

BAB II LANDASAN TEORI

2.1 Peneliti Terkait

Penelitian terkait adalah studi yang telah sudah ada sebelum penelitian ini dilakukan. Penelitian - penelitian tersebut digunakan sebagai acuan atau pedoman, dimana dari penelitian - penelitian sebelumnya dapat diambil teori - teori yang dibutuhkan[5]. Berikut penelitian terkait:

Penelitian pertama yang digunakan dalam menentukan kelayakan kredit dengan menggunakan metode *naive bayes*. Data yang digunakan adalah data Koperasi Simpan Pinjam Mekar Jaya Maleber. Tujuan dari penelitian ini adalah menentukan prioritas pemberian pinjaman kepada nasabah Koperasi Simpan Pinjam Mekar Jaya Maleber. Pada penelitian ini menggunakan 6 atribut. Hasil dari penelitian menghasilkan akurasi sebesar 76,76 % [5].

Penelitian menggunakan metode *naive bayes* dalam menentukan calon anggota kredit. Penelitian ini menggunakan 7 atribut dengan total data 522 menghasilkan akurasi terbesar 80,00 %. Atribut yang digunakan antara lain umur, jenis kelamin, jumlah pinjaman, jangka waktu pengembalian, jaminan, jenis pekerjaan, dan penghasilan[6].

Penelitian terkait ketiga adalah Dalam ini membahas tentang algoritma *naive bayes*, *decission tree*, dan SVM untuk klasifikasi persetujuan pembiayaan koperasi. Pada penelitian ini menggunakan data dari Koperasi Syariah di Jakarta dengan kriteria jenis kelamin, umur, status, status tempat tinggal, jumlah tanggungan, pendidikan, pekerjaan, penghasilan, jangka waktu. Hasil dari penelitian ini adalah SVM memiliki hasil akurasi sebesar 89,86 %, *naive bayes* 77, 29 %, dan *decission tree* 89,02 % [7].

Tabel 2. 1 Penelitian Terkait Mengenai Pemberian Pinjaman

No	Penelitian	Metode	Atribut	Akurasi
1.	Penelitian mengenai kelayakankredit nasabah pada Koperasi Simpan Pinjam Mekar Jaya Maleber	<i>Naïve Bayes</i>	6 atribut (jumlah pengajuan, jangka waktu, jaminan, pendapatan, pengeluaran, pinjaman sebelumnya) Dan 2 label ya dan tidak.	76,76 %
2.	Penelitian mengenai penentuan calon anggota kredit	<i>Naïve bayes</i>	7 atribut (umur, jenis kelamin, jumlah pinjaman, jangka waktu pengembalian, jaminan, jenis pekerjaan, dan penghasilan). Dengan 3 label yaitu lancar, kurang lancar dan macet.	80 %
3.	Klasifikasi persetujuan pembiayaan koperasi syariah di Jakarta	<i>Naïve Bayes</i> , SVM, dan Descission Tree	9 atribut (Jenis kelamin, umur, status, status tempat tinggal, jumlah tanggungan, pendidikan, pekerjaan, penghasilan, jangka waktu). Dengan 2 label lancar dan macet.	Naïve bayes 77,29 %, SVM 89,86 %, dan descission tree 89,02 %

Pada penelitian ini merupakan penelitian penentuan pemberian pinjaman pada koperasi Bumi Putera Lolowau dengan jumlah data 2000 data dan 8 atribut (umur, Jenis Kelamin, jenis pekerjaan, jumlah pinjaman, jangka waktu pengembalian, jaminan, pendapatan). Dengan 2 label yaitu, layak dan tidak layak.

2.2 Koperasi

Pengertian koperasi menurut Undang-undang No.25 Tahun 1992 adalah badan usaha yang beranggotakan orang-orang atau badan hukum koperasi, dengan melandaskan kegiatannya berdasarkan prinsip koperasi sekaligus sebagai Gerakan ekonomi rakyat yang berdasar atas asas kekeluargaan yang bertujuan untuk memajukan kesejahteraan anggota pada khususnya dan masyarakat pada umumnya serta turut serta membangun tatanan perekonomian nasional dalam rangka mewujudkan masyarakat yang maju, adil dan Makmur[8].

