Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI: 10.25126/jtiik.202296737
Vol. 9, No. 7, Desember 2022, hlm. 1341-1348 p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 36/E/KPT/2019 e-ISSN: 2528-6579

STUDI KOMPARASI NAIVE BAYES, K-NEAREST NEIGHBOR, DAN RANDOM FOREST UNTUK PREDIKSI CALON MAHASISWA YANG DITERIMA ATAU MUNDUR

Puteri Sejati[1], Munawar*[2], Marzuki Pilliang[3], Habibullah Akbar[4]

[1,2,3,4]Universitas Esa Unggul, Jakarta Barat
Email: [1]puterisejati@esaunggul.ac.id, [2]moenawar@gmail.com, [3]marzuki.pilliang@ieee.org, [4]habibullah.akbar@esaunggul.ac.id
*Penulis Korespondensi

(Naskah masuk: 01 Desember 2022, diterima untuk diterbitkan: 26 Desember 2022)

Abstrak

Penelitian ini bertujuan untuk mendapatkan model prediksi terbaik dari data Penerimaan Mahasiswa Baru tahun 2014 hingga 2019 dengan membandingkan Naive Bayes, K-Nearest Neighbor, dan Random Forest. Penelitian ini menggunakan metode klasifikasi untuk memprediksi calon mahasiswa. Mereka diterima atau mundur. Dalam penelitian ini digunakan 19.603 data latih dan 4.901 data uji. Hasil penelitian menunjukkan bahwa algoritma Random Forest adalah yang terbaik dengan akurasi 73,61%, dibandingkan dengan K-Nearest Neighbor dengan akurasi 72,08%, dan Naive Bayes dengan akurasi 70,47%. Disimpulkan juga bahwa optimasi model dengan teknik *Hyperparameter* menghasilkan nilai akurasi yang lebih baik. Hasil penelitian ini dapat digunakan untuk mendukung bagian pemasaran dalam meminimalisir jumlah calon mahasiswa yang mengundurkan diri.

Kata kunci: *calon mahasiswa, komparasi, k-nearest neighbor, naive bayes, prediksi, random forest*

COMPARATIVE STUDY OF NAIVE BAYES, K-NEAREST NEIGHBOR, AND RANDOM FOREST FOR THE PREDICTION OF PROSPECTIVE STUDENTS ACCEPTED OR WITHDRAWN

Abstract

This study aimed to obtain the best predictive model from New Student Admissions data for 2014 to 2019 by comparing Naive Bayes, K-Nearest Neighbor, and Random Forest. This study used the classification method to predict prospective students. They are accepted or withdrawn. In this study, 19,603 training data and 4,901 test data were used. The results showed that the Random Forest algorithm was the best with an accuracy of 73.61%, compared to K-Nearest Neighbor with an accuracy of 72.08%, and Naive Bayes with an accuracy of 70.47%. It is also concluded that optimizing the model with the Hyperparameter technique produces better accuracy values. This study's results can be used to support the marketing department in minimizing the number of withdrawn prospective students.

Keywords: *comparative, k-nearest neighbor, naive bayes, prediction, prospective students, random forest*

1. PENDAHULUAN

Penerimaan mahasiswa menjadi salah satu hal penting dalam kegiatan akademik dimana banyaknya calon mahasiswa yang mendaftar dan diantara para pendaftar tersebut ada yang jadi masuk ke universitas dan tidak jadi masuk universitas. Kegiatan (Penerimaan Mahasiswa Baru) PMB adalah rutinitas tahunan untuk merekrut calon mahasiswa (Alviana and Kurniawan, 2019). Kegiatan tersebut menghasilkan data yang bertambah banyak tiap tahunnya, seiring dengan pemanfaatan teknologi informasi dalam PMB ini.

Permasalahan yang timbul saat ini adalah data tersebut tidak dimanfaatkan secara optimal oleh para pemangku kepentingan, dalam studi kasus penelitian ini adalah Universitas Esa Unggul (UEU). Data tersebut belum digunakan untuk prediksi calon mahasiswa yang diterima atau tidak meregistrasi ulang (mundur). Sehingga jumlah calon mahasiswa yang mundur dapat diminimalisir dengan adanya pendekatan pemasaran yang lebih masif dan pertimbangan penerimaan jalur beasiswa.

Untuk menutup *gap* tersebut dilakukan penelitian untuk mengindentifikasi pola data PMB dan menentukan model prediksi calon mahasiswa

dengan objek penelitian adalah data PMB dari tahun 2014 sampai tahun 2019. Sehingga penelitian ini menghasilkan suatu kebaruan dalam ilmu pengetahuan dan bisa dimanfaatkan untuk kemajuan kegiatan PMB.

