cara membandingkan data aktual dengan hasil prediksi yang didapat dari *code* program pada Gambar 4.1. Perhitungan *confusion matrix* digunakan persamaan dengan nilai yang didapat dari dataset, lalu menghitung nilai-nilai yang telah dimasukkan tersebut untuk dihitung tingkat akurasi, presisi, dan *recall*.

```
from sklearn.metrics import confusion_matrix
cm_test = confusion_matrix(y_test, y_pred_test)
cm_train = confusion_matrix(y_train, y_pred_train)
print(f"\nData Uji ")
print(cm_test)
print(f"\nData Latih ")
print(cm_train)
```

Data yang telah diklasifikasi dapat dilihat pada Tabel 4.3 dan Tabel 4.4 yang didapat dari Gambar 4.3, aktual menunjukkan arah samping (kolom), sedangkan prediksi menunjukkan arah atas (baris). Terdapat 24 data pada data *training* dengan klasifikasi yaitu 7 data “Cukup Puas”, 3 data “Puas”, 4 data “Sangat Puas”, 5 data “Tidak Puas”, akan tetapi terdapat 4 data yang seharusnya masuk dalam klasifikasi “Cukup Puas” namun 2 data masuk klasifikasi “Tidak puas”, 1 data “Puas”, dan 1 data “ Sangat Puas”, dan terdapat 1 data yang seharusnya masuk dalam klasifikasi “Sangat Puas” namun data tersebut masuk klasifikasi “Cukup Puas”. Pada data *testing* terdapat 7 data diantaranya, ada 1 data “Cukup Puas”, 1 data “Sangat Puas”, dan 2 data “Tidak Puas”, akan tetapi terdapat 3 data yang seharusnya masuk dalam klasifikasi “Cukup Puas” tetapi 1 data masuk klasifikasi “Tidak Puas”, dan 2 data masuk klasifikasi “Puas”.

Tabel 4. Hasil Klasifikasi Model Data
Training

Data Training	
Data Aktual	Data Prediksi

Tidak Puas	Cukup Puas	Puas	Sangat Puas	Tidak Puas	Cukup Puas	Puas	Sangat Puas
7	8	4	5	5	11	3	5

Tabel 4.6 Hasil Klasifikasi Model Data *Testing*

Data <i>Testing</i>							
Data Aktual				Data Prediksi			
Tidak Puas	Cukup Puas	Puas	Sangat Puas	Tidak Puas	Cukup Puas	Puas	Sangat Puas
3	1	2	1	2	4	0	1

Hasil klasifikasi yang telah didapatkan dapat dilihat pada Tabel 4.5 untuk data *training*, dan Tabel 4.6 untuk data *testing* terdapat perbedaan data antara data aktual dan data prediksi hasil klasifikasi pada kedua Tabel. Sehingga dapat disimpulkan terdapat 3 kesalahan klasifikasi pada dataset untuk data *testing* dan terdapat 5 kesalahan klasifikasi pada dataset untuk data *training*.

1. *Confusion Matrix* pada Data *Training*

TP = 14 TN = 5 FP = 3 FN = 2

Jumlah Data : 24

$$\begin{aligned}
 \text{Akurasi} &= \frac{(TP+TN)}{(TP+TN+FP+FN)} \times 100\% \\
 &= \frac{(14+5)}{(14+5+3+2)} \times 100\% \\
 &= 0,79 = 79\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Presisi} &= \frac{TP}{(TP+FP)} \times 100\% \\
 &= \frac{14}{(14+3)} \times 100\% \\
 &= 0,82 = 82\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Recall} &= \frac{TP}{(TP+FN)} \times 100\% \\
 &= \frac{14}{(14+2)} \times 100\% \\
 &= 0,87 = 87\%
 \end{aligned}$$

Perhitungan *Confusion matrix* terhadap data training menghasilkan nilai akurasi sebesar 79%, presisi sebesar 82%, *recall* sebesar 87%. Perhitungan *F1 Score* menggunakan Persamaan yaitu dengan cara hasil *recall* dikali presisi dibagi *recall* ditambah presisi maka akan di dapatkan hasil perbandingan rata-rata presisi dan *recall* yang telah dibobotkan sebelumnya.

$$\begin{aligned}
 F1\ Score &= 2 \times \frac{(\text{Presisi} \times \text{Recall})}{(\text{Presisi} + \text{Recall})} \times 100\% \\
 &= 2 \times \frac{(82\% \times 87\%)}{(82\% + 87\%)} \times 100\% \\
 &= 0,84 = 84\%
 \end{aligned}$$