2.2.1 Prinsip – Prinsip Pemberian Pinjaman

Sebelum memberikan jalur kredit, debitur harus memastikan bahwa pinjaman yang ditawarkan benar-benar dilunasi. Sebuah keyakinan diperoleh dari peringkat kredit pra-pembayaran. Saat mengevaluasi kriteria dan aspek, skor tetap sama. Demikian pula, skala mapan telah menjadi kriteria penilaian semua lembaga keuangan. Secara umum kriteria penilaian yang harus dipenuhi debitur untuk mendapatkan nasabah yang benar-benar menguntungkan dilakukan dengan menggunakan analisis 5C dan 7P [9]. analisis 5C antara lain :

1. *Character*

Karakter dan temperamen orang yang dapat dipercaya benar-benar dapat tercermin dari latar belakangnya, baik dalam kehidupan profesional maupun pribadimereka. Contoh: Hobi.

2. *Capacity*

Peluang Bisnis juga diukur pada berdasarkan kemampuan mereka untuk memahami peraturan pemerintah untuk memastikan kemampuan klien.

3. *Capital*

Untuk memastikan penggunaan modal yang efisien, laporan keuangan (neraca dan laporan laba rugi) ditinjau menggunakan indikator seperti likuiditas, solvabilitas, profitabilitas dan indikator lainnya. Penting juga untuk menentukan sumber pendanaan mana yang saat ini tersedia.

4. *Collaterall*

Yang dimaksud dengan *collateral* disini merupakan jaminan dari calon peminjam. Hal tersebut juga menjadi bahan pertimbangan untuk penentuan pemberian pinjaman.

5. *Condition*

Yang dimaksud dengan kondisi ini merupakan kondisi perekonomian. Kondisi perekonomian calon peminjam juga termasuk menjadi perhatian khususnya untuk debitur.

2.2.2 Koperasi Bumi Putera Lolowau

Koperasi Bumi Putera Lolowau merupakan koperasi yang di dirikan di Desa Nituwuboho, Kecamatan Lolowau, Kabupaten Nias Selatan pada tahun 1895. Dalam perkembangannya hingga saat ini koperasi Bumi Putera Lolowau, dalam prediksi kelayakkan pemberian pinjaman secara *self-adjustment* atau penilaian secara langsung dari pihak pengurus dengan mempertimbangkan 7 atribut, diantaranya Jenis Kelamin, umur, jenis pekerjaan, jumlah pinjaman, jangka waktu pengembalian, jaminan, pendapatan pada koperasi dan status keanggotaan. Namun, dalam pelaksanaannya pemberian pinjaman kepada anggota dengan sel-adjustment ternyata tidak berjalan sesuai yang diharapkan. Terdapat pengambilan keputusan yang tidak sesuai bahkan dapat dinilai tidak adil [10]. Sehingga, terdapat kemacetan dalam pembayaran angsuran. Hal tersebut yang disebabkan kurang akuratnya management pihak koperasi dalam pemberian pinjaman kepada anggota. Oleh karena itu, penulis menggunakan 7 atribut tersebut dalam penelitian ini dalam membuat aplikasi prediksi kelayakkan pemberian pinjaman dengan kategori layak dan tidak layak.

2.3 Prediksi

Pengertian Prediksi sama dengan ramalan atau perkiraan, Prediksi adalah suatu proses memperkirakan secara sistematis tentang sesuatu yang paling mungkin terjadi di masa depan berdasarkan informasi masa lalu dan sekarang yang dimiliki, agar selisih antara sesuatu yang terjadi dengan hasil perkiraan dapat diperkecil. Prediksi tidak harus memberikan jawaban secara pasti kejadian yang akan terjadi, melainkan berusaha untuk mencari jawaban sedekat mungkin yang akan terjadi [11].

2.4 Pinjaman

Pengertian pinjaman menurut peraturan pemerintahan Republik Indonesia Nomor 9 Tahun 1995 tentang kegiatan usaha Simpan Pinjam, yaitu sebagai berikut: “pinjaman uang atau tagihan yang dapat dipersamakan dengan itu, berdasarkan persetujuan atau kesepakatan pinjam-meminjam antara koperasi dengan pihak lain

yang mewajibkan pihak peminjam untuk melunasi hutangnya setelah jangka waktu tertentu dan disertai dengan pembayaran sejumlah imbalan”. Dalam lingkup koperasi pinjaman adalah merupakan sejumlah dana yang dipinjamkan oleh koperasi kepada anggota, dan anggota wajib mengembalikannya dalam suatu jangka waktu tertentu melalui angsuran pembayaran berupa pokok pinjaman ditambah dengan bunga pinjaman[12].

