Prediksi Mahasiswa Mengundurkan Diri Menggunakan Metode *Support Vector Machine*

1st I Gusti Ngurah Bagus Putra Adnyana
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
putraadnyana@student.telkomuniversity.ac.id

2nd Oktariani Nurul Pratiwi
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
onurulp@telkomuniversity.ac.id

3rd Irfan Darmawan
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
irfandarmawan@telkomuniversity.ac.id

***Abstrak* — Tingkat keberhasilan mahasiswa adalah salah satu cara untuk mengukur kualitas dari sebuah perguruan tinggi, dan salah satu masalah yang sering menyebabkan mahasiswa gagal adalah berhenti kuliah. Dari data yang diperoleh 8.483.213 mahasiswa terdaftar pada tahun 2020, 602.208 mahasiswa berhenti kuliah yang berasal dari perguruan tinggi swasta. Telkom University sebagai salah satu perguruan tinggi swasta akan dilakukan penelitian untuk memprediksi mahasiswa yang berhenti kuliah, terutama pada program studi S1 Sistem Informasi. Karena pada Telkom University berhenti kuliah dikategorikan sebagai mengundurkan diri, maka penelitian ini dilakukan untuk memprediksi mahasiswa mengundurkan diri atau tidak. Mengundurkan diri pada program studi S1 Sistem Informasi merupakan salah satu *key peformance indicator* yang nilainya harus dapat ditekan, oleh karena itu menggunakan machine learning dengan metode SVM dapat menyelesaikan permasalahan pada penelitian ini. Pada penelitian ini menunjukkan bahwa model SVM mendapatkan akurasi tinggi sebesar 98,30% sebelum dilakukan metode oversampling dengan SMOTE, namun menurun menjadi 92,34% setelah penerapan metode oversampling dengan SMOTE untuk mengatasi ketidakseimbangan data. Meskipun akurasinya menurun, tetapi dari nilai recall, precision, serta F1-Score meningkat yang mengindikasikan SVM setelah dilakukan oversampling lebih baik dalam mengklasifikasikan mahasiswa yang mengundurkan diri. Dengan akurasi yang tinggi, maka metode SVM terbukti efektif dalam memprediksi mahasiswa yang terindikasi mengundurkan diri atau tidak.**

Kata Kunci* - SVM, *Machine Learning*, Mahasiswa Mengundurkan Diri, Prediksi, *CRISP-DM

I. PENDAHULUAN

A. Latar Belakang

Tingkat tinggi dan rendahnya mahasiswa dapat mencerminkan kualitas dari suatu perguruan tinggi, salah satu indikator kegagalan mahasiswa adalah kasus *drop out* [1]. *Drop out* merupakan proses pencabutan status mahasiswa yang disebabkan oleh berbagai penyebab yang telah ditentukan oleh universitas [2]. Maka dari itu sangatlah penting untuk mendeteksi siswa yang memiliki resiko drop out untuk dapat mencegahnya.

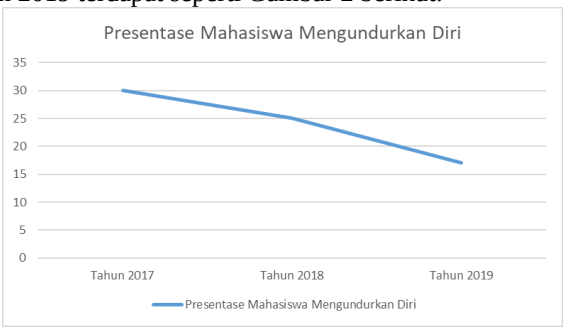
Berdasarakan data yang diperoleh dari opedata.jabar.go.id pada tahun 2020 tercatat 8.483.213 mahasiswa yang terdaftar diberbagai perguruan tinggi di Indonesia. Dari jumlah tersebut, 602.208 mahasiswa memutuskan untuk berhenti kuliah. Angka ini didominasikan oleh mahasiswa dari perguruan tinggi swasta dengan jumlah 478.826 mahasiswa atau 79,5%.

GAMBAR 1.
Jumlah *Drop Out* Pada Tahun 2020

Telkom University merupakan salah satu perguruan tinggi swasta di Provinsi Jawa Barat yang didirikan pada tahun 2013. Mengingat jumlah kasus berhenti kuliah paling tinggi berada pada perguruan tinggi swasta, maka dari itu peneliti ingin melakukan penelitian terkait dengan adanya indikasi berhenti kuliah yang berada pada Telkom University, lebih khususnya pada program studi S1 Sistem Informasi.

Menurut Kepala Program Studi S1 Sistem Informasi, mahasiswa-mahasiswa yang berhenti kuliah pada Telkom University disebut dengan mengundurkan diri atau *resign*. Terdapat beberapa alasan yang menyebabkan mahasiswa diharuskan untuk mengundurkan diri, seperti mata kuliah tingkat 1 (semester 1, 2) dan tingkat 2 (semester 3, 4) maksimal harus diselesaikan pada semester 4 jika tidak dapat diselesaikan pada semester 4 maka mahasiswa yang bersangkutan dipaksa untuk mengundurkan diri, jika indeks prestasi kumulati (IPK) pada tingkat 1 kurang dari 2 maka diharuskan untuk mengundurkan diri, tidak melakukan registrasi sebanyak dua kali, menerima pelanggaran etik dalam kategori berat, masa studi melebihi 12 semester, dan mengundurkan diri karena alasan pribadi.