Beberapa peneliti sebelumnya yang telah melakukan penelitian tentang analisis data PMB atau komparasi dari metode klasifikasi. Diantaranya penelitian yang dilakukan (Aribowo and Setiadi, 2018) mengenai analisis komparasi algoritma untuk klasifikasi calon mahasiswa STMIK Widya Pratama menggunakan metode K-Nearest Neighbor (KNN), Naive Bayes dan Decision Tree C4.5. Dalam penelitian tersebut menghasilkan tingkat akurasi algoritma Decision Tree C4.5 merupakan yang terbaik yaitu 80,72%.

Kajian yang dilakukan (Yahya and Jananto, 2019) mengenai komparasi kinerja algoritma C4.5 dan Naive Bayes untuk memprediksi kegiatan PMB pada Universitas STIKUBANG Semarang. Hasil untuk algoritma C4.5 adalah akurasi sebesar 87,5 % dan tingkat kesalahan prediksi sebesar 12,5%.

Kajian (Yulianti, 2020) dihasilkan bahwa Naive Bayes memiliki tingkat akurasi yang baik dibanding dengan Decision Tree dengan 87,19% dan 82,11 %. Sehingga Naive Bayes sangat cocok diterapkan untuk prediksi penjurusan siswa SMA.

Penelitian komparasi algoritma klasifikasi C4.5, Naive Bayes, dan Random Forest dalam menentukan kelulusan mata kuliah di universitas dilakukan oleh (Frastian, Hendrian and Valentino, 2018). Berdasarkan hasil pengukuraan tingkat akurasi algoritma tersebut, diketahui bahwa nilai akurasi C4.5 adalah 98,89% dan nilai AUC adalah 0,500.

Studi (Maulana, Sabarudin and Nugraha, 2019) pada metode C4.5 menghasilkan akurasi sebesar 90,85% dengan AUC sebesar 0,904. Naive Bayes akurasi sebesar 85,52% dengan AUC sebesar 0,891. KNN akurasi sebesar 81,68% dengan AUC sebesar 0,500. Rule Induction akurasi sebesar 90,57%, dengan AUC sebesar 0,906. Random Forest akuasi sebesar 79,96% dengan AUC sebesar 0,800.

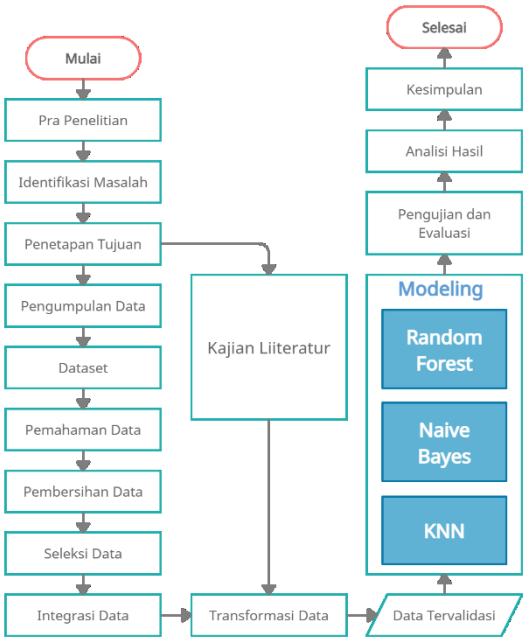
Studi (Kadafi, 2018) menggunakan teknik Cross Validation, dengan hasil bahwa Naive Bayes memiliki nilai akurasi paling tinggi yakni 79,51%, kemudian diikuti oleh KNN pada nilai 79,51%. Diposisi ketiga dari 4 algoritma yang dikomparasi adalah C4.5 dengan nilai akurasi 77,09%, algoritma dengan akurasi paling rendah adalah Rule Induction dengan nilai akurasi 75,57%.

Kemudian penelitian (Fernandez-Garcia et al., 2020) mengenai sistem rekomendasi untuk membantu siswa dengan pilihan mata pelajaran, memprediksi risiko putus sekolah atau prestasi siswa, dan memaksimalkan tingkat kelulusan. Pada penelitian tersebut salah satunya menggunakan metode Random Forest mendapatkan nilai akurasi sebesar 72,3% dan KNN mendapatkan nilai akurasi sebesar 72,2%, dengan peringkat 6 besar terbaik.

Mengacu pada penelitian-penelitian tersebut, maka dalam penelitian ini menggunakan metode klasifikasi KNN, Naive Bayes, dan Random Forest karena metode tersebut menghasilkan nilai akurasi yang lebih baik. Random Forest merupakan kombinasi dari Decision Tree yang digunakan untuk menghindari *overfitting* (Denisko and Hoffman, 2018).

2. METODE PENELITIAN

Pendekatan kuantitatif dengan metode komparatif digunakan dalam studi ini dengan objek penelitian berupa data PMB periode tahun 2014 sampai dengan tahun 2019. Penelitian ini melewati beberapa tahapan seperti yang dilihat pada Gambar 1.