2.5 Algoritma Naives Bayes

Merupakan salah satu algoritma yang terdapat pada Teknik klasifikasi. *Naïve Bayes* merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuan inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman dimasa sebelumnya sehingga dikenal sebagai teorema *Bayes*. Teorema tersebut dikombinasikan dengan *naïve* dimana diasumsikan kondisi antar atribut saling bebas[13].

Klasifikasi *naïve bayes* diasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada hubungannya dengan ciri dari kelas lainnya. Prediksi *bayes* didasarkan pada formula teorema *bayes* dengan formula umum sebagai berikut:

$$P(C|X) = \frac{P(X|C) \times P(C)}{P(X)}$$

Dimana :

X : Data dengan kelas yang belum diketahui

C : Hipotesis data X merupakan suatu kelas spesifik

P(C|X) : Probabilitas Hipotesis C berdasar kondisi X (probabilitas posteriori)

P(C) : Probabilitas hipotesis C (probabilitas prior)

P(X|C) : Probabilitas X berdasarkan kondisi pada hipotesis C

P(X) : Probabilitas X

Untuk menjelaskan teorema *naïve bayes* perlu diketahui bahwa proses kalsifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut. Karena itu, teorema bayes di atas di sesuaikan sebagai berikut :

$$P(C|X_1 \dots X_n) = \frac{P(C)P(X_1 \dots X_n|C)}{P(X_1 \dots X_n)}$$

Dimana variable C mempresentasikan kelas, sementara variable $X_1 \dots X_n$ merepresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi atau kriteria. Maka rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas C (Posterior) adalah peluang munculnya kelas C (sebelum masuknya sampel tersebut, seringkali disebut prior), dikali dengan peluang kemunculan karakteristik – karakteristik sampel pada kelas C (disebut juga *likelihood*), dibagi dengan peluang kemunculan karakteristik – karakteristik sampel secara global (disebut juga *evidence*). Karena itu, rumus diatas dapat pula ditulis secara sederhana sebagai berikut :

$$\text{Posterior} = \frac{\text{Prior} \times \text{likelihood}}{\text{Evidence}}$$

Nilai *Evidence* selalu tetap untuk setiap kelas pada satu sampel. Nilai dari posterior tersebut nantinya akan dibandingkan dengan nilai – nilai posterior kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan. Penjabaran lebih lanjut rumus *bayes* tersebut dilakukan dengan menjabarkan $(C|X_1 \dots X_n)$ menggunakan aturan perkalian sebagai berikut :

$$P(C|X_1, \dots, X_n) = P(C) P(X_1, \dots, X_n|C)$$

$$\begin{aligned} &P(C) P(X_1|C)P(X_2, \dots, X_n|C, X_1) \\ &P(C) P(X_1|C)P(X_2|C, X_1)P(X_3, \dots, X_n|C, X_1, X_2) \\ &P(C) P(X_1|C)P(X_2|C, X_1)P(X_3, \dots, X_n|C, X_1, X_2, X_3) \\ &P(C) P(X_1|C) \dots P(X_n|C, X_1, X_3, \dots, X_{n-1}) \end{aligned}$$

Dapat dilihat bahwa hasil penjabaran tersebut menyebabkan semakin banyak dan semakin kompleksnya faktor – faktor syarat yang mempengaruhi nilai probabilitas, yang hampir mustahil untuk dianalisis satu persatu. Akibatnya, perhitungan tersebut menjadi sulit untuk dilakukan. Disinilah digunakan asumsi independensi yang sangat tinggi (naif), bahwa masing – masing petunjuk

(X_1, \dots, X_n) saling bebas (independen) satu sama lain. Dengan asumsi tersebut, maka berlaku suatu kesamaan sebagai berikut:

$$P(X_i | X_j) = \frac{P(X_i \cap X_j)}{P(X_j)} = \frac{P(X_i)P(X_j)}{P(X_j)}$$

Untuk $i \neq j$, sehingga: $P(X_i | C, X_j) = P(X_i | C)$

Dari persamaan di atas dapat disimpulkan bahwa asumsi independensi naif tersebut membuat syarat peluang menjadi sederhana, sehingga perhitungan menjadi mungkin untuk dilakukan.