Presentase mahasiswa mengundurkan diri pada Program Studi S1 Sistem Informasi pada rentang tahun 2017 sampai dengan 2019 terdapat seperti Gambar 2 berikut:



GAMBAR 2.
Presentase Mahasiswa Mengundurkan Diri

Dari grafik pada Gambar 2 diatas dapat dilihat bahwa jumlah mahasiswa yang mengundurkan diri dari tahun 2017 sampai dengan 2019 mengalami penurunan, yang awalnya pada tahun 2017 memiliki jumlah sebanyak 30 mahasiswa menurun menjadi 25 mahasiswa pada tahun 2018 dan kemudian menurun lagi pada tahun 2019 menjadi 17 mahasiswa. Angka ini menunjukan bahwa jumlah mahasiswa mengundurkan diri pada program studi S1 Sistem Informasi ini mengalami penurunan selama 3 tahun terakhir.

Penelitian ini dilakukan karena jumlah mahasiswa mengundurkan diri mempengaruhi *Key Performance Indicator* (KPI) program studi yang erat kaitannya dengan akreditasi. Untuk memprediksi mahasiswa yang berpotensi mengundurkan diri, penerapan machine learning, khususnya metode *Support Vector Mechine* (SVM) sangatlah releven atau berpengaruh. Metode SVM dipilih karena memiliki akurasi yang tinggi dalam memprediksi mahasiswa yang *drop out*, seperti permasalahan dalam penelitian terkait [1] dengan hasil penelitian tersebut menunjukan bahwa SVM mampu memberikan prediksi dengan tingkat akurasi 98,06% dan error yang rendah. Sehingga dengan menggunakan *machine learning* khususnya metode *Support Vector Machine* (SVM) dapat membantu dalam memprediksi mahasiswa yang berpotensi menggundurkan diri yang sudah terbukti efektif berdasarkan penelitian sebelumnya yang mendapatkan hasil yang efektif dengan nilai akurasi sebesar 98.06% serta nilai error yang kecil.

B. Rumusan Masalah

Rumusan masalah yang mendasari penelitian ini adalah:

1. Bagaimana implementasi metode *support vector machine* untuk memprediksi mahasiswa mengundurkan diri?
2. Bagaimana tingkat akurasi *metode support vector machine* untuk memprediksi mahasiswa mengundurkan diri?

C. Tujuan Tugas Akhir

Berikut merupakan tujuan dari penelitian ini:

1. Membuat implementasi program dengan menggunakan metode *Support Vector Machine* (SVM) untuk memprediksi mahasiswa mengundurkan diri
2. Menganalisis hasil dan mengevaluasi model dari penerapan metode *Support Vector Machine* (SVM) dalam memprediksi mahasiswa mengundurkan diri

D. Manfaat Tugas Akhir

Manfaat penelitian ini:

1. Bagi mahasiswa Program Studi S1 Sistem Informasi penelitian ini diharapkan mampu membantu untuk mengetahui apakah mereka terindikasi mengundurkan diri atau tidak.
2. Bagi Program Studi S1 Sistem Informasi penelitian ini diharapkan dapat membantu untuk mengetahui mahasiswa yang terindikasi mengundurkan diri atau tidak

II. KAJIAN TEORI

A. *Data Mining*

Data Mining adalah proses menemukan pola, model, dan jenis pengetahuan lain yang menarik dalam kumpulan data yang besar. Data mining melibatkan penggunaan alat analisis data canggih untuk menemukan pola dan hubungan yang sebelumnya tidak diketahui. Alat-alat ini dapat mencakup model statistik, algoritma matematis, dan metode pembelajaran mesin algoritma yang meningkatkan kinerjanya secara otomatis melalui pengalaman.

B. *Machine Learning*

Machine learning atau pembelajaran mesin adalah cabang dari kecerdasan buatan (*artificial intelligence*) yang berfokus pada pengembangan algoritma dan model dalam prediksi. *Machine learning* atau pembelajaran mesin adalah cabang dari kecerdasan buatan (*artificial intelligence*) yang berfokus pada pengembangan algoritma dan model [3].

C. *Supervised Learning*

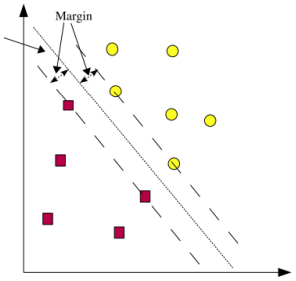
Supervised learning adalah metode dalam machine learning di mana algoritma dilatih menggunakan data yang telah diberi label. Supervised learning sangat umum digunakan dalam berbagai aplikasi, seperti pengenalan gambar dan prediksi hasil berdasarkan data historis [3].

D. Klasifikasi

Klasifikasi adalah pembelajaran mesin yang termasuk ke dalam *supervised learning*. Tujuan dari klasifikasi adalah untuk membangun sebuah pemodelan yang dapat memprediksi kelas atau label dari data baru berdasarkan atribut – atribut prediktor yang ada serta klasifikasi dibagi menjadi 2 tahap yaitu fase pelatihan (*training*) dan fase pengujian (*testing*) [4].

