

Laporan Skripsi Raphael KH Turntitin Januari 2024

by turnitin turnitin

Submission date: 09-Jan-2024 02:28PM (UTC+0700)

Submission ID: 2268282250

File name: Laporan_Skripsi_Raphael_KH_Turntitin_Januari_2024-1.docx (604.76K)

Word count: 10434

Character count: 62132

PENDAHULUAN

1.1 Latar Belakang

Edukasi adalah faktor penting perkembangan bangsa dan untuk mencapai tingkat edukasi yang baik, maka tingkat kelulusan siswa juga harus tinggi [1][2]. Masih banyak siswa yang belum lulus dikarenakan faktor-faktor tertentu, misalnya karena mengalami kesulitan pada suatu mata pelajaran tertentu [3][4]. Maka dari itu dibutuhkan cara untuk mengatasi isu ini.

Machine learning merupakan sebuah ilmu komputer yang dapat membuat suatu sistem komputer yang melakukan pembelajaran dari data [5][6]. Proses ini menggunakan metode matematika dan statistika. *Machine learning* merupakan bagian dari teknologi *artificial intelligence*. *Machine learning* dapat digunakan untuk menganalisa faktor-faktor yang memengaruhi performa siswa, sehingga dengan mengetahui faktor-faktor penting tersebut sebuah institusi pendidikan dapat meningkatkan performa siswa dengan memperbaiki faktor-faktor penting yang sudah dianalisa dengan sistem *machine learning* [7]. *Machine learning* memiliki berbagai macam metode yang dapat digunakan, diantaranya seperti ⁴² *Decision Tree* (DT), *Naive Bayes* (NB), *Support Vector Machine* (SVM), dan *K-Nearest Neighbour* (KNN) [8][9]. Beberapa studi tentang prediksi performa mahasiswa sudah berhasil menggunakan *machine learning* pada penelitian mereka [10][11][12][13][14][15] dan seperti penelitian-penelitian sebelumnya, penelitian ini menggunakan *machine learning* untuk memprediksi performa siswa.

Penelitian ini difokuskan pada perbandingan algoritma *machine learning* dalam membuat prediksi performa siswa. Hal ini karena pada mata pelajaran tertentu ⁷⁷ yang ada di sekolah, masih banyak siswa yang belum lulus. Berdasarkan alasan itu maka perlu dibuat penelitian tentang prediksi performa siswa dengan menggunakan algoritma DT, KNN, SVM, dan NB. Secara khusus sistem prediksi dibuat dengan sistem *binary classification* (*passed* or *failed*). Hasil dari keempat algoritma ⁶³ ini kemudian dibandingkan satu sama lain untuk menentukan algoritma yang memiliki akurasi prediksi terbaik. Ada beberapa penelitian sebelumnya yang berfokus pada prediksi performa siswa. Penelitian pertama

melakukan prediksi performa siswa pada SMAN 3 Ambon dengan menggunakan 3 teknik *machine learning*, yaitu KNN, DT, dan NB. Hasil penelitian ini menunjukkan bahwa DT memiliki akurasi tertinggi dengan nilai sebesar 99,6% [9]. Hasil ini lebih baik dibanding KNN dan NB. Penelitian lainya melakukan penelitian untuk membandingkan performa DT dan NB. Penelitian ini menggunakan *machine learning* untuk untuk membandingkan kecepatan dan akurasi dari sistem klasifikasi untuk menentukan penerima beasiswa. Hasil yang didapat dari penelitian menunjukkan bahwa DT memiliki performa lebih baik dengan akurasi sebesar 96,40%, yang dimana hasil tersebut lebih baik dari NB yang mendapatkan akurasi sebesar 95,11% [10].

Penelitian dari Universitas AMIKOM Yogyakarta memiliki tujuan untuk memprediksi tingkat kelulusan mahasiswa Universitas AMIKOM Yogyakarta dengan menggunakan DT. Secara khusus, penelitian ini membandingkan tiga algoritma DT (C4.5, ID3, dan CART). Hasil yang didapat pada penelitian ini menunjukkan bahwa DT C4.5 memiliki performa terbaik dengan nilai akurasi sebesar 76,10% dengan pembagian data yang digunakan adalah 70:30, yaitu 70% untuk *training data* dan 30% untuk *testing data* [11]. Mirip dengan penelitian sebelumnya, penelitian berikutnya membuat penelitian untuk memprediksi performa mahasiswa pada Universitas Duhok, provinsi Duhok, daerah Kurdistan, Iraq. Penelitian ini menggunakan lima algoritma yaitu DT, SVM, KNN, NB, dan *Random Forest* (RF). Hasil yang didapatkan menunjukkan bahwa DT memiliki performa yang paling baik [12].

Berdasarkan penelitian-penelitian sebelumnya, penelitian ini dilakukan untuk memprediksi tingkat kelulusan siswa di SMKN 4 Kota Tangerang pada mata pelajaran Matematika, Bahasa Inggris dan PKn dengan menggunakan algoritma DT, KNN, SVM dan NB. Mata pelajaran ini ini dipilih karena mata pelajaran ini merupakan mata pelajaran yang penting pada jenjang pendidikan SMK, dengan tujuan untuk mencari algoritma yang memiliki tingkat akurasi prediksi yang tertinggi. Penelitian ini menggunakan lebih banyak variabel dibandingkan penelitian-penelitian sebelumnya dan juga penelitian ini dilakukan di SMK sehingga penelitian ini memiliki perbedaan dengan penelitian-penelitian sebelumnya.

8 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka terdapat permasalahan yang akan dikerjakan pada penelitian ini, yaitu:

1. Bagaimana cara memprediksi performa siswa menggunakan teknik *machine learning*?
2. Apa algoritma terbaik yang bisa digunakan untuk membuat prediksi performa siswa?

52 1.3 Tujuan Penelitian

Tujuan dari penelitian yang hendak dicapai, yaitu:

1. Menerapkan *machine learning* untuk sistem prediksi performa siswa.
2. Mengevaluasi algoritma *machine learning* terbaik yang bisa digunakan untuk membuat prediksi performa siswa.

1.4 Manfaat Penelitian

Adapun beberapa manfaat yang ingin dicapai dalam penelitian berupa:

1. Penelitian ini diharapkan dapat memperoleh bahan informasi yang dapat diberikan kepada staff kependidikan untuk meningkatkan performa siswa.
2. Sebagai acuan untuk penelitian selanjutnya terkait algoritma *machine learning* mana yang dapat digunakan untuk membuat prediksi performa siswa.

1.5 Batasan Masalah

Berdasarkan tujuan penelitian yang telah dipaparkan sebelumnya, maka ruang lingkup penelitian ini dibatasi sebagai berikut.

1. *Dataset* yang digunakan didapatkan dari *private dataset* yang kemudian dibuat sebagai model untuk sistem prediksi performa siswa.
2. Hasil keluaran yang didapat berupa nilai hasil prediksi performa siswa dengan menggunakan 4 algoritma *machine learning* (DT, KNN, SVM, dan NB).
3. Mata pelajaran yang dijadikan target prediksi adalah Matematika, Bahasa Inggris, dan PKn.

75 BAB II TINJAUAN PUSTAKA

2.1 *Decision Tree*

Decision Tree (DT) adalah sebuah algoritma dengan skema pohon keputusan. DT adalah sebuah alat pendukung yang menggunakan model pohon keputusan dan kemungkinan-kemungkinan hasilnya. DT sudah sering digunakan untuk sistem klasifikasi dan prediksi. Metode DT ini mengubah suatu data yang sangat besar menjadi sebuah pohon yang dapat merepresentasikan aturan yang dihasilkan. DT sangat berguna untuk mengeksplorasi data, dan menemukan suatu hubungan tersembunyi antara sejumlah variabel dengan variabel target [14][15].

Classification and Regression Trees (CART) merupakan salah satu jenis dari algoritma DT. Algoritma CART adalah jenis algoritma pohon keputusan yang dapat digunakan untuk klasifikasi dan regresi. Algoritma CART bekerja dengan cara membagi data menjadi subset-subset yang berdasarkan nilai dari sebuah atribut dalam satu waktu. Tujuan dari pembagian subset-subset ini adalah untuk memaksimalkan homogenitas dari setiap subset-subset hasil dari pembagian data. Proses yang dilakukan ini akan terus diulang sampai kriteria yang diinginkan terpenuhi, seperti sudah mencapai batas kedalaman dari pohon keputusan atau mencapai jumlah minimum dari jumlah sampel yang ada pada setiap *leaf node* [16].

Algoritma CART saat digunakan untuk sistem klasifikasi menentukan setiap kelas dari *leaf node* berdasarkan mayoritas data dari *training data* yang ada pada *node* itu, sedangkan saat Algoritma CART digunakan untuk sistem regresi maka CART membuat kalkulasi nilai *mean* dari target atribut untuk *training data* pada setiap *leaf node*. Algoritma CART ini memiliki beberapa kelebihan. Kelebihan itu adalah mudah dimengerti manusia, kesederhanaan sistemnya, dan kemampuan untuk menangani data atribut dalam jenis atribut kategori dan atribut *continuous*. Algoritma CART rawan terhadap *overfitting* jika pohon keputusannya terlalu dalam dan data yang ada terkontaminasi dengan *noise*. *Overfitting* dapat diselesaikan dengan beberapa cara. Cara itu adalah *pruning*, *regularization*, dan

ensembling. *Pruning* adalah memotong *leaf node* pada pohon keputusan yang tidak meningkatkan performa akurasi pada *validation set*. *Regularization* adalah memberikan sistem *penalty* ke *splitting criterion* untuk membuat pohon keputusan menjadi lebih simpel. *Ensembling* adalah menggabungkan beberapa pohon keputusan, seperti dengan menggunakan teknik *bagging* atau *boosting* untuk meningkatkan akurasi dan ketahanan dari sebuah model.

2.2 Naive Bayes

Naive Bayes (NB) adalah sebuah algoritma SL. NB adalah sebuah algoritma yang simpel tetapi sangat efektif. NB akan mengklasifikasikan suatu objek berdasarkan teori *Naive Bayes*. Algoritma NB akan mengasumsikan bahwa suatu atribut pada suatu kelas tidak berhubungan dengan atribut lainnya. NB menggunakan teori probabilitas yang berfokus pada nilai parameter dari suatu kelas [16]. Sistem klasifikasi NB bekerja dengan cara mengklasifikasikan data baru sebagai nilai dari sebuah target label A, jika nilai target label B diketahui [17]. Data baru itu kemudian diklasifikasikan ke kelas yang memiliki probabilitas data yang lebih besar. Teori dari NB dapat dituliskan dalam bentuk persamaan seperti Persamaan (2.1) dari teori NB, yaitu:

Teori NB

$$A|B = \frac{P(B|A) * P(A)}{P(B)} \quad (2.1)$$

Keterangan pada Persamaan (2.1) yaitu, $A|B$ = peluang kejadian A jika kejadian B diketahui, $P(B|A)$ = peluang kejadian B jika A diketahui, $P(A)$ = peluang kejadian A saat B tidak diketahui, $P(B)$ = peluang kejadian B saat A tidak diketahui. Hasil dari $A|B$ bisa didapatkan dengan mengalikan $P(B|A)$ dengan $P(A)$ yang dibagi dengan $P(B)$.

2.3 Support Vector Machine

Support Vector Machine (SVM) merupakan teknik klasifikasi yang bisa digunakan dalam *machine learning*. Cara kerja SVM adalah dengan memisahkan data menjadi kelas-kelas yang berbeda dengan menggunakan *plane* atau *hyperplane*. SVM bekerja dengan mencari garis yang dapat memisahkan kelas-

kelas yang ada, dengan memaksimalkan margin atau jarak antara garis atau bidang dan titik-titik yang terdekat dari masing-masing kelas [18]. Teori dari SVM dapat dituliskan dalam bentuk persamaan seperti Persamaan (2.2) dari teori SVM, yaitu:

Teori SVM

$$y = \text{sign}(w * x + b) \quad (2.2)$$

Keterangan pada Persamaan (2.2) yaitu, y = adalah kelas prediksi yang diketahui, w = adalah vektor bobot, x = adalah vektor fitur dari data yang ingin diprediksi, b = b adalah bias, $\text{sign} = \text{sign}$ adalah fungsi signum, yaitu fungsi yang akan menghasilkan argumen positif atau 0, dan nilai -1 jika hasilnya negatif. Perhitungan w dikalikan x ditambah b kemudian dikalikan dengan sign perlu dilakukan untuk melakukan klasifikasi data baru.

2.4 *K-Nearest Neighbour (KNN)*

KNN adalah sebuah algoritma *supervised learning non-parametric* yang dapat digunakan untuk tugas klasifikasi dan regresi. Sistem klasifikasi KNN ini bekerja dengan cara mengelompokan data baru ke kelas terbanyak pada k tetangga terdekat (k adalah *integer* positif). Jika $k=2$, maka data baru akan diklasifikasikan ke kelas dengan dua tetangga terdekatnya [19]. Tahapan dalam KNN dapat dilakukan dengan langkah berikut:

1. Melakukan perhitungan setiap data pada *training* data dan *testing* data pada jarak *Euclidean* masing-masing data.
2. Melakukan pengurutan jarak terdekat dari hasil perhitungan setiap data.
3. Menentukan k tetangga terdekat.
4. Menentukan hasil klasifikasi dari mayoritas k tetangga terdekat.

2.5 **Python**

Python adalah bahasa pemrograman yang dibuat pertama kalinya oleh Guido van Rossum yang saat itu seorang peneliti di Universitas Amsterdam, Belanda. Python dirilis ke publik untuk yang pertama pada awal dekade 1990. Lisensi Python yang *open source* dipegang oleh Python Software Foundation (PSF), lembaga yang didirikan oleh Guido dan rekan-rekannya di awal 2000-an.

Pembuatan *machine learning* tentunya memerlukan bahasa pemrograman yang mendukung proses *machine learning*. Salah satu Bahasa pemrograman yang mendukung hal ini adalah Python. Python merupakan bahasa pemrograman berorientasi objek yang dapat digunakan dalam banyak pengembangan perangkat lunak terutama dalam bidang *data science*. Python lebih populer digunakan karena didukung oleh beragam *open library* yang berkaitan dengan penelitian di bidang kecerdasan buatan [7].

A. *Artificial Intelligence*

Artificial intelligence atau yang biasa dikenal sebagai AI dan juga kecerdasan buatan adalah jalan untuk membuat mesin agar dapat berpikir dan memiliki kemampuan intelegensi. Mesin fisik AI menggunakan algoritma tertentu agar dapat memahami situasi dan kondisi sebagaimana yang manusia pikirkan [20].

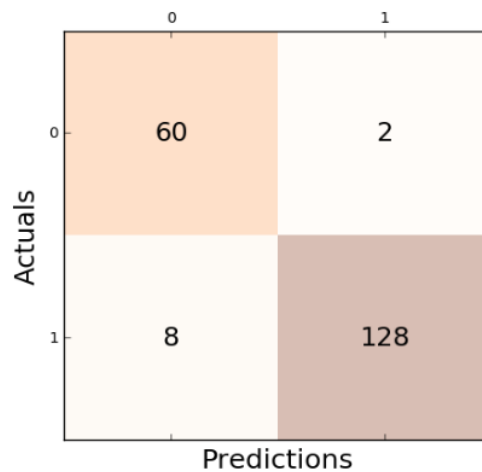
B. *Machine Learning*

Proses pembelajaran mesin atau yang biasa dikenal sebagai *machine learning* merupakan sebuah metode untuk membuat mesin dapat berpikir layaknya manusia. *Machine learning* merupakan cabang dari ilmu kecerdasan buatan atau *artificial intelligence*. Pembelajaran pada *machine learning* ini dimaksudkan agar mesin dapat menyelesaikan masalah. Masalah yang dapat diselesaikan oleh *machine learning* biasanya bersifat spesifik agar dapat memberikan jaminan solusi terbaik dalam penyelesaiannya [21]. *Machine learning* model sendiri merupakan sebuah algoritma yang telah dilakukan siklus pelatihan pada data tertentu, sehingga telah siap untuk dilakukan uji data untuk mengetahui keakuratan dari algoritma yang telah dibangun.