2. *Confusion Matrix* pada Data Testing

TP = 2 TN = 2 FP = 1 FN = 2

Jumlah Data : 7

$$\begin{aligned}
 \text{Akurasi} &= \frac{(TP+TN)}{(TP+TN+FP+FN)} \times 100\% \\
 &= \frac{(2+2)}{(2+2+1+2)} \times 100\% \\
 &= 0,57 = 57\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Presisi} &= \frac{TP}{(TP+FP)} \times 100\% \\
 &= \frac{2}{(2+1)} \times 100\% \\
 &= 0,66 = 66\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Recall} &= \frac{TP}{(TP+FN)} \times 100\% \\
 &= \frac{2}{(2+2)} \times 100\% \\
 &= 0,5 = 50\%
 \end{aligned}$$

Perhitungan *Confusion matrix* terhadap data testing menghasilkan nilai akurasi sebesar 57%, presisi sebesar 66%, *recall* sebesar 50%. Perhitungan *F1 Score* dapat dilakukan setelah

mendapatkan nilai *sensitivity (recall)*, presisi, dan akurasi sama seperti pada *confusion matrix* data *training*.

$$\begin{aligned} F1 \text{ Score} &= 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \times 100\% \\ &= 2 \times \frac{66\% \times 50\%}{66\% + 50\%} \times 100\% \\ &= 0,568 = 57\% \end{aligned}$$

3.Perhitungan Akurasi Model Klasifikasi Menggunakan Metode *k- Neirest Neighbor*

Selanjutnya akan dilakukan perhitungan *confusion matrix* terhadap model klasifikasi untuk mengetahui seberapa baik model yang telah dibuat berdasarkan dataset yang telah di inputkan sebanyak 31 data. Berikut perhitungan *confusion matrix*.

$$TP = 16 \quad TN = 7 \quad FP = 4 \quad FN = 4$$

$$\text{Jumlah Data} = 31$$

$$\begin{aligned} \text{Akurasi} &= \frac{(TP+TN)}{(TP+TN+FP+FN)} \times 100\% \\ &= \frac{(16+7)}{(16+7+4+4)} \times 100\% \\ &= 0,74 = 74\% \end{aligned}$$

$$\begin{aligned} \text{Presisi} &= \frac{TP}{(TP+FP)} \times 100\% \\ &= \frac{16}{(16+4)} \times 100\% \\ &= 0,8 = 80\% \end{aligned}$$

$$\begin{aligned} \text{Recall} &= \frac{TP}{(TP+FN)} \times 100\% \\ &= \frac{16}{(16+4)} \times 100\% \\ &= 0,8 = 80\% \end{aligned}$$

4.2.3 Diskusi / Pembahasan

Berdasarkan perhitungan *confusion matrix* algoritma ini memiliki tingkat akurasi sebesar 77%, yang didapat dari persamaan (2.2), terdapat 16 data *True Positive*, 7 data *True Negative*, 4 data *False Positive*, dan 4 data *False Negative* dari 31 dataset, sehingga dapat direkomendasikan kepada PT. Telkom agar dapat mengetahui tindakan yang akan diambil, apakah seorang teknisi dapat dipertahankan atau tidak didalam perusahaan. Terdapat 31 dataset yang diklasifikasi kedalam 4 kelas yaitu “Tidak Puas”, “Cukup Puas”, “Puas”, dan “Sangat Puas”.

Terdapat 3 klasifikasi yang menjadi rekomendasi untuk PT. Telkom yaitu “Sangat Puas”, “Puas”, dan “Cukup Puas”, teknisi yang memiliki nilai tertinggi terhadap 5 teknisi yang berklasifikasi “Sangat Puas” adalah Ariyanto bernilai 14,92, lalu teknisi yang memiliki nilai tertinggi terhadap 5 teknisi berklasifikasi “Puas” adalah Adi Putera bernilai 11,17, dan teknisi yang memiliki nilai

tertinggi terhadap 8 teknisi berklasifikasi “Cukup Puas” adalah Zazali Nanda bernilai 6,36, sedangkan terdapat 1 klasifikasi yang tidak direkomendasikan kepada PT. Telkom yaitu “Tidak Puas” yang memiliki nilai tertinggi hanya 3,91 bernama Rusdi Haryanto terhadap 9 teknisi berklasifikasi “Tidak Puas” lainnya. Setiap data objek memiliki 4 atribut yaitu jumlah perbaikan, waktu perbaikan (jam), disiplin, dan kerjasama tim. Terdapat 23 data diklasifikasi dengan benar dan 8 data diklasifikasi tidak sama (tidak sesuai) dari 31 data.