E. *Support Vector Machine* (SVM)

Konsep *support vector machine* ini merupakan usaha untuk mencari garis pemisah optimal atau *hyperplane* yang digunakan untuk memisahkan dua class, dua class ini disimbolkan menjadi +1 dan -1 [5]



GAMBAR 3.
Support Vector Machine

Garis …. pada Gambar 3 merepresentasikan hyperplane atau pemisan antara kedua kelas, garis --- merepresentasikan margin atau jarak antara support vector terdekat dengan hyperplane. Hyperlane didefinisikan sebagai berikut:

$$w.\ x_i + b = 0 \ldots (1)$$

Data xi yang tergolong ke dalam class negative adalah yang class yang memenuhi pertidaksamaan.

$$w.\ x_i + b \leq -1 \ldots . (2)$$

Serta yang tergolong ke class positive adalah yang memenuhi pertidaksamaan [5]

$$w.\ x_i + b \geq -1 \ldots . (3)$$

Keterangan:
w: Vector bobot (*Weight Vector*)
x_i: Vector fitur dari suatu titik data
b: Nilai pergeseran atau bias

Pada SVM terdapat sebuah metode untuk melakukan transformasi data, untuk melakukan transformasi ini diperlukan sebuah *Kernel Fuction*. Berikut adalah beberapa jenis kernel [6]:

a. *Linear Kernel*
b. *Polynomial Kernel Fuction*
c. *Radial Basis Function*
d. *Sigmoid Fuction*

Dalam SVM terdapat sebuah *hyperparameter*, *hypermarameter* merupakan parameter yang nilainya ditetapkan sebelum proses pelatihan model dan digunakan untuk mengontrol proses pelatihan. Berikut merupakan *hyperparameter* pada SVM:

1. C (Regularization parameter)
   Parameter C mengontrol trade-off antara besar margin dan klasifikasi yang benar dari titik data pelatihan [7].
2. Kernel
   Fungsi kernel menentukan ruang fitur tempat data akan diproyeksikan. Beberapa jenis kernel yang umum digunakan dalam SVM adalah linear, polynomial, Radial Basis Function (RBF), dan sigmoid [7].
3. Gamma
   Gamma adalah parameter untuk kernel RBF yang menentukan seberapa jauh pengaruh dari satu titik data [8].
4. Degree
   Degree adalah parameter untuk kernel polynomial yang menentukan derajat dari polynomial [9].

F. Grid Search

Grid Search merupakan metode pencarian hyperparameter yang dirancang untuk menemukan kombinasi *hyperparameter* optimal dalam ruang hyperparameter yang sudah ditentukan sehingga model yang dibuat dapat memprediksi data yang tidak dikenal dengan akurasi tinggi [10]. Dalam *support vector machine grid search* bekerja dengan cara mengoptimalkan *hyperparameter* seperti C, *gamma*, *degree*, dan kernel dengan menggunakan teknik *cross validation*. *Cross validation* digunakan untuk mengevaluasi peforma dari setiap hyperparameter yang diujikan sehingga dapat mengurangi overfitting [10].

G. CRISP-DM

CRISP-DM merupakan salah satu *framework* data mining yang merupakan singkatan dari *Cross-Industry Standard Process for Data Mining* yang memiliki enam fase berulang – ulang yaitu tahap *business understanding*, *data understanding*, *data preparation*, *modeling*, *evaluation* dan *deployment* [11].

H. Python

Python ini merupakan bahasa pemograman yang beorientasi pada objek dan diinterpretasikan, yang berarti kodenya langsung dieksekusi oleh interpreter tanpa perlu dikompilasi menjadi kode mesin terlebih dahulu. *Python* banyak memiliki *framework* dan pustaka-pustaka yang memudahkan dalam pengembangan aplikasi di berbagai bidang, seperti *matplotlib*, *pandas*, *NumPy*, dan *SciPy* [12].

I. Google Colab

Google Colab atau *colaboratory* merupakan produk dari *google research*. *Goolgle colab* memungkinkan penggunanya untuk menulis dan menjalakan kode *python* langsung melalui *browser* dan sangat cocok untuk *machine learning* serta analisis data [13].

J. Imbalance Handling

Imbalance handling merupakan teknik yang digunakan untuk mengatasi masalah dalam *dataset* yang tidak seimbang, yang mana ketika satu kelas memiliki jumlah contoh yang lebih banyak dibandingkan kelas lainnya. Tahap ini dibagi menjadi 2 metode yaitu *oversampling* dan *undersampling* [14].

K. Oversampling

Oversampling merupakan proses meningkatkan jumlah contoh atau sampel dari kelas minoritas dengan cara menghasilkan sampel baru atau mengulangi beberapa sample yang sudah ada [15].

L. *Synthetic Minority Over-sampling Technique* (SMOTE)

SMOTE adalah sebuah metode untuk menangani ketidakseimbangan kelas dalam dataset dengan cara oversampling kelas minoritas dengan membuat contoh replika atau sintetik [16].

M. Confusion Matrix

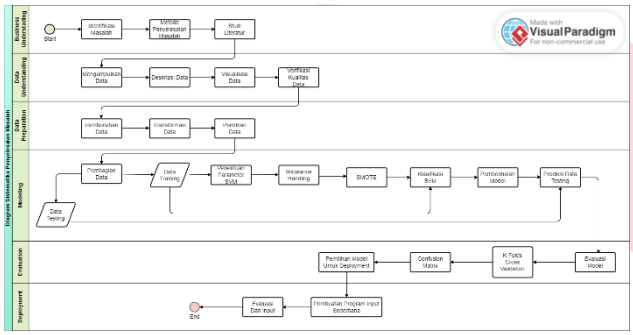
Confusion matrix digunakan untuk mengevaluasi kinerja dari model klasifikasi dengan cara menunjukan *true positive* (TP), *true negative* (TN), *false positive* (FP), false negative (FN) dan juga ukuran yang digunakan untuk menilai klasifikasi prediksi seperti *accuracy*, *recall*, *precision*, *f1-score* [17].