2.6 *Confusion Matrix*

Confusion matrix dalam *machine learning* dikenal juga dengan nama *error matrix*. *Confusion matrix* adalah sebuah susunan tabel yang secara spesifik memvisualisasikan performa dari suatu algoritma, yang biasanya algoritma ini merupakan algoritma *Supervised Learning* (SL). *Confusion matrix* dalam

unsupervised learning disebut dengan nama *matching matrix*. Setiap baris pada *confusion matrix* ini merepresentasikan sebuah data dalam kelas yang sebenarnya, sedangkan kolom pada *confusion matrix* merepresentasikan kelas hasil prediksi dari algoritma yang digunakan (bisa juga sebaliknya yaitu setiap baris merepresentasikan kelas prediksinya dan setiap kolom merepresentasikan kelas yang sebenarnya). Nama dari *confusion matrix* ini diambil karena *confusion matrix* ini memudahkan untuk melihat apakah sistem yang dibuat ini salah atau benar berdasarkan data asli dan data hasil prediksi yang ada [22]. Gambar dari *confusion matrix* dapat dilihat pada Gambar 2.1 di bawah ini.



Gambar 2.1 Confusion Matrix

Berdasarkan Gambar 2.1 di atas, dapat dilihat bahwa *confusion matrix* di atas ini menggunakan pengaturan baris pada *confusion matrix* untuk data yang sebenarnya, dan kolom pada *confusion matrix* ini untuk menampilkan data hasil prediksi dari algoritma yang digunakan.

Evaluasi pada algoritma sistem klasifikasi *machine learning* merupakan cara untuk melihat sebaik apa performa suatu sistem model klasifikasi yang sudah dibuat. Tujuan dari evaluasi sistem ini adalah untuk melihat sebaik apa performa suatu model klasifikasi dalam memprediksi kelas data yang benar. Ada beberapa metode evaluasi yang umum digunakan untuk mengevaluasi model *machine learning*, yaitu *accuracy*,

precision, *recall*, *confusion matrix* dan *F1score*. *Accuracy* adalah cara evaluasi model dengan mengukur seberapa tepat suatu model *machine learning* dalam mengklasifikasikan keseluruhan *testing* data dengan benar. persamaan dari *Accuracy* dapat dilihat pada Persamaan (2.3) di bawah ini [23].

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \quad (23)$$

Berdasarkan Persamaan (2.3) *true positive* (TP) adalah jumlah *testing* data positif yang berhasil diklasifikasikan dengan benar dan *true negative* (TN) adalah jumlah *testing* data negatif yang berhasil diklasifikasikan dengan benar. *False negative* (FN) adalah jumlah *testing* data positif yang tidak diklasifikasikan dengan benar dan *false positive* (FP) adalah jumlah *testing* data negatif yang tidak diklasifikasikan dengan benar. *Precision* dalam model evaluasi digunakan untuk mengukur seberapa presisi model dalam mengidentifikasi data yang benar-benar positif dari semua data yang diprediksi positif. *Precision* mengukur seberapa tepat data yang diprediksikan positif dari data yang aslinya benar positif. Persamaan dari *precision* dapat dilihat pada Persamaan (2.4) di bawah ini [23].

$$Precision = \frac{TP}{TP+FP} \quad (24)$$

Berdasarkan Persamaan (2.4) *precision* dapat dihitung dengan membagi nilai TP dengan nilai TP yang ditambah dengan nilai FP. Dalam kasus *imbalanced data*, *precision* bisa digunakan untuk melihat performa dari suatu model pada kelas data minoritas dan kelas mayoritas sehingga bisa diketahui bagaimana performa model *machine learning* tersebut saat memprediksi kelas minoritas dan mayoritas. *Recall* merupakan metode evaluasi yang memberikan gambaran seberapa baik suatu model klasifikasi dalam mengklasifikasikan semua kelas data yang diprediksikan positif dari semua kelas data yang benar-benar positif. Persamaan dari *recall* dapat dituliskan dalam Persamaan (2.5) di bawah ini [23].

$$Recall = \frac{TP}{TP+FN} \quad (25)$$

Berdasarkan persamaan (2.5) di atas, nilai dari *recall* dapat dicari dengan membagi nilai TP dengan nilai TP ditambah nilai FN. *Recall* bisa juga digunakan untuk mengevaluasi performa suatu model algoritma yang datanya memiliki *imbalanced data*. *F1score* adalah suatu metode evaluasi yang mengukur *harmonic* dari nilai *recall* dan *precision*. *F1score* ini juga memberikan gambaran tentang

kinerja suatu model secara keseluruhan. Persamaan dari $F1score$ dapat dilihat pada Persamaan (2.6) di bawah ini [23].

$$F1score = \frac{2 \times precision \times recall}{precision + recall} \quad (26)$$

Berdasarkan Persamaan (2.6) di atas, $F1score$ dapat dicari dengan cara dua dikalikan dengan $precision$ dan $recall$, kemudian dibagi dengan nilai dari $precision$ ditambah $recall$. $F1score$ dalam prakteknya memberikan sebuah gambaran keseimbangan antara $precision$ dan $recall$ [23].

Selain metode evaluasi yang dijelaskan di atas, masih ada metode evaluasi lainnya yaitu *matthews correlation coefficient* (MCC). MCC merupakan suatu metode evaluasi yang sangat cocok digunakan untuk evaluasi model *machine learning* terutama pada sistem *binary classification*. MCC disebut merupakan metode evaluasi performa model yang paling baik karena MCC ini hanya akan menghasilkan nilai yang tinggi jika model *machine learning* tersebut memiliki performa yang bagus dari keempat nilai pada *confusion matrix* (TP, TN, FN, FP). MCC ini disebutkan sebagai model yang sesuai dan paling baik untuk mengevaluasi performa model yang memiliki *class imbalanced*. Persamaan perhitungan dari MCC dapat dilihat pada Persamaan (2.7) di bawah ini [24].

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (27)$$

Berdasarkan Persamaan (2.7) di atas, dapat dilihat bahwa nilai MCC menggunakan semua nilai dari TP, TN, FP, dan FN. Hasil dari MCC ini juga memiliki nilai yang artinya dapat dilihat pada Tabel 2.1 di bawah ini.

Tabel 2.1 Rentang Nilai Skor MCC

Matthews Correlation Coefficient	
Nilai skor MCC	Keterangan
-1	Performa prediksi model terburuk
0	Performa prediksi model standar yang menghasilkan nilai yang acak
1	Performa prediksi model yang sempurna

Berdasarkan Tabel 2.1 di atas, dapat dilihat rentang nilai yang menggambarkan performa suatu model *machine learning* dengan menggunakan metode evaluasi MCC. Semakin mendekati 1 maka performa model itu semakin

baik. Sebaliknya semakin dekat ke angka -1 maka performa model itu semakin buruk. Nilai 0 jika didapatkan berarti performa model itu merupakan model yang tidak istimewa atau bisa disebut hanya mendapat nilai 50 dari 100 [24][25].

2.7 Kajian Pustaka

Penelitian ini berlandaskan dari penelitian-penelitian terdahulu, baik dari landasan teori, metode atau teknik penelitian yang digunakan, maupun jenis penelitiannya. Berikut ini lima penelitian yang menjadi landasan dari penelitian yang sedang dilakukan, yaitu:

Penelitian pertama, penelitian ini dilakukan di sebuah universitas di negara Turki. Algoritma yang dibandingkan pada penelitian ini adalah *random forests*, *Neural Network*, *support vector machine*, *logistic regression*, NB, dan KNN. Hasil yang didapatkan menyatakan bahwa *random forest* dan *neural network* memiliki performa terbaik dengan akurasi sebesar 74,6%. [8].

Penelitian kedua, penelitian kedua ini berfokus pada prediksi performa siswa. Penelitian ini melakukan prediksi performa siswa pada SMAN 3 Ambon dengan menggunakan tiga teknik *data mining*, yaitu KNN, DT, dan NB. Hasil penelitian ini menunjukkan bahwa DT memiliki akurasi tertinggi dengan nilai sebesar 99,6%. Hasil ini lebih baik dibanding KNN dan NB [10].

Penelitian ketiga, penelitian ketiga melakukan penelitian untuk membandingkan performa DT dan NB. Penelitian ini menggunakan *data mining* untuk membandingkan kecepatan dan akurasi dari sistem klasifikasi untuk menentukan penerima beasiswa. Hasil yang didapat dari penelitian menunjukkan bahwa DT memiliki performa lebih baik dengan akurasi sebesar 96,40%. Lebih baik dari NB yang mendapatkan akurasi sebesar 95,11% [11].

Penelitian keempat, mirip dengan penelitian ketiga, penelitian keempat membuat penelitian untuk memprediksi performa mahasiswa pada Universitas Duhok, provinsi Duhok, daerah Kurdistan, Iraq. Berdasarkan penelitian ini, digunakan lima algoritma yaitu DT, SVM, KNN, NB, dan *Random Forest* (RF). Hasil yang didapatkan menunjukkan bahwa DT memiliki performa yang paling baik [13].

Penelitian kelima, penelitian ini dilakukan di sekolah dasar negeri 4 Trimulyo sekampung. Penelitian ini membuat prediksi performa siswa dengan menggunakan algoritma DT C4.5. Hasil Penelitian yang didapatkan adalah sebuah sistem prediksi performa siswa dengan akurasi sebesar 94,43% [14].

Berdasarkan beberapa hasil penelitian di atas, dapat diketahui bahwa *machine learning* dapat digunakan untuk membuat prediksi performa siswa. Namun penelitian di atas belum ada yang melakukan prediksi performa siswa pada tingkat SMK, maka dari itu penelitian ini menggunakan *machine learning* untuk memprediksi performa siswa pada tingkat SMK.

METODOLOGI PENELITIAN**3.1 Metodologi Penelitian**

Pada bab ini dijelaskan tahap-tahap yang dilakukan dalam melaksanakan penelitian. Tahapan penelitian yang dilakukan adalah sebagai berikut:

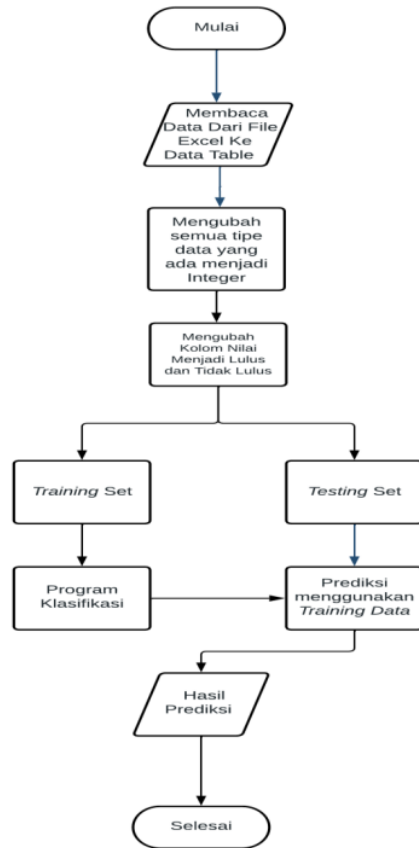
1. Melakukan studi literatur yaitu mengumpulkan referensi seperti artikel, buku, dan laporan penelitian yang berhubungan dengan permasalahan yang dibahas.
2. Melakukan pencarian *dataset* untuk dibuat penelitian performa siswa.
3. Melakukan transformasi data dari nilai angka menjadi sistem *binary classification* (*Passed* dan *Failed*).
4. Membuat program klasifikasi dengan empat algoritma *machine learning* (DT, KNN, SVM, dan NB).
5. Menarik kesimpulan dari hasil prediksi dan evaluasi performa dari setiap algoritma *machine learning* yang digunakan.

3.2 Instrumen Penelitian

Perangkat pendukung yang digunakan pada penelitian ini untuk membuat sistem klasifikasi adalah Jupyter Notebook. Jupyter Notebook merupakan sebuah aplikasi yang biasa digunakan untuk membuat program AI, *machine learning*, dan analisa data. Jupyter Notebook dapat bekerja dengan baik untuk membantu dalam mencari makna dan memberikan informasi penting yang terdapat pada data yang digunakan.

3.3 Perancangan Penelitian

Secara garis besar, perancangan penelitian untuk membuat sistem klasifikasi dibagi menjadi tiga, yaitu pembacaan data, mengubah nilai angka menjadi sistem *binary classification*, dan pembuatan sistem klasifikasi dengan *machine learning*. Diagram alir perancangan penelitian dapat dilihat pada Gambar 3.1 di bawah ini.



Gambar 3.1 Diagram Alir Penelitian

Berdasarkan Gambar 3.1 di atas, bisa dilihat tahapan penelitian yang dilakukan dari awal mulai hingga didapatkan nilai akurasi dari percobaan setiap algoritma. Tahapan penelitian seperti pada gambar di atas ini harus dilakukan dengan baik dan secara berurutan sehingga bisa didapatkan hasil prediksi setiap algoritma yang diharapkan. Perancangan model *machine learning* pada penelitian ini menggunakan tahapan sebagai berikut.

1. Melakukan pembacaan data penelitian.
2. Melakukan Transformasi data dari tipe *string* ke *integer*.
3. Melakukan pembagian *training* dan *testing* data.
4. Membuat sistem *tuning* parameter untuk mencari parameter terbaik dari masing-masing model *machine learning* (DT, KNN, SVM, dan NB).

5. Setelah didapatkan hasil parameter terbaik, selanjutnya parameter terbaik tersebut digunakan untuk membuat model *machine learning* yang dilatih dengan *training* data untuk setiap algoritma yang digunakan.
6. Hasil dari setiap model di atas kemudian digunakan untuk membuat prediksi dari *testing* data. Hasil dari model ini juga menghasilkan *confusion matrix*, *accuracy*, *precision*, *recall*, *F1score*, dan *MCC*.

3.4 Pengambilan Dataset

Dataset merupakan sekumpulan data yang berasal dari informasi-informasi di masa lalu dan disimpan dalam memori pada kondisi tidak terhubung. *Dataset* dapat ditambahkan, dihapus, atau memperbarui baris dalam memori sehingga siap untuk dikelola menjadi suatu informasi yang baru. *Dataset* terdiri dari dua jenis, yakni *private dataset* dan *public dataset*. *Private dataset* merupakan *dataset* yang diambil dari suatu organisasi untuk dijadikan sebagai objek penelitian, misalnya data rumah sakit, sekolah, bisnis dan sebagainya. Sedangkan *public dataset* merupakan *dataset* yang diambil dari *public repository* yang telah disepakati oleh pakar peneliti *data mining*.

