BAB 2

TINJAUAN PUSTAKA

2.1. Prediksi Keterlambatan Pembayaran SPP

Prediksi adalah proses untuk memperkirakan sesuatu yang paling mungkin terjadi pada masa depan. Prediksi dapat mengarah pada wawasan teoritis, misalnya pengetahuan suatu area atau objek yang dituju atas dasar dugaan perkiraan yang ada sebelumnya (Fadila *dkk,* 2020).

SPP adalah sumbangan pembinaan pendidikan (SPP) yang dibayarkan mahasiswa di perguruan tinggi swasta. Tujuan dari SPP adalah untuk mendanai kebutuhan pembelajaran sehingga kegiatan belajar mengajar dapat berlangsung dengan baik. SPP biasanya dibayarkan setiap semester oleh mahasiswa (Fadlan, 2020).

Berdasarkan penjelasan diatas dapat penulis simpulkan bahwa prediksi keterlambatan pembayaran SPP adalah suatu proses untuk memperkirakan sesuatu yang paling mungkin terjadi tentang suatu pembayaran yang sudah melawati batas waktu yang telah disepakati sebelumnya. Prediksi keterlambatan pembayaran SPP sangat penting karena sumber pendapatan memerlukan perhatian dan pengawasan yang baik. Apalagi pendapatan yang berasal dari SPP menjadi sumber pendapatan utama, terutama perguruan tinggi swasta yang digunakan untuk biaya operasional sehingga kegiatan belajar mengajar dapat berlangsung dengan baik. Berikut adalah tabel penelitian tentang prediksi keterlambatan pembayaran SPP yang dapat dilihat pada Tabel 2.1.

Tabel 2.1 Prediksi Keterlambatan Pembayaran SPP

No.	Penulis	Judul	Metode	Hasil
1.	(Rohmayani, 2020)	*Analysis of Student Tuition Fee Pay Delay Prediction* (Case Study :	*Naïve Bayes algorithm with particle swarm*	Hasil penelitian yang didapatkan mendapatkan tingkat akurasi sebesar 73,94%

Tabel 2.1 Prediksi Keterlambatan Pembayaran SPP (Lanjutan)

No.	Penulis	Judul	Metode	Hasil
		Politeknik Tedc Bandung)	*optimation optimazation*	
2.	(Kusrini *et al.*, 2019)	*Tuition Fee Payment Overdue Prediction*	*Naïve Baye*s dan *K-Nearest Neighbor*	Penelitian ini membuat perbandingan empat metode, yaitu Metode *Naïve Bayes, Naïve Bayes dan Information Gain,* K NN, dan K-NN dan *Information Gain*, K NN, dan K-NN dan *Information Gain*. Metode terbaik adalah diperoleh dari kombinasi algoritma *Naïve Bayes* dengan pemilihan fitur perolehan informasi, yang menghasilkan nilai akurasi = 67%, presisi = 64%, *recall* = 59% dan *f measure* = 61%
3.	(Abdullah, 2019)	Sistem Prediksi Keterlambatan	*K-Nearest Neighbor*	Hasil penelitian yang didapatkan

Tabel 2.1 Prediksi Keterlambatan Pembayaran SPP (Lanjutan)

No.	Penulis	Judul	Metode	Hasil
		Pembayaran Spp Sekolah (Studi Kasus: Smk Al-Islam Surakarta)		mendapatkan rata-rata akurasi sebesar 65% dengan pengujian k fold 5 dengan nilai presisi=63%, recall=59% dan *F-measure* sebesar 62% dimana tingkat akurasi tertinggi tertinggi yaitu pada pengujian k=3 yang menghasilkan nilai akurasi sebesar 68% dengan nilai presisi sebesar 65% , *recall* 73% , *F-measure* 76%
4.	(Fadlan, 2020)	Notifikasi Keterlambatan Pembayaran Sumbangan Pembinaan Pedidikan (SPP) Berbasis Android Di Smk N 1 Baso	*Waterfall*	Hasil dari uji validitas menghasilkan rata-rata 0,91 yang memiliki kriteria valid, uji praktikalitas menghasilkan rata-rata 0,97

Tabel 2.1 Prediksi Keterlambatan Pembayaran SPP (Lanjutan)