2.6 Python

Python adalah bahasa pemrograman yang ditafsirkan, tingkat tinggi, dan tujuan umum. Dibuat oleh Guido Van Rossum dan pertama kali rilis pada Tahun 1991, Filosofi desain *python* menekankan keterbacaan kode dengan penggunaan spasi putih yang signifikan. konstruksi bahasanya dan pendekatan berorientasi objek bertujuan untuk membantu pemrograman menulis kode yang jelas dan logis untuk proyek skala kecil dan besar[13].

Python juga merupakan Bahasa pemrograman yang berbasis OOP yang mampu mengimplementasikan arsitektur MVC (*model view controller*). MVC merupakan sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen – komponen aplikasi, seperti : manipulasi data, *controller*, dan *user interface*

2.7 Django Framework

Django adalah sebuah web *framework* yang menggunakan bahasa pemrograman *Python* yang mendukung pembuatan website dengan konsep *rapid development*[14]. Wiki-Python menyatakan bahwa *framework* yang sekarang telah berada di versi 2.1 ini sebagai web *framework* paling populer dikalangan *high-level frameworks*. Karena menggunakan bahasa pemrograman *python* yang telah memiliki banyak library siap pakai, mengembangkan sistem dengan *django* berarti dapat menggunakan *library* tersebut sesuai dengan keperluan

pengembangan. *Django* bertujuan untuk memudahkan pengembangan situs web dan basis data yang kompleks. Salah satu keunggulan *django* adalah, *framework* ini telah merepresentasikan ORM (*Object Relational Mapper*) sehingga tidak perlu menyesuaikan query jika terjadi perubahan database yang digunakan. Berikut ini adalah beberapa keuntungan menggunakan *Django*:

1. *Object-Relational Mapping (ORM) Support.*
2. *Automatic Admin Interface.*
3. *Elegant URL Design.*
4. *Template System.*
5. *Cache system.*
6. *Internationalization.*
7. *A light weight web server for development and test.*

2.7.1 Peralatan yang digunakan untuk membuat *Django*

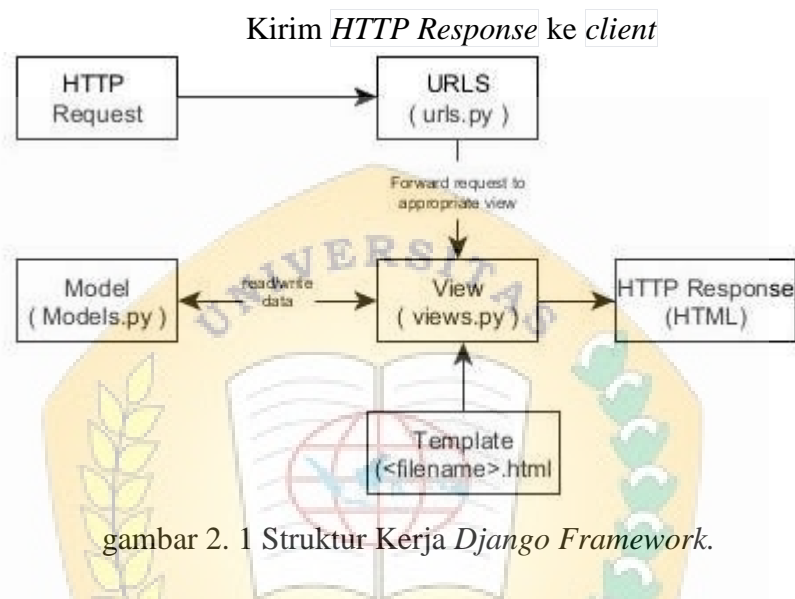
- Virtualenv
Virtualenv adalah sebuah *tool* yang berfungsi untuk membuat lingkungan virtual yang terisolasi
- Pip.
sebuah *tool* yang akan kita gunakan untuk manajemen paket *python*. Termasuk juga menginstall *Django*.

2.7.2 Struktur Direktori *Project Django*

- Direktori *mysite/* , adalah *root* direktori yang berisi seluruh file dari project.
- File *manage.py* , program untuk mengelolah *project Django*.
- File *mysite/__init.py*, program untuk mengelolah *project Django*.
- File *mysite/settings.py*, tempat mengkonfigurasi *project*.
- File *mysite/Urls.py*, tempat untuk mendeklarasikan *URLS*.
- File *mysite/wsgi.py*, *entri point* untuk *WSGI – compatible*.