Dataset yang digunakan dalam penelitian ini adalah *private dataset*. Data percobaan dikumpulkan dari siswa SMKN 4 Kota Tangerang. Hasilnya didapatkan satu *dataset* dengan data target yang berisi data nilai siswa SMKN 4 Kota Tangerang pada mata pelajaran Matematika, PKn, dan Bahasa Inggris.

3.5 Dataset

Dataset ini berisi data siswa yang berasal dari SMKN 4 Kota Tangerang. *Dataset* ini memiliki 32 variabel yang dapat digunakan untuk penelitian. Variabel yang digunakan pada penelitian ini merupakan variabel yang bisa digunakan untuk membuat prediksi performa siswa. Variabel dari faktor ekonomi, faktor keluarga, jenis kelamin, dan umur bisa digunakan untuk membuat prediksi performa siswa. Variabel nilai siswa pada mata pelajaran semester sebelumnya juga bisa digunakan untuk membuat prediksi performa siswa. Variabel yang digunakan ini dapat dilihat pada Tabel 3.1 di bawah ini.

Tabel 3.1 Variabel Penelitian

No	Variabel	Penjelasan
1	Sex	Jenis kelamin siswa (laki-laki atau perempuan)
2	Age	Umur siswa (numerik)
3	Medu	Tingkat edukasi ibu (1 s.d. 5)
4	Fedu	Tingkat edukasi ayah (1 s.d. 5)
5	Studytime	Waktu belajar siswa dalam satu minggu (numerik)
6	Famrel	Hubungan dalam keluarga (1-5)
7	Freetime	Waktu luang (numerik)
8	Health	Tingkat kesehatan siswa (numerik)
9	Famsize	Jumlah anggota keluarga (lebih kecil dari 3 atau lebih dari tiga)
10	Pstatus	Status tinggal dengan orang tua (ya atau tidak)
11	Mjob	Pekerjaan Ibu (numerik)
12	Fjob	Pekerjaan Ayah (numerik)
13	reason	Alasan memilih sekolah ini (nominal: dekat dengan rumah, reputasi sekolah, pembelajaran yang baik atau lainnya)
14	schoolsup	Mengikuti pembelajaran tambahan seperti bimbel di luar sekolah (ya atau tidak)
15	Traveltime	Waktu perjalanan dari rumah ke sekolah (numerik)
16	Famsud	Biaya dukungan dari keluarga (Nominal)
17	Activities	Mengikuti ekstrakurikuler (ya atau tidak)
18	Higher	Apakah ingin melanjutkan pendidikan ke jenjang yang lebih tinggi (ya atau tidak)
19	Internet	Akses internet di rumah (ya atau tidak)
20	Romantic	Memiliki hubungan dengan lawan jenis (ya atau tidak)
21	Goout	Apakah sering bermain dengan teman (ya atau tidak)
22	Sosmed	Waktu bermain sosial media (numerik)
23	Onlinegame	Waktu bermain game online (numerik)
24	Fisikas2	Nilai Fisika semester 2
26	Kimias2	Nilai Kimia semester 2
27	PKNs2	Nilai PKn semester 2
28	Binggriss2	Nilai Bahasa Inggris semester 2
29	Matematikas2	Nilai matematika semester 2
30	PKNs3 (Target data)	Nilai PKn semester 3
31	Binggriss3 (Target data)	Nilai Bahasa Inggris semester 3
32	Matematikas3 (Target data)	Nilai Matematika semester 3

Berdasarkan Tabel 3.1 di atas, digunakan 32 variabel untuk membuat sistem prediksi performa siswa. Tabel 3.1 tersebut juga menjelaskan apa isi dari setiap variabel yang digunakan lengkap dengan rentang data yang ada pada setiap variabel. Variabel yang memiliki penjelasan s.d. menandakan bahwa itu adalah rentang data dari variabel tersebut, yang menyatakan jika tertulis 1 s.d. 5 maka jika nilainya 1 itu menandakan sangat buruk dan jika nilainya adalah 5 maka itu

menandakan sangat baik. Jumlah data yang berhasil dikumpulkan pada penelitian ini ada sebanyak 231 data.

3.6 Predikat Nilai Siswa

Predikat nilai siswa merupakan sebuah nilai kategori yang menyatakan nilai prestasi atau nilai akademis siswa berdasarkan rentang skor tertentu. Penelitian ini menggunakan pembagian predikat siswa seperti pada Tabel 3.2 di bawah ini.

Tabel 3.2 Predikat Nilai Siswa

Predikat Nilai	Keterangan	Rentang Skor
A	Lulus	80 s.d.100
B	Tidak Lulus	0 s.d. 79

Berdasarkan Tabel 3.2 di atas, penelitian ini membuat sistem prediksi performa siswa dengan *machine learning* untuk memprediksi tingkat kelulusan siswa dengan predikat lulus untuk nilai diatas 80, sehingga pada hasil prediksi dengan *machine learning* akan menghasilkan 2 keluaran. Keluaran yang pertama yaitu adalah lulus (diprediksikan mendapat nilai 80 atau lebih), dan yang kedua adalah tidak lulus (diprediksikan mendapat nilai antara 0 sampai 79). Batas nilai pembagian dari nilai 80 dipilih karena nilai 80 merupakan standar yang ingin dicapai oleh guru pada lembaga pendidikan di SMKN 4 Kota Tangerang.

3.7 Training Data dan Testing Data

Tujuan dari penelitian ini adalah untuk mencari model dengan performa prediksi terbaik. Penelitian ini menggunakan satu jenis pembagian data, yaitu sebesar 60% untuk *training* data dan 40% untuk *testing* data. Pembagian data untuk mata pelajaran Matematika dapat dilihat pada Tabel 3.3 di bawah ini.

Tabel 3.3 *Binary Classification Training and Testing Data* (Matematika)

<i>Binary Classification</i> (Matematika)	
Nilai	Total Data
Lulus	73
Tidak Lulus	158
Total Data	231

¹⁰ Berdasarkan Tabel 3.3 di atas, dapat dilihat bahwa total data yang berhasil dikumpulkan ada sebanyak 231 data. Berdasarkan data yang berhasil dikumpulkan tersebut, sebanyak 73 siswa dinyatakan lulus, sedangkan sisanya yang tidak lulus ada sebanyak 158 data. Hasil dari data tersebut kemudian dibagi untuk *training data* dan *testing data*. Hasil pembagian untuk *training data* dengan pembagian sebesar 60% dapat dilihat pada Tabel 3.4 di bawah ini.

Tabel 3.4 Matematika *Binary Classification Training Data* (60%)

<i>Binary Classification Training Data</i> (Matematika)	
Nilai	Total Data
Lulus	41
Tidak Lulus	97
Total Data	138

²⁰ Berdasarkan Tabel 3.4 di atas, dapat dilihat bahwa pada *training data* dengan pembagian sebesar 60% terdapat 138 data. Data untuk siswa yang lulus ada sebanyak 41 data, sedangkan untuk siswa tidak lulus ada sebanyak 97 data. Hasil pembagian data untuk *testing data* dengan pembagian sebesar 40% dapat dilihat pada Tabel 3.5 di bawah ini.

Tabel 3.5 Matematika *Binary Classification Testing Data* (40%)

<i>Binary Classification Testing Data</i> (Matematika)	
Nilai	Total Data
Lulus	32
Tidak Lulus	61
Total Data	93

²⁰ Berdasarkan Tabel 3.5 di atas, dapat dilihat bahwa pada *testing data* sebesar 40% terdapat total 93 data. Terdapat juga 32 data untuk siswa yang lulus dan ada 61 data untuk siswa yang tidak lulus. Hasil pembagian pada mata pelajaran Bahasa Inggris dapat dilihat pada Tabel 3.6 di bawah ini.

Tabel 3.6 Bahasa Inggris *Binary Classification Training and Testing Data*.

<i>Binary Classification</i> (Bahasa Inggris)	
Nilai	Total Data
Lulus	48
Tidak Lulus	183
Total Data	231

¹⁰ Berdasarkan Tabel 3.6 di atas, dapat dilihat bahwa total data yang berhasil dikumpulkan untuk mata pelajaran bahasa Inggris ada sebanyak 231 data. Dari data yang berhasil dikumpulkan tersebut, sebanyak 48 siswa lulus, sedangkan sisanya adalah siswa yang tidak lulus ada sebanyak 183 data. Hasil dari data tersebut kemudian akan dibagi untuk *training* data dan *testing* data. Hasil pembagian untuk *training* data dengan pembagian 60% ³⁹ ini dapat dilihat pada Tabel 3.7 di bawah ini.

Tabel 3.7 Bahasa Inggris *Binary Classification Training Data* (60%).

<i>Binary Classification Training Data</i> (Bahasa Inggris)	
Nilai	Total Data
Lulus	27
Tidak Lulus	111
Total Data	138

¹⁸ Berdasarkan Tabel 3.7 di atas, dapat dilihat bahwa pada *training* data dengan pembagian sebesar 60% terdapat 138 data. Data untuk siswa yang lulus dengan ada sebanyak 27 data, sedangkan untuk siswa yang tidak lulus ada sebanyak 111 data. Hasil pembagian data untuk *testing* data dengan pembagian sebesar 40% ²³ dapat dilihat pada Tabel 3.8 di bawah ini.

Tabel 3.8 Bahasa Inggris *Binary Classification Testing Data* (40%).

<i>Binary Classification Testing Data</i> (Bahasa Inggris)	
Nilai	Total Data
Lulus	21
Tidak Lulus	72
Total Data	93

²⁰ Berdasarkan Tabel 3.8 di atas, dapat dilihat bahwa pada *testing* data sebesar 40% terdapat total 93 data. Terdapat juga 21 data untuk siswa yang lulus dan ada 72 data untuk siswa yang tidak lulus. Hasil pembagian pada mata pelajaran PKn dapat dilihat pada Tabel 3.9 di bawah ini.

Tabel 3.9 PKn *Binary Classification Training and Testing Data*.

<i>Binary Classification (PKn)</i>	
Nilai	Total Data
Lulus	166
Tidak Lulus	65
Total Data	231

¹⁰ Berdasarkan Tabel 3.9 di atas, dapat dilihat bahwa total data yang berhasil dikumpulkan untuk mata pelajaran PKn ada sebanyak 231 data. Dari data yang berhasil dikumpulkan tersebut, sebanyak 166 siswa lulus, sedangkan sisanya adalah siswa yang tidak lulus ada sebanyak 65 data. Hasil dari ²⁹ data tersebut kemudian dibagi untuk *training data* dan *testing data*. Hasil pembagian untuk *training data* dengan pembagian 60% ini dapat dilihat pada Tabel 3.10 di bawah ini.

Tabel 3.10 PKn *Binary Classification Training Data (60%)*.

<i>Binary Classification Training Data (PKn)</i>	
Nilai	Total Data
Lulus	100
Tidak Lulus	38
Total Data	138

⁷ Berdasarkan Tabel 3.10 di atas, dapat dilihat bahwa pada *training data* dengan pembagian sebesar 60% terdapat 138 data. Data untuk siswa yang lulus ada sebanyak 100 data, sedangkan untuk siswa yang tidak lulus ada sebanyak 38 data. Hasil pembagian data untuk *testing data* dengan pembagian sebesar 40% ¹⁵ dapat dilihat pada Tabel 3.11 di bawah ini.

Tabel 3.11 PKn *Binary Classification Testing Data (40%)*.

<i>Binary Classification Testing Data (PKn)</i>	
Nilai	Total Data
Lulus	66
Tidak Lulus	27
Total Data	93

⁷ Berdasarkan Tabel 3.11 di atas, dapat dilihat bahwa pada *testing data* sebesar 40% terdapat total 93 data. Terdapat juga 66 data untuk siswa yang lulus dan ada 27 data untuk siswa yang tidak lulus.

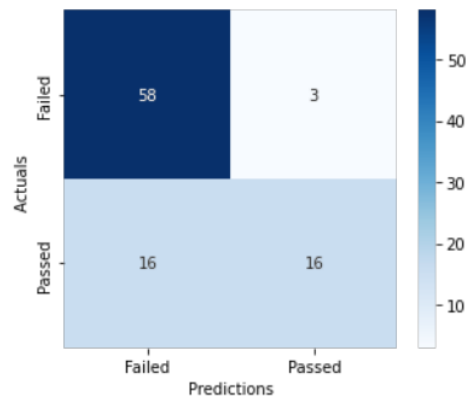
HASIL DAN PEMBAHASAN

4.1 Hasil Model *Machine Learning* pada Mata Pelajaran Matematika

Hasil model *machine learning* pada mata pelajaran Matematika menghasilkan empat model *machine learning*. Hasil klasifikasi lulus (*passed*) merepresentasikan nilai siswa diatas 80 dan tidak lulus (*failed*) merepresentasikan siswa mendapat nilai kurang dari 80. Semua model menggunakan sistem *binary classification*, kemudian empat model tersebut menggunakan pembagian *training* data dan *testing* data sebesar 60:40.

4.1.1 Hasil Model DT pada Mata Pelajaran Matematika

Berikut ini *confusion matrix* DT dengan pembagian data sebesar 60:40. Hasilnya dapat dilihat pada Gambar 4.1 di bawah ini.



Gambar 4.1 Matematika DT *Confusion Matrix* 60:40

Berdasarkan Gambar 4.1 di atas, dapat dilihat hasil prediksi dari algoritma DT pada mata pelajaran Matematika. Berdasarkan gambar tersebut DT berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 74 data. DT berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 16 data. DT juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 58 data. DT juga melakukan kesalahan klasifikasi sebanyak 19 kali. DT salah

memprediksi data yang nilai aslinya lulus tetapi diprediksikan sebagai tidak lulus sebanyak 16 kali, dan DT salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 3 kali. Hasil *classification report* untuk DT pada mata pelajaran Matematika dengan pembagian data 60:40 dapat dilihat pada Tabel 4.1 di bawah ini.

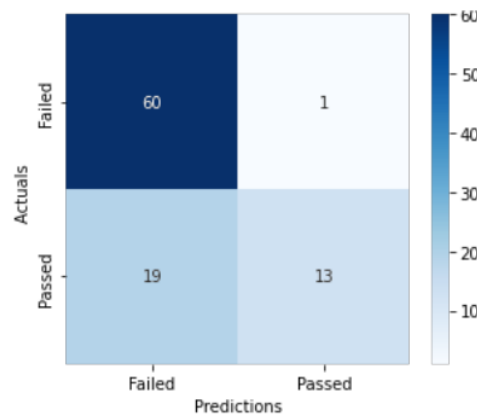
Tabel 4.1 Matematika DT

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	78	95	86	61
P	84	50	63	32
<i>Accuracy</i>			80	93

Berdasarkan Tabel 4.1 di atas, dapat dilihat nilai *precision*, *recall* dan *F1score* dari DT. DT memiliki akurasi yang baik dengan nilai sebesar 80%. DT juga bekerja dengan baik saat memprediksi nilai yang gagal yang hasilnya dapat dilihat pada nilai *precision*, *recall* dan *F1score* yang cukup baik. DT kurang bisa bekerja dengan baik pada saat memprediksi kelas lulus karena hanya memiliki *recall* sebesar 50% dan *F1score* yang hanya sebesar 63%.

4.1.2 Hasil Model KNN pada Mata Pelajaran Matematika

Berikut ini *confusion matrix* untuk algoritma KNN dengan pembagian data sebesar 60:40. Hasilnya dapat dilihat pada Gambar 4.2 di bawah ini.