5. Kesimpulan

Dari hasil penelitian dapat diambil kesimpulan bahwa untuk melakukan klasifikasi kinerja teknisi dapat menggunakan metode *k-Nearest Neighbor*. Ada beberapa tahapan yang perlu dilakukan, diantaranya mencari jarak 2 (dua) titik dengan menggunakan persamaan *Euclidean Distance*, mengurutkan nilai terendah sebagai tetangga terdekat, menghitung jumlah tetangga berdasarkan 5 (lima) tetangga terdekat dengan klasifikasi “Tidak Puas”, “Cukup Puas”, “Puas”, dan “Sangat Puas”, lalu menghitung jumlah klasifikasi terbanyak sebagai hasil klasifikasi. Berdasarkan hasil, algoritma ini memiliki tingkat akurasi sebesar 74%, presisi sebesar 80%, dan *recall* sebesar 80% yang didapat dari Tabel 4.4 dan Tabel 4.5, terdapat 16 data *True Positive*, 7 data *True Negative*, 4 data *False Positive*, dan 4 data *False Negative* dari 31 dataset, sehingga dapat direkomendasikan kepada PT. Telkom agar dapat mengetahui tindakan yang akan diambil, akan tetapi terdapat perbedaan pada penelitian terdahulu yang berjudul

“IMPLEMENTASI METODE K-NEAREST NEIGHBOR (KNN) UNTUK SELEKSI CALON KARYAWAN BARU (Nugraha dkk., 2020)” memiliki tingkat akurasi sebesar 91%, nilai presisi sebesar 87%, dan nilai *recall* sebesar 100%.

Terdapat 3 klasifikasi yang menjadi rekomendasi untuk PT. Telkom yaitu “Sangat Puas”, “Puas”, dan “Cukup Puas”, teknisi yang memiliki nilai tertinggi terhadap 5 teknisi yang berklasifikasi “Sangat Puas” adalah Ariyanto bernilai 14,92, lalu teknisi yang memiliki nilai tertinggi terhadap 5 teknisi berklasifikasi “Puas” adalah Adi Putera bernilai 11,17, dan teknisi yang memiliki nilai tertinggi terhadap 8 teknisi berklasifikasi “Cukup Puas” adalah Zazali Nanda bernilai 6,36, sedangkan terdapat 1 klasifikasi yang tidak direkomendasikan kepada PT. Telkom yaitu “Tidak Puas” yang memiliki nilai tertinggi hanya 3,91 bernama Rusdi Haryanto terhadap 9 teknisi berklasifikasi “Tidak Puas” lainnya. Setiap data objek memiliki 4 atribut yaitu jumlah perbaikan, waktu perbaikan (jam), disiplin, dan kerjasama tim. Terdapat 23 data diklasifikasi dengan benar dan 8 data diklasifikasi tidak sama (tidak sesuai) dari 31 data.

Referensi

- [1] Admojo, F. T. (2020). Klasifikasi Aroma Alkohol Menggunakan Metode KNN. *Indonesian Journal of Data and Science*, 1(2), 34-38.
- [2] Ali, N., Neagu, D., & Trundle, P. (2019). Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets. *SN Applied Sciences*, 1, 1-15.
- [3] Amilia, R., & Prasetyo, E. (2021). KLASIFIKASI DIAGNOSA PENYAKIT DEMAM BERDARAH DENGUE PADA ANAK MENGGUNAKAN METODE K-NEAREST