Gambar 1. Alur Penelitian

Skrip Python digunakan dalam penelitian ini dengan platform Kaggle Notebook, sebuah antar muka komputasi awan yang dimiliki oleh perusahaan Google (Tummers et al., 2020). Sejumlah besar kumpulan data serta lingkungan ilmu data berbasis cloud juga disediakan di Kaggle (Quaranta, Calefato and Lanubile, 2021).

2.1. Persiapan Data

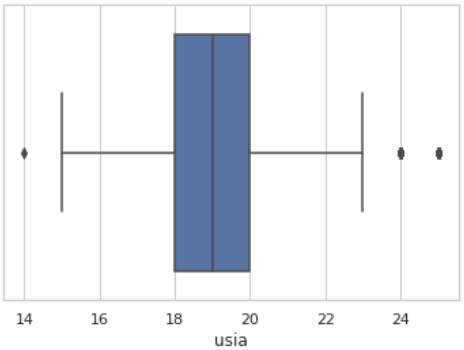
Data yang digunakan dalam penelitian ini sebanyak 24.817 data PMB diterima dan sebanyak 13.316 data PMB mundur (tidak didaftarkan ulang). Fitur label yang ada pada *dataset* tersebut telah melalui proses validasi dari tim marketing UEU. Label tersebut diisi 0 (nol) yang berarti masuk menjadi mahasiswa UEU, dan 1 (satu) yang berarti tidak menjadi mahasiswa UEU. Berikut ini atribut-atribut yang digunakan dalam *data mining*:

- **Data PMB diterima**: id; sex; tgllahir; provinsi; agama; periodemasuk; prodi; asalsmu; label.
- **Data PMB mundur**: id; jeniskelamin; alamat; agama; periodedaftar; prodi; agama; asalsmu; label.

2.2. Pra-pemprosesan Data

Dari dua *dataset* tersebut kemudian dilakukan proses *preprocessing* dengan tahapan sebagai berikut:

1. Filterisasi dan penanganan nilai atribut yang kosong (*Handling Missing Value*). Sehingga data PMB diterima menjadi 18.065 baris, dan data PMB mundur menjadi 13.314 baris.
2. Merubah isi dari kolom tanggal lahir menjadi usia, dihitung sampai dengan tahun pendaftaran.
3. Membuang data *noise* dengan merubah isi *string* '[NULL]' menjadi variabel NaN terlebih dahulu. Sehingga data PMB mundur menjadi 9.161, sedangkan data PMB diterima tetap 18.065 baris.
4. Mengambil data provinsi dari kolom alamat.
5. Seluruh data dijadikan *lowercase*.
6. Mengganti isi kolom asal SMU menjadi kelas SMA dan SMK menggunakan *Regular Expression*.
7. Kemudian menggabungkan dua data tersebut menjadi satu dengan jumlah baris 27.226.
8. Membuang *outlier* (pencilan) data usia dengan IQR (*Inter Quartile Range*) yang merupakan ukuran statistik untuk mengukur variabilitas dalam data tertentu dengan mengambil perbedaan antara kuartil ketiga dan kuartil pertama dalam kumpulan data (IQR = Q3 – Q1). Dimana Q3 = nilai persentil ke-75 (ini adalah nilai tengah antara median dan nilai terbesar di dalam *dataset*). Q1 = nilai persentil ke-25 (ini adalah nilai tengah antara median dan nilai terkecil di dalam *dataset*). Sehingga jumlah baris di *dataset* menjadi 24.504. Visualisasi dengan *boxplot* setelah normalisasi dapat dilihat pada Gambar 2.



Gambar 2. Setelah normalisasi

2.3. Transformasi Data

Dataset yang telah melalui proses *preprocessing* kemudian dilakukan proses *Transform Encode* menggunakan librari *Sklearn*. Proses transformasi data ini bertujuan untuk membuat kelas berbentuk numerik, yang akan digunakan pada tahap *modeling*.

Kelas-kelas hasil transformasi tersebut terlihat pada Tabel 1.