Gambar 4.2 Matematika KNN *Confusion Matrix* 60:40

³ Berdasarkan Gambar 4.2 di atas, dapat dilihat hasil prediksi dari algoritma KNN pada mata pelajaran Matematika. Berdasarkan gambar tersebut KNN berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 73 data. KNN berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 13 data. KNN juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 60 data. KNN juga melakukan kesalahan klasifikasi sebanyak 20 kali. KNN salah memprediksi data yang nilai aslinya lulus sebanyak 19 kali, dan KNN salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 1 kali. Hasil *classification report* untuk KNN pada mata pelajaran Matematika dengan pembagian data 60:40 ²⁸ ini dapat dilihat pada Tabel 4.2 di bawah ini.

Tabel 4.2 Matematika KNN

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	76	98	86	61
P	93	41	57	32
<i>Accuracy</i>			78	93

⁶⁰ Berdasarkan Tabel 4.2 di atas, dapat dilihat nilai *precision*, *recall* dan *F1score* dari KNN. KNN memiliki akurasi yang baik dengan nilai sebesar 78%. KNN juga bekerja dengan baik saat memprediksi nilai yang gagal yang hasilnya dapat dilihat pada nilai *precision*, *recall* dan *F1score* yang cukup baik. KNN kurang bisa bekerja dengan baik pada saat memprediksi kelas lulus karena hanya memiliki *recall* sebesar 41% dan *F1score* yang hanya sebesar 57%.

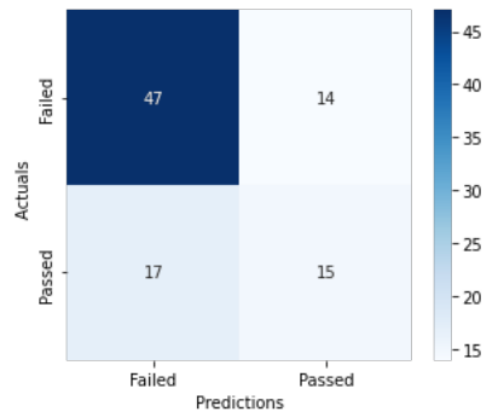
4.1.3 Hasil Model SVM pada Mata Pelajaran Matematika

Berikut *classification report* untuk SVM pada mata pelajaran Matematika dengan pembagian data 60:40. ⁷ Hasilnya dapat dilihat pada Tabel 4.3 di bawah ini.

Tabel 4.3 Matematika SVM

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	73	77	75	61
P	52	47	49	32
<i>Accuracy</i>			67	93

¹⁵ Berdasarkan Tabel 4.3 di atas, dapat dilihat nilai *precision*, *recall*, dan *F1score* dari SVM. SVM memiliki akurasi yang baik dengan nilai sebesar 67%. SVM dapat bekerja dengan cukup baik meskipun nilai *precision*, *recall*, dan *F1score* yang didapatkan SVM tidak sebaik DT dan KNN. Hasil *confusion matrix* untuk SVM ⁴ dapat dilihat pada Gambar 4.3 di bawah ini.

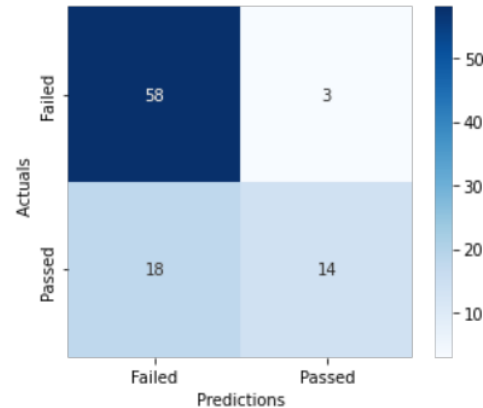


Gambar 4.3 Matematika SVM *Confusion Matrix* 60:40

³ Berdasarkan Gambar 4.3 di atas, dapat dilihat hasil prediksi dari algoritma SVM pada mata pelajaran Matematika. Berdasarkan gambar tersebut SVM berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 62 data. SVM berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 15 data. SVM juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 47 data. SVM juga melakukan kesalahan klasifikasi sebanyak 31 kali. SVM salah memprediksi data yang nilai aslinya lulus sebanyak 17 kali, dan SVM salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 14 kali.

4.1.4 Hasil Model NB pada Mata Pelajaran Matematika

Berikut *confusion matrix* untuk algoritma NB dengan pembagian data sebesar 60:40. Hasilnya ⁴ dapat dilihat pada Gambar 4.4 di bawah ini.



Gambar 4.4 Matematika NB *Confusion Matrix* 60:40

³ Berdasarkan Gambar 4.4 di atas, dapat dilihat hasil prediksi dari algoritma SVM pada mata pelajaran Matematika. Berdasarkan gambar tersebut NB berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 72 data. NB berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 14 data. NB juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 58 data. NB juga melakukan kesalahan klasifikasi sebanyak 31 kali. NB salah memprediksi data yang nilai aslinya lulus sebanyak 18 kali, dan NB salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 3 kali. Hasil *classification report* untuk NB pada mata pelajaran Matematika dengan pembagian data 60:40 ⁶ ini dapat dilihat pada Tabel 4.4 di bawah ini.

Tabel 4.4 Matematika NB

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1Score (%)</i>	<i>Support</i>
F	76	95	85	61
P	82	44	57	32
<i>Accuracy</i>			77	93

¹⁹ Berdasarkan Tabel 4.4 di atas, dapat dilihat nilai *precision*, *recall* dan *F1Score* dari NB. NB memiliki akurasi yang baik dengan nilai sebesar 77%. NB juga bekerja dengan baik saat memprediksi nilai yang gagal yang hasilnya dapat dilihat pada nilai *precision*, *recall*, dan *F1Score* yang cukup baik. NB kurang bisa

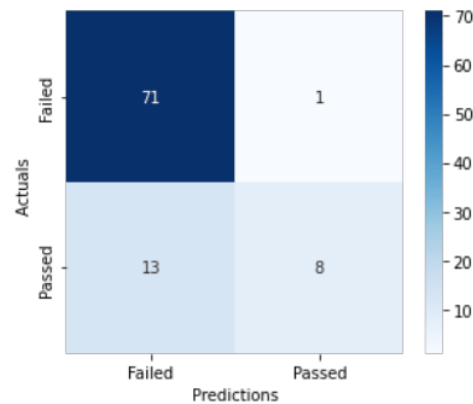
bekerja dengan baik pada saat memprediksi kelas lulus karena hanya memiliki *recall* sebesar 44% dan *F1score* yang hanya sebesar 57%.

4.2 Hasil Model *Machine Learning* pada Mata Pelajaran Bahasa Inggris

Hasil model *machine learning* pada mata pelajaran Bahasa Inggris menghasilkan empat model *machine learning*. Hasil klasifikasi lulus (*passed*) merepresentasikan nilai siswa diatas 80 dan tidak lulus (*failed*) merepresentasikan siswa mendapat nilai kurang dari 80. Semua model menggunakan sistem *binary classification*, kemudian empat model tersebut menggunakan pembagian data *training* dan *testing* sebesar 60:40.

4.2.1 Hasil Model DT pada Mata Pelajaran Bahasa Inggris

Berikut ini *confusion matrix* DT dengan pembagian data sebesar 60:40. Hasilnya dapat dilihat pada Gambar 4.5 di bawah ini.



Gambar 4.5 Bahasa Inggris DT *Confusion Matrix* 60:40

Berdasarkan Gambar 4.5 di atas, dapat dilihat hasil prediksi dari algoritma DT pada mata pelajaran Bahasa Inggris. Berdasarkan gambar 4.5 tersebut DT berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 79 data. DT berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 8 data. DT juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 71 data. DT juga melakukan kesalahan klasifikasi sebanyak 14 kali. DT salah

memprediksi data yang nilai aslinya lulus tetapi diprediksikan sebagai tidak lulus sebanyak 13 kali dan DT salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 1 kali. Hasil *classification report* untuk DT pada mata pelajaran Bahasa Inggris dengan pembagian data 60:40 dapat dilihat pada Tabel 4.5 di bawah ini.

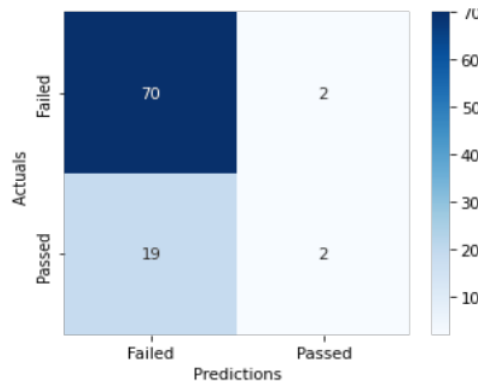
Tabel 4.5 Bahasa Inggris DT

	Precision (%)	Recall (%)	F1score (%)	Support
F	85	99	91	72
P	89	38	53	21
Accuracy			85	58

Berdasarkan Tabel 4.5 di atas, dapat dilihat nilai *precision*, *recall* dan *F1score* dari DT. DT memiliki akurasi yang baik dengan nilai sebesar 85%. DT juga bekerja dengan baik saat memprediksi nilai yang gagal yang hasilnya dapat dilihat pada nilai *precision*, *recall*, dan *F1score* yang cukup baik. DT kurang bisa bekerja dengan baik pada saat memprediksi kelas lulus karena hanya memiliki *recall* sebesar 38% dan *F1score* yang hanya sebesar 53%.

4.2.2 Hasil Model KNN pada Mata Pelajaran Bahasa Inggris

Berikut ini *confusion matrix* KNN dengan pembagian data sebesar 60:40. Hasilnya dapat dilihat pada Gambar 4.6 di bawah ini.

Gambar 4.6 Bahasa Inggris KNN *Confusion Matrix* 60:40

³ Berdasarkan Gambar 4.6 di atas, dapat dilihat hasil prediksi dari algoritma KNN pada mata pelajaran Bahasa Inggris. Berdasarkan gambar tersebut KNN berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 72 data. KNN berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 2 data. KNN juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 70 data. KNN juga melakukan kesalahan klasifikasi sebanyak 21 kali. KNN salah memprediksi data yang nilai aslinya lulus sebanyak 19 kali, dan KNN melakukan kesalahan saat memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 2 kali. Hasil *classification report* untuk KNN ²⁵ pada mata pelajaran Bahasa Inggris dengan pembagian data 60:40 ²⁷ ini dapat dilihat pada Tabel 4.6 di bawah ini.

Tabel 4.6 Bahasa Inggris KNN

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	79	97	87	72
P	50	10	16	21
<i>Accuracy</i>			77	93

³⁰ Berdasarkan Tabel 4.6 di atas, dapat dilihat nilai *precision*, *recall* dan *F1score* dari KNN. KNN memiliki akurasi sebesar 76%. KNN juga bekerja dengan baik saat memprediksi nilai yang gagal yang hasilnya dapat dilihat pada nilai *precision*, *recall*, dan *F1score* yang cukup baik. KNN kurang bisa bekerja dengan baik pada saat memprediksi kelas lulus karena hanya memiliki *recall* sebesar 10% dan *F1score* yang hanya sebesar 16%.

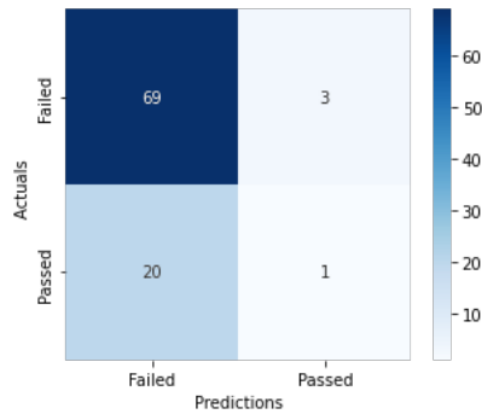
²² 4.2.3 Hasil Model SVM pada Mata Pelajaran Bahasa Inggris

Berikut *classification report* untuk SVM ²⁵ pada mata pelajaran Bahasa Inggris ⁶ dengan pembagian data 60:40. Hasilnya dapat dilihat pada Tabel 4.7 di bawah ini.

Tabel 4.7 Bahasa Inggris SVM

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	78	96	86	72
P	25	5	8	21
<i>Accuracy</i>			75	93

³⁰ Berdasarkan Tabel 4.7 di atas, dapat dilihat nilai *precision*, *recall*, dan *F1score* dari SVM. SVM memiliki akurasi sebesar 75%. SVM bekerja dengan buruk pada kelas lulus karena nilai *recall* dan *F1score* dari SVM sangat buruk. Hasil *confusion matrix* untuk SVM dapat dilihat pada Gambar 4.7 di bawah ini.

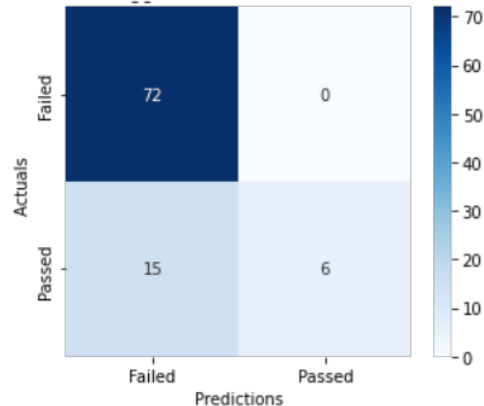


Gambar 4.7 Bahasa Inggris SVM *Confusion Matrix* 60:40

³ Berdasarkan Gambar 4.7 di atas, dapat dilihat hasil prediksi dari algoritma SVM pada mata pelajaran Bahasa Inggris. Berdasarkan gambar tersebut SVM berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 70 data. SVM berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 1 data. SVM juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 69 data. SVM juga melakukan kesalahan klasifikasi sebanyak 23 kali. SVM salah memprediksi data yang nilai aslinya lulus sebanyak 20 kali, dan SVM salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 3 kali.

4.2.4 Hasil Model NB pada Mata Pelajaran Bahasa Inggris

² Berikut ini *confusion matrix* NB dengan pembagian data sebesar 60:40. Hasilnya dapat dilihat pada Gambar 4.8 di bawah ini.



Gambar 4.8 Bahasa Inggris NB *Confusion Matrix* 60:40

Berdasarkan Gambar 4.8 di atas, dapat dilihat hasil prediksi dari algoritma NB pada mata pelajaran Bahasa Inggris. Berdasarkan gambar tersebut NB berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 78 data. NB berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 6 data. NB juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 72 data. NB juga melakukan kesalahan klasifikasi sebanyak 15 kali. NB tidak melakukan kesalahan saat memprediksi data yang nilai aslinya lulus tetapi diklasifikasikan sebagai tidak lulus, dan NB salah memprediksi nilai yang seharusnya lulus tetapi diklasifikasikan sebagai tidak lulus sebanyak 15 kali. Hasil *classification report* untuk NB pada mata pelajaran Bahasa Inggris dengan pembagian data 60:40 ini dapat dilihat pada Tabel 4.8 di bawah ini.

