PENERAPAN ALGORITMA *K-NEAREST NEIGHBOR* DAN C4.5 UNTUK KLASIFIKASI PENYAKIT KANKER SERVIKS

Muhammad Ridzki Ramadhan
Universitas Buana Perjuangan Karawang
Karawang, Indonesia
if18.muhammadramadhan@mhs.ubpkarawang.ac.id

Yana Cahyana
Universitas Buana Perjuangan Karawang
Karawang, Indonesia
yana.cahyana@ubpkarawang.ac.id

Ayu Ratna Juwita
Universitas Buana Perjuangan Karawang
Karawang, Indonesia
ayurj@ubpkarawang.ac.id

Abstract*— Kanker serviks merupakan penyebab kematian nomor dua pada perempuan di dunia setelah kanker payudara, sedangkan di Indonesia kanker serviks menduduki peringkat pertama, hal tersebut yang menjadikan masalah kesehatan reproduksi di Indonesia masih menjadi sorotan utama. *K-Nearest Neighbor* (KNN) adalah suatu metode algoritma *supervised learning*, di mana kelas yang paling banyak muncul (mayoritas) yang akan menjadi kelas hasil klasifikasi. Sedangkan Algoritma C4.5 merupakan sebuah algoritma klasifikasi yang digunakan untuk membangun *decision tree* (pohon keputusan). Penelitian kali ini bertujuan untuk mengklasifikasikan data resiko kebiasaan kanker serviks dengan menerapkan algoritma KNN dan C4.5. Data diambil dari *website UCI Machine Learning* sebanyak 72 data dan 19 atribut setelah dilakukan seleksi menjadi 63 data dan 5 atribut yang diantaranya adalah dukungan sosial instrumental, pengetahuan pemberdayaan, kemampuan pemberdayaan dan keinginan pemberdayaan lalu untuk kanker serviks dijadikan untuk atribut kelas. Pengujian ini dilakukan dengan cara manual, pemrograman *python* dan *rapidminer*. Penghitungan algoritma KNN telah dilakukan pada pengujian menggunakan *rapidminer* dengan *cross validation* kemudian menghasilkan akurasi 80.95% dan dengan *split validation* membagi data menjadi data *training* dan *testing* sebesar 80 : 20 menghasilkan akurasi 83.33%, sedangkan algoritma C4.5 dengan *cross validation* kemudian menghasilkan akurasi 76.19% dan dengan *split validation* membagi data menjadi data *training* dan *testing* sebesar 80 : 20 menghasilkan akurasi 75.00%. Untuk pengujian dengan pemrograman *python* dengan *split validation* membagi data menjadi data *training* dan *testing* sebesar 80 : 20 kemudian algoritma KNN mendapatkan hasil akurasi 84.00%, sedangkan algoritma C4.5 menghasilkan akurasi 69.00%. Sehingga algoritma KNN dengan pengujian *Python* mendapatkan akurasi terbaik pada penelitian ini dengan nilai akurasi 84.00%.

Kata Kunci : Klasifikasi, Kanker Serviks, K-Nearest Neighbor (KNN), C4.5

I. PENDAHULUAN

Meningkatnya angka pengidap kanker jadi fenomena menakutkan yang sukses menarik atensi warga dunia dalam waktu pendek. Dikala ini, telah jadi pembicaraan universal bahwa wanita sudah jadi sasaran sebagian kategori kanker ganas. Tetapi pada realitasnya, para wanita masih saja menyepelehkan serta tidak menguasai bagaimana kiat melaksanakan pola hidup sehat dan merawat badan dengan benar. Kanker serviks banyak berlangsung di Negara yang tengah berkembang, Perihal ini dikarenakan buruknya pola hidup warga yang kurang mencermati kesehatan badannya [1]. Sebagaimana [2] dalam penelitiannya menyebutkan bahwa adanya data histori rekam medis pasien yang tidak disertai dengan diakukannya ekstraksi data menjadi sebuah informasi yang berguna bagi keputusan klinis, mengakibatkan kurangnya pengetahuan wanita tehadap bahaya kanker serviks, sehingga sampai saat ini kasus kanker serviks terus meningkat.

Setelah memahami permasalahan diatas, maka solusinya adalah penanggulangan kanker serviks harus dilakukan sejak dini, dengan cara mengolah data menjadi informasi yang akurat dengan menggunakan metode klasifikasi pada data mining sehingga berguna bagi keputusan klinis ataupun dapat dijadikan pengetahuan bagi wanita akan bahaya nya kanker serviks. Setidaknya pada sepuluh hingga dua puluh tahun kedepan ketika mencapai usia rentan gejala kanker serviks timbul, diharapkan upaya ini dapat meminimalisir jumlah angka penderita kanker serviks [1].