Tabel 1. Kelas hasil encode

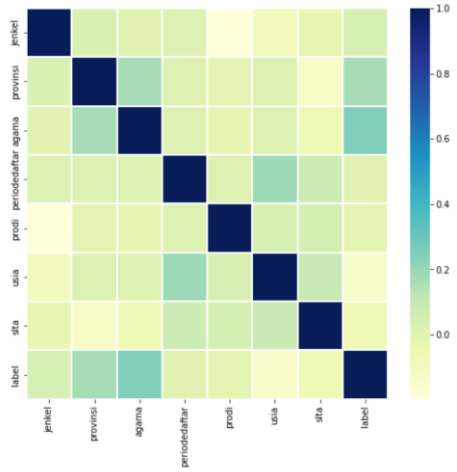
Fitur	Kelas	Kode
jenkel (Gender)	L (Laki-laki)	0
	P (Perempuan)	1
provinsi (Asumsi: Tempat tinggal masih dalam satu provinsi sebagai faktor yang mempengaruhi)	aceh	0
	bali	1
	bangka belitung	2
	banten	3
	bengkulu	4
	di yogyakarta	5
	dki jakarta	6
	gorontalo	7
	jambi	8
	jawa barat	9
	jawa tengah	10
	jawa timur	11
	kalimantan barat	12
	kalimantan selatan	13
	kalimantan tengah	14
	kalimantan timur	15
	kepulauan riau	16
	lampung	17
	luar negeri	18
	maluku	19
	maluku utara	20
	nusa tenggara barat	21
	nusa tenggara timur	22
	papua	23
	papua barat	24
	riau	25
	sulawesi barat	26
	sulawesi selatan	27
	sulawesi tengah	28
	sulawesi tenggara	29
	sulawesi utara	30
	sumatera barat	31
	sumatera selatan	32
	sumatera utara	33
agama (Asumsi: Waktu dan jadwal kegiatan keagamaan mingguan terhadap jadwal perkuliahan, sebagai faktor yang mempengaruhi)	budha	0
	hindu	1
	islam	2
	katolik	3
	kong hu cu	4
	kristen	5
	protestan	6
	undefined	7

Fitur	Kelas	Kode
periodedaftar (Asumsi: Mahasiswa baru regular hanya dibuka semester ganjil, semester genap untuk mahasiswa pindahan atau kelas karyawan)	1 (semester ganjil)	0
	2 (semester genap)	1
prodi (Jurusan Perkuliahan)	akuntansi	0
	bahasa inggris	1
	bioteknologi	2
	broadcasting	3
	desain interior	4
	desain komunikasi visual	5
	desain produk	6
	farmasi	7
	fisioterapi	8
	hubungan masyarakat	9
	ilmu gizi	10
	ilmu hukum	11
	ilmu keperawatan	12
	jurnalistik	13
	kesehatan masyarakat	14
	komunikasi pemasaran	15
	magister administrasi publik	16
	magister administrasi rumah sakit	17
	magister akuntansi	18
	magister hukum	19
	magister ilmu komputer	20
	magister ilmu komunikasi	21
	magister manajemen	22
	manajemen	23
	manajemen informasi kesehatan (d4)	24
	manajemen informasi kesehatan (s1)	25
	pendidikan bahasa inggris	26
	pendidikan guru sd	27
	perencanaan wilayah dan kota	28
	profesi fisioterapi	29
	profesi keperawatan	30
	psikologi	31
	rekam medis dan informasi kesehatan	32
	sistem informasi	33
	survei dan pemetaan	34
	teknik industri	35
	teknik informatika	36
	tv dan radio broadcasting	37
usia	14 15 16 17 18 19 20 21 22 23 24 25	
slta (Asumsi: Asal sekolah SLTA, sebagai faktor yang mempengaruhi dalam ketepatan pilihan jurusan perkuliahan)	sma	0
	smk	1
label (Klasifikasi diterima atau mundur)	diterima	0
	mundur	1

Untuk memeriksa hubungan antara fitur digunakan Matriks Korelasi, berikut ini berupa kode semu (*pseudocode*) dari *script* yang digunakan:

Var cor = dataFrame fungsi Corr
Var f, ax = matPlot fungsi *subplots* ukuran 9, 8
Var cg = sns fungsi *heatmap* (Var cor, ax, cmap="YlGnBu", *linewidths*=0.1)

Dengan teknik *heatmap* tersebut, sehingga diperoleh hasil seperti Gambar 3, dimana nilai 0 (nol) menunjukkan tidak ada korelasi dan nilai 1 (satu) menunjukkan korelasi maksimum.



Gambar 3. Kolerasi antar fitur

2.4. Pembentukan Data Latih

Kemudian *dataset* hasil dari *Transform Encode* dibagi menjadi dua bagian, yaitu data latih dan data uji. Dengan proporsi data latih sebanyak 80% dan data uji sebanyak 20% dari total *dataset* dengan nilai *random state* sebesar 42. Berdasarkan penelitian (Yahya and Jananto, 2019) pembagian dengan jumlah yang tepat dapat menghasilkan tingkat akurasi yang baik. Sehingga total dari masing-masing *dataset* menjadi 19.603 baris data latih, dan 4.901 baris data uji.

2.5. Pembentukan Model

Dalam tahap pemodelan ini memproses klasifikasi data menggunakan tiga metode, yaitu Naive Bayes, K-Nearest Neighbor (KNN), dan Random Forest.