No.	Penulis	Judul	Metode	Hasil
				menghasilkan kriteria sangat tinggi, dan uji efektivitas menghasilkan rata-rata 0,87 menghasilkan kategori kefektifitasan sangat efektiv.
5.	(Istiana & Waspada, 2019)	*Predict Students Monthly Payment On Islamic Boarding School*	C4.5	Hasil penelitian yang didapatkan mendapatkan akurasi rata-rata 81,15%, rata-rata presisi 77,62% dan nilai *recall* 91,90%

2.2. *Data Mining*

Menurut (Santosa & Umam, 2018) *data mining* merupakan aktivitas pengumpulan informasi atau pengetahuan penting yang berasal dari suatu set data yang besar. Sedangkan menurut Han et al. (2012) *data mining* adalah salah satu proses perhitungan untuk menemukan pola dalam kumpulan data yang besar (Scholz, 2017). Menurut Fayyad et al. (1996) proses *data mining* digambarkan sebagai penemuan pengetahuan dalam database antara lain penggunaan algoritma, alat statistik, dan pembelajaran mesin untuk mengekstrak pola yang sebelumnya tidak diketahui (Scholz, 2017). Metode *data mining* yang digunakan dalam penelitian ini adalah *random forest.*

Secara umum *data mining* dapat dibagi menjadi dua kategori yaitu sebagai berikut (Haristu, 2019) :

1) *Descriptive*

   Descriptive berfungsi sebagai informasi umum tentang data yang terdapat dalam database. Database biasanya digunakan untuk merepresentasikan data.

2) *Predictive*

   Predictive berfungsi dalam memprediksi data yang ada dalam database.

Data Mining biasanya dilakukan melalui 3 langkah yaitu sebagai berikut (Arhami & Nasir, 2020):

1) Eksplorasi

   Tahap yang dilakukan adalah mempersiapkan data, selanjutnya data dibersihkan sesuai dengan yang diperlukan, dan menghapus data yang sama sehingga data yang tersisa merupakan data yang benar-benar dapat digunakan.

2) Permodelan

   Tahap yang dilakukan adalah membuat model statistik dengan tujuan mengevaluasinya untuk dapat membuat prediksi yang terbaik dan paling akurat. Proses ini dapat memakan waktu karena model yang berbeda diterapkan pada kumpulan data yang sama dan dijalankan secara berulang-ualng untuk kemudian membandingkan hasilnya.

3) Penerapan

   Pada langkah terakhir ini, model yang digunakan diuji pada data *training* dan pada data *testing* untuk membuat prediksi hasil yang konsisten dengan apa yang diterapkan.

2.3. Prapemrosesan Data

Prapemrosesan data dapat dilakukan dengan cara membersihkan data, mengintegrasikan data, reduksi, penambahan data, dan transformasi data yang artinya dapat melakukan prapemrosesan data menggunakan pembersihan data dan reduksi data secara bersamaan (Suyanto, 2017). Prapemroses data adalah tahapan memasukan data kosong ke dalam data, menghilangkan duplikasi data, memeriksa ketidakkonsistenan data, membersihkan data, dan memperbaiki

kesalahan pada data. Data kosong biasanya disebabkan oleh data baru tanpa informasi (Pristyanto, 2019).

Tujuan prapemrosesan data terbagi menjadi 3 tujuan yaitu sebagai berikut (Suyanto, 2017):

1) Untuk mempermudah pemahaman data dan untuk mempermudah pemilihan teknik dan metode data mining yang tepat.
2) Untuk meningkatkan kualitas data sehigga proses data mining bekerja lebih baik.
3) Untuk meningkatkan efektivitas dan kemudahan proses data mining.

Berikut adalah tahapan dalam prapemrosesan data yaitu sebagai berikut (Prasojo & Haryatmi, 2021):

1) *Data cleaning*

   Data cleaning yang digunakan yaitu mengisi *missing value,* mengidentifiaksi outlier, memperbaiki noise data, mengoreksi inkonsistensi data, dan mengatasi dampak dari integrasi data yang mengakitbatkan redudansi.

2) *Data integration*

   Data Integration adalah tahapan untuk menyatukan data dari berbagai sumber. Integrasi data dilakukan ketika terdapat data yang berasal dari lokasi yang berbeda. Tahapan yang dilakukan yaitu integrasi skema, identifikasi masalah dengan entitas, dan menangani konflik pada nilai data.

3) *Data Transformation*

   Data Transformation adalah memodifikasi data untuk mendapatkan data yang berkualitas. Adapun tahapan dalam transformasi data yaitu menghapus noise data, mengagregasi data, normalisasi data, dan membentuk atribut.