Telah dilakukan penelitian oleh [3] yang menerapkan algoritma SVM untuk klasifikasi penyakit kanker serviks, pada *website Archive.com* data yang akan di ambil sejumlah 72 data dan 19 atribut dengan menggunakan data latih sejumlah 59 data dan 4 atribut, pengujian dengan menggunakan *orange* yang membagi data 80:20 menghasilkan akurasi senilai 92,9%, dan *python* senilai 87%. Kemudian [4] dalam penelitiannya yang menerapkan algoritma KNN untuk melakukan klasifikasi pada data ISPU di wilayah Jabodetabek dengan menggunakan algoritma KNN, dengan data latih sejumlah 304 serta data uji sejumlah 1 data saja sehingga diperoleh nilai akurasi senilai 95.78% dengan menetapkan K=7. Selanjutya [5] dalam penelitiannya menggunakan Algoritma C4.5 untuk klasifikasi pengidap penyakit diabetes, Pengujian pada penelitian dengan algoritma C4.5 ini menciptakan akurasi yang cukup besar yakni 97,12 %, *Precision* senilai 93,02%, serta *Recall* senilai 100,00%. Ada pula Kurva ROC (*Reciver Operating Characteristic*) menampilkan algoritma C4.5 mempunyai nilai AUC sebesar 0.994 yang artinya *Excellent Classification*.

Berdasarkan masalah serta metode pada penelitian sebelumnya, sehingga penelitian ini bertujuan untuk mengetahui tingkat akurasi algoritma *K-Nearest Neighbor* dan C4.5, agar dapat dilakukan evaluasi dengan maksud untuk menilai performa dari masing-masing algoritma. Sehingga dari kedua algoritma yang akan di uji tersebut, diketahui algoritma yang memiliki kinerja paling baik dalam mengklasifikasikan data resiko kebiasaan kanker serviks.

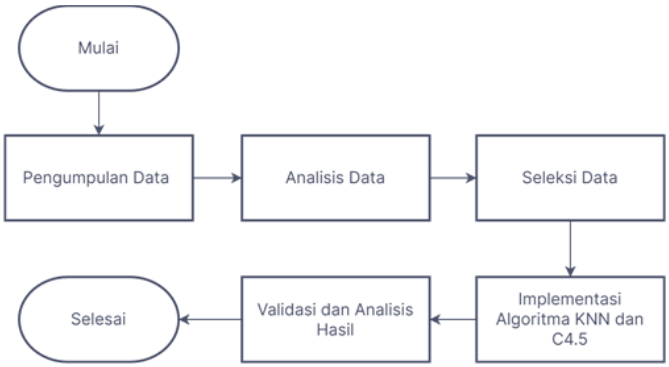
II. METODE PENELITIAN

A. Objek Penelitian

Pada penelitian ini hal penting yang akan penulis lakukan yaitu mempersiapkan bahan penelitian, dalam hal ini penulis akan menggunakan dataset berupa data kanker serviks yang di dapat dari *website UCI Machine Learning Repository*. Sehingga data tersebut dapat penulis klasifikasikan dengan menerapkan algoritma KNN dan C4.5.

B. Prosedur Penelitian

Pada penelitian ini hal penting yang akan penulis lakukan yaitu mempersiapkan bahan penelitian, dalam hal ini penulis akan menggunakan dataset berupa data kanker serviks yang di dapat dari *website UCI Machine Learning Repository*. Sehingga data tersebut dapat penulis klasifikasikan dengan menerapkan algoritma KNN dan C4.5.



Gambar 1 Alur Penelitian

1. Pengumpulan Data

   Pengumpulan data adalah tahapan yang bertujuan untuk memperoleh informasi maupun data yang berkaitan dengan penelitian. Bahan atau data yang akan digunakan dalam penelitian ini yaitu dataset yang diperoleh dari *website UCI Machine Learning Repository* oleh DR. Sobar, Prof. Rizanda Machmud, dan Adi Wijaya, PhD candidate. Yang berisi data resiko kebiasaan penyakit kanker serviks dengan jumlah total atribut 19 dari 72 data, sehingga diharapkan dapat mendukung untuk proses klasifikasi penyakit kanker serviks.

2. Analisis Data

   Analisis data bertujuan untuk memahami data agar dapat diseleksi dan dikelola. Sehingga memudahkan dalam menentukan kesalahan atau *error* dalam mengakurasikan algoritma KNN dan C4.5 terhadap klasifikasi penyakit kanker serviks.