- **Naive Bayes**
  Metode Naive Bayes adalah menghitung probabilitas dari masing-masing fitur data yang sebelumnya telah ditentukan, menghitung nilai *likelihood* dengan mengalikan nilai dari setiap probabilitas untuk selanjutnya dilakukan prediksi data berdasarkan label yang telah ditentukan sebelumnya (Mustofa and Mahfudh, 2019). Pada persamaan (1) merupakan rumus untuk menghitung nilai probabilitas:

$$\boldsymbol{P(c|x)} = \frac{P(x|c)}{P(x)}\boldsymbol{P(c)} \qquad (1)$$

P(c|x) adalah probabilitas posterior kelas (target) yang diberikan prediktor (atribut). **P(c)** adalah peluang kelas sebelumnya. *P(x|c)* adalah peluang yang merupakan peluang dari kelas yang diberikan oleh prediktor. *P(x)* adalah probabilitas sebelumnya dari prediktor.

- **K-Nearest Neighbor**
  Metode KNN adalah menentukan jumlah tetangga (*k*) dan mencari nilai jarak kedekatan ketetanggaan dilakukan dengan perhitungan *Euclideann Distance* (Annisa, 2019). Kemudian hasil dari perhitungan *Euclidean Distance* tersebut diurutkan berdasarkan nilai terbesar (*descending*). Setelah itu dilakukan prediksi data berdasarkan label yang telah ditentukan. Berikut merupakan rumus perhitungan jarak menggunakan teknik *Euclidean Distance* seperti pada persamaan (2).

$$dist(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \qquad (2)$$

  n adalah jumlah variabel, x_i dan y_i masing-masing adalah variabel vektor *x* dan *y*, dalam ruang vektor dua dimensi. yaitu $x = (x_1, x_2, x_3, \ldots)$ dan $y = (y_1, y_2, y_3, \ldots)$.

- **Random Forest**
  Metode Random Forest merupakan metode pelatihan berbasis *ensemble learning* yang menggunakan algoritma Decision Tree (Rahayuningsih, 2019). Kemudian dari beberapa model Decision Tree tersebut dilakukan prediksi menggunakan data uji yang sama pada disetiap model. Hasil dari prediksi kemudian dilakukan proses *majority voting* menggunakan metode modus, dan akhirnya kelas dari modus tersebut menjadi kelas prediksi. Pada persamaan (3) merupakan rumus untuk menghitung *Entropy*, dan persamaan (4) merupakan rumus untuk menghitung *Gain* pada metode Random Forest.

$$E = -\sum_{i=1}^{n} p_i * log_2(p_i) \qquad (3)$$

  p_i adalah peluang terpilihnya contoh secara acak di kelas *i*.

$$Gain = E_{parent} - E_{children} \qquad (4)$$

  E_{parent} adalah entropi dari node induk dan $E_{children}$ adalah entropi rata-rata dari node anak.

2.6. Optimasi Model

Optimisasi di dalam *machine learning* adalah proses menentukan nilai parameter model yang akan disesuaikan berdasarkan data latih (*training data*). Proses ini akan menghasilkan model yang lebih akurat untuk memetakan *input* menjadi *output* sesuai pola yang ditemukan di dalam data latih.

Teknik *Hyperparameter* digunakan dalam penelitian ini, guna mendapatkan model yang paling optimal. Telah diakui secara luas bahwa teknik penyetelan pada *Hyperparameter* mendapatkan hasil yang lebih baik ketimbang pengaturan default yang disediakan oleh librari *machine learning* (Gressling, 2020; Yang and Shami, 2020). Librari yang digunakan adalah *GridSearchCV* dari Sklearn, dan khusus algoritma Naive Bayes ditambahkan librari *RandomizedSearchCV* sebagai perbandingan.

3. HASIL DAN PEMBAHASAN

Bagian ini merupakan pembahasan metode Naive Bayes, K-Nearest Neighbor dan Random Forest yang digunakan pada klasifikasi penerimaan mahasiswa. *Dataset* dan *script* Python yang dihasilkan dalam penelitian ini dapat diakses di https://www.kaggle.com/code/marzukipilliang/study-of-naive-bayes-knn-random-forest.

Diambil dua baris data yang digunakan untuk pembuktian model yang telah dibuat. Dua data tersebut dapat dilihat pada Tabel 2.

Klasifikasi di makalah ini adalah kelas label 0 (diterima) dan kelas label 1 (mundur), hasil prediksi tanpa proses optimasi di *Hyperparameter* pada Naive Bayes, KNN, dan Random Forest, beserta nilai akurasi *Training*, dan nilai akurasi *Test Data* ditampilkan pada Tabel 3.

Tabel 2. Data untuk pembuktian

Fitur	Kelas	
	Data#1	Data#2
jenkel	1 (Perempuan)	0 (Laki-laki)
provinsi	28 (Sulawesi Tengah	3 (Banten)
agama	2 (Islam)	5 (Kristen)
periode daftar	0 (Semester Ganjil)	1 (Semester Genap)
prodi	3 (Broadcasting)	33 (Sistem Informasi)
usia	17 (Tahun)	25 (Tahun)
slta	1 (SMK)	0 (SMA)
label	1 (Mundur)	0 (Diterima)

Hasil prediksi label dan nilai akurasi model ditampikan pada tabel dibawah ini.