4) *Data Reduction*

   Data reduction adalah langkah untuk pengurangan dimensi data, atribut atau jumlah data. Adapaun tahapannya yaitu agregasi data, pengurangan di mensi, diskretisasi, dan kompresi data.

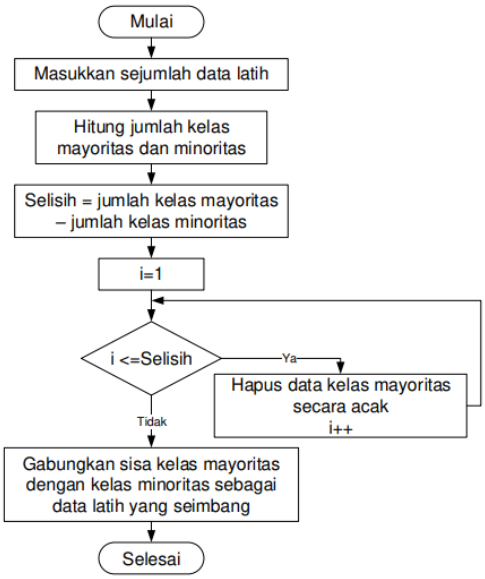
5) *Data Reduction*

   Data reduction adalah langkah untuk pengurangan dimensi data, atribut atau

jumlah data. Adapaun tahapannya yaitu agregasi data, pengurangan dimensi, diskretisasi, dan kompresi data.

2.4. *Random Under Sampling*

Random under sampling secara acak memilih data dari kelas mayoritas yang akan dihapus dari data *training.* Dengan melakukan *random under sampling* akan mengurangi jumlah data *training* dari kelas mayoritas. Kelemahan dari *random under sampling* adalah data yang dihapus dari kelas mayoritas merupakan data acak bisa jadi data tersebut penting untuk membuat model klasifikasi yang baik. (Sabilla & Vista, 2021) Algoritrma *random under sampling* dapat dilihat pada gambar 2.1.



Gambar 2.1 *Flowchart* Algoritma *Random Under sampling*

Sumber: (Saifudin *dkk*, 2015)

Berikut adalah tahapan *random under sampling:*

1) Masukkan jumlah data latih.
2) Menghitung jumlah dari kelas mayoritas dan minoritas.
3) Menghitung selisih jumlah dari kelas mayoritas dan jumlah dari kelas minoritas.
4) Menghapus data dari kelas mayoritas sampai sama dengan minoritas.
5) Menggabungkan data yang tersisa dari kelas mayoritas dengan kelas minoritas dan menghasilkan data latih yang seimbang.

2.5. *Random Forest*

Random Forest adalah salah satu algoritma dari *machine learning* untuk mengembangkan *decision tree. Random Forest* dapat dianggap sebagai kombinasi dari beberapa buah *decision tree* (Primartha, 2021). *Random Forest* merupakan salah satu bentuk yang berasal dari metode *ensemble* yang bertujuan untuk meningkatkan akurasi klasifikasi data dari sebuah pemilah tunggal yang tidak stabil melalui kombinasi dari banyak jenis metode yang sama sebagai proses *majority voting* untuk menghasilkan prediksi tentang klasifikasi akhir (Putra, 2019). Metode *random forest* pada saat ini banyak digunakan untuk mengklasifikasikan berbagai studi kasus yang ditunjukan pada tabel 2.2.

Tabel 2.2 Penelitian *Random Forest*

No.	Penulis	Judul	Metode	Hasil
1.	(Putra, 2019)	Sistem Rekomendasi Kelayakan Kredit pada BRI Kantor Cabang Pelaihari	*Random Forest*	Hasil penelitian yang didapatkan mendapatkan tingkat akurasi rata-rata sebesar 96,47%
2.	(Adnyana, 2015)	Prediksi Lama Studi Mahasiswa (Studi Kasus: Stikom Bali)	*Random Forest*	Hasil penelitian yang didapatkan mendapatkan tingkat akurasi sebesar 83,54%
3.	(Nalatissifa *dkk*, 2020)	Prediksi Ketidakhadiran Di Tempat Kerja	*Naïve Bayes, Support Vector Machine* (SVM) dan	Pada hasil penelitian ini, algoritma *Random Forest* menghasilkan nilai akurasi, presisi, dan *recall* tertinggi dibandingkan dengan