3. Seleksi Data

   Seleksi data adalah proses pembersihan data dari atribut-atribut yang tidak diperlukan dan mengeliminasi data jika memiliki *missing value/duplicate value* yang tidak diperlukan dalam pengolahan *data mining*. Data sebelum dan setelah di seleksi terdapat pada gambar berikut:

	perilaku seksual risiko	perilaku makan	perilaku kebersihan pribadi	agregasi niat	komitmen niat	konsistensi sikap	spontanitas sikap	norma orang penting	penerimaan norma	kerentanan persepsi	keparahan persepsi	kekuatan motivasi	motivasi kemauan	dukungan sosial emosionalitas	apr... dul...
0	10	13	12	4	7	9	10	1	8	7	3	14	8	5	
1	10	11	11	10	14	7	7	5	5	4	2	15	13	7	
2	10	15	3	2	14	8	10	1	4	7	2	7	3	3	
3	10	11	10	10	15	7	7	1	5	4	2	15	13	7	
4	8	11	7	8	10	7	8	1	5	3	2	15	5	3	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
67	10	14	14	10	15	6	7	5	15	14	10	15	13	9	
68	10	12	15	10	15	8	8	5	15	14	8	12	14	11	
69	10	8	11	6	10	6	4	3	13	9	8	14	12	9	
70	9	12	13	10	13	6	6	5	14	13	10	13	12	11	
71	10	14	14	6	12	7	8	5	15	12	10	10	13	11	

72 rows × 20 columns

Gambar 2 Data Sebelum di Seleksi

	dukungan sosial instrumental	pengetahuan pemberdayaan	kemampuan pemberdayaan	keinginan pemberdayaan	kanker serviks
0	12	12	11	8	positif
1	5	5	4	4	positif
2	11	3	3	15	positif
3	4	4	4	4	positif
4	12	5	4	7	positif
...	...	...	...	...	...
58	12	12	11	9	negatif
59	13	15	11	14	negatif
60	11	12	10	10	negatif
61	12	11	13	15	negatif
62	14	13	15	15	negatif

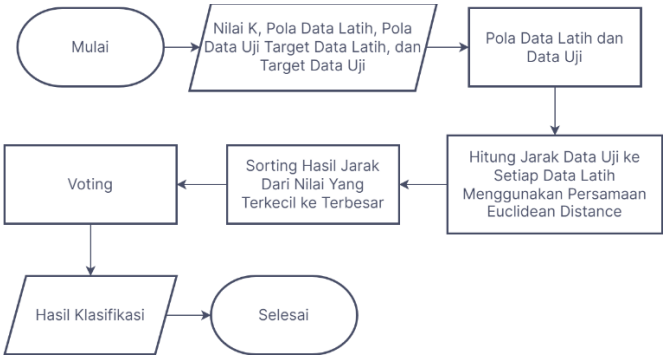
63 rows × 5 columns

Gambar 3 Data Setelah di Seleksi

Pada Gambar 3 terdapat hasil seleksi berdasarkan data yang memiliki *missing value* atau data yang tidak lengkap memiliki nilai *default* dengan menggunakan nilai median. Sehingga data yang sebelumnya sebanyak 72 data dan 19 atribut menjadi 63 data dan 5 atribut yang diantaranya adalah dukungan sosial instrumental, pengetahuan pemberdayaan, kemampuan pemberdayaan dan keinginan pemberdayaan lalu untuk kanker serviks dijadikan untuk atribut kelas jika data bernilai 1 maka data dinyatakan positif jika 0 maka dinyatakan negatif.

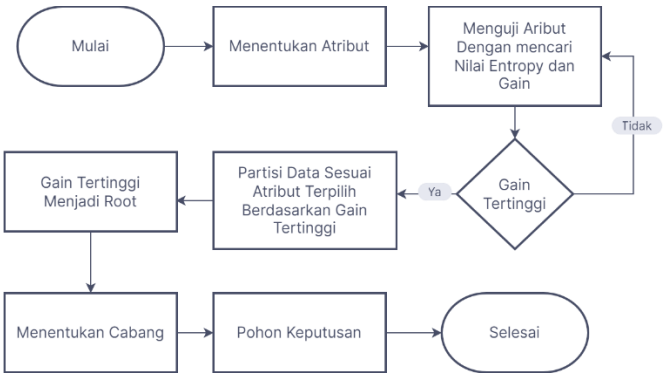
4. Implementasi Algoritma KNN dan C4.5

Pada proses ini yaitu menerapkan algoritma KNN, dengan penghitungan menggunakan *platform RapidMiner* dan program *Python*. Selain menerapkan pada *flatform*, pada tahap ini juga akan dilakukan penghitungan secara manual dengan menggunakan rumus algoritma KNN dan C4.5.



Gambar 4 *Flowchart* KNN

Gambar 4 Merupakan cara kerja algoritma KNN, yang diawali dengan menentukan nilai K, menentukan pola data latih/data uji, dan menetukan target data latih/data uji. Kemudian proses pola data uji dan data latih. Lalu hitung jarak data uji ke data latih dengan *Euclidean Distance*. Selanjutnya urutkan dari jarak yang tekecil hingga terbesar. Kemudian dilakukan *voting* dan kelas yang paling banyak muncul (mayoritas) yang akan menjadi kelas hasil klasifikasi.