N. K-Folds Cross Validation

K-Folds Cross Validation merupakan metode untuk melakukan validasi yang membagi dataset menjadi bagian K (*folds*) yang sama besar. Metode ini bertujuan untuk memberikan estimasi yang lebih akurat mengenai *peforma* model [18].

III. METODE

Pada bagian ini dijelaskan mengenai kerangka pemecahan masalah yang digunakan, membahas sistematika penyelesaian masalah, pengumpulan data, pengolahan data atau proses pengembangan produk/ artifak, serta membahas metode evaluasi yang digunakan.



GAMBAR 4.
Diagram Sistematika Penyelesaian Masalah

A. Pengumpulan Data

Penelitian ini menggunakan data primer yang didapatkan dari pihak Program Studi S1 Sistem Informasi. Data ini berisikan informasi - informasi terkait dengan status mahasiswa dari program studi S1 Sistem Informasi.

B. Pengolahan dan Pengembangan Data

Pengolahan data didalam penelitian ini dilakukan dari tahap *data understaing* sampai dengan *data preparation*. *Data understaing* dilakukan untuk menggali informasi lebih dalam mengenai data dan mencari pengetahuan awal yang mungkin terkandung di dalamnya, sedangkan *data preparation* dilakukan pembersihan pada data yang memiliki kualitas data buruk, karena data yang buruk ini dapat mempengaruhi proses *modeling*.

C. Metode Evaluasi

Pada tahap ini dilakukan evaluasi performa model berdasarkan pengujian model untuk melihat sejauh mana kemampuannya memprediksi mahasiswa mengundurkan diri. Selanjutnya dilakukan *confusion matrix* untuk mengukur akurasi, presisi, *recall*, dan *F1-score* yang digunakan mengukur performa model. Sedangkan *K-Folds Cross Validation* tahap yang digunakan Metode ini berguna untuk validasi yang bekerja dengan cara membagi dataset menjadi bagian K (*folds*) yang sama besar. Metode ini bertujuan untuk memberikan estimasi yang lebih akurat mengenai peforma model.

IV. HASIL DAN PEMBAHASAN

A. *Business Understanding*

Dalam fase ini akan berfokus pada pemaparan kebutuhan yang diperlukan untuk merancang model yang dapat memprediksi mahasiswa yang terindikasi akan mengundurkan diri.

B. *Data Understanding*

Pada fase ini fokus untuk mengumpulkan, mengidentifikasi serta menelaah data mengenai mahasiswa S1 Sistem Informasi.

1. Pengumpulan Data

Pada tahap pengumpulan data, akan dilakukan pengambilan data ke pihak program studi S1 Sistem Informasi

2. Deskripsi Data

Pada tahap ini data yang sudah diambil sebelumnya pada tahap pengumpulan data akan dideskripsikan. Data yang diambil merupakan data akademik dan demografis mahasiswa dengan total 1186 baris dan 49 kolom dengan

3. Verifikasi Kualitas Data

Pada fase ini akan dilakukan pemeriksaan terkait dengan kualitas data yang mencakup kelengkapan dan konsistensi dari data yang bertujuan untuk memastikan bahwa data yang digunakan dalam pemodelan sudah memiliki kualitas yang baik

C. *Data Preparation*

1. Pembersihan Data

Dilakukan pembersihan pada variabel-variabel data terkait dengan kualitas data seperti mengatasi nilai yang hilang dengan menghapus atau menggantinya, memperbaiki data yang tidak konsisten, menghapus data yang terduplikasi, dan mengatasi data yang terdapat kesalahan pada saat pengambilan data seperti terdapat kesalahan penulisan, atau ketidaksesuaian format.

2. Transformasi Data

Melakukan modifikasi, perubahan atau membangun kembali data dengan tujuan untuk meningkatkan kualitas data seperti dengan cara melakukan normalisasi data dengan mengubah tipe data.

3. Pemilihan Data (*Feature Selection*)

Melakukan pemilihan data yang relevan sesuai dengan tujuan untuk memprediksi mahasiswa yang terindikasi mengundurkan diri dengan cara menghapus variabel – variabel yang tidak ada keterkaitan dengan variabel target dan melakukan pengecekan dengan menggunakan *correlation matrix*.

D. *Modeling*

1. Pembagian Data

Tahap ini menerapkan metode *k-folds cross validation* yang bekerja dengan cara membagi dataset menjadi 2 bagian, yaitu data pelatihan (*data training*) yang digunakan untuk melatih model dan data pengujian (*data testing*) untuk menguji kinerja model yang dibuat. Maka pembagian dataset dilakukan dengan rasio 80:20 yang berarti 80% dari total dataset akan digunakan untuk data pelatihan, dan 20% sebagai data pengujian.

2. Penentuan Parameter

Tahapan ini dilakukan untuk mendapatkan hasil yang optimal pada parameter *karnel*, C, *gamma*, dan *degree*. Seperti *grid search* dengan menggunakan pustaka sklearn pada *python* dengan mengimport *GridSearchCV*. Berikut merupakan Hasil dari *GridSearch* yang dilakukan.