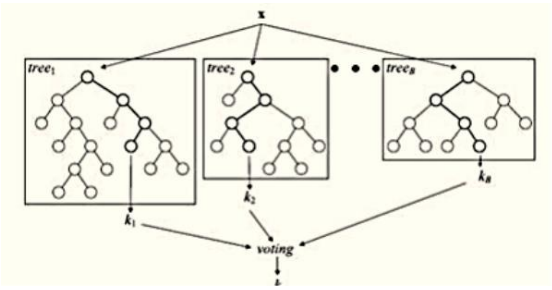
Tabel 2.2 Penelitian *Random Forest* (Lanjutan)

No.	Penulis	Judul	Metode	Hasil
			Random Forest	algoritma *Naïve Bayes* dan SVM, yaitu menghasilkan tingkat akurasi sebesar 99,38%, presisi 99,42% dan *recall* 99,39%
4.	(Haristu, 2019)	Prediksi Win Ratio Pemain Pl*ayer Unknown Battleground*	*Random Forest*	Hasil penelitian yang didapatkan mendapatkan tingkat akurasi sebesar 88,19%
5.	(Sandag, 2020)	Prediksi *Rating* Aplikasi *App Store*	*Random Forest*	Hasil Penelitian yang didapatkan mendapatkan tingkat *accuracy* 86,27%,
6.	(Prasojo & Haryatmi, 2021)	Analisa Prediksi Kelayakan Pemberian Kredit Pinjaman	*Random Forest*	Hasil penelitian yang didapatkan mendapatkan tingkat akurasi sebesar 83%
7.	(Renata & Ayub, 2020)	Analisis Risiko pada dataset *Peer to peer lending*	*Random Forest* dan *Logistic Regression*	Hasil penelitian yang didapatkan mendapatkan tingkat akurasi menggunakan metode *Random Forest* sebesar 0.924012 sedangkan *Logistic Regression* hanya mendapatkan tingkat akurasi sebesar 0.923995

Tabel 2.2 Penelitian *Random Forest* (Lanjutan)

No.	Penulis	Judul	Metode	Hasil
8.	(Zailani & Hanun, 2020)	Penentuan Kelayakan Pemberian Kredit Di Koperasi Mitra Sejahtera	*Random Forest*	Hasil penelitian yang didapatkan mendapatkan tingkat akurasi sebesar 87,88%
9.	(Putra, 2019)	Sistem Rekomendasi Kelayakan Kredit pada BRI Kantor Cabang Pelaihari	*Random Forest*	Hasil penelitian yang yang didapatkan mendapatkan tingkat rata-rata *accuracy* sebesar 96,57%, *precision* sebesar 96,46% dan *recall* sebesar 100%
10.	(Siburian & Mulyana, 2018)	Prediksi Harga Ponsel	*Random Forest*	Hasil penelitian yang didapatkan mendapatkan tingkat akurasi sebesar 81%

Berdasarkan penelitian pada tabel 2.2, *random forest* menghasilkan nilai akurasi yang tinggi. Oleh karena itu, peneliti tertarik untuk *menggunakan random forest* dalam memprediksi keterlambatan pembayaran SPP di Universitas Muhammadiyah Kalimantan Timur.



Gambar 2.2 Contoh *Random Forest*

Sumber: Putra (2019)

Langkah-langkah *random forest* sebagai berikut :

a) Melakukan pengambilan contoh acak yang berukuran n dengan perbaikan pada gugus data. Tahapan ini merupakan tahapan dari *bootstrap.*
b) Melakukan pengambilan sampel data secara acak (*random*) dengan kemungkinan pengambilan sampel data yang sama (tahapan ini disebut tahapan *bootstrap*).
c) Ulangi langkah 1 dan 2 sebanyak k kali, sehingga terbentuk sebuah hutan yang terdiri atas k pohon.