NEIGHBOR STUDI KASUS RUMAH SAKIT PKU MUHAMMADIYAH UJUNG PANGKAH

- GRESIK. Indexia: Informatics and Computational Intelligent Journal, 2(2).
- [4] Argina, A. M. (2020). Penerapan Metode Klasifikasi K-Nearest Neighbor pada Dataset Penderita Penyakit Diabetes. *Indonesian Journal of Data and Science*, 1(2), 29- 33.
 - [5] Barupal, D. K., & Fiehn, O. (2019). Generating the blood exposome database using a comprehensive text mining and database fusion approach. *Environmental Health Perspectives*, 127(9), 2825–2830.
 - [6] Gad, A. F. (2021). Pygad: An intuitive genetic algorithm python library. *arXiv preprint arXiv:2106.06158*.
 - [7] Gunawan, W., & Syamsudin, A. (2022). Pengaruh budaya organisasi dan gaya kepemimpinan terhadap kinerja pegawai melalui motivasi kerja DLHKP Kabupaten Gunung Mas. *Journal of Environment and Management*, 3(1), 16-25.
 - [8] Handayani, C., & Fitrianie, T. (2018). Klasifikasi Hasil Pembelajaran Siswa dengan Metode K-Mean Clustering di SMPN 2 Kertasemaya. *Jurnal Web Informatika Teknologi*, 3(1), 12-18.
 - [9] Harismawan, A. F. (2017). Analisis Perbandingan Performa Web Service Menggunakan Bahasa Pemrograman Python, PHP, dan Perl pada Client Berbasis Android (Doctoral dissertation, Universitas Brawijaya).
 - [10] Harrou, F., Zeroual, A., & Sun, Y. (2020). Traffic congestion monitoring using an improved kNN strategy. *Measurement*, 156, 107534.
 - [11] Hutahaean, J., & Hutagalung, J. (2022). Sistem Pendukung Keputusan Pemilihan Teknisi Terbaik Menggunakan Metode Fuzzy Tsukamoto. *JURIKOM (Jurnal Riset Komputer)*, 9(4), 846-856.
 - [12] Jumadi, A., Istiqomah, N. N., & Tentua, M. N. (2020, April). Klasifikasi Evaluasi Asisten Pengajar dengan Menggunakan Metode KNN dan Naive Bayes. In *Seri Prosiding Seminar Nasional Dinamika Informatika* (Vol. 4, No. 1).
 - [13] Khesya, N. (2021). Mengenal Flowchart Dan Pseudocode Dalam Algoritma Dan Pemrograman
 - [14] Laeliah, N., & Subekti, H. (2017). Waktu tunggu pelayanan rawat jalan dengan kepuasan pasien terhadap pelayanan di rawat jalan RSUD Kabupaten Indramayu. *Jurnal Kesehatan Vokasional*, 1(2), 102-112.
 - [15] Ma'rifah, E., Rahmawati, L., & Juairiah, J. (2023). Analisis pemanfaatan aplikasi iHSS oleh pemustaka pada Dinas Perpustakaan dan Kearsipan Kabupaten Hulu Sungai Selatan. *Pustaka Karya: Jurnal Ilmiah Ilmu Perpustakaan dan Informasi*, 11(1), 1-11.
 - [16] Mary, T. R., & Sebastian, S. (2019). Predicting heart ailment in patients with varying number of features using data mining techniques. *International Journal of Electrical & Computer Engineering* (2088-8708), 9(4).
 - [17] Mutawalli, L., Zaen, M. T. A., & Bagye, W. (2019). Klasifikasi Teks Sosial Media Twitter Menggunakan Support Vector Machine (Studi Kasus Penusukan Wiranto). *Jurnal Informatika dan Rekayasa Elektronik*, 2(2), 43-51.