TABEL 1
Hasil Gridsearch

Hasil *GridSearch*			
Kernel	**C**	***Gamma***	***Degree***
Polynomial	0.1	0.01	5

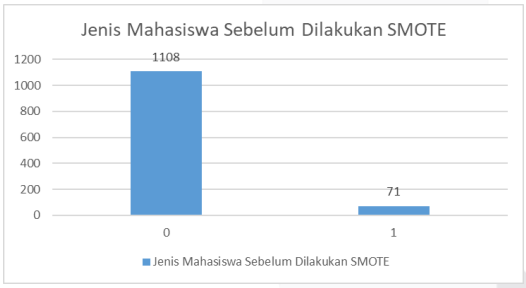
Maka parameter performa terbaik adalah *kernel polynomial* dengan nilai C sebesar 0,1 dan nilai *degree* sebesar 5. Sedangkan *gamma* tidak akan digunakan karena tidak relevan dengan tipe *kernel polynomial*.

3. Proses Klasifikasi SVM Sebelum *Oversampling*

Tahapan ini melakukan pembuatan model metode SVM menggunakan pustaka *sklearn* pada *python* dengan mengimport SVC. Parameter C dengan nilai sebesar 0.1 dan parameter degree sebesar 5. *Data training* sebanyak 80% (0.8) dan *data testing* sebanyak 20% (0.2) serta akan dilakukan *K-folds cross validation* sebanyak K = 5.

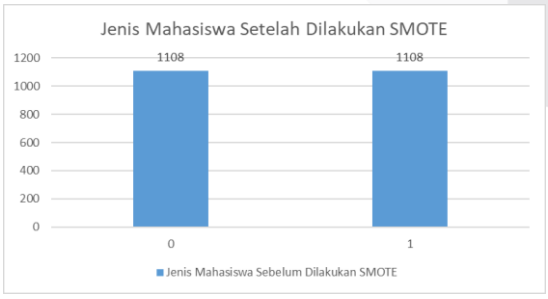
4. Proses Klasifikasi SVM Setelah *Oversampling*

Pada variabel "Jenis Mahasiswa" terdapat *Imbalance Data* yang jumlah nilai mahasiswa mengundurkan diri dan tidak mengundurkan diri yang sangat berbeda, yaitu 1108 untuk mahasiswa kelas 0 (Tidak Mengundurkan Diri) dan 71 mahasiswa kelas 1 (Mengundurkan Diri), maka dari itu akan dilakukan *oversampling* untuk menganani *imbalance* data ini. Berikut merupakan hasil variabel "Jenis Mahasiswa" sebelum dan setelah dilakukan *oversampling* menggunakan metode SMOTE.



GAMBAR 5.
Jenis Mahasiswa Sebelum Dilakukan Smote

Selanjutnya melakukan klasifikasi variabel "Jenis Mahasiswa" setelah dilakukan SMOTE.



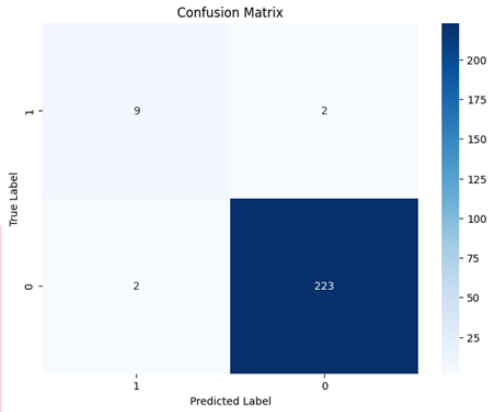
GAMBAR 6.
Jenis Mahasiswa Setelah Dilakukan Smote

Setelah dilakukan SMOTE, dari kelas 0 (Tidak Mengundurkan Diri) dan kelas 1 (Mengundurkan Diri) berada pada jumlah yang sama yaitu sebanyak 1108 data. SMOTE ini dilakukan dengan cara memanggil *toolbox imblearn* pada *python* kemudian mengimport SMOTE.

Kemudian setelah dilakukan SMOTE, maka akan dilakukan pembuatan model metode SVM dengan parameter – paramter dan *data splitting* seperti sebelumnya.

E. Evaluasi Hasil Pemodelan Sebelum SMOTE

1. *Confusion Matrix* Sebelum SMOTE



GAMBAR 7.
Confusion Matrix Sebelum Smote

Didapatkan visualisasi dari *confusion matrix* yang belum dilakukan SMOTE, untuk nilai *True Positive* (TP) sebanyak 9, *False Positive* (FP) sebanyak 2, *True Negative* (TN) sebanyak 223, dan *False Negative* (FN) sebanyak 2. Dari hasil tersebut berarti 9 data (TP) yang merupakan kelas 1 atau mengundurkan diri dapat diprediksi oleh pemodelan dengan benar dan 2 data (FP) yang seharusnya kelas 1 atau mengundurkan diri diprediksi salah oleh pemodelan, selanjutnya terdapat 223 data (TN) pada kelas 0 atau tidak mengundurkan diri yang dapat diprediksi oleh pemodelan dengan benar dan 2 data (FN) yang seharusnya kelas 0 atau tidak mengundurkan diri diprediksi salah oleh pemodelan.