2.7.3 Konsep Kerja Django

Setiap *request* dari *client* akan diproses pertama kali oleh *urls.py*, karena di sini berisi definisi alamat URL (*route*) dan fungsi yang akan dieksekusi di setiap rute. Berikutnya, fungsi yang ada di *views.py* akan melakukan pemrosesan seperti: Tulis data atau ambil data dari Model Racik tampilan data dengan template *HTML*[15].



gambar 2. 1 Struktur Kerja *Django Framework*.

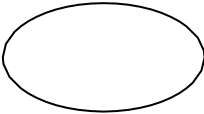
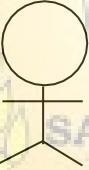
2.8 Alat Bantu Analisa


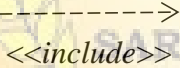
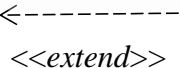
Unified Modelling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak[16]. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem

a. *Use case diagram*

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi – fungsi tersebut.

Tabel 2. 2 Simbol *Use Case Diagram*






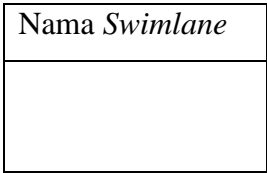
Symbol	Keterangan
	<p><i>Use Case</i></p> <p>menggambarkan fungsionalitas yang disediakan sistem sebagai unit – unit yang bertukar pesan antar unit dengan akhir, yang dinyatakan dengan menggunakan kata kerja.</p>
	<p><i>Actor</i></p> <p>Aktor adalah <i>Abstraction</i> dari orang atau sistem lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi n akhir, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem . Orang atau sistem bisa muncul dalam beberapa peran, perlu dicatat bahwa aktor berinteraksi dengan <i>Use Case</i>, tetapi tidak memiliki kontrol terhadap <i>use case</i>.</p>

	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanda panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data.</p>
	<p>Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.</p>
	<p><i>Include</i> Merupakan di dalam <i>use case</i> lain (required) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.</p>
	<p><i>Extend</i> Merupakan perluasan dari <i>usecase</i> lain jika kondisi atau syarat terpenuhi.</p>

b. Diagram Aktivitas (*Activity Diagram*)

Aktivitas Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Adapun simbol – simbol *activity Diagram* dapat dilihat pada tabel 2.3

Tabel 2. 3 Simbol *Activity Diagram*

No	Simbol	Fungsi
1		Status Awal/Initial Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2		Aktivitas / <i>Activity</i> Aktivitas yang dilakukan sistem, biasanya diawali dengan kata kerja.
3		Percabangan/ <i>Decision</i> Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu.
4		Penggabungan/ <i>Join</i> Asosiasi Penggabungan Dimana Lebih Dari Satu Aktvitas Lebih Dari Satu.
5		Status Akhir (<i>Final</i>) Status Akhir Yang Dilakukan Sistem, Sebuah Diagram Aktivitas Memiliki Sebuah Status Satu.
6		<i>Swimlane</i> Memisahkan organisasi yang bertanggung jawab terhadap aktivitas yang terjadi.

c. *Class Diagram*

Class Diagram menggambarkan jenis objek dalam sistem dan berbagai jenis hubungan statis yang ada di antara mereka. *Class Diagram* juga menunjukkan sifat – sifat dan operasi dari sebuah kelas dan kendala yang berlaku untuk cara objek yang terhubung.

Tabel 2. 4 *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku ada di atasnya objek induk (<i>anscestor</i>)
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek
6		<i>Dependeng</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

2.9 Pengujian *White-Box*

White-Box Testing adalah tes yang menggunakan kompleksitas siklomatik. Salah satu cara untuk mengujinya adalah dengan melihat kode yang ada dengan melihat modul dan menganalisis kesalahannya. Jika keluaran yang dihasilkan oleh modul tidak sesuai dengan proses yang benar. Baris program, variabel dan parameter yang terdapat dalam unit kemudian diperiksa atau dikoreksi satu per satu dan kemudian dikompilasi ulang[17].

2.10 Pengujian *Black-Box*

Black-Box merupakan metode pengujian perangkat lunak tanpa memperhatikan detail perangkat lunak. Pengujian ini hanya memeriksa nilai keluaran berdasarkan masing-masing nilai masukan. Tidak ada upaya yang dilakukan untuk mengetahui kode program mana yang digunakan hasilnya. Proses pengujian *black box* dilakukan dengan menguji aplikasi yang dibangun dengan mencoba memasukkan data pada setiap halaman. Tes ini digunakan untuk menentukan apakah aplikasi berfungsi sesuai kebutuhan[18].