Gambar 5 *Flowchart* C4.5

Gambar 5 Merupakan cara kerja algoritma C45, yang diawali dengan menentukan atribut yang akan digunakan. Kemudian mencari *gain* tertinggi berdasarkan hasil penghitungan *entropy* dari masing masing atribut agar dapat dilakukan pengujian. Apabila ditemukan *gain* tertinggi, maka *gain* tersebut akan menjadi *root* awal. Selanjutnya mencari *gain* tertinggi dari hasil setiap partisi untuk menentukan cabang. Kemudian pembuatan pohon keputusan untuk mengetahui hasil dari klasifikasi.

5. Validasi dan Analisis Hasil

Pada tahap validasi ini diambil dari hasil akurasi algoritma KNN dan C4.5. kemudian algoritma yang mempunyai nilai akurasi yang tinggi merupakan algoritma terbaik pada proses klasifikasi ini. untuk mendapatkan nilai akurasi dalam klasifikasi kanker serviks.

III. HASIL DAN PEMBAHASAN

A. Persiapan Data

Hal pertama yang dilakukan yaitu mempersiapkan dataset yang diperoleh dari website UCI MACHINE LEARNING, yang berisi data resiko kebiasaan penyakit kanker serviks dengan jumlah total 19 atribut dari 72 data , yang kemudian setelah di seleksi menjadi 63 data dan 5 atribut diantaranya adalah dukungan sosial instrumental, pengetahuan pemberdayaan, kemampuan pemberdayaan dan keinginan pemberdayaan lalu untuk kanker serviks dijadikan sebagai atribut kelas. Sehingga hal tersebut diharapkan dapat mendukung untuk proses klasifikasi penyakit kanker serviks dengan algoritma K-Nearest Neighbor dan C4.5.

Tabel 1 Dataset Resiko Kebiasaan Kanker Serviks

No	dukungan sosial instrumental	pengetahuan pemberdayaan	kemampuan pemberdayaan	keinginan pemberdayaan	kanker serviks
1	12.0	12.0	11.0	8.0	positif
2	5.0	5.0	4.0	4.0	positif
3	11.0	3.0	3.0	15.0	positif
4	4.0	4.0	4.0	4.0	positif
5	12.0	5.0	4.0	7.0	positif
6	7.0	13.0	9.0	6.0	positif
7	15.0	3.0	3.0	5.0	positif
.					
.					
63	14.0	13.0	15.0	15.0	negatif

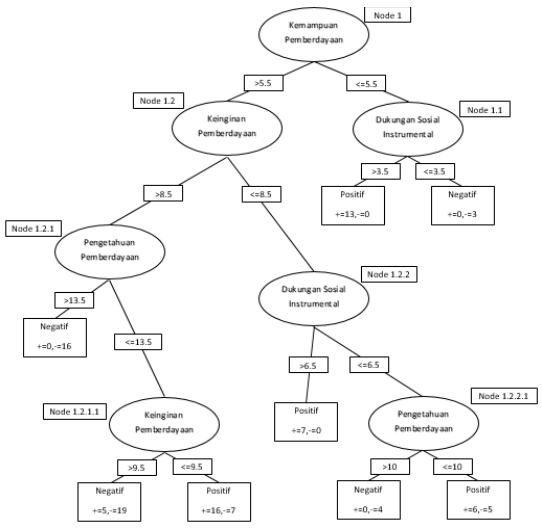
B. Hasil Penghitungan Manual

1. Algoritma *K-Nearest Neighbor* (KNN)

   Data *training* diambil dari *dataset* nomor urut 1 sampai 62 sedangkan data *testing* diambil dari *dataset* nomor urut 63. Dengan dihitung menggunakan metode *euclidean distance*, memperoleh hasil "Positif " = 1 dan "Negatif " = 9 dengan K = 7, sesuai *voting* maka dapat disimpulkan bahwa *dataset* nomor urut 63 yaitu (14.0, 13.0, 15.0, 15.0), mendapatkan klasifikasi Negatif karena dari 10 data (K=7) yang cocok dengan dataset lebih dominan ke "Negatif" dengan jumlah 9 data, sedangkan "Positif" hanya 1 data saja.

2. Algoritma C4.5

   Dataset yang digunakan dalam penghitungan ini, menggunakan 63 data resiko kebiasaan kanker serviks. Dari 63 data tersebut terdapat 21 data dengan kelas "Positif" dan 42 data dengan kelas "Negatif". Penghitungan algoritma C4.5, hal pertama yang dilakukan yaitu dengan menentukan *gain* tertinggi. Untuk menentukan *gain* tertinggi yaitu dengan menghitung *entropy* keseluruhan. Setelah mendapatkan *entropy*, selanjutnya adalah menentukan *gain* dari setiap atribut. Kemudian menentukan *gain* tertinggi yang akan dijadikan akar dari cabang pohon keputusan. Sehingga meghasilkan pohon keputusan seperti pada Gambar 6.