Setelah dihitung nilai *data testing* yang diprediksi, maka selanjutnya akan dihitung untuk nilai *accuracy*, *recall*, *precision* dan *F-1 Score*.

$$Accuracy = \frac{TP+TN}{P+N} = \frac{9+223}{9+2+223+2} = \frac{232}{236} = 0{,}9830 = 98{,}30\%$$

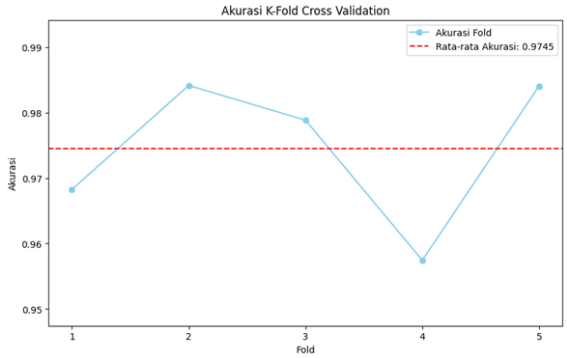
$$Recall = \frac{TP}{TP+FN} = \frac{9}{9+2} = \frac{9}{11} = 0{,}8181 = 81{,}81\%$$

$$Precision = \frac{TP}{TP+FP} = \frac{9}{9+2} = \frac{9}{11} = 0{,}8181 = 81{,}81\%$$

$$F\text{-}1 \quad Score = 2x\frac{Precision\ x\ Recall}{Precision+Recall} = 2x\ \frac{0{,}8181\ x\ 0{,}8181}{0{,}8181+\ 0{,}8181} = 2x\ \frac{0{,}6692}{1{,}6362} = 0.8177 = 81{,}77\%$$

2. *K-Fold Cross Validation* Sebelum SMOTE

Melakukan gambaran umum tentang performa model, sehingga memberikan evaluasi yang lebih stabil dan akurat.



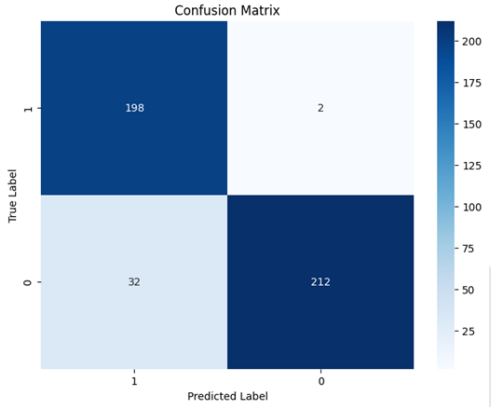
GAMBAR 8.
K-Fold Cross Validation Sebelum Smote

Hasil dari *K-fold cross validation* yang dimana pada *fold* pertama atau k = 1 didapatkan hasil sebesar 96,83% (0.9683), pada *fold* kedua atau k = 2 didapatkan hasil sebesar 98,41% (0.9841), pada *fold* ketiga atau k = 3 didapatkan hasil sebesar 97,88% (0.9788), pada *fold* keempat atau k = 4 didapatkan hasil sebesar 95,74% (0.9574), dan pada *fold* terkahir yaitu *fold* 5 didapatkan hasil sebesar 98,40% (0.9840). Dari ke 5 *fold* yang sudah dilakukan pengujian didapatkan rata – rata sebesar 97,45% (0.9745) yang model memiliki peforma yang sangat baik dan konsisten dalam memprediksi data pada berbagai *fold* yang diuji.

F. Evaluasi Hasil Pemodelan Setelah SMOTE

1. *Confusion Matrix* Setelah SMOTE

Berikut merupakan *confusion matrix* sesudah dilakukan *oversampling*.



GAMBAR 9.
Confusion Matrix Setelah Smote

Didapatkan visualisasi dari *confusion matrix* yang sudah dilakukan SMOTE, untuk nilai untuk *True Positive* (TP) sebanyak 198, *False Positive* (FP) sebanyak 2, *True Negative* (TN) sebanyak 212, dan *False Negative* (FN) sebanyak 32. Dari hasil tersebut berarti 198 data (TP) yang merupakan kelas 1 atau mengundurkan diri dapat diprediksi oleh pemodelan dengan benar dan 2 data (FP) yang seharusnya kelas 1 atau mengundurkan diri diprediksi salah oleh pemodelan, selanjutnya terdapat 212 data (TN) pada kelas 0 atau tidak mengundurkan diri yang dapat diprediksi oleh pemodelan dengan benar dan 32 data (FN) yang seharusnya 0 atau tidak mengundurkan diri diprediksi salah oleh pemodelan.

Setelah dihitung nilai data testing yang diprediksi, maka selanjutnya akan dihitung untuk nilai *accuracy*, *recall*, *precision* dan *F-1 Score*.

$$Accuracy = \frac{TP+TN}{P+N} = \frac{198+212}{198+2+212+32} = \frac{410}{444} = 0{,}9234 = 92{,}34\%$$