Tabel 4.8 Bahasa Inggris NB

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	83	100	91	61
P	100	29	44	32
<i>Accuracy</i>			84	93

Berdasarkan Tabel 4.8 di atas, dapat dilihat nilai *precision*, *recall*, dan *F1score* dari NB. NB memiliki akurasi yang baik dengan nilai sebesar 84%. NB juga bekerja dengan baik saat memprediksi nilai yang gagal yang hasilnya dapat dilihat pada nilai *precision*, *recall*, dan *F1score* yang cukup baik. NB kurang bisa

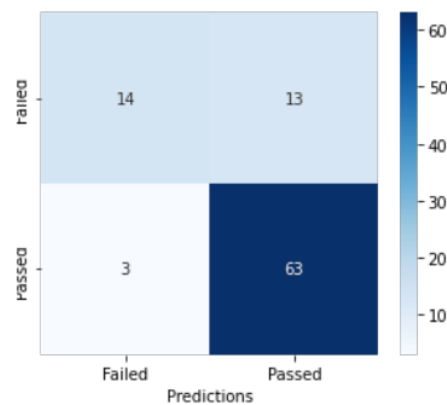
bekerja dengan baik pada saat memprediksi kelas lulus karena hanya memiliki *recall* sebesar 29% dan *F1score* yang hanya sebesar 44%.

4.3 Hasil Model *Machine Learning* pada Mata Pelajaran PKn

Hasil model *machine learning* pada mata pelajaran PKn menghasilkan empat model *machine learning*. Hasil klasifikasi lulus (*passed*) merepresentasikan nilai siswa diatas 80 dan tidak lulus (*failed*) merepresentasikan siswa mendapat nilai kurang dari 80. Semua model menggunakan sistem *binary classification*, kemudian empat model tersebut menggunakan pembagian data *training* dan *testing* sebesar 60:40.

4.3.1 Hasil Model DT pada Mata Pelajaran PKn

Berikut ini *confusion matrix* DT dengan pembagian data sebesar 60:40. Hasilnya dapat dilihat pada Gambar 4.9 di bawah ini.



Gambar 4.9 PKn DT *Confusion Matrix* 60:40

Berdasarkan Gambar 4.9 di atas, dapat dilihat hasil prediksi dari algoritma DT pada mata pelajaran PKn. Berdasarkan gambar tersebut DT berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 77 data. DT berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 63 data. DT juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 14 data. DT juga melakukan kesalahan klasifikasi sebanyak 16 kali. DT salah

memprediksi data yang nilai aslinya lulus tetapi diprediksikan sebagai tidak lulus sebanyak 3 kali, dan DT salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 13 kali. Hasil *classification report* untuk DT pada mata pelajaran PKn dengan pembagian data 60:40 ini dapat dilihat pada Tabel 4.9 di bawah ini.

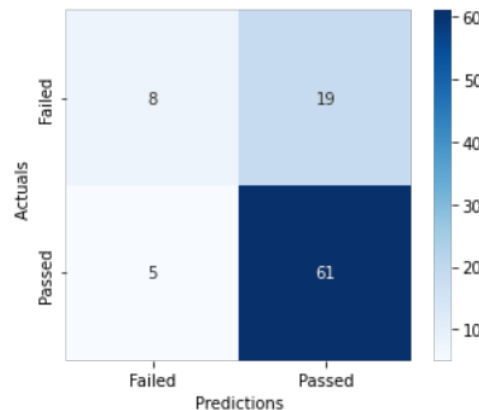
Tabel 4.9 PKn DT

	Precision (%)	Recall (%)	F1score (%)	Support
F	82	52	64	27
P	83	95	89	66
Accuracy			83	93

Berdasarkan Tabel 4.9 di atas, dapat dilihat nilai *precision*, *recall*, dan *F1score* dari DT. DT memiliki akurasi sebesar 83%. DT juga bekerja dengan baik saat memprediksi nilai yang lulus yang hasilnya dapat dilihat pada nilai *precision*, *recall*, dan *F1score* yang cukup baik. DT kurang bisa bekerja dengan baik pada saat memprediksi nilai yang tidak lulus karena hanya memiliki *recall* sebesar 52% dan *F1score* yang hanya sebesar 64%.

4.3.2 Hasil Model KNN pada Mata Pelajaran PKn

Berikut ini *confusion matrix* KNN dengan pembagian data sebesar 60:40. Hasilnya dapat dilihat pada Gambar 4.10 di bawah ini.



Gambar 4.10 PKn KNN Confusion Matrix 60:40

3 Berdasarkan Gambar 4.10 di atas, dapat dilihat hasil prediksi dari algoritma KNN pada mata pelajaran PKn. Berdasarkan gambar tersebut KNN berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 69 data. KNN berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 61 data. KNN juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 8 data. KNN juga melakukan kesalahan klasifikasi sebanyak 24 kali. KNN salah memprediksi data yang nilai aslinya lulus sebanyak 5 kali, dan KNN melakukan kesalahan saat memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 19 kali. Hasil *classification report* untuk KNN pada mata pelajaran PKn dengan pembagian data 60:40 7 ini dapat dilihat pada Tabel 4.10 di bawah ini.

Tabel 4.10 PKn KNN

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	62	30	40	27
P	76	92	84	66
<i>Accuracy</i>			74	93

5 Berdasarkan Tabel 4.10 di atas, dapat dilihat nilai *precision*, *recall* dan *F1score* dari KNN. KNN memiliki akurasi sebesar 74%. KNN juga bekerja dengan baik saat memprediksi nilai yang lulus yang hasilnya dapat dilihat pada nilai *precision*, *recall*, dan *F1score* yang cukup baik. KNN kurang bisa bekerja dengan baik pada saat memprediksi nilai yang tidak lulus karena hanya memiliki *recall* sebesar 30% dan *F1score* yang hanya sebesar 40%.

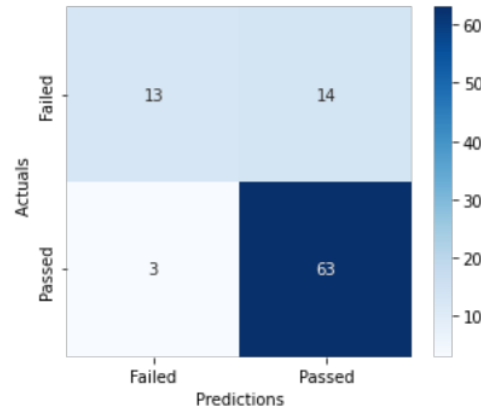
4.3.3 Hasil Model SVM Pada Mata Pelajaran PKn

Berikut *classification report* untuk SVM pada mata pelajaran PKn dengan pembagian data 60:40. Hasilnya dapat dilihat pada Tabel 4.11 di bawah ini. 6

Tabel 4.11 PKn SVM

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	81	48	60	27
P	82	95	88	66
<i>Accuracy</i>			82	93

¹⁴ Berdasarkan Tabel 4.11 di atas, dapat dilihat nilai *precision*, *recall* dan *F1score* dari SVM. SVM memiliki akurasi sebesar 82%. SVM memiliki hasil yang sangat baik terutama pada prediksi untuk kelas lulus. Hasil *confusion matrix* untuk SVM ⁴ dapat dilihat pada Gambar 4.11 di bawah ini.

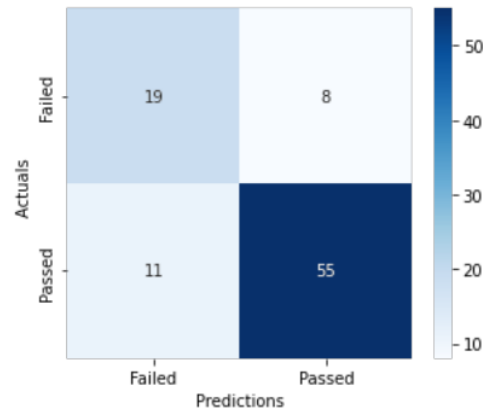


Gambar 4.11 PKn SVM *Confusion Matrix* 60:40

³ Berdasarkan Gambar 4.11 di atas, dapat dilihat hasil prediksi dari algoritma SVM pada mata pelajaran PKn. Berdasarkan gambar tersebut SVM berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 76 data. SVM berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 63 data. SVM juga berhasil mengklasifikasikan data yang diprediksi tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 13 data. SVM juga melakukan kesalahan klasifikasi sebanyak 17 kali. SVM salah memprediksi data yang nilai aslinya lulus tetapi diklasifikasikan sebagai tidak lulus sebanyak 3 kali, dan SVM salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 14 kali.

4.3.4 Hasil Model NB pada Mata Pelajaran PKn

³⁷ Berikut ini *confusion matrix* NB dengan pembagian data sebesar 60:40. Hasilnya dapat dilihat pada Gambar 4.12 di bawah ini.



Gambar 4.12 PKn NB *Confusion Matrix* 60:40

Berdasarkan Gambar 4.12 di atas, dapat dilihat hasil prediksi dari algoritma SVM pada mata pelajaran PKn. Berdasarkan gambar tersebut NB berhasil melakukan klasifikasi dengan tepat sasaran sebanyak 74 data. NB berhasil mengklasifikasikan data yang diprediksikan lulus dan data aslinya benar-benar lulus sebanyak 55 data. NB juga berhasil mengklasifikasikan data yang diprediksikan tidak lulus dan data aslinya benar-benar tidak lulus sebanyak 19 data. NB juga melakukan kesalahan klasifikasi sebanyak 19 kali. NB salah memprediksikan data yang nilai aslinya lulus tetapi diklasifikasikan sebagai tidak lulus sebanyak 11 kali, dan NB salah memprediksi nilai yang seharusnya tidak lulus tetapi diklasifikasikan sebagai lulus sebanyak 8 kali. Hasil *classification report* untuk NB pada mata pelajaran PKn dengan pembagian data 60:40 ini dapat dilihat pada Tabel 4.12 di bawah ini.

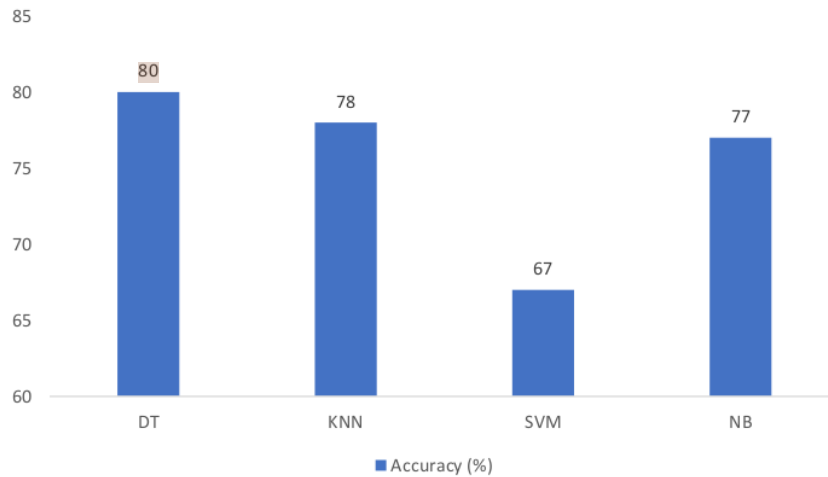
Tabel 4.12 PKn NB

	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>F1score (%)</i>	<i>Support</i>
F	63	70	67	27
P	87	83	85	66
<i>Accuracy</i>			80	93

Berdasarkan Tabel 4.12 di atas, dapat dilihat nilai *precision*, *recall*, dan *F1score* dari NB. NB memiliki akurasi sebesar 80% dan memiliki performa yang baik saat memprediksi kelas lulus.

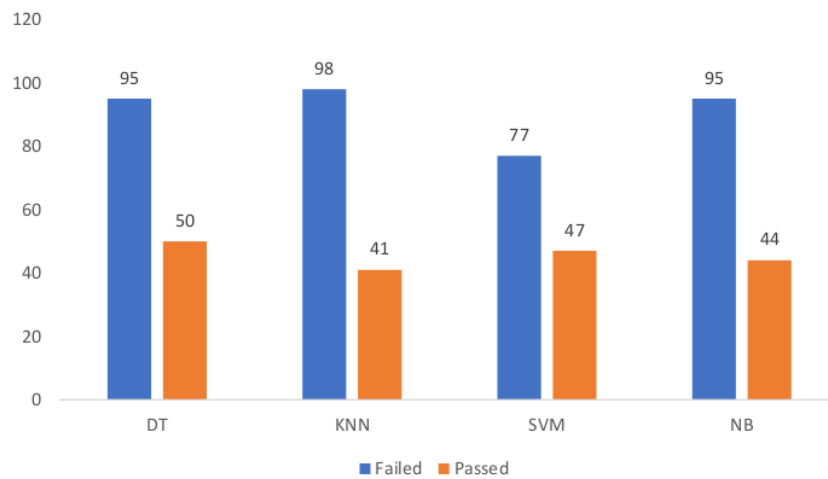
4.4 Evaluasi Model *Machine Learning* pada Mata Pelajaran Matematika

Hasil model *machine learning* pada mata pelajaran Matematika menghasilkan empat model *machine learning*. Perbandingan dalam hal akurasi empat model tersebut dapat dilihat pada Gambar 4.13 di bawah ini.



Gambar 4.13 Grafik Perbandingan Nilai Akurasi Pada Mata Pelajaran Matematika

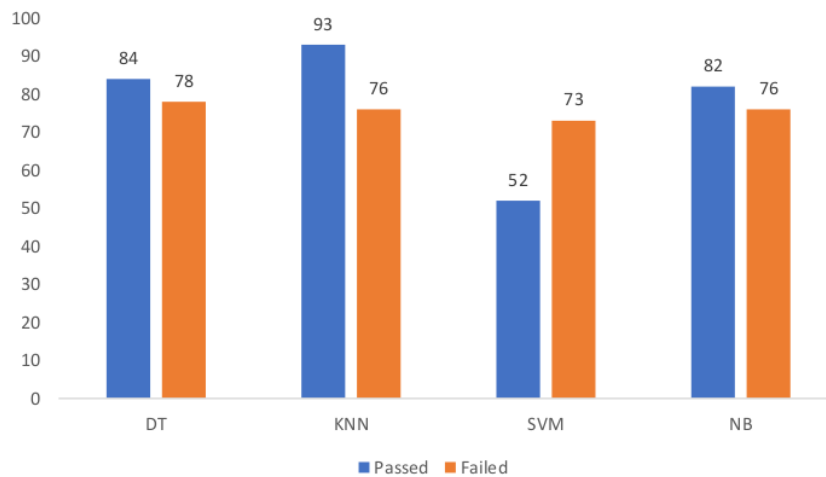
Berdasarkan Gambar 4.13 di atas dapat dilihat perbandingan akurasi pada mata pelajaran Matematika. Mengacu pada akurasi saja maka bisa dikatakan bahwa DT memiliki performa terbaik dengan akurasi sebesar 80%. Beberapa kasus memang menitikberatkan performa suatu algoritma berdasarkan akurasi [8] [9] tetapi pada kasus *imbalanced* data akurasi saja tidak cukup untuk menentukan algoritma mana yang memiliki performa terbaik. Akurasi pada model *machine learning* yang memiliki *class imbalanced* akan cenderung memberikan akurasi yang tinggi kepada model yang memprediksikan banyak nilai ke kelas mayoritas atau bahkan dengan memprediksi semua nilai ke kelas mayoritas maka akurasi yang tinggi bisa didapatkan. Model evaluasi *machine learning* yang lainya seperti *precision*, *recall*, *F1score*, dan *MCC* bisa memberikan sebuah gambaran yang lebih baik terkait performa suatu algoritma yang memiliki *imbalanced* data. Gambar perbandingan *recall* dari setiap algoritma *machine learning* pada mata pelajaran Matematika dapat dilihat pada Gambar 4.14 di bawah ini.