- [18] Onyezewe, A., Kana, A. F., Abdullahi, F. B., & Abdulsalami, A. O. (2021). An enhanced adaptive K-nearest neighbor classifier using simulated annealing. *International Journal of Intelligent Systems and Applications*, 13(1), 34-44.
- [19] Pratiwi, B. P., Handayani, A. S., & Sarjana, S. (2020). Pengukuran Kinerja Sistem Kualitas Udara Dengan Teknologi Wsn Menggunakan Confusion Matrix. *Jurnal Informatika Upgris*, 6(2).
- [20] Permana, M. A., Darwiyanto, E., & Bijaksana, M. A. (2021). Pembobotan Dan Pemingkatan Ayat Al-quran Berdasarkan Compound, Term Frequency Dan Prinsip Pareto Untuk Membantu Hafalan. *EProceedings ...*, 8(2), 3352–3360.
- [21] Price-Whelan, A. M., Lim, P. L., Earl, N., Starkman, N., Bradley, L., Shupe, D. L., ... & Astropy Collaboration. (2022). The Astropy Project: sustaining and growing a community-oriented open-source project and the latest major release (v5. 0) of the core package. *The Astrophysical Journal*, 935(2), 167.
- [22] Python, W. (2021). Python. Python Releases for Windows, 24.
- [23] Qiudandra, E., & Akram, R. (2022). SISTEM PAKAR DIAGNOSA PENYAKIT OSTEOARTHRITIS DENGAN MENGGUNAKAN METODE K-NEAREST NEIGHBOR. *Jurnal Ilmiah Teknik Informatika METHOTIKA*, 2(2), 37-48.
- [24] Rady, E. H. A., & Anwar, A. S. (2019). Prediction of kidney disease stages using data mining algorithms. *Informatics in Medicine Unlocked*, 15, 100178.
- [25] Romli, I., & Zy, A. T. (2020). Penentuan jadwal overtime dengan klasifikasi data karyawan menggunakan algoritma C4. 5. *J-SAKTI (Jurnal Sains Komputer dan Informatika)*, 4(2), 694-702.
- [26] Rosaly, R., & Prasetyo, A. (2019). Pengertian Flowchart Beserta Fungsi dan Simbol- simbol Flowchart yang Paling Umum Digunakan. *Program Studi Teknik Informatika Politeknik Purbaya*.
- [27] Schuh, G., Reinhart, G., Prote, J. P., Sauermann, F., Horsthofer, J., Oppolzer, F., & Knoll, D. (2019). Data mining definitions and applications for the management of production complexity. *Procedia CIRP*, 81, 874-879.
- [28] Septiawan, I., & Topiq, S. (2021). PENILAIAN KINERJA TEKNISI PSB & MAINTENANCE MENGGUNAKAN METODE WEIGHTED PRODUCT DI PT TELKOM AKSES AHMAD YANI. *eProsiding Teknik Informatika (PROTEKTIF)*, 2(1), 148-153.
- [29] Shah, K., Patel, H., Sanghvi, D., & Shah, M. (2020). A comparative analysis of logistic regression, random forest and KNN models for the text classification. *Augmented Human Research*, 5, 1-16.
- [30] Silaen, N. R., Syamsuriansyah, S., Chairunnisah, R., Sari, M. R., Mahriani, E., Tanjung, R., ... & Putra, S. (2021). Kinerja Karyawan.
- [31] Silalahi, A. P., Simanullang, H. G., & Hutapea, M. I. (2023). Supervised Learning Metode K-Nearest Neighbor untuk Prediksi Diabetes pada Wanita. *METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi*, 7(1), 144-149.

- [32] Sri Wulandari KN, A., & Irmawati, I. (2017). Pengaruh Quality Of Work Life, Self Determination, Dan Job Performance Terhadap Work Engagement Karyawan (Studi Pada Karyawan Pt. Pamor Spinning Mill's Di Karanganyar) (Doctoral dissertation, Universitas Muhammdiyah Surakarta).
- [33] Syamsiah, S. (2019). Perancangan Flowchart dan Pseudocode Pembelajaran Mengenal Angka dengan Animasi untuk Anak PAUD Rambutan. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, 4(1), 86-93.
- [34] Thaker, N., & Shukla, A. (2020). Python as multi paradigm programming language. *International Journal of Computer Applications*, 177(31), 38-42.
- [35] Wazery, Y. M., Saber, E., Houssein, E. H., Ali, A. A., & Amer, E. (2021). An efficient slime mould algorithm combined with k-nearest neighbor for medical classification tasks. *IEEE Access*, 9, 113666-113682.
- [36] Xing, W., & Bei, Y. (2019). Medical health big data classification based on KNN classification algorithm. *IEEE Access*, 8, 28808-28819.
- [37] Yapentra, A. (2021). Pengaruh Kepemimpinan Terhadap Kinerja Karyawan Pada Divisi Pasang Baru PT. Telkom Akses Pekanbaru. *Eko dan Bisnis: Riau Economic and Business Review*, 12(1), 97-104.
- [38] Ziogas, A. N., Schneider, T., Ben-Nun, T., Calotoiu, A., De Matteis, T., de Fine Licht, J., ... & Hoefler, T. (2021, November). Productivity, portability, performance: Data-centric Python. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (pp. 1-13).
- [39] Z. R. Mair, W. Cholil, E. Yulianti, D. Marcelina, Theresiawati and I. N. Isnainiyah, "Convolutional Neural Network Analysis on Handwriting Patterns and Its Relationship to Personality: A Systematical Review," 2023 International Conference on Informatics, Multimedia, Cyber and Informations System (ICIMCIS), Jakarta Selatan, Indonesia, 2023, pp. 308-312, doi: 10.1109/ICIMCIS60089.2023.10348999.
- [40] Sari, N., Gustriasyah, R. and Mair, Z.R., 2023. Analisis User Experience pada Aplikasi Prodeskel Menggunakan Metode System Usability Scale (SUS), User Experience Questionnaire (UEQ), dan Heart Metrics (Doctoral dissertation, Universitas Indo Global Mandiri)

Copyrights

Copyright for this article is retained by the author(s), with first publication rights granted to the journal.

This is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>)