Tabel 3. Nilai akurasi tanpa optimasi model

Model	Prediksi		Akurasi %	
	Data#1	Data#2	Latih	Uji
Naive Bayes	1 Diterima	0 Mundur	67,92	67,86
KNN	1 Diterima	1 Mundur	75,57	70,06
Random Forest	1 Diterima	0 Mundur	83,14	70,67

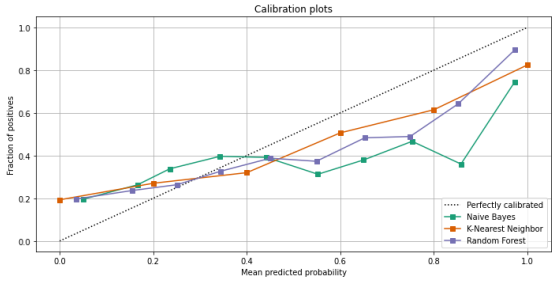
Jika dilihat perbandingan antara Tabel 2 dan Tabel 3, maka model Naive Bayes dan Random Forest menghasilkan prediksi kelas label yang sesuai, sedangkan model KNN memprediksikan label yang tidak sesuai pada Data#2. Pada model KNN didapat 75,57% akurasi terhadap data latih, dan 70,06% terhadap data uji.

Untuk akurasi data latih model Random Forest menghasilnya nilai yang terbaik yaitu 83,14%. Namun akurasi terhadap data uji turun jauh menjadi 70,67%. Hal tersebut dimungkinkan karena pada model Random Forest terjadi *overfitting*.

Akurasi dari model Naive Bayes merupakan yang paling rendah, yaitu 67,92% terhadap data latih, dan 67,86% terhadap data uji.

Dari tiga model yang dihasilkan dari data latih, kemudian dibuat *plot* kurva kalibrasi (juga dikenal sebagai diagram keandalan) menggunakan probabilitas yang diprediksi dari kumpulan data pengujian. *Binning* probabilitas prediksi membuat kurva kalibrasi, kemudian di-*plot* probabilitas prediksi rata-rata di setiap nampan terhadap frekuensi yang diamati ('fraksi positif'), seperti terlihat pada Gambar 4.

Berdasarkan pada Gambar 4 dapat dipastikan bahwa masing-masing model masih tidak cukup dekat dengan garis kalibrasi yang sempurna. Untuk itulah dilakukan optimasi model yang dibahas sub bagian berikut.



Gambar 4. Perbandingan kalibrasi model

3.1. Hasil Optimasi Naive Bayes

Setelah dilakukan penetapan parameter *var_smoothing* dengan nilai *logspace*(0, 9, *num* = 100), serta *cv_method* berupa *n_splits* = 5, *n_repeats* = 3, *random_state* = 999 terhadap *GridSearchCV*. Maka menghasilkan *fitting* 15 *folds* untuk setiap 100 kandidat, sehingga total perulangan *fitting* model menjadi 1.500 kali terhadap data latih. Didapat parameter *var_smoothing* sebesar 1,87381742 x 10^{-2} merupakan yang terbaik, dengan nilai akurasi dan prediksi pada Data#1 dan Data#2 seperti terlihat di Tabel 4.

Tabel 4. Hasil optimasi Naive Bayes

Optimasi	Prediksi		Akurasi %	
	Data#1	Data#2	Latih	Uji
Naive Bayes	1 Diterima	0 Mundur	70,45	70,47

3.2. Hasil Optimasi KNN

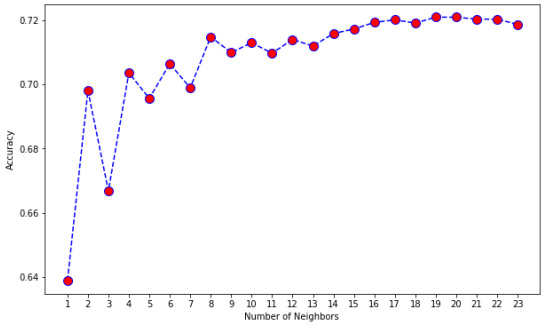
Parameter yang digunakan untuk *GridSearchCV* KNN berupa *leaf_size* = *list* (*range* (1,50)), *n_neighbors* = *list* (*range* (1,30)), dan p = [1,2]. Sehingga didapat parameter *leaf_size* = 35, *n_neighbor* = 20, dan p = 1 merupakan yang terbaik untuk KNN. Tabel 5 menunjukkan nilai akurasi dan prediksi yang dihasilkan.

Tabel 5. Hasil optimasi KNN

Optimasi	Prediksi		Akurasi %	
	Data#1	Data#2	Latih	Uji
KNN	1 Diterima	0 Mundur	73,76	72,08

Bila dibandingkan dengan prediksi sebelumnya di Tabel 3, maka terdapat perubahan prediksi label terhadap Data#2 dengan hasil yang tepat.