Gambar 4.14 Grafik Perbandingan Nilai *Recall* Pada Mata Pelajaran Matematika

Gambar 4.14 di atas memberikan gambaran tentang performa setiap algoritma *machine learning* pada mata pelajaran Matematika dengan metode evaluasi *recall*. *Recall* merupakan persentase data suatu model *machine learning* yang berhasil diklasifikasikan dengan positif dari total data yang benar-benar data positif. Dapat dilihat pada Gambar 4.14 bahwa DT, KNN, dan NB memiliki performa yang hampir setara saat mengklasifikasikan kelas *passed* dan *failed*, tetapi KNN memiliki *recall* paling tinggi untuk kelas *failed* dengan nilai sebesar 98%. Mengacu pada Tabel 4.2 tentang *classification report* dari KNN, hal ini berarti dari 61 siswa yang benar-benar tidak lulus, hanya satu siswa yang diprediksikan sebagai lulus, sedangkan pada DT dan NB yang memiliki *recall* sebesar 95% dan mengacu pada Tabel 4.1 dan Tabel 4.4, ini berarti DT dan NB berhasil memprediksi dengan tepat sasaran 58 siswa dari 61 siswa yang benar-benar gagal dan melakukan kesalahan prediksi siswa yang gagal diprediksikan sebagai lulus sebanyak tiga data. Hasil yang diperoleh DT, KNN, dan SVM sangat bagus karena pada kasus prediksi performa siswa, model algoritma ini dirancang untuk pencegahan sehingga siswa yang benar-benar tidak lulus bisa diberikan bimbingan dan pengarahan yang lebih sehingga kedepannya siswa tersebut bisa berubah menjadi siswa yang memiliki nilai lulus. *Recall* dari KNN memiliki nilai sebesar 41% dan *recall* dari NB sebesar 44%. Hasil ini memang

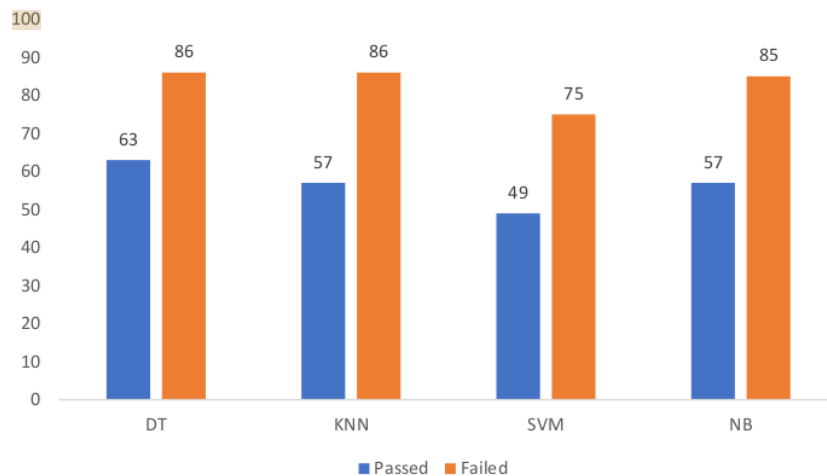
masih kalah jika dibandingkan *recall* dari DT yang sebesar 50% untuk siswa yang diprediksikan lulus, tetapi tetap dapat dikatakan bahwa DT, KNN, dan NB bisa digunakan untuk prediksi performa siswa pada mata pelajaran Matematika. Hasil *recall* dari DT, KNN dan NB pada kelas lulus memang bukan hasil yang baik karena itu berarti hampir 50% dari siswa yang sebenarnya lulus diprediksikan sebagai tidak lulus, tetapi dampak negatif dari siswa yang sebenarnya lulus tetapi diprediksikan sebagai tidak lulus itu sendiri tidak besar karena nanti hasil akhirnya siswa tersebut sebenarnya tetap lulus pada akhirnya. Sebaliknya yaitu siswa yang tidak lulus diprediksikan sebagai lulus, hal ini akan membuat masalah karena pada akhirnya siswa tersebut akan benar-benar tidak lulus. Hasil perbandingan nilai *precision* yang didapat pada mata pelajaran Matematika bisa dilihat pada Gambar 4.15 di bawah ini.



Gambar 4.15 Grafik Perbandingan Nilai *Precision* Pada Mata Pelajaran Matematika.

Gambar 4.15 di atas menggambarkan perbandingan nilai *precision* untuk mata pelajaran Matematika. Dapat dilihat bahwa berdasarkan *precision*, KNN memiliki nilai yang terbaik untuk kelas *failed* dan DT memiliki nilai *precision* terbaik untuk kelas *passed*. *Precision* menjawab untuk pertanyaan berapa banyak prediksi positif yang benar dari seluruh prediksi positif yang dilakukan. KNN berdasarkan grafik berhasil memprediksi data untuk siswa yang diprediksikan lulus dengan nilai *precision* sebesar 93%, yang artinya dari seluruh hasil prediksi

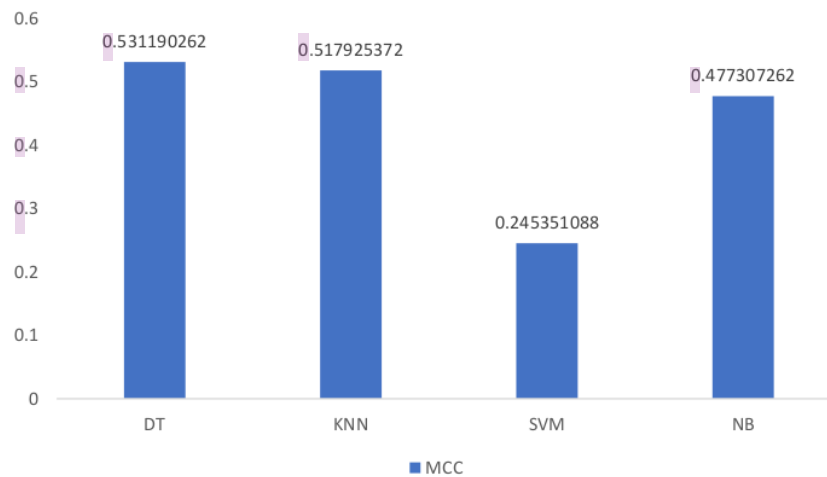
siswa yang lulus yang dibuat oleh KNN, siswa yang benar-benar lulus sebesar 93%, dan jika melihat lagi kepada Tabel 4.2 tentang *classification report* dari KNN, dapat dilihat bahwa dari 14 prediksi siswa yang lulus, hanya 1 prediksi siswa yang salah yang artinya hal ini sangat bagus. Siswa-siswa yang diprediksikan lulus tersebut ternyata benar-benar lulus hampir semuanya. *Precision* KNN untuk kelas *failed* sebesar 76%, yang artinya dari seluruh data yang diprediksikan tidak lulus, sebesar 76% dari siswa tersebut benar-benar tidak lulus. Hasil ini bukan hasil yang buruk mengingat karena walaupun siswa yang lulus diprediksikan sebagai tidak lulus maka tidak ada dampak negatif yang didapat siswa, sebaliknya siswa tersebut yang diprediksikan gagal padahal aslinya dia lulus bisa membuat performanya lebih meningkat lagi karena siswa yang aslinya sudah lulus diberikan bimbingan tambahan lagi yang mana hal ini merupakan hal positif yang baik. Hasil perbandingan nilai *F1score* yang didapat bisa dilihat pada Gambar 4.16 di bawah ini.



Gambar 4.16 Grafik Perbandingan Nilai *F1score* pada Mata Pelajaran Matematika.

Gambar 4.16 di atas memberikan gambaran perbandingan performa algoritma *machine learning* pada mata pelajaran Matematika dengan metode evaluasi *F1score*. *F1score* adalah metode evaluasi yang menggambarkan nilai rata-rata *harmonic* dari *recall* dan *precision*. Berdasarkan grafik perbandingan tersebut dapat dilihat bahwa DT merupakan algoritma dengan performa terbaik

karena nilai *F1score*-nya paling tinggi, yaitu 86% untuk kelas *failed* dan 63% untuk kelas *passed*. Berdasarkan *recall* maka DT, KNN, dan NB memiliki performa baik yang hampir setara. Selain dengan menggunakan *accuracy*, *recall*, *precision*, dan *F1score*, Sebuah model *machine learning* juga dapat menggunakan MCC untuk metode evaluasinya. Hasil dari MCC untuk setiap algoritma *machine learning* pada mata pelajaran Matematika dapat dilihat pada Gambar 4.17 di bawah ini.



Gambar 4.17 Grafik Perbandingan Nilai MCC pada Mata Pelajaran Matematika.

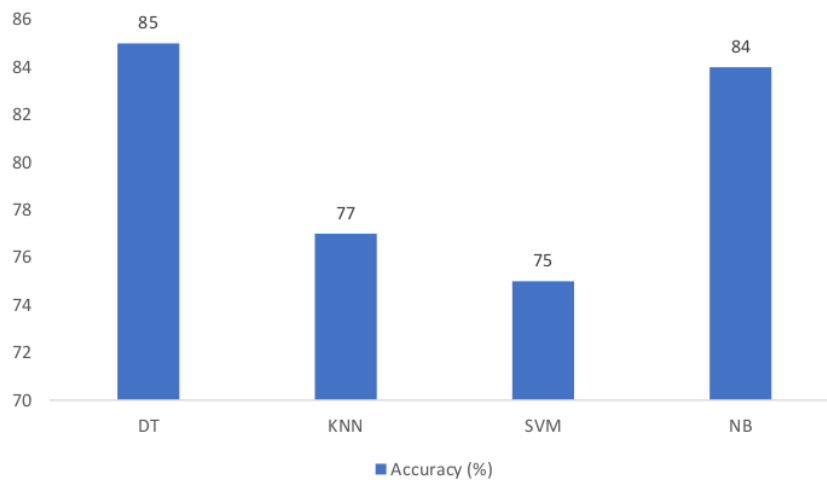
Berdasarkan Gambar 4.17 di atas, dapat hasil nilai MCC untuk empat algoritma yang digunakan untuk prediksi pada mata pelajaran Matematika. MCC merupakan metode evaluasi yang mengukur hasil prediksi model berdasarkan nilai dari *confusion matrix* dan cocok untuk model yang memiliki *class imbalanced*. Berdasarkan Gambar 4.17 di atas, dapat dilihat bahwa MCC yang paling tinggi dimiliki oleh DT. Nilai 0,531190262 merupakan nilai MCC yang paling besar dan paling mendekati nilai 1, sehingga bisa dikatakan bahwa DT adalah algoritma yang memiliki performa prediksi terbaik berdasarkan nilai MCC, tetapi KNN dan NB juga bekerja dengan baik karena nilai MCC yang hampir setara dengan KNN.

Berdasarkan hasil evaluasi dari akurasi, *recall*, *precision*, *F1score* dan MCC di atas, didapatkan kesimpulan bahwa DT, KNN, dan NB memiliki

performa yang hampir setara dan bisa digunakan untuk prediksi performa siswa pada mata pelajaran Matematika. SVM dapat dikatakan sebagai algoritma terburuk karena perbedaan nilai yang cukup jauh dibandingkan dengan ketiga algoritma lainnya.

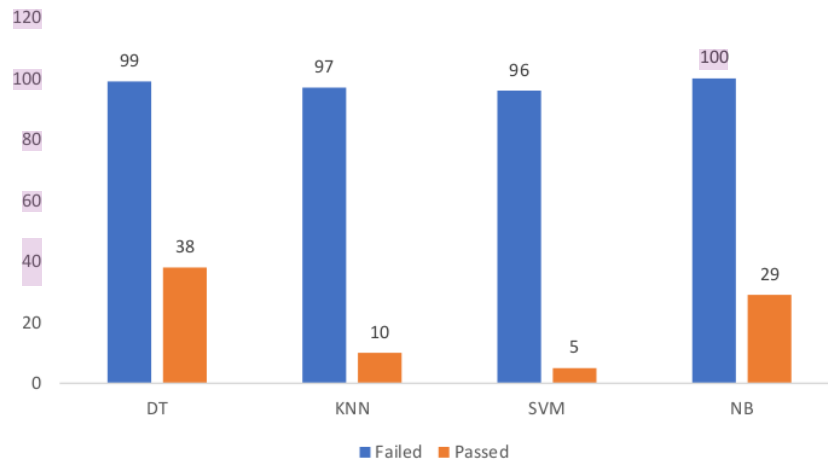
4.5 Evaluasi Model *Machine Learning* pada Pelajaran Bahasa Inggris

Hasil model *machine learning* pada mata pelajaran Bahasa Inggris menghasilkan empat model *machine learning*. Perbandingan dalam hal akurasi empat model tersebut dapat dilihat pada Gambar 4.18 di bawah ini.



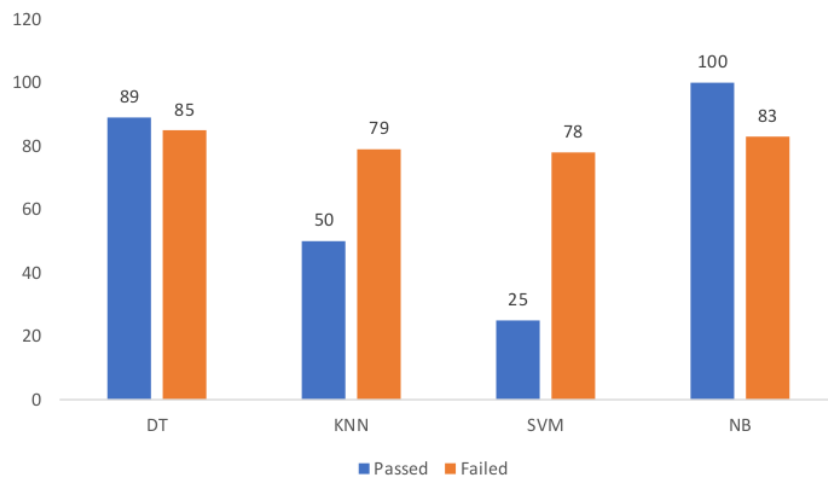
Gambar 4.18 Grafik Perbandingan Nilai Akurasi pada Mata Pelajaran Bahasa Inggris

Berdasarkan Gambar 4.18 di atas dapat dilihat perbandingan akurasi pada mata pelajaran Bahasa Inggris. Berdasarkan gambar tersebut DT bisa dikatakan memiliki performa terbaik karena memiliki akurasi sebesar 85%. NB memiliki performa terbaik kedua dengan akurasi sebesar 84%, kemudian diikuti oleh KNN dengan akurasi sebesar 77% dan SVM dengan akurasi sebesar 75%. Gambar perbandingan nilai *recall* pada mata pelajaran bahasa Inggris dapat dilihat pada Gambar 4.19 di bawah ini.