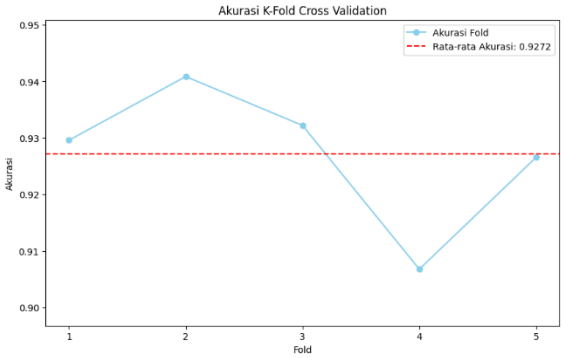
$$Recall = \frac{TP}{TP+FN} = \frac{198}{198+32} = \frac{198}{230} = 0{,}86 = 86\%$$

$$Precision = \frac{TP}{TP+FP} = \frac{198}{198+2} = \frac{198}{200} = 0{,}99 = 99\%$$

$$F\text{-}1 \quad Score = 2x\frac{Precision\ x\ Recall}{Precision+Recall} = 2x\ \frac{0{,}99\ x\ 0{,}86}{0{,}99+\ 0{,}86} = 2x\ \frac{0{,}8514}{1{,}85} = 0.92 = 92\%$$

2. *K-Fold Cross Validation* Setelah SMOTE

Melakukan gambaran umum tentang performa model, sehingga memberikan evaluasi yang lebih stabil dan akurat.



GAMBAR 10.
K-Fold Cross Validation Setelah Smote

Hasil dari *K-fold cross validation* yang dimana pada *fold* pertama atau k = 1 didapatkan hasil sebesar 92,96% (0.9296), pada *fold* kedua atau k = 2 didapatkan hasil sebesar 94,08% (0.9408), pada *fold* ketiga atau k = 3 didapatkan hasil sebesar 93,22% (0.9322), pada *fold* keempat atau k = 4 didapatkan hasil sebesar 90,68% (0.9068), dan pada *fold* terkahir yaitu *fold* 5 didapatkan hasil sebesar 92,66% (0.9266). Dari ke 5 fold yang sudah dilakukan pengujian didapatkan rata – rata sebesar 92,72% (0.9272) yang model memiliki peforma yang sangat baik dan konsisten dalam memprediksi data pada berbagai *fold* yang diuji.

TABEL 2.
Evaluasi Model

	Accuracy	*Precision*	*Recall*	*F1-Score*	*K – Folds Cross Validation* (rata – rata)
Sebelum SMOTE	98,30%	81,81%	81,81%	81,77%	97,45%
Setelah SMOTE	92,34%	99%	86%	92%	92,72%

Maka, didapatkan bahwa peforma model lebih baik sebelum dilakukan *oversampling*, namun jumlah kelas 1 (mengundurkan diri) pada variable target sangatlah signifikan perbedaanya yaitu sebanyak 71 data untuk kelas 1 (mengundurkan diri) dan sebanyak 1108 data untuk kelas 0 (tidak mengundurkan diri).

Setelah melakukan *oversampling* dengan menggunakan metode SMOTE perbedaan pada kelas 1 (mengundurkan diri) dan kelas 0 (tidak mengundurkan diri) dapat diatasi dengan membuat replika atau data sintetis dari data kelas 1 (mengundurkan diri), sehingga untuk data training yang dipelajari oleh model dapat bertambah yang membuat model

dapat memprediksi lebih baik terkait dengan kelas 1 (mengundurkan diri) yang sesuai dengan tujuan penelitian ini.

Meskipun dari nilai *accuracy model* menurun setelah dilakukan *oversampling*, nilai *precision*, *recall* serta *F1-score* meningkat yang mengindikasikan bahwa pemodelan dapat lebih baik dalam mendeteksi kelas 1 (mengundurkan diri) dan mengurangi kesalahan dalam prediksi kelas 1 (mengundurkan diri). Dari hasil rata – rata kedua *k-folds cross validation* sebelum dilakukan SMOTE dan sesudah dilakukan SMOTE didapatkan rata – rata yang mirip dengan nilai akurasi yang menandakan peforma yang sangat baik dan konsisten dalam memprediksi data pada berbagai fold yang diuji. Oleh karena itu pemodelan yang telah melalui proses *oversampling* akan digunakan untuk pembuatan program input sederhana yang dapat memprediksi input dari mahasiswa, untuk mengetahui apakah mahasiswa tersebut terindikasi mengundurkan diri atau tidak mengundurkan diri

G. Program Input Sederhana

Pada tahap ini, akan dibuat program input sederhana yang dapat memprediksi mahasiswa terindikasi mengundurkan diri atau tidak mengundurkan diri. Program input sederhana ini bekerja dengan cara memprediksi data baru yang diinputkan oleh mahasiswa kemudian diklasifikasikan oleh model yang sebelumnya sudah dibuat.

V. KESIMPULAN

A. Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka didapatkan kesimpulan sebagai berikut:

1. Metode *Support Vector Machine* (SVM) dalam penelitian ini menunjukkan performa yang sangat baik dengan akurasi mencapai 98,30%. Namun, akurasi tersebut dicapai sebelum penanganan ketidakseimbangan data pada variabel target. Setelah menerapkan teknik *oversampling* untuk mengatasi ketidakseimbangan data, akurasi model menurun menjadi 92,34%. Meskipun terjadi penurunan dalam akurasi setelah *oversampling*, nilai akurasi ini masih sangat tinggi. Dengan menggunakan metode *oversampling*, model SVM menjadi lebih efektif dalam memprediksi mahasiswa yang berisiko mengundurkan diri, menunjukkan perbaikan dalam kemampuan klasifikasinya terhadap mahasiswa mengundurkan diri.
2. Implementasi metode *Support Vector Machine* (SVM) untuk memprediksi mahasiswa yang terindikasi mengundurkan diri dilakukan dengan mengembangkan program input sederhana. Program ini dirancang untuk menerima input dari mahasiswa dan kemudian menggunakan model SVM yang telah dibuat untuk melakukan prediksi. Berdasarkan input yang diberikan, program akan menghasilkan hasil prediksi yang menunjukkan apakah mahasiswa tersebut terindikasi mengundurkan diri atau tidak mengundurkan diri. Dengan cara ini, program input sederhana ini dapat membantu dalam mendeteksi mahasiswa yang berpotensi mengundurkan diri dan memungkinkan tindakan pencegahan yang lebih awal.