Proses optimasi tersebut juga menghasilkan nilai *k* yang paling tepat. Perbandingan antara jumlah *k* dengan nilai akurasi yang dihasilkan dapat dilihat pada Gambar 5.



Gambar 5. Perbandingan k dan akurasinya

3.3. Hasil Optimasi Random Forest

Khusus optimasi pada Random Forest dilakukan dengan dua metode, yaitu:

- *RandomizedSearchCV* (RFC Rand)
  Parameter yang digunakan adalah *n_estimators* antara 200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800; *max_features* antara auto atau sqrt; *max_depth* antara 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, None; *min_samples_split* antara 2, 5, 10; *min_samples_leaf* antara 1, 2, 4; *bootstrap* antara True atau False. Dari proses ini dihasilkan model Random Rorest terbaik

dengan parameter *n_estimators* = 1200, *max_depth* = 10, *min_samples_split* = 5.

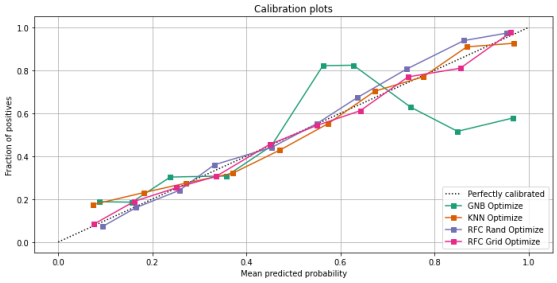
- *GridSearchCV* (RFC Grid)
  Parameter yang dipakai adalah *bootstrap* = True; *max_depth* antara 10 sampai 15; *max_features* antara 2 atau 3; *min_samples_leaf* antara 3, 4, 5, 6; *min_samples_split* antara 3, 4, 5, 6; *n_estimators* antara 1150, 1200, 1250, 1300, 1350. Didapat parameter *max_depth* = 15, *max_features* = 3, *min_samples_leaf* = 6, *min_samples_split* = 3, *n_estimators* = 1200 merupakan yang terbaik untuk pemodelan Random Forest.

Dari dua model tesebut dilakukan prediksi terhadap Data#1 dan Data#2 dengan hasil akurasi terlihat pada Tabel 6.

Tabel 6. Hasil optimasi Random Forest

Optimasi	Prediksi		Akurasi %	
	Data#1	Data#2	Latih	Uji
RFC Rand	1 Diterima	0 Mundur	75,24	73,47
RFC Grid	1 Diterima	0 Mundur	76,22	73,61

Tingkat akurasi dari masing-masing model menjadi meningkat dan yang terpenting adalah *overfitting* pada Random Forest dapat diatasi. Gambar 6 menunjukkan kalibrasi tiap-tiap model yang sudah mendekati sempurna, hanya model Naive Bayes yang tidak ada peningkatan secara signifikan.



Gambar 6. Plot kalibrasi setelah optimasi

4. KESIMPULAN DAN SARAN

Pembersihan *dataset* dari *noise* dan *outlier,* sangat penting dilakukan pada tahap pra-pemprosesan untuk mendapatkan akurasi model yang baik.

Penentuan nilai *k* yang tepat pada KNN, dapat menghasilkan nilai akurasi yang tinggi, dan mengindari terjadinya *overfitting* atau *underfitting*.

Optimasi model dengan teknik *Hyperparameter* terbukti dapat meningkatkan nilai akurasi, dan mengatasi *overfitting* pada Random Forest.

Penelitian ini menggunakan 19.603 data latih, dan 4,901 data uji. Tabel 7 menunjukkan bahwa Random Forest merupakan algoritma terbaik dengan validasi akurasi 73,61%, 1,5% lebih tinggi dibandingkan dengan KNN dengan validasi akurasi 72,08%. Sedangkan algoritma Naive Bayes menghasilkan akurasi yang paling rendah yaitu 70,47%. Hal tersebut menunjukkan bahwa algoritma Random Forest layak dijadikan model untuk sistem pendukung keputusan para pemangku kepentingan dalam mempertimbangkan calon mahasiswa baru.

Tabel 7. Hasil Penelitian

Algoritma	Akurasi %	
	Latih	Uji
Naïve Bayes	70,45	70,47
K-Nearest Neighbor	73,76	72,08
Random Forest	76,22	73,61

Namun proses optimasi di Random Forest memakan waktu yang lama. Sehingga masih diperlukan penelitian lanjutan agar mendapatkan model optimal dengan waktu yang relatif singkat.