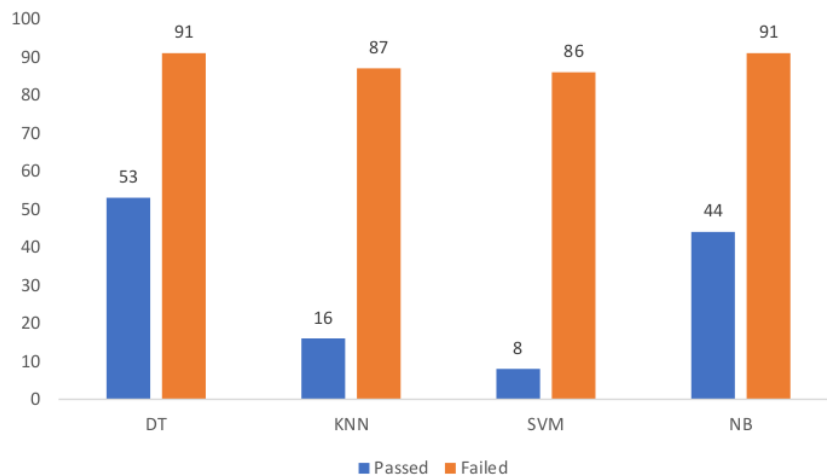
Gambar 4.19 Grafik Perbandingan Nilai *Recall* pada Mata Pelajaran Bahasa Inggris

Gambar 4.19 di atas memberikan gambaran tentang performa setiap algoritma *machine learning* pada mata pelajaran Bahasa Inggris dengan metode evaluasi *recall*. Berdasarkan Gambar 4.19 NB dapat dikatakan sebagai algoritma dengan *recall* terbaik pada kelas *failed* dengan nilai sebesar 100%, dan DT merupakan algoritma dengan *recall* terbaik untuk kelas *passed* dengan nilai *recall* sebesar 38%. Hasil perbandingan nilai *precision* yang didapat dari pelajaran Bahasa Inggris bisa dilihat pada Gambar 4.20 di bawah ini.



Gambar 4.20 Grafik Perbandingan Nilai *Precision* pada Mata Pelajaran Bahasa Inggris

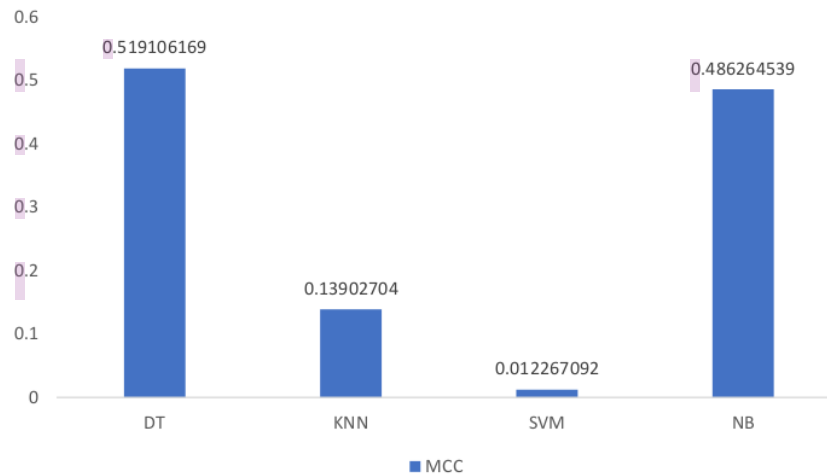
Gambar 4.20 di atas menggambarkan perbandingan nilai *precision* untuk mata pelajaran Bahasa Inggris. Dapat dilihat bahwa berdasarkan *precision*, NB memiliki performa terbaik dengan nilai *precision* pada prediksi *passed* sebesar 100% dan pada prediksi *failed* dengan nilai sebesar 83%. DT juga memiliki hasil yang mirip dengan NB. NB pada prediksi *passed* memiliki *precision* 100% karena berhasil menghasilkan 6 hasil prediksi *passed* yang semuanya nilai aslinya benar-benar *passed*, sedangkan DT pada prediksi *passed* memiliki *precision* 89% karena berhasil menghasilkan sembilan hasil prediksi *passed* dan dari sembilan hasil prediksi *passed* itu hanya satu nilai prediksi yang data aslinya itu bukan merupakan nilai *passed*. Hasil perbandingan nilai *F1score* yang didapat pada mata pelajaran Bahasa Inggris bisa dilihat pada Gambar 4.21 di bawah ini.



Gambar 4.21 Grafik Perbandingan Nilai *F1score* pada Mata Pelajaran Bahasa Inggris.

Gambar 4.21 di atas memberikan gambaran perbandingan performa algoritma *machine learning* pada mata pelajaran Bahasa Inggris dengan metode evaluasi *F1score*. *F1score* adalah metode evaluasi yang menggambarkan nilai rata-rata *harmonic* dari *recall* dan *precision*. Berdasarkan grafik perbandingan tersebut dapat dilihat bahwa NB dan DT masih tetap merupakan algoritma dengan performa terbaik dengan nilai *F1score* untuk kelas *failed* dan *passed*, karena nilai *F1score*-nya paling tinggi, NB memiliki nilai *F1score* 91% untuk kelas *failed* dan 44% untuk kelas *passed*, sedangkan DT memiliki nilai *F1score* sebesar 91%

untuk kelas *failed* dan 53% untuk kelas *passed*. Selain dengan menggunakan *accuracy*, *recall*, *precision*, dan *F1score*, Sebuah model *machine learning* juga menggunakan MCC untuk evaluasi performa. Hasil dari MCC untuk setiap algoritma *machine learning* pada mata pelajaran Bahasa Inggris dapat dilihat pada Gambar 4.22 di bawah ini.



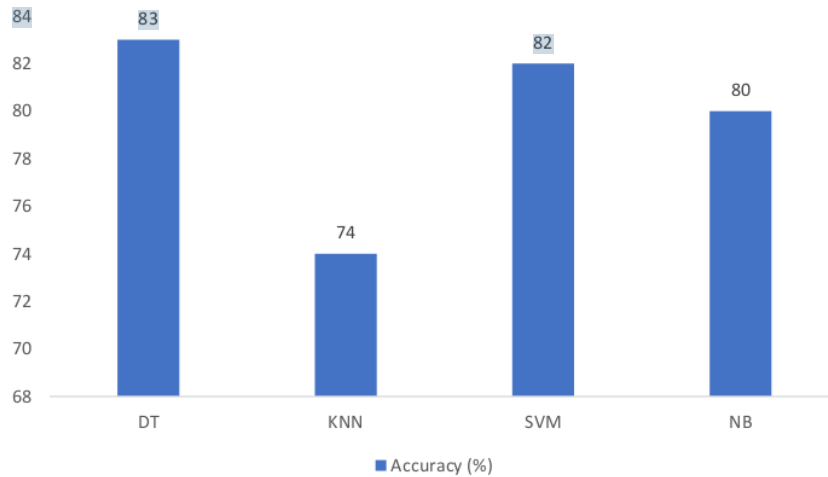
Gambar 4.22 Grafik Perbandingan Nilai MCC Pada Mata Pelajaran Bahasa Inggris.

Berdasarkan Gambar 4.22 di atas, dapat hasil nilai MCC untuk empat algoritma yang digunakan untuk prediksi pada mata pelajaran Bahasa Inggris. MCC merupakan metode evaluasi yang mengukur hasil prediksi model berdasarkan nilai dari *confusion matrix* dan cocok untuk model yang memiliki *class imbalanced*. Berdasarkan Gambar 4.22 di atas, dapat dilihat bahwa MCC yang paling tinggi dimiliki oleh DT dengan nilai 0.519106169 yang merupakan nilai MCC yang paling besar dan paling mendekati nilai 1, sehingga bisa dikatakan bahwa DT adalah algoritma yang memiliki performa prediksi terbaik berdasarkan nilai MCC.

Berdasarkan hasil evaluasi dari akurasi, *recall*, *precision*, *F1score*, dan MCC di atas, didapatkan kesimpulan bahwa DT dan NB merupakan algoritma yang memiliki performa prediksi terbaik. Hal ini karena DT dan NB berhasil mendapatkan nilai tertinggi dibandingkan dengan kedua algoritma lainnya pada semua metode evaluasi yang sudah dilakukan.

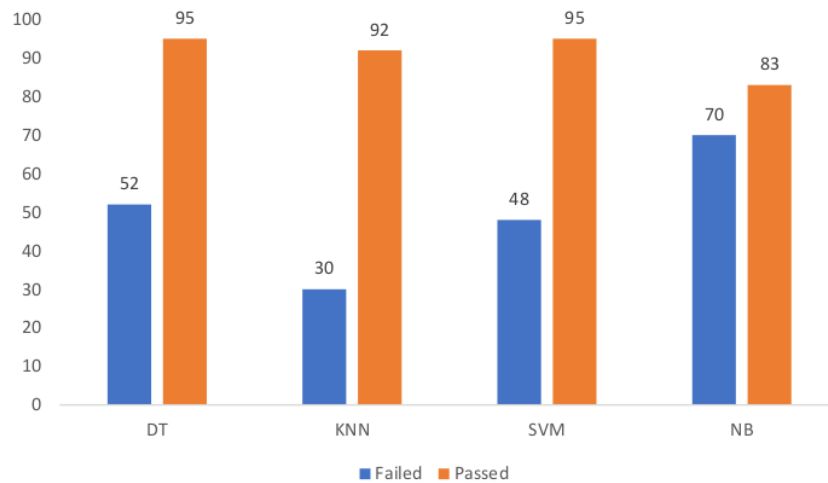
4.6 Evaluasi Model *Machine Learning* pada Pelajaran PKn

Hasil model *machine learning* pada mata pelajaran PKn menghasilkan empat model *machine learning*. Perbandingan dalam hal akurasi empat model tersebut dapat dilihat pada Gambar 4.23 di bawah ini.



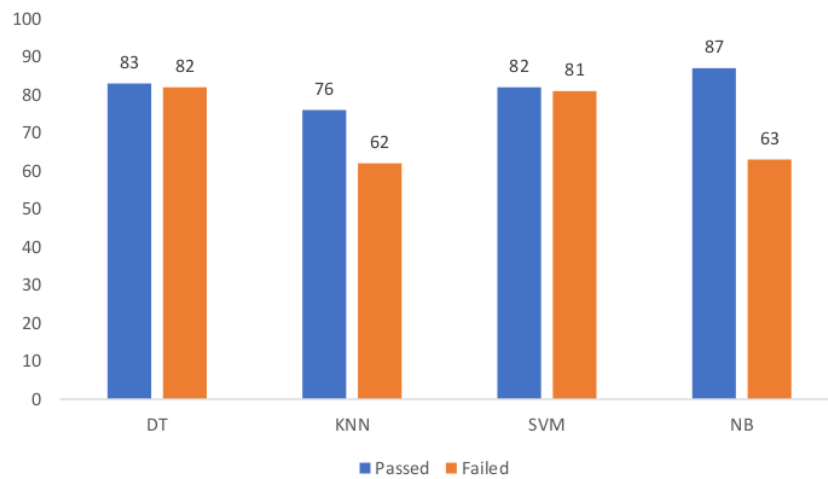
Gambar 4.23 Grafik Perbandingan Nilai Akurasi Pada Mata Pelajaran PKn

Berdasarkan Gambar 4.23 di atas dapat dilihat perbandingan akurasi pada mata pelajaran PKn. Berdasarkan gambar tersebut DT bisa dikatakan memiliki performa akurasi terbaik karena memiliki akurasi sebesar 83%. SVM memiliki performa terbaik kedua dengan akurasi sebesar 82%, kemudian diikuti oleh NB dengan akurasi sebesar 80% dan KNN dengan akurasi sebesar 74%. Berdasarkan nilai akurasi maka DT, SVM, dan NB merupakan algoritma *machine learning* yang sesuai untuk prediksi performa siswa pada mata pelajaran PKn, tetapi karena pada sistem prediksi ini data yang digunakan adalah *imbalanced dataset*, maka dibutuhkan metode evaluasi lain yang lebih mencerminkan performa dari suatu model. Gambar perbandingan nilai *recall* pada mata pelajaran PKn dapat dilihat pada Gambar 4.24 di bawah ini.



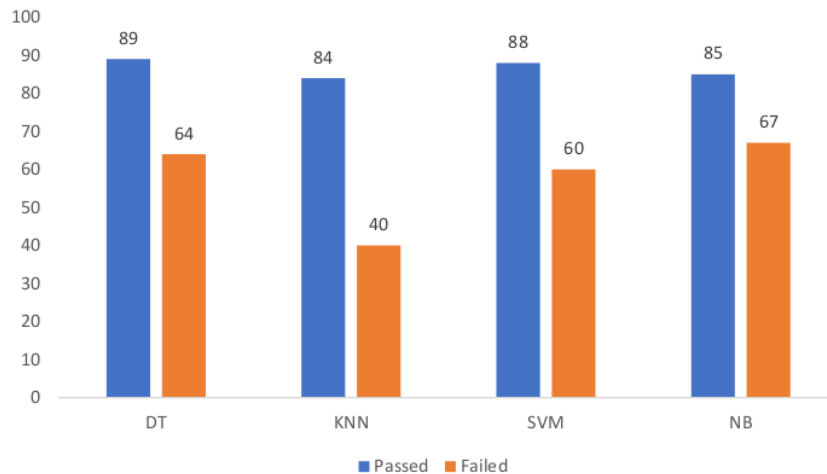
Gambar 4.24 Grafik Perbandingan Nilai *Recall* pada Mata Pelajaran PKn

Gambar 4.24 di atas memberikan gambaran tentang performa setiap algoritma *machine learning* pada mata pelajaran PKn dengan metode evaluasi *recall*. Berdasarkan Gambar 4.24 DT memiliki *recall* tertinggi untuk kelas *passed* dengan nilai *recall* sebesar 100%, dan NB memiliki *recall* tertinggi untuk kelas *failed* dengan nilai 70%. Hasil perbandingan nilai *precision* yang didapat dari pelajaran PKn ²⁸ bisa dilihat pada Gambar 4.25 di bawah ini.



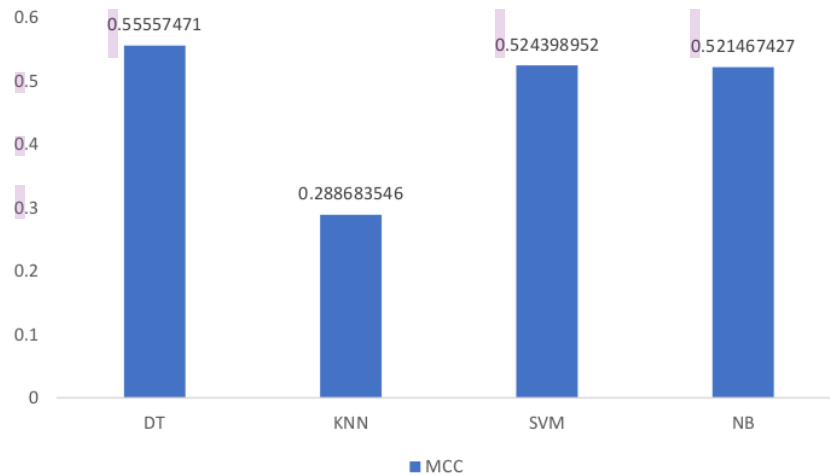
Gambar 4.25 Grafik Perbandingan Nilai *Precision* pada Mata Pelajaran PKn

Gambar 4.25 di atas menggambarkan perbandingan nilai *precision* untuk mata pelajaran PKn. Dapat dilihat bahwa berdasarkan *precision*, NB memiliki performa terbaik dengan *precision* pada prediksi *passed* sebesar 87%, dan DT memiliki nilai *precision* terbesar pada prediksi *failed* dengan nilai sebesar 82%. Hasil perbandingan nilai *F1score* yang didapat bisa dilihat pada Gambar 4.26 di bawah ini.



Gambar 4.26 Grafik Perbandingan Nilai *F1score* Pada Mata Pelajaran PKn.