B. Saran

Adapun saran untuk penelitian selanjutnya:

1. Penelitian ini hanya mengkategorikan apakah mahasiswa terindikasi mengundurkan diri atau tidak mengundurkan diri, disarankan untuk penelitian selanjutnya dapat lebih memberikan informasi lebih rinci kepada mahasiswa seperti pada semester berapa mahasiswa tersebut akan berpotensi mengundurkan diri.
2. Melakukan perancangan aplikasi yang dapat memprediksi mahasiswa yang terindikasi mengundurkan diri dengan lebih optimal, bukan hanya input sederhana.

REFERENSI

[1] S. Nurhayati dan E. T. Luthfi, *"Prediksi Mahasiswa Drop Out Menggunakan Metode Support Vector Machine," Sisfotenika,* vol. 5, no. 1, hlm. 82–93, 2015.

[2] S. D. Purba, L. Harahap, dan J. F. R. Panggabean, *"Prediction of Students Drop Out with Support Vector Machine Algorithm," Jurnal Mantik, vol. 6, no. 1, hlm. 582–586, 2022.*

[3] V. Nasteski, *"An overview of the supervised machine learning methods," Horizons. b, vol. 4, no. 51–62, hlm. 56, 2017.*

[4] S. Neelamegam dan E. Ramaraj, *"Classification algorithm in data mining: An overview," International Journal of P2P Network Trends and Technology (IJPTT), vol. 4, no. 8, hlm. 369–374, 2013.*

[5] A. S. Nugroho, "Pengantar *support vector machine," J. Data Mining*, Jakarta, hlm. 3, 2007.

[6] A. Patle dan D. S. Chouhan, *"SVM kernel functions for classification," dalam 2013 International Conference on Advances in Technology and Engineering (ICATE), 2013, hlm. 1–9. doi: 10.1109/ICAdTE.2013.6524743.*

[7] A. Rosales-Pérez, H. J. Escalante, J. A. Gonzalez, C. A. Reyes-Garcia, dan C. A. Coello Coello, *"Bias and Variance Multi-objective Optimization for Support Vector Machines Model Selection," dalam Pattern Recognition and Image Analysis, J. M. Sanches, L. Micó, dan J. S. Cardoso, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, hlm. 108–116.*

[8] M. Fajri dan A. Primajaya, *"Komparasi Teknik Hyperparameter Optimization pada SVM untuk Permasalahan Klasifikasi dengan Menggunakan Grid Search dan Random Search," Journal of Applied Informatics and Computing, vol. 7, no. 1, Jul 2023, doi: 10.30871/jaic.v7i1.5004.*

[9] C. Kim dan H. Kim, *"A Hybrid Deep Q-Network for the SVM Lagrangian," dalam Information Science and Applications 2018, K. J. Kim dan N. Baek, Ed., Singapore: Springer Singapore, 2019, hlm. 643–651.*

[10] I. Syarif, A. Prugel-Bennett, dan G. Wills, *"SVM parameter optimization using grid search and genetic algorithm to improve classification performance," TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 14, no. 4, hlm. 1502–1509, 2016.*

[11] C. Schröer, F. Kruse, dan J. M. Gómez, *"A Systematic Literature Review on Applying CRISP-*

DM Process Model," Procedia Comput Sci, vol. 181, hlm. 526–534, 2021, doi: https://doi.org/10.1016/j.procs.2021.01.199.

[12] A. J. Dhruv, R. Patel, dan N. Doshi, *"Python: the most advanced programming language for computer science applications," Science and Technology Publications, Lda*, hlm. 292–299, 2021.

[13] P. Naik, G. Naik, dan M. Patil, *"Conceptualizing Python in Google COLAB," India: Shashwat Publication, 2022.*

[14] S. Kotsiantis, D. Kanellopoulos, dan P. Pintelas, *"Handling imbalanced datasets: A review," GESTS international transactions on computer science and engineering, vol. 30, no. 1*, hlm. 25–36, 2006.

[15] R. Mohammed, J. Rawashdeh, dan M. Abdullah, *"Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results," dalam 2020 11th International Conference on Information and Communication Systems (ICICS), 2020*, hlm. 243–248. doi: 10.1109/ICICS49469.2020.239556.

[16] N. V Chawla, K. W. Bowyer, L. O. Hall, dan W. P. *Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," Journal of artificial intelligence research, vol. 16, hlm. 321–357, 2002.*

[17] J. Han, J. Pei, dan H. Tong, *Data mining: concepts and techniques. Morgan kaufmann, 2022.*

[18] J. D. Rodriguez, A. Perez, dan J. A. Lozano, *"Sensitivity Analysis of k-Fold Cross Validation in Prediction Error Estimation," IEEE Trans Pattern Anal Mach Intell*, vol. 32, no. 3, hlm. 569–575, 2010, doi: 10.1109/TPAMI.2009.187.