Gambar 6 Pohon Keputusan C4.5 (manual)

Pada gambar 6 dijelaskan, jika "Kemampuan Pemberdayaan ≤ 5.5" maka harus dilihat dulu ke atribut "Dukungan Sosial Instrumental", jika Dukungan Sosial Instrumental > 3.5 = Positif dan jika Dukungan Sosial Instrumental ≤ 3.5 = Negatif. Sedangkan jika "Kemampuan Pemberdayaan > 5.5" maka harus dilihat dulu ke atribut "Keinginan Pemberdayaan" jika Keinginan Pemberdayaan > 8.5 maka harus dilihat dulu ke atribut "Pengetahuan Pemberdayaan", jika Pengetahuan Pemberdayaan > 13.5 = Negatif dan jika Pengetahuan Pemberdayaan ≤ 13.5 maka harus dilihat dulu ke atribut "Keinginan Pemberdayaan", jika Keinginan Pemberdayaan > 9.5 = Negatif dan jika Keinginan Pemberdayaan ≤ 9.5 = Positif. Sedangkan jika "Kemampuan Pemberdayaan > 5.5" maka harus dilihat dulu ke atribut "Keinginan Pemberdayaan" jika Keinginan Pemberdayaan ≤ 8.5 maka harus dilihat dulu ke atribut "Dukungan Sosial Instrumental" , jika Dukungan Sosial Instrumental > 6.5 = Positif dan jika Dukungan Sosial Instrumental ≤ 6.5 maka harus dilihat dulu ke atribut "Pengetahuan Pemberdayaan", jika Pengetahuan Pemberdayaan > 10 = Negatif dan jika Pengetahuan Pemberdayaan ≤ 10 = Positif.

C. Implementasi *RapidMiner*

1. Algoritma *K-Nearest Neighbor* (KNN)

Berikut merupakan tahapan dalam mengolah dataset dengan algoritma KNN (*cross validation*) pada *rapidminer studio*:

a. *Import dataset* yang sudah disiapkan pada *tool rapidminer*

b. *Drag and drop dataset* pada *frame desain*

c. *Drag and drop* operator yang diperlukan untuk mengolah *dataset*. yaitu *set role* dan *cross validation* (*fold*=10) pada *frame desain*

d. Klik dua kali operator *cross validation*, kemudian *drag and drop* operator KNN (K=7) pada *frame training* dan *apply model*, *performance* pada *frame testing*

e. *Setting* dan hubungkan masing-masing operator dengan operator lain yang saling berkaitan

f. Klik *run* untuk melihat hasilnya

Hasil akurasi yang diperoleh dari *dataset* yang telah diproses oleh *rapidminer* (*cross validation*) dengan menggunakan algoritma KNN adalah 80.95%. seperti pada Gambar 7.

accuracy: 80.95% +/- 14.50% (micro average: 80.95%)

	true positif	true negatif	class precision
pred. positif	15	6	71.43%
pred. negatif	6	36	85.71%
class recall	71.43%	85.71%	

Gambar 7 Hasil Akurasi *Rapidminer* (*cross validation* KNN)

Berikut merupakan tahapan dalam mengolah dataset dengan algoritma KNN (*split validation*) pada *rapidminer studio*:

a. *Import dataset* yang sudah disiapkan pada *tool rapidminer*

b. *Drag and drop dataset* pada *frame desain*

c. *Drag and drop* operator yang diperlukan untuk mengolah *dataset*. yaitu *set role*, *split data* (80% *training* 20% *testing*), KNN (K=7), *apply model*, dan *performace* pada *frame desain*

d. *setting* dan hubungkan masing-masing operator dengan operator lain yang saling berkaitan

e. Klik *run* untuk melihat hasilnya

Hasil akurasi yang diperoleh dari *dataset* yang telah diproses oleh *rapidminer* (*split validation*) dengan menggunakan algoritma KNN adalah 83.33%. seperti pada Gambar 8.

accuracy: 83.33%

	true positif	true negatif	class precision
pred. positif	2	0	100.00%
pred. negatif	2	8	80.00%
class recall	50.00%	100.00%	

Gambar 8 Hasil Akurasi *Rapidminer* (*split validation* KNN)

2. Algoritma C4.5

Berikut merupakan tahapan dalam mengolah dataset dengan algoritma C4.5 (*cross validation*) pada *rapidminer studio*:

a. *Import dataset* yang sudah disiapkan pada *tool rapidminer*

b. *Drag and drop dataset* pada *frame desain*

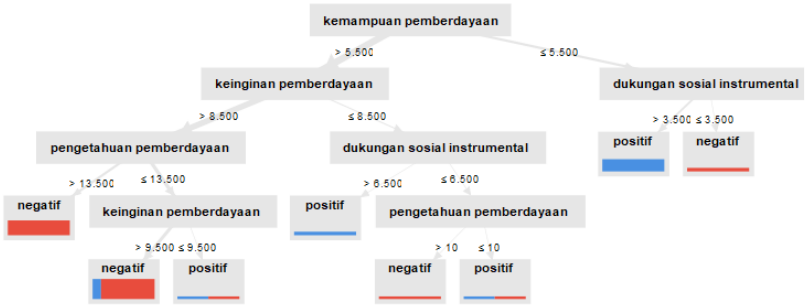
c. *Drag and drop* operator yang diperlukan untuk mengolah dataset. yaitu *set role* dan *cross validation* (*fold*=10) pada *frame desain*

d. Klik dua kali operator *cross validation*, kemudian *drag and drop* operator *decision tree* pada *frame training dan apply model*, *performance* pada *frame testing*

e. *Setting* dan hubungkan masing-masing operator dengan operator lain yang saling berkaitan

f. Klik *run* untuk melihat hasil dan pohon keputusannya.

Hasil pohon keputusan yang didapatkan dari *dataset* yang diproses oleh *rapidminer* (*cross validation*) dengan algoritma C4.5 bisa dilihat pada Gambar 9.



Gambar 9 Hasil Pohon Keputusan (*rapidminer*)

Tree

```
kemampuan pemberdayaan > 5.500
|   keinginan pemberdayaan > 8.500
|   |   pengetahuan pemberdayaan > 13.500: negatif {positif=0, negatif=16}
|   |   pengetahuan pemberdayaan ≤ 13.500
|   |   |   keinginan pemberdayaan > 9.500: negatif {positif=3, negatif=19}
|   |   |   keinginan pemberdayaan ≤ 9.500: positif {positif=1, negatif=1}
|   keinginan pemberdayaan ≤ 8.500
|   |   dukungan sosial instrumental > 6.500: positif {positif=3, negatif=0}
|   |   dukungan sosial instrumental ≤ 6.500
|   |   |   pengetahuan pemberdayaan > 10: negatif {positif=0, negatif=2}
|   |   |   pengetahuan pemberdayaan ≤ 10: positif {positif=1, negatif=1}
kemampuan pemberdayaan ≤ 5.500
|   dukungan sosial instrumental > 3.500: positif {positif=13, negatif=0}
|   dukungan sosial instrumental ≤ 3.500: negatif {positif=0, negatif=3}
```

Gambar 10 Deskripsi Pohon Keputusan (*rapidminer*)

Kemudian untuk hasil akurasi yang diperoleh dari *dataset* yang telah diproses oleh *rapidminer* (*cross validation*) dengan menggunakan algoritma C4.5 adalah 76.19%. Seperti pada Gambar 11.

accuracy: 76.19% +/- 11.39% (micro average: 76.19%)

	true positif	true negatif	class precision
pred. positif	16	10	61.54%
pred. negatif	5	32	86.49%
class recall	76.19%	76.19%	

Gambar 11 Hasil Akurasi *Rapidminer* (*cross validation* C4.5)

Berikut merupakan tahapan dalam mengolah dataset dengan algoritma C4.5 (*split validation*) pada *rapidminer studio*:

a. *Import dataset* yang sudah disiapkan pada *tool rapidminer*

b. *Drag and drop dataset* pada *frame desain*

c. *Drag and drop* operator yang diperlukan untuk mengolah *dataset*. yaitu *set role*, *split data* (80% *training* 20% *testing*),

decision tree, *apply model*, dan *performace* pada *frame desain*

d. *Setting* dan hubungkan masing-masing operator dengan operator lain yang saling berkaitan

e. Klik *run* untuk melihat hasilnya

Hasil akurasi yang diperoleh dari *dataset* yang telah diproses oleh *rapidminer* (*split validation*) dengan menggunakan algoritma C4.5 adalah 75.00%. seperti pada Gambar 12.

accuracy: 75.00%

	true positif	true negatif	class precision
pred. positif	2	1	66.67%
pred. negatif	2	7	77.78%
class recall	50.00%	87.50%	

Gambar 12 Hasil Akurasi *Rapidminer* (*split validation* C4.5)

D. Implementasi *Python*

1. Algoritma *K-Nearest Neighbor* (KNN)

Pada proses implementasi *python* dengan algoritma KNN ini, *tools* yang digunakan adalah *google colaboratory*. Untuk pengoperasiannya dilakukan proses *coding* dengan menggunakan bahasa pemrograman *python*. *Library* yang digunakan yaitu *numpy* dan *pandas*. Seperti pada Gambar 13.

```
[1]  import numpy as np
     import pandas as pd

[2]  ks = pd.read_excel("kankerserviks.xlsx")

[3]  atr_ks = ks.drop(columns = 'kanker serviks')

[4]  cls_ks = ks['kanker serviks']

[5]  from sklearn.model_selection import train_test_split
     from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
     from sklearn.neighbors import KNeighborsClassifier

[6]  xtrain, xtest, ytrain, ytest = train_test_split(atr_ks, cls_ks, test_size=0.2, random_state=0)
     classifier_ks = KNeighborsClassifier(n_neighbors=7, metric= 'euclidean', p=2)
     classifier_ks.fit(xtrain, ytrain)

     KNeighborsClassifier(metric='euclidean', n_neighbors=7)
```

Gambar 13 *Library* dan Model *K-Nearest Classifier*

```
[7]  y_pred = classifier_ks.predict(xtest)
     cm = confusion_matrix(ytest, y_pred)
     print("CONFUSION MATRIX")
     print(cm)
     akurasi = classification_report(ytest, y_pred)
     print("                    TINGKAT AKURASI ALGORITMA KNN")
     print(akurasi)
     akurasi = accuracy_score(ytest, y_pred)
     print("TINGKAT AKURASI: %d persen" %(akurasi*100))

     CONFUSION MATRIX
     [[8 2]
      [0 3]]
                         TINGKAT AKURASI ALGORITMA KNN
                    precision    recall  f1-score   support

          negatif       1.00      0.80      0.89        10
          positif       0.60      1.00      0.75         3

         accuracy                           0.85        13
        macro avg       0.80      0.90      0.82        13
     weighted avg       0.91      0.85      0.86        13

     TINGKAT AKURASI: 84 persen
```

Gambar 14 Hasil Akurasi Python KNN

Pada Gambar 14 dijelaskan proses *coding* yang dilakukan yaitu dengan menggunakan algoritma KNN dengan nilai K=7, dengan mendapatkan hasil akurasi sebesar 84.00%.

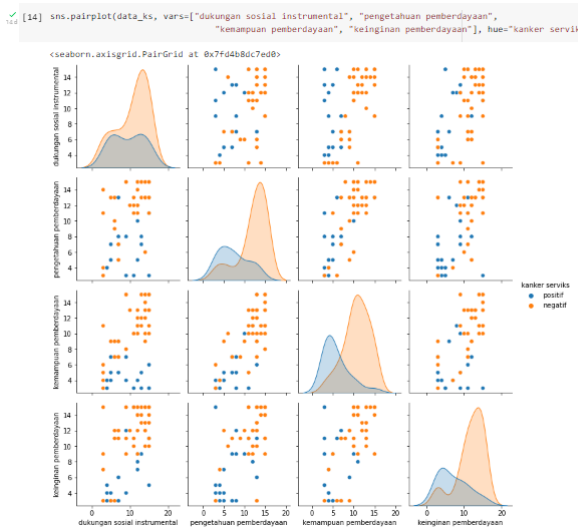
Pada Gambar 15 dan 16 terdapat baris kode pemrograman *python* yang akan memvisualisasikan data dengan sebuah *pairplot.*

```
[12] import seaborn as sns
     from matplotlib import pyplot as plt
     from sklearn.neighbors import KNeighborsClassifier

[13] data_ks = pd.read_excel("kankerserviks.xlsx")
     y = data_ks["kanker serviks"]
     x = data_ks.drop('kanker serviks', axis=1)
```

Gambar 15 *Library* dan Proses Pengolahan Visualisasi Data

Pada gambar 15 merupakan Proses *importing library* serta pengolahan data yang selanjutnya data akan di visualisasikan dengan menggunakan *pairplot.* Seperti pada Gambar 16.



Gambar 16 Proses dan Hasil *Pairplot*

2. Algoritma C4.5

Untuk proses implementasi *python* dengan algoritma C4.5 ini, *tools* yang digunakan sama dengan algoritma KNN yaitu *google colaboratory*. Untuk pengoperasiannya dilakukan proses *coding* dengan menggunakan bahasa pemrograman *python*. *Library* yang digunakan yaitu *numpy* dan *pandas.* Seperti pada Gambar 17.

```
[1] import numpy as np
    import pandas as pd

[2] ks = pd.read_excel("kankerserviks.xlsx")

[3] atr_ks = ks.drop(columns = 'kanker serviks')

[4] cls_ks = ks['kanker serviks']

[5] from sklearn.model_selection import train_test_split
    from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
    from sklearn.tree import DecisionTreeClassifier

[6] xtrain, xtest, ytrain, ytest = train_test_split(atr_ks, cls_ks, test_size=0.2, random_state=0)
    tree_ks = DecisionTreeClassifier(random_state=0)
    tree_ks.fit(xtrain, ytrain)

    DecisionTreeClassifier(random_state=0)
```

Gambar 17 *Library* dan Model *Decision Tree Classifier*

```
[7]  y_pred = tree_ks.predict(xtest)
     cm = confusion_matrix(ytest, y_pred)
     print("CONFUSION MATRIX")
     print(cm)
     akurasi = classification_report(ytest, y_pred)
     print("                    TINGKAT AKURASI ALGORITMA C4.5")
     print(akurasi)
     akurasi = accuracy_score(ytest, y_pred)
     print("TINGKAT AKURASI: %d persen" %(akurasi*100))

     CONFUSION MATRIX
     [[6 4]
      [0 3]]
                         TINGKAT AKURASI ALGORITMA C4.5
                    precision    recall  f1-score   support

         negatif       1.00      0.60      0.75        10
         positif       0.43      1.00      0.60         3

        accuracy                           0.69        13
       macro avg       0.71      0.80      0.67        13
    weighted avg       0.87      0.69      0.72        13

     TINGKAT AKURASI: 69 persen
```

Gambar 18 Hasil Akurasi Python C4.5

Pada Gambar 18 merupakan proses *coding* akurasi algoritma C4.5 dengan hasil akurasi 69.00%.

E. Validasi dan Analisis Hasil

Dari hasil pengujian menggunakan *Rapidminer* dan *Python*, didapatkan akurasi yang lebih baik merupakan algoritma KNN dibandingkan dengan algoritma C4.5. Adapun untuk operator validasi yang digunakan pada pada *tool RapidMiner* menggunakan operator *cross validation* dan *split validation*, sedangkan pada *Python* hanya menggunakan operator *split validation* saja. Maka algoritma KNN dengan pengujian *Python* menghasikan akurasi terbaik dengan nilai persentase 84.00%. untuk nilai akurasi bisa dilihat pada Tabel 2.

Tabel 2 Hasil Pengujian *RapidMiner* dan *Python*

Pengujian		Algoritma	Akurasi
Rapidminer	*Cross V*	KNN	80.95%
		C4.5	76.19%
	Split V	KNN	83.33%
		C4.5	75.00%
Python	*Split V*	KNN	84.00%
		C4.5	69.00%

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari hasil dan pembahasan pada penelitian yang telah dilakukan ini, dapat diambil beberapa kesimpulan sebagai berikut:

1) Penerapan algoritma *K-Nearest Neighbor* (KNN) dan C4.5 pada proses klasifikasi penyakit kanker serviks menggunakan penghitungan manual sebagai pengujian dan membentuk pohon keputusan. Lalu penghitungan dengan menggunakan *rapidminer* dan pemrograman *python* untuk menghasilkan akurasi dari kedua algoritma.

2) Penghitungan algoritma KNN telah dilakukan pada pengujian menggunakan *rapidminer* dengan *cross validation* kemudian menghasilkan akurasi 80.95% dan dengan split validation membagi data menjadi data *training* dan *testing* sebesar 80 : 20 menghasilkan akurasi 83.33%, sedangkan algoritma C4.5 dengan *cross validation* kemudian menghasilkan akurasi 76.19% dan dengan split validation membagi data menjadi data *training* dan *testing* sebesar 80 : 20 menghasilkan akurasi 75.00%. Untuk pengujian dengan pemrograman *python* dengan *split validation* membagi data menjadi data *training* dan *testing* sebesar 80 : 20 kemudian algoritma KNN mendapatkan hasil akurasi 84.00%, sedangkan algoritma C4.5 menghasilkan akurasi 69.00%. Sehingga algoritma KNN dengan pengujian *Python* mendapatkan akurasi terbaik pada penelitian ini dengan nilai akurasi 84.00%.

B. Saran

Adapun saran pada penelitian ini yaitu agar dapat dilakukan penelitian lebih lanjut dengan menggunakan pengujian ataupun algoritma lain, sehingga dapat diperoleh perbandingan tingkat akurasi pada klasifikasi penyakit kanker serviks.

PENGAKUAN

Naskah ilmiah ini adalah sebagai dari penelitian Tugas Akhir milik Muhammad Ridzki Ramadhan, dengan judul Klasifikasi Penyakit Kanker Serviks Dengan Algoritma K-Nearest Neighbor dan C4.5. Yang dibimbing oleh Yana Cahyana dan Ayu Ratna Juwita.

DAFTAR PUSTAKA

[1] R. M. Andanni, “Perancangan Strategi Kampanye Bahaya Kanker Serviks Bagi Remaja Putri Usia 13-18 Tahun,” 2016, [Online]. Available: https://repository.its.ac.id/id/eprint/75601%0A.

[2] T. Praningki and I. Budi, “Sistem Prediksi Penyakit Kanker Serviks Menggunakan CART, Naive Bayes, dan k-NN,” *Creat. Inf. Technol. J.*, vol. 4, no. 2, p. 83, 2018, doi: 10.24076/citec.2017v4i2.100.

[3] S. S. Arifin, A. M. Siregar, A. Ratna, and T. Al Mudzakir, “Klasifikasi Penyakit Kanker Serviks Menggunakan Algoritma Support Vector Machine (SVM),” no. Ciastech, pp. 521–528, 2021.

[4] S. Nurjanah, A. M. Siregar, and D. S. Kusumaningrum, “Penerapan Algoritma K – Nearest Neighbor (Knn) Untuk Klasifikasi Pencemaran Udara Di Kota Jakarta,” *Sci. Student J. Information, Technol. Sci.*, vol. 1, no. 2, pp. 71–76, 2020.

[5] F. M. Hana, “Klasifikasi Penderita Penyakit Diabetes Menggunakan Algoritma Decision Tree C4.5,” *J. SISKOM-KB (Sistem Komput. dan Kecerdasan Buatan)*, vol. 4, no. 1, pp. 32–39, 2020, doi: 10.47970/siskom-kb.v4i1.173.