Gambar 4.26 di atas memberikan gambaran perbandingan performa algoritma *machine learning* pada mata pelajaran PKn dengan metode evaluasi *F1score*. *F1score* adalah metode evaluasi yang menggambarkan nilai rata-rata *harmonic* dari *recall* dan *precision*. Berdasarkan grafik perbandingan tersebut dapat dilihat bahwa DT, SVM, dan NB memiliki nilai *F1score* yang baik dan hampir sama sehingga bisa dikatakan bahwa DT, SVM, dan NB memiliki performa yang baik berdasarkan nilai *F1score*. Selain dengan menggunakan *accuracy*, *recall*, *precision*, dan *F1score*, Sebuah model *machine learning* juga menggunakan MCC untuk evaluasi performa. Hasil dari MCC untuk setiap algoritma *machine learning* pada mata pelajaran PKn dapat dilihat pada Gambar 4.27 di bawah ini.



Gambar 4.27 Grafik Perbandingan Nilai MCC Pada Mata Pelajaran PKn.

Berdasarkan Gambar 4.27 di atas, dapat dilihat hasil nilai MCC untuk empat algoritma yang digunakan untuk prediksi pada mata pelajaran PKn. MCC merupakan metode evaluasi yang mengukur hasil prediksi model berdasarkan nilai dari *confusion matrix* dan cocok untuk model yang memiliki *class imbalanced*. Berdasarkan Gambar 4.27 di atas, dapat dilihat bahwa MCC yang paling tinggi dimiliki oleh DT. Nilai 0,55557471 merupakan nilai MCC yang paling besar dan paling mendekati nilai 1, sehingga bisa dikatakan bahwa DT adalah algoritma yang memiliki performa prediksi terbaik berdasarkan nilai MCC. NB dan SVM juga memiliki nilai MCC yang baik karena nilainya tidak berbeda jauh dengan MCC dari DT.

Berdasarkan hasil evaluasi dari akurasi, *recall*, *precision*, *F1score*, dan MCC di atas, didapatkan kesimpulan bahwa NB merupakan algoritma yang memiliki performa prediksi terbaik. Hal ini karena walaupun pada akurasi NB masih kalah dari DT dan SVM tetapi NB memiliki kestabilan antara prediksi *passed* dan *failed* yang lebih baik. Model prediksi ini bertujuan untuk membantu siswa yang gagal sehingga nilai dari prediksi *failed* harus memiliki nilai yang baik. Pada hasil perbandingan di atas NB memiliki nilai prediksi *failed* yang terbaik sehingga NB menjadi algoritma yang terbaik pada prediksi performa siswa pada mata pelajaran PKn.

Berdasarkan seluruh hasil penelitian dan evaluasi dari setiap model *machine learning* di atas, dapat diambil kesimpulan bahwa pada prediksi mata pelajaran Matematika DT, KNN, dan NB merupakan algoritma yang baik yang bisa digunakan untuk prediksi performa, sedangkan pada mata pelajaran Bahasa Inggris DT dan NB memiliki performa yang terbaik. Mata pelajaran PKn memberikan hasil yang cukup berbeda karena NB yang memiliki akurasi lebih rendah dibandingkan dengan SVM dan DT menjadi algoritma yang terbaik. Hal ini karena performa NB pada prediksi *failed* jauh lebih baik dibanding dengan DT dan SVM.

Hasil penentuan algoritma yang terbaik di atas didasarkan pada prediksi kelas *failed* yang lebih diutamakan dibanding prediksi performa nilai *passed*, tetapi jika penentuan performa berdasarkan kemampuan prediksi semua kelas, maka nilai dari MCC bisa dijadikan patokan yang lebih baik. Berdasarkan nilai MCC maka pada mata pelajaran Matematika DT, KNN, dan NB merupakan algoritma terbaik yang bisa digunakan. Berdasarkan nilai MCC pada mata pelajaran Bahasa Inggris maka DT dan NB merupakan algoritma terbaik yang bisa digunakan. Berdasarkan nilai MCC pada mata pelajaran PKn maka DT, SVM, NB merupakan algoritma terbaik yang bisa digunakan. Secara keseluruhan DT merupakan algoritma yang terbaik karena DT bisa bekerja dengan baik pada semua mata pelajaran yang diuji, yaitu mata pelajaran Matematika, Bahasa Inggris, dan PKn.

PENUTUP

5.1. Kesimpulan

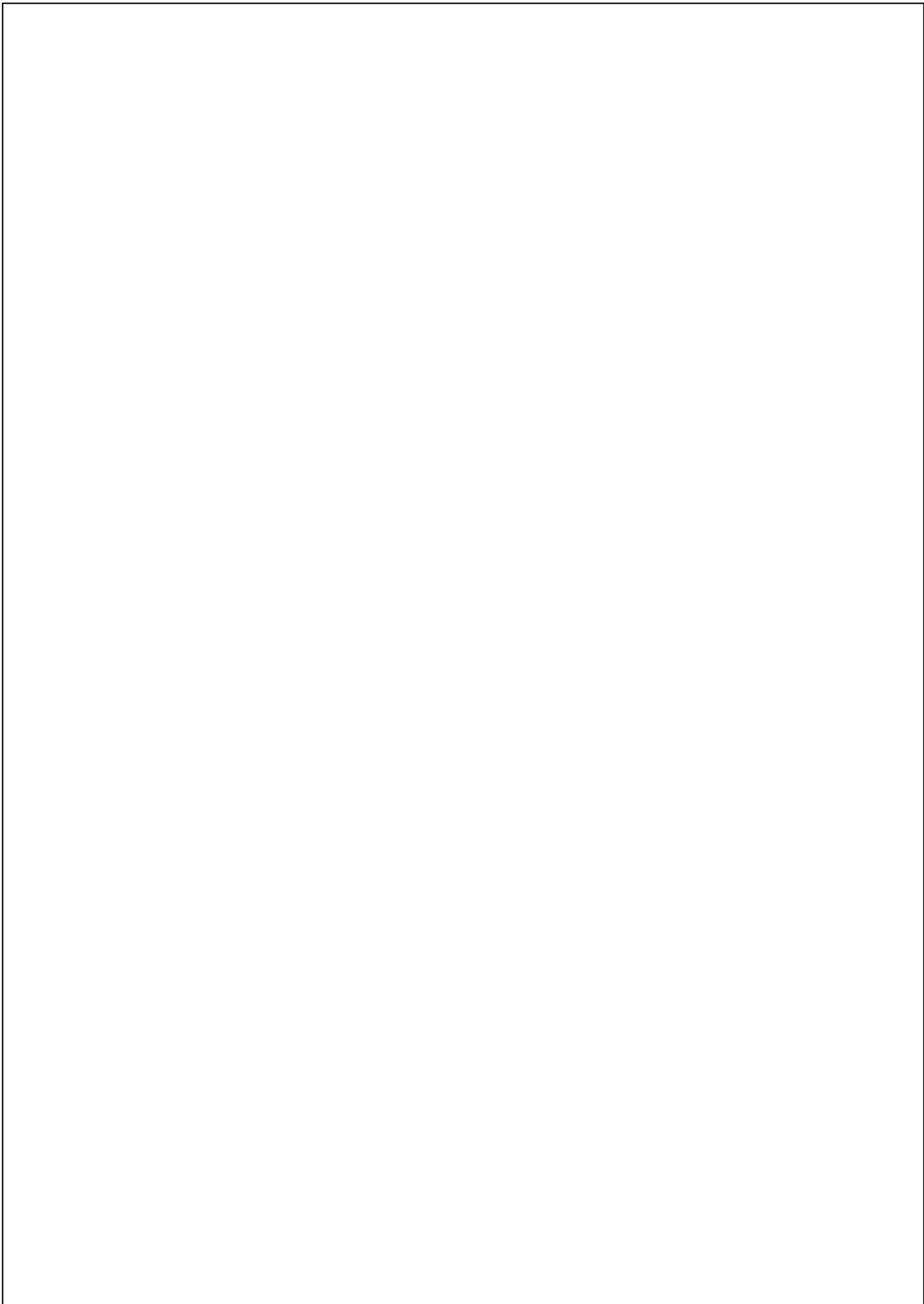
Berdasarkan kegiatan penelitian yang telah dilakukan maka dapat disimpulkan sebagai berikut:

1. Sistem prediksi performa siswa dengan menggunakan *machine learning* dapat dibuat dengan menggunakan empat algoritma *machine learning*, yaitu DT, KNN, SVM dan NB.
2. Secara keseluruhan DT memiliki performa terbaik karena memiliki performa yang stabil dan nilai evaluasi metrik seperti *accuracy*, *precision*, *recall*, *F1score*, dan MCC yang terbaik pada ketiga mata pelajaran yang diuji. DT juga mampu menangani fitur non-linear lebih baik dibandingkan ketiga algoritma lainnya sehingga walaupun menggunakan struktur data yang sama, hasil yang didapatkan oleh DT lebih baik.

5.2. Saran

Sistem yang dirancang pada penelitian ini masih memiliki beberapa kekurangan, dan karena itu saran untuk pengembangan penelitian ini adalah

1. Prediksi performa mahasiswa juga dapat dilakukan dengan menggunakan algoritma lainnya, seperti *logistic regression*, *random forest*, dan *neural network*.
2. *Dataset* serta kasus yang lebih rele



Laporan Skripsi Raphael KH Turntitin Januari 2024

ORIGINALITY REPORT

17%

SIMILARITY INDEX

16%

INTERNET SOURCES

6%

PUBLICATIONS

9%

STUDENT PAPERS

PRIMARY SOURCES

1	mbkm.unila.ac.id Internet Source	2%
2	docplayer.info Internet Source	1%
3	e-campus.iainbukittinggi.ac.id Internet Source	1%
4	repository.dinamika.ac.id Internet Source	1%
5	www.scribd.com Internet Source	1%
6	repository.its.ac.id Internet Source	1%
7	repository.uinjkt.ac.id Internet Source	1%
8	theses.uin-malang.ac.id Internet Source	<1%
9	dokumen.tips Internet Source	<1%

10	text-id.123dok.com Internet Source	<1 %
11	Submitted to Universitas Negeri Surabaya The State University of Surabaya Student Paper	<1 %
12	repository.ub.ac.id Internet Source	<1 %
13	id.123dok.com Internet Source	<1 %
14	repository.ibs.ac.id Internet Source	<1 %
15	repository.unilak.ac.id Internet Source	<1 %
16	repository.usd.ac.id Internet Source	<1 %
17	www.researchgate.net Internet Source	<1 %
18	eprints.undip.ac.id Internet Source	<1 %
19	core.ac.uk Internet Source	<1 %
20	repositori.usu.ac.id Internet Source	<1 %

21	Submitted to Forum Komunikasi Perpustakaan Perguruan Tinggi Kristen Indonesia (FKPPTKI) Student Paper	<1 %
22	journal.umpo.ac.id Internet Source	<1 %
23	digilib.uinsby.ac.id Internet Source	<1 %
24	jurnal.umj.ac.id Internet Source	<1 %
25	repository.radenintan.ac.id Internet Source	<1 %
26	Submitted to UIN Syarif Hidayatullah Jakarta Student Paper	<1 %
27	eprints.uns.ac.id Internet Source	<1 %
28	thesis.binus.ac.id Internet Source	<1 %
29	Sri Juniarsih, Eva Faja Ripanti, Enda Esyudha Pratama. "Implementasi Naive Bayes Classifier pada Opinion Mining Berdasarkan Tweets Masyarakat Terkait Kinerja Presiden dalam Aspek Ekonomi", Jurnal Sistem dan Teknologi Informasi (Justin), 2020 Publication	<1 %

30	docobook.com Internet Source	<1 %
31	repository.ar-raniry.ac.id Internet Source	<1 %
32	repository.ppns.ac.id Internet Source	<1 %
33	satriyaindra46.wordpress.com Internet Source	<1 %
34	Rismayanti Rismayanti, Fera Damayanti, Khairunnisa Khairunnisa. "Penerapan Data Mining Algoritma C4.5 dalam Menentukan Rekam Jejak Kinerja Dosen STT Harapan Medan", SinkrOn, 2018 Publication	<1 %
35	kinetik.umm.ac.id Internet Source	<1 %
36	Submitted to School of Business and Management ITB Student Paper	<1 %
37	Submitted to Universitas Negeri Malang Student Paper	<1 %
38	id.scribd.com Internet Source	<1 %
39	repository.uksw.edu Internet Source	<1 %

40	repository.unej.ac.id Internet Source	<1 %
41	repository.unhas.ac.id Internet Source	<1 %
42	www.mdpi.com Internet Source	<1 %
43	Submitted to Universitas Brawijaya Student Paper	<1 %
44	Submitted to Universitas Muria Kudus Student Paper	<1 %
45	Submitted to Universitas Pendidikan Indonesia Student Paper	<1 %
46	dspace.uii.ac.id Internet Source	<1 %
47	library.stmikgici.ac.id Internet Source	<1 %
48	Submitted to Universiti Malaysia Pahang Student Paper	<1 %
49	digilib.uin-suka.ac.id Internet Source	<1 %
50	digilib.unimed.ac.id Internet Source	<1 %
51	repositori.uin-alauddin.ac.id	

Internet Source

<1 %

52

repository.trisakti.ac.id

Internet Source

<1 %

53

Pungkas Subarkah. "Penerapan Algoritme Klasifikasi Classification And Regression Trees (Cart) Untuk Diagnosis Penyakit Diabetes Retinopathy", MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer, 2020

Publication

<1 %

54

Raluca Elena Radu, Octavian Grigorescu, Razvan Victor Rughinis. "Security News Aggregator", 2019 18th RoEduNet Conference: Networking in Education and Research (RoEduNet), 2019

Publication

<1 %

55

cheshuma.wordpress.com

Internet Source

<1 %

56

ecampus.iainbatusangkar.ac.id

Internet Source

<1 %

57

ejournal.unp.ac.id

Internet Source

<1 %

58

eprints.umm.ac.id

Internet Source

<1 %

es.scribd.com

59	Internet Source	<1 %
60	etd.iain-padangsidimpuan.ac.id Internet Source	<1 %
61	jimfeb.ub.ac.id Internet Source	<1 %
62	media.neliti.com Internet Source	<1 %
63	repository.isi-ska.ac.id Internet Source	<1 %
64	repository.library.uksw.edu Internet Source	<1 %
65	repository.paramadina.ac.id Internet Source	<1 %
66	repository.unisba.ac.id Internet Source	<1 %
67	repository.unj.ac.id Internet Source	<1 %
68	u-aizu.repo.nii.ac.jp Internet Source	<1 %
69	Ketut Jaya Atmaja, Ida Bagus Gede Anandita, Ni Kadek Ceryna Dewi. "Penerapan Data Mining Untuk Memprediksi Potensi Pendonor Darah Menjadi Pendonor Tetap	<1 %

Menggunakan Metode Decision Tree C.45", S@CIES, 2018

Publication

70	academic-accelerator.com Internet Source	<1 %
71	atimunyigr.blogspot.com Internet Source	<1 %
72	docslide.us Internet Source	<1 %
73	eprints.binadarma.ac.id Internet Source	<1 %
74	fr.scribd.com Internet Source	<1 %
75	javit.ppj.unp.ac.id Internet Source	<1 %
76	jurnal.stmikroyal.ac.id Internet Source	<1 %
77	ramadhonschoi.blogspot.com Internet Source	<1 %
78	qdoc.tips Internet Source	<1 %

Exclude quotes Off

Exclude matches Off

Exclude bibliography On