DAFTAR PUSTAKA

ALVIANA, S. AND KURNIAWAN, B., 2019. Analisis Data Penerimaan Mahasiswa Baru Untuk Meningkatkan Potensi Pemasaran Universitas Menggunakan Business Intelligence (Studi Kasus Universitas XYZ). *Infotronik : Jurnal Teknologi Informasi dan Elektronika*, [online] 4(1), pp.10–15. https://doi.org/10.32897/infotronik.2019.4.1.2.

ANNISA, R., 2019. Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Penderita Penyakit Jantung. *Jurnal Teknik Informatika Kaputama (JTIK)*, [online] 3(1), pp.22–28. Available at: <https://jurnal.kaputama.ac.id/index.php/JTIK/article/view/141/156>.

ARIBOWO, D. AND SETIADI, A.E.H., 2018. Analisa Komparasi Algoritma Data Mining untuk Klasifikasi Heregistrasi Calon Mahasiswa STMIK Widya Pratama. *IC-Tech*, [online] 13(2), pp.1–6. https://doi.org/10.47775/ictech.v13i2.30.

DENISKO, D. AND HOFFMAN, M.M., 2018. *Classification and interaction in random forests*. *Proceedings of the National Academy of Sciences of the United States of America*, https://doi.org/10.1073/pnas.1800256115.

FERNANDEZ-GARCIA, A.J., RODRIGUEZ-ECHEVERRIA, R., PRECIADO, J.C., MANZANO, J.M.C. AND SANCHEZ-FIGUEROA, F., 2020. Creating a Recommender System to Support Higher Education Students in the Subject Enrollment Decision. *IEEE Access*, [online] 8, pp.189069–189088.

https://doi.org/10.1109/ACCESS.2020.3031572.

FRASTIAN, N., HENDRIAN, S. AND VALENTINO, V.H., 2018. Komparasi Algoritma Klasifikasi Menentukan Kelulusan Mata Kuliah Pada Universitas. *Faktor Exacta*, [online] 11(1), p.66. https://doi.org/10.30998/faktorexacta.v11i1.1826.

GRESSLING, T., 2020. 84 Automated machine learning. In: *Data Science in Chemistry*. [online] De Gruyter. pp.409–411. https://doi.org/10.1515/9783110629453-084.

KADAFI, A.R., 2018. Perbandingan Algoritma Klasifikasi Untuk Penjurusan Siswa SMA. *Jurnal ELTIKOM*, [online] 2(2), pp.67–77. https://doi.org/10.31961/eltikom.v2i2.86.

MAULANA, M.S., SABARUDIN, R. AND NUGRAHA, W., 2019. Prediksi Ketepatan Kelulusan Mahasiswa Diploma dengan Komparasi Algoritma Klasifikasi. *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, [online] 7(3), p.202. https://doi.org/10.26418/justin.v7i3.33316.

MUSTOFA, H. AND MAHFUDH, A.A., 2019. Klasifikasi Berita Hoax Dengan Menggunakan Metode Naive Bayes. *Walisongo Journal of Information Technology*, [online] 1(1), p.1. https://doi.org/10.21580/wjit.2019.1.1.3915.

QUARANTA, L., CALEFATO, F. AND LANUBILE, F., 2021. KGTorrent: A Dataset of Python Jupyter Notebooks from Kaggle. In: *2021 IEEE/ACM 18th International Conference on Mining Software Repositories (MSR)*. [online] IEEE. pp.550–554. https://doi.org/10.1109/MSR52588.2021.00072.

RAHAYUNINGSIH, P.A., 2019. Analisis Komparasi Algoritma Klasifikasi Data Mining. *Jurnal Teknik Informatika Kaputama (JTIK)*, [online] 3(1). Available at: <http://jurnal.kaputama.ac.id/index.php/JTIK/article/view/169>.

TUMMERS, J., CATAL, C., TOBI, H., TEKINERDOGAN, B. AND LEUSINK, G., 2020. Coronaviruses and people with intellectual disability: an exploratory data analysis. *Journal of Intellectual Disability Research*, [online] 64(7), pp.475–481. https://doi.org/10.1111/jir.12730.

YAHYA, N. AND JANANTO, A., 2019. Komparasi Kinerja Algoritma C4.5 Dan Naive Bayes Untuk Prediksi Kegiatan Penerimaan Mahasiswa Baru (Studi Kasus : Universitas Stikubank Semarang). *Prosiding SENDI*, [online] (2014), pp.978–979. Available at: <https://www.unisbank.ac.id/ojs/index.php/sendi_u/article/view/7389>.

YANG, L. AND SHAMI, A., 2020. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, [online] 415, pp.295–316. https://doi.org/10.1016/j.neucom.2020.07.061.

YULIANTI, H., 2020. Analisis Komparasi Algoritma Klasifikasi Data Mining untuk Prediksi Penjurusan Siswa Sekolah Menengah Atas (SMA) Pramitra Karawaci Tangerang. *LENSA*, [online] 2(48), pp.1–6. https://doi.org/10.33050/lns.v2i48.1277.