Untuk menghitung nilai *entropy* menggunakan rumus pada persamaan 2.1 dan menghitung nilai *informasi Gain* menggunakan persamaan 2.2.

$$Entropy(S) = \sum_{i=1}^{n} -pi^{*}log_{2}pi \qquad (21)$$

Keterangan :

S : Himpunan kasus

A : Fitur

n : Jumlah partisi S

pi : Proporsi dari Si terhadap S

$$Gain\ (S,A) = Entropy(S) \sum_{i=1}^{n} \frac{|\ S_i\ |}{|\ S\ |} * entropy(Si) \qquad (2.2)$$

Keterangan :

S : Himpunan kasus

A : Atribut

n : Jumlah partisi atribut A

|Si| : Jumlah kasus pada partisi ke-i

|S| : Jumlah kasus dalam S

2.6. Seleksi Atribut *Information Gain Ratio*

Seleksi atribut digunakan untuk menghapus atribut yang tidak terkait dengan klasifikasi sentimen teks untuk meningkatkan akurasi yang lebih tinggi dan untuk mengurangi waktu proses eksekusi algoritma (Grandis & Arumsari, 2021).

Gain Ratio merupakan proses untuk meningkatkan perolehan informasi dengan memberikan nilai kontribusi fitur yang dinormalisasi untuk klasifikasi yang optimal.

Berikut adalah tahapan perhitungan *gain ratio* yaitu sebagai berikut (Suyanto, 2017):

a) Menghitung nilai *Entropy*

Untuk rumus menghitung nilai *entropy* dapat dilihat pada persamaan 2.1.

b) Menghitung nilai *Information Gain*

Setelah mendapatkan nilai *entropy,* tahap selanjutnya adalah menghitung nilai *information gain. Information gain* adalah ukuran efektivitas fitur dalam mengelompokan data. Perhititungan *information gain* memerlukan inputan berupa nilai *entropy* dan nilai *information gain,* sehingga perhitungan nilai *information gain* dilakukan setelah perhitungan nilai *entropy*, dan hasilnya kemudian digunakan sebagai inputan untuk perhitungan *gain ratio.* Untuk rumus menghitung *information gain* dapat dilihat pada persamaan 2.2.

c) Menghitung nilai *gain ratio*

Untuk menghitung gain ratio memerlukan *split information*. Rumus untuk menghitung *split information* menggunakan persamaan 2.3.

$$SplitInfo(S,A) = -\sum_{j=1}^{c} \frac{|S_j|}{|S|} log_2 \frac{|S_j|}{|S|} \quad (2.3)$$

Keterangan:

S_1 sampai S_c adalah c *subset* yang dihasilkan dengan mempartisi S menggunakan fitur A dengan banyak nilai c.

Rumus untuk menghitung *gain ratio* menggunakan persamaan 2.4.

$$GainRatio(S,A) = \frac{Gain(S,A)}{SplitInfo(S,A)} \quad (2.4)$$

2.7. *Confusion Matrix*

Confusion Matrix adalah alat pengukuran yang melakukan pengukuran saat menganalisis klasifikasi. Pengklasifikasi baik dalam hal mengidentifikasi tupel dari kelas yang berbeda. Ketika mengklasifiaksi dan memiliki data bernilai benar, maka nilai *True-Positive* dan *True-Negative* berfungsi untuk memberikan informasi tersebut. Sedangkan jika pengklasifikasi memiliki kesalahan saat mengklasifikasi

data, maka nilai dari *False-Positive* dan *False-Negative* akan memberikan informasi tersebut (Putra, 2019).

Confusion Matrix digunakan untuk melihat seberapa besar akurat model klasifikasi yang didapatkan berdasarkan dari model klasifikasi yang dibuat untuk memprediksi suatu kelas berdasarkan dari data *testing.* Rincian dari hasil klasifikasi untuk prediksi kelas ditampilkan di atas, dan kelas sebenarnya ditampilkan di kiri bawah (Haristu, 2019). Bentuk *Confusion Matrix* serta penjelasannya dapat dilihat pada tabel 2.3:

Tabel 2.3 *Confusion Matrix*

Kelas Prediksi			
Kelas Aktual		Kelas = Positif	Kelas = Negatif
	Kelas = Positif	TP (*True Positive*)	FN (*False Negative*)
	Kelas = Negatif	FP (*False Positive*)	TN (*True Negative*)

Nilai akurasi dapat dihitung menggunakan rumus sebagai berikut:

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FP+FN} * 100\% \quad (3)$$

Keterangan :

True Positive : Jumlah kelas positif yang diklasifikasn sebagai positif

False Postive : Jumlah kelas negatif yang diklasifikasn sebagai positif

True Negative : Jumlah kelas positif yang diklasifikasn sebagai negatif

False Negative : Jumlah kelas negatif yang diklasifikasn sebagai negatif