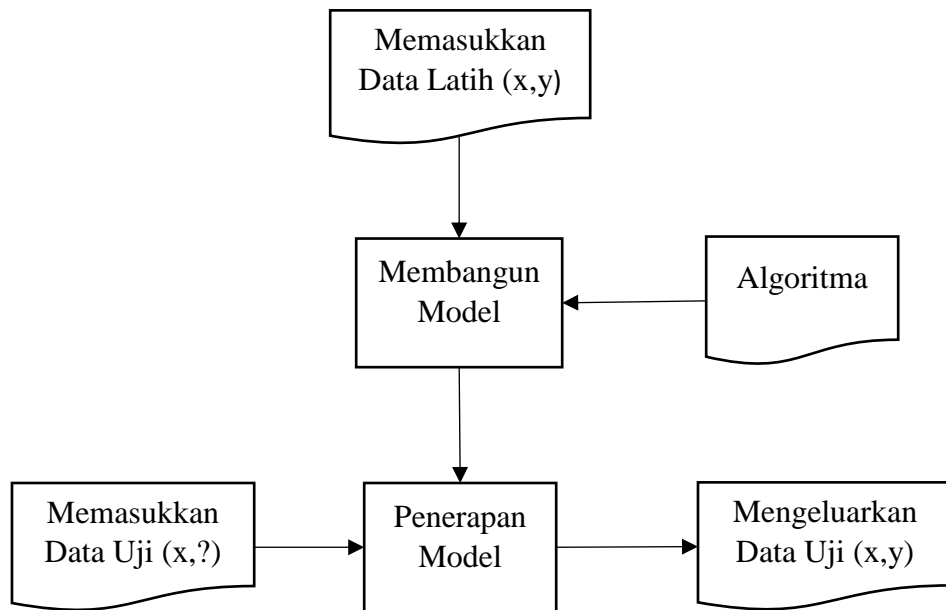


BAB II

LANDASAN TEORI

2.1 Konsep Klasifikasi

Klasifikasi dapat didefinisikan secara detail sebagai suatu pekerjaan yang menggunakan pelatihan atau pembelajaran terhadap fungsi target f yang memetakan setiap vektor (sub fitur) x ke dalam satu dari sejumlah label kelas y yang tersedia. Pekerjaan pelatihan tersebut akan menghasilkan suatu model yang kemudian disimpan sebagai memori (Prasetyo, 2012).



Gambar 2.1 Proses Klasifikasi .

Kerangka kerja klasifikasi ditunjukkan pada gambar 2.2 Pada gambar tersebut disediakan sejumlah data latih (x,y) untuk digunakan sebagai data pembangun model, kemudian menggunakan model tersebut untuk memprediