Journal of Artificial Intelligence and Data Engineering (JAIDE)
Vol. 1, No. 1, July 2024, page 28-34
E-ISSN: xxxx-xxxx, P-ISSN: xxx-xxxx
DOI: https://doi.org/xx.xxxxx/jaide.xxxx.x
https://journal.beta-academia.com/index.php/jaide

Perbandingan Metode *C45* dan *Naive Baiyes* untuk Sistem Prediksi Pemilihan Jurusan di SMK Muhammadiyah 10 Kisaran

Dina Pertiwi[1], Khairunnisa[2], Sri Damayanti[3]
[1,2,3] Mahasiswa Program Studi Sistem Informasi, Sekolah Tinggi Manajemen Informatika dan Komputer Royal
[1] ceritadinapertiwi@gmail.com *

* Email Koresponden

INFO ARTIKEL
Histori Artikel
Diterima: 26/Juli/2024
Ditinjau: 28/Juli/2024
Disetujui: 31/Juli/2024

ABSTRAK

Penelitian ini dilatarbelakangi oleh banyaknya calon siswa yang gegabah memilih jurusan ketika hendak masuk SMK tanpa mempertimbangkan kemampuan. Berdasarkan hal ini, dinilai perlu dalam membuat sistem prediksi di SMK Muhammadiyah 10 Kisaran dengan metode perbandingan C45 dan Naive Baiyes. Metode Pohon Keputusan atau C45 digunakan karena mampu membuat pohon keputusan yang mudah digambarkan, serta mempunyai tingkat efisiensi dalam menangangi data atribut diskret atau numerik. Sementara metode Naive Bayes digunakan karena memiliki hasil akurasi yang tinggi. Penelitian ini dilakukan berdasarkan data siswa SMK Muhammadiyah 10 Kisaran yang berisi pertanyaan mengenai merasa salah jurusan, minat, dan faktor penentu jurusan lain, data tersebut dibagi menjadi 2, yakni variabel target (y) dan variabel fitur (x). Dataset yang berjumlah 316 dibagi menjadi data training dan data testing dengan perbandingan 70:30 di kedua metode untuk mendapatkan tingkat akurasi. Dari hasil yang diberikan, dapat diketahui bahwa algoritma C45 memiliki akurasi sebesar 85% dan algoritma Naive Bayes hanya memiliki akurasi sebesar 26%. Hal ini menunjukkan bahwa algoritma C45 lebih efektif dalam mengklasifikasikan dataset yang tersedia dibandingkan dengan algoritma Naive Bayes.

Kata Kunci: Metode C45, Naïve Bayes, Pyhton, Akurasi

ABSTRACT

This research is motivated by the large number of prospective students who simply choose a major when they want to enter a vocational school without considering their abilities. The Decision Tree or C45 method is used because it is able to make decision trees that are easy to describe, and has a level of efficiency in handling discrete and numeric attribute data. While the Naive Bayes method is used because it has a high accuracy of results. This research was conducted based on data from students of SMK Muhammadiyah 10 Kisaran which contained questions about feelings of wrong majors, interests, and determinants of other majors. Data is divided into 2 labels, namely free labels (y) and bound labels (x). Followed by dividing the dataset into training data and testing data with a ratio of 70:30 in both methods to get the level of accuracy. From the results given, it can be seen that the C45 algorithm has an accuracy of 85% and the Naive Bayes algorithm has an accuracy of 26%. This shows that the C45 algorithm is more effective in classifying the available datasets compared to the Naive Bayes.

Keywords: C45 Method, Naïve Bayes, Python, Accuracy

Journal of Artificial Intelligence and Data Engineering (JAIDE)
Vol. 1, No. 1, July 2024, page 28-34
E-ISSN: xxxx-xxxx, P-ISSN: xxx-xxxx
DOI: https://doi.org/xx.xxxxx/jaide.xxxx.x
https://journal.beta-academia.com/index.php/jaide

PENDAHULUAN

Sebagai bagian dari sekolah yang mengusung pendidikan formal dengan menyelenggarakan pendidikan kejuruan yang setara dengan pendidikan menengah atas, Sekolah Menengah Kejuruan (SMK) selalu berusaha meningkatkan mutu pendidikan sehingga lulusan dapat memiliki keahlian sehingga bisa bersaing di dunia kerja [1]. Namun sejalan dengan besarnya antusias lulusan SMP dalam melanjutkan pendidikan ke SMK, ternyata tidak dibarengi dengan kematangan siswa dalam memilih jurusan yang menyebabkan banyak permasalahan dalam menjalani pendidikannya [1].

Banyak calon siswa yang asal memilih jurusan ketika hendak masuk SMK tanpa mempertimbangkan kemampuan dan keberlanjutan masa pembelajaran secara jangka panjang. Hal ini mengakibatkan rendahnya *grade* nilai yang diperoleh siswa, permasalahan dalam memahami mata pelajaran, rendahnya minat belajar, hingga menghambat kelulusan siswa tersebut [2].

Penjurusan di SMK Muhammadiyah 10 Kisaran terdiri dari Teknik Komputer dan Jaringan (TKJ), Teknik Audio Video (TAV), serta Teknik dan Bisnis Sepeda Motor (TBSM) dimana penjurusan akan disesuaikan dengan kemampuan akademik, minat, serta bakat yang dimiliki siswa dengan tujuan agar siswa bisa mengikuti proses pembelajaran, serta mampu meningkatkan prestasi dan kenyamanan kepada calon siswa [3]. Selain itu, penjurusan yang tidak tepat dapat menurunkan minat belajar, kegaduhan di dalam kelas, kelesuan, permasalahan bolos, penurunan prestasi, serta hilangnya gairah belajar [3].

Berdasarkan permasalahan di atas, dinilai perlu dalam membuat sistem prediksi di SMK Muhammadiyah 10 Kisaran dengan metode perbandingan C45 dan *Naive Baiyes* dimana kedua metode akan diterapkan kepada data yang sudah ada untuk mengambil metode dengan nilai akurasi terbaik.

Metode Pohon Keputusan atau C45 digunakan karena mampu membuat pohon keputusan yang mudah digambarkan, mempunyai tingkat akurasi yang bisa diterima, serta mempunyai tingkat efisiensi dalam menangangi data atribut diskret atau numerik [4]. Sementara metode *Naive Bayes* digunakan karena memiliki hasil akurasi yang tinggi [5]

Penelitian ini dilakukan guna membandingkan hasil akurasi pemilihan jurusan di SMK Muhammadiyah 10 Kisaran dengan menggunakan metode C45 dan *Naive Baiyes.*

METODE

Algoritma C4.5

Algoritma C4.5 adalah algoritma yang digunakan untuk membangun pohon keputusan dari sebuah dataset. Algoritma C4.5 merupakan evolusi dari algoritma ID3, yang juga merupakan algoritma pohon keputusan [6]. Algoritma C4.5 mengunjungi setiap node keputusan secara rekursif, memilih cabang optimal hingga tidak ada lagi cabang yang memungkinkan. Algoritma C4.5 merupakan salah satu algoritma yang banyak digunakan khususnya dalam bidang *machine learning* yang memiliki berbagai peningkatan dibanding dengan ID3 [7].

Terdapat beberapa tahapan perhitungan C4.5, yaitu:

1. Menyiapkan data training yang diambil dari histori data yang pernah terjadi, kemudian data ini dijadikan ke dalam satu kelompok yang berada di kelas-kelas tertentu.
2. Menghitung akar pohon yang didapatkan melalui atribut yang dipilih dan akan dihitung nilai entropi kemudian menghitung nilai gainnya. Akar pertama ditentukan dari atribut yang memiliki nilai gain paling tinggi.

Nilai entropy dihitung dengan menggunakan rumus:

$$Entopy(S) = \sum_{i=0}^{n} - pi * Log2\ pi \qquad (1)$$

Dengan keterangan:
S: Himpunan Kasus
N: Jumlah Partisi S
Pi: Proporsi dari Si terhadap S

Setelah nilai entropy selesai dihitung, setiap atribut akan dihitung nilai gain dengan

Journal of Artificial Intelligence and Data Engineering (JAIDE)
Vol. 1, No. 1, July 2024, page 28-34
E-ISSN: xxxx-xxxx, P-ISSN: xxx-xxxx
DOI: https://doi.org/xx.xxxxx/jaide.xxxx.x
https://journal.beta-academia.com/index.php/jaide

menggunakan rumus:

$$Gain(S,A) = Entropy(S) - \sum_{i=1}^{n} \frac{|si|}{|s|} * Entropy\ (Si) \qquad (2)$$

Dengan keterangan:
S: Himpunan Kasus
A: Fitur
N: Jumlah partisi atribut A
|Si| : Jumlah kasus pada partisi ke-i
|S| : Jumlah kasus dalam S

Naive Baiyes

Naive bayes adalah metode pengklasifikasian yang menggunakan statistik dan probabilitas untuk memprediksi peluang yang ada di masa depan dengan menggunakan data pengalaman di masalalu yang juga disebut dengan teorema Bayes [8]. Bayes dikombinasikan dengan naive yang memiliki asumsi bahwa seriap atribut bebas dan tidak saling terikat [9].

Dalam sebuah dataset, setiap baris atau dokumen I dianggap sebagai vektor dari nilai-nilai atribut <x1, x2, …, xn>, di mana masing-masing nilai mewakili atribut Xi (i ∈ [1, n]). Setiap baris memiliki label kelas ci ∈ {c1, c2, …, ck} yang mewakili nilai variabel kelas C. Untuk melakukan klasifikasi, dapat dihitung probabilitas p(C=ci|X=xj). Karena pada Naïve Bayes diasumsikan bahwa setiap atribut bersifat independen, maka persamaan yang digunakan adalah sebagai berikut:

Pertama, probabilitas p(C=ci|X=xj) menunjukkan kemungkinan terjadinya atribut Xi dengan nilai xi jika diberikan kelas c. Dalam metode Naïve Bayes, kelas C merupakan kategori kualitatif, sementara atribut Xi bisa berupa data kualitatif maupun kuantitatif [10].

Jika atribut Xi berupa data kuantitatif, maka probabilitas p(X=xi|C=cj) biasanya sangat kecil, sehingga persamaan tersebut mungkin tidak efektif untuk atribut kuantitatif. Untuk mengatasi masalah ini, beberapa pendekatan dapat digunakan, seperti menggunakan distribusi normal (Gaussian).

Phyton

Python adalah bahasa pemrograman yang digunakan untuk menulis skrip (*scripting language*) dan berorientasi obyek [11]. Python adalah bahasa pemrograman yang dapat digunakan untuk berbagai aplikasi pengembangan perangkat lunak dan kompatibel dengan berbagai sistem operasi. Python adalah perangkat lunak sumber terbuka, yang berarti tidak ada batasan dalam hal penyalinan atau distribusi. [12].

Bahasa Python sangat populer karena memiliki beberapa keunggulan seperti:

1. Mudah penggunannya dalam pengembangan produk perangkat lunak, perangkat keras, IOT, web appalication, dan *video game.*
2. kode mudah dipahami, serta memiliki *library* yang sangat banyak dan luas.
3. Dukungan ekosistem *Internet of Things* sangat baik.

HASIL DAN PEMBAHASAN

Berdasarkan data yang diperoleh dari siswa SMK Muhammadiyah 10 Kisaran yang berisi pertanyaan mengenai merasa salah jurusan, minat, dan faktor penentu jurusan lain, data tersebut dibagi menjadi 2 label, yakni label target (y) dan label fitur (x). Data *pre-processing* ini kemudian dibersihkan hingga bisa terbaca di *Phyton* untuk diolah menggunakan *libraries* dari *Phyton* untuk *data science*, di antaranya pandas, sklearn, dan matplotlib dengan menggunakan C45 dan *Naive Bayes*.

Berikut proses olah data dengan metode C45 dan *Naive Bayes* menggunakan *Phyton*. Langkah pertama yang dilakukan adalah memuat data dengan pandas serta memasukkan *libraries* yang terkait. Setelah data terbaca, periksa kelengkapan data. Jika terdapat data yang tidak sesuai dengan jumlah, data

Journal of Artificial Intelligence and Data Engineering (JAIDE)
Vol. 1, No. 1, July 2024, page 28-34
E-ISSN: xxxx-xxxx, P-ISSN: xxx-xxxx
DOI: https://doi.org/xx.xxxxx/jaide.xxxx.x
https://journal.beta-academia.com/index.php/jaide

akan dibersihkan.

```
In [10]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 321 entries, 0 to 320
Data columns (total 10 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   JENIS KELAMIN         320 non-null    object
 1   SALAH JURUSAN         320 non-null    object
 2   ASAL SEKOLAH          318 non-null    object
 3   MINAT                 320 non-null    object
 4   PENGARUH TEMAN        320 non-null    object
 5   SARAN ORANG TUA       320 non-null    object
 6   NILAI MATEMATIKA      320 non-null    float64
 7   NILAI BAHASA INGGRIS  320 non-null    float64
 8   NILAI IPA             320 non-null    float64
 9   JURUSAN SAAT INI      320 non-null    object
dtypes: float64(3), object(7)
memory usage: 25.2+ KB
```

Gambar 1. Memeriksa kelengkapan data

Kemudian, melakukan eliminasi pada kolom dataset yang tidak digunakan untuk membersihkan data.

```
In [12]: df=df.dropna()
         df
```

Out[12]:

	JENIS KELAMIN	SALAH JURUSAN	ASAL SEKOLAH	MINAT	PENGARUH TEMAN	SARAN ORANG TUA	NILAI MATEMATIKA	NILAI BAHASA INGGRIS	NILAI IPA	JURUSAN SAAT INI
0	L	Tidak	SMP	Sistem Audio Vidio	Tidak Berpengaruh	Cukup Menyarankan	95.0	76.0	91.0	TAV
1	L	Ya	SMP	Programming	Berpengaruh	Tidak Menyarankan	95.0	90.0	90.0	TAV
2	L	Ya	SMP	Jaringan	Berpengaruh	Cukup Menyarankan	84.0	77.0	81.0	TAV
3	L	Tidak	MTs	Listrik dan Elektronika	Tidak Berpengaruh	Cukup Menyarankan	94.0	88.0	95.0	TAV
4	L	Ya	SMP	Jaringan	Berpengaruh	Cukup Menyarankan	76.0	87.0	92.0	TAV
...	...	...	...	...	...	...	...	...	...	...
316	L	Tidak	MTs	Otomotif	Cukup Berpengaruh	Menyarankan	88.0	96.0	79.0	TBSM
317	L	Tidak	SMP	Otomotif	Tidak Berpengaruh	Menyarankan	93.0	86.0	86.0	TBSM
318	L	Ya	SMP	Listrik dan Elektronika	Cukup Berpengaruh	Tidak Menyarankan	78.0	83.0	82.0	TBSM
319	L	Ya	MTs	Komunikasi Digital	Berpengaruh	Cukup Menyarankan	76.0	88.0	93.0	TBSM
320	L	Ya	SMP	Sistem Audio	Berpengaruh	Cukup	76.0	93.0	87.0	TBSM

Gambar 2. Mengeliminasi kolom pada dataset

Setelah semua data sudah bersih, proses dilanjutkan dengan membagi dataset menjadi data training dan data testing dengan perbandingan 70:30 di kedua metode untuk mendapatkan tingkat akurasi. Berikut pembagian data training dan testing yang bisa dilakukan:

```
In [33]: #split data

         from sklearn.model_selection import train_test_split
         x_train, x_test, y_train, y_test=train_test_split(x, y, test_size=0.3, random_state=0)

In [34]: x_train.shape
Out[34]: (222, 9)

In [35]: x_test.shape
Out[35]: (96, 9)
```

Gambar 3. Membagi data training dan data testing

Setelah pembagian data training dan data testing selesai, selanjutnya adalah mengimport sklearn, library *Decision Tree* dan *Naive Bayes*, serta memasukkan y_prediksi seperti gambar berikut:

Journal of Artificial Intelligence and Data Engineering (JAIDE)
Vol. 1, No. 1, July 2024, page 28-34
E-ISSN: xxxx-xxxx, P-ISSN: xxx-xxxx
DOI: https://doi.org/xx.xxxxx/jaide.xxxx.x
https://journal.beta-academia.com/index.php/jaide

```
In [36]: # Import Decision Tree Classifier

         from sklearn.tree import DecisionTreeClassifier

         pohon = DecisionTreeClassifier()
         pohon.fit(x_train, y_train)

Out[36]: DecisionTreeClassifier()

In [37]: #prediksi

         y_prediksi = pohon.predict(x_test)

         y_prediksi

Out[37]: array(['TBSM', 'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TKJ', 'TBSM', 'TBSM',
                'TBSM', 'TBSM', 'TKJ', 'TKJ', 'TKJ', 'TBSM', 'TBSM', 'TBSM', 'TKJ',
                'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TBSM',
                'TKJ', 'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TKJ',
                'TKJ', 'TBSM', 'TBSM', 'TKJ', 'TKJ', 'TKJ', 'TBSM', 'TBSM', 'TAV',
                'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TAV', 'TKJ', 'TKJ', 'TBSM', 'TBSM',
                'TBSM', 'TKJ', 'TKJ', 'TBSM', 'TKJ', 'TKJ', 'TBSM', 'TAV', 'TBSM',
                'TAV', 'TBSM', 'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TBSM', 'TAV', 'TKJ',
                'TKJ', 'TKJ', 'TBSM', 'TBSM', 'TBSM', 'TKJ', 'TKJ', 'TKJ', 'TBSM',
                'TBSM', 'TBSM', 'TBSM', 'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TBSM',
                'TKJ', 'TBSM', 'TBSM', 'TKJ', 'TBSM', 'TBSM', 'TKJ', 'TKJ', 'TKJ'],
               dtype=object)
```

Gambar 4. Decision Tree Clasifier

```
In [15]: #import naive bayes training model

         from sklearn.naive_bayes import GaussianNB

         klasifikasi = GaussianNB()
         klasifikasi.fit(x_train, y_train)

Out[15]: GaussianNB()

In [16]: #Testing

         y_prediksi = klasifikasi.predict(x_test)
         y_prediksi

Out[16]: array(['TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TKJ', 'TKJ', 'TAV', 'TAV',
                'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TKJ',
                'TAV', 'TKJ', 'TKJ', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV',
                'TKJ', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TKJ', 'TAV', 'TAV',
                'TKJ', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV',
                'TKJ', 'TAV', 'TAV', 'TAV', 'TKJ', 'TAV', 'TAV', 'TAV', 'TAV',
                'TAV', 'TAV', 'TKJ', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV', 'TAV',
```

Gambar 5. *Naive Bayes*

Kemudian untuk membandingkan kedua metode, proses dilanjutkan dengan menghitung akurasi tiap-tiap metode terhadap data yang telah tersedia mana metode yang paling tepat penggunaannya. Berikut proses perhitungan akurasi dari setiap metode:

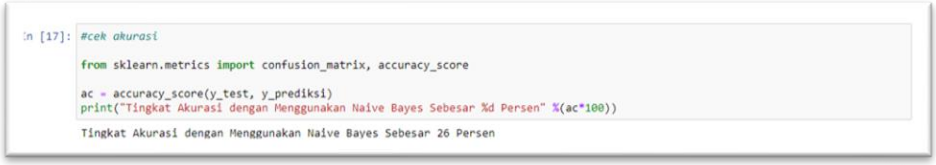
```
n [38]: #cek akurasi

        from sklearn.metrics import confusion_matrix, accuracy_score

        ac = accuracy_score(y_test, y_prediksi)
        print("Tingkat Akurasi dengan Menggunakan C45 Sebesar %d Persen" %(ac*100))

        Tingkat Akurasi dengan Menggunakan C45 Sebesar 85 Persen
```

Gambar 6. Akurasi Metode C45



```
in [17]: #cek akurasi

         from sklearn.metrics import confusion_matrix, accuracy_score

         ac = accuracy_score(y_test, y_prediksi)
         print("Tingkat Akurasi dengan Menggunakan Naive Bayes Sebesar %d Persen" %(ac*100))

         Tingkat Akurasi dengan Menggunakan Naive Bayes Sebesar 26 Persen
```

Gambar 7. Akurasi Metode *Naive Bayes*

Journal of Artificial Intelligence and Data Engineering (JAIDE)
Vol. 1, No. 1, July 2024, page 28-34
E-ISSN: xxxx-xxxx, P-ISSN: xxx-xxxx
DOI: https://doi.org/xx.xxxxx/jaide.xxxx.x
https://journal.beta-academia.com/index.php/jaide

KESIMPULAN

Kedua algoritma yang dibandingkan dalam kasus ini adalah C45 dan *Naive Bayes*. Dari hasil yang diberikan, dapat diketahui bahwa algoritma C45 memiliki akurasi sebesar 85% dan algoritma *Naive Bayes* memiliki akurasi sebesar 26%. Hal ini menunjukkan bahwa algoritma C45 lebih efektif dalam mengklasifikasikan dataset yang tersedia dibandingkan dengan algoritma *Naive Bayes*.

Akurasi merupakan salah satu ukuran yang digunakan untuk mengukur kemampuan suatu algoritma dalam mengklasifikasikan data. Semakin tinggi akurasi suatu algoritma, maka semakin baik pula kemampuannya dalam mengklasifikasikan data. Dengan demikian, dapat disimpulkan bahwa algoritma C45 memiliki kemampuan yang lebih baik dalam mengklasifikasikan data dibandingkan dengan algoritma *Naive Bayes*.

Oleh karena itu, jika membutuhkan suatu algoritma yang dapat mengklasifikasikan data dengan akurasi yang tinggi, maka algoritma C45 merupakan pilihan yang tepat. Namun, perlu diingat bahwa keputusan terkait pemilihan algoritma selalu tergantung pada kondisi dan tujuan yang ingin dicapai. Sebagai contoh, jika memiliki dataset yang relatif kecil, maka algoritma *Naive Bayes* mungkin lebih cocok untuk digunakan karena memiliki kebutuhan komputasi yang lebih rendah dibandingkan dengan algoritma C45. Namun, jika dataset yang akan diklasifikasikan cukup besar, maka algoritma C45 merupakan pilihan yang lebih tepat karena memiliki akurasi yang lebih tinggi.

DAFTAR PUSTAKA

[1] Rusdiansyah, "Analisis Keputusan Menentukan Jurusan Pada Sekolah Menengah dengan Netode Simple Additive Weighting," *J. Techno Nusa Mandiri*, vol. XIV, no. 1, pp. 49–56, 2017.

[2] Julizal, Lukman, and I. Sunoto, "Sistem Pendukung Keputusan Pemilihan Jurusan Smk Adi Luhur," *Julizal al / Sist. Pendukung Keputusan Pemilihan*, vol. 2, no. 1, pp. 366–375, 2021.

[3] M. Rahmayu and R. K. Serli, "Sistem Pendukung Keputusan Pemilihan Jurusan Pada Smk Putra Nusantara Jakarta Menggunakan Metode Analytical Hierarchy Process (AHP)," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 9, no. 1, pp. 551–564, 2019, [Online]. Available: https://jurnal.umk.ac.id/index.php/simet/article/view/2022

[4] V. Anestiviya, A. Ferico, and O. Pasaribu, "Analisis Pola Menggunakan Metode C4.5 Untuk Peminatan Jurusan Siswa Berdasarkan Kurikulum (Studi Kasus : Sman 1 Natar)," *J. Teknol. dan Sist. Inf.*, vol. 2, no. 1, pp. 80–85, 2021, [Online]. Available: http://jim.teknokrat.ac.id/index.php/JTSI

[5] M. F. Rifai, H. Jatnika, and B. Valentino, "Penerapan Algoritma Naïve Bayes Pada Sistem Prediksi Tingkat Kelulusan Peserta Sertifikasi Microsoft Office Specialist (MOS)," *Petir*, vol. 12, no. 2, pp. 131–144, 2019, doi: 10.33322/petir.v12i2.471.

[6] P. B. N. Setio, D. R. S. Saputro, and Bowo Winarno, "Klasifikasi Dengan Pohon Keputusan Berbasis Algoritme C4.5," *Prism. Pros. Semin. Nas. Mat.*, vol. 3, pp. 64–71, 2020.

[7] M. W. Prihatmono and A. F. Watratan, "Implementasi Algoritma C4.5 Menggunakan Python Untuk Klasifikasi Kepuasan Konsumen," *Progres*, pp. 49–55, 2019, [Online]. Available: https://jurnal.stmikprofesional.ac.id/index.php/Progress/article/view/146/22

[8] F. Ekawati, F. T. Informasi, U. Islam, K. Muhammad, A. Al, and B. Banjarmasin, "ALGORITMA NAÏVE BAYES UNTUK PENENTUAN JURUSAN," vol. 9, no. 1, 2018.

[9] S. Syarli and A. A. Muin, "Metode *Naive Bayes* Untuk Prediksi Kelulusan (Studi Kasus : Data Mahasiswa Baru Perguruan Tinggi)," vol. 2, no. 1, pp. 22–26, 2016.

[10] R. A. Anggraini, G. Widagdo, A. S. Budi, and M. Qomaruddin, "Penerapan Data Mining Classification untuk Data Blogger Menggunakan Metode Naïve Bayes," *J. Sist. dan Teknol. Inf.*, vol. 7, no. 1, p. 47, 2019, doi: 10.26418/justin.v7i1.30211.

[11] I Komang Setia Buana, "Implementasi Aplikasi Speech to Text untuk Memudahkan Wartawan Mencatat Wawancara dengan Python," *J. Sist. dan Inform.*, vol. 14, no. 2, pp. 135–142, 2020, doi: 10.30864/jsi.v14i2.293.

Journal of Artificial Intelligence and Data Engineering (JAIDE)
Vol. 1, No. 1, July 2024, page 28-34
E-ISSN: xxxx-xxxx, P-ISSN: xxx-xxxx
DOI: https://doi.org/xx.xxxxx/jaide.xxxx.x
https://journal.beta-academia.com/index.php/jaide

[12] F. S. Pamungkas, B. D. Prasetya, and I. Kharisudin, "Perbandingan Metode Klasifikasi Supervised Learning pada Data Bank Customers Menggunakan Python," *Prism. Pros. Semin. Nas. Mat.*, vol. 3, pp. 692–697, 2020, [Online]. Available: https://journal.unnes.ac.id/sju/index.php/prisma/article/view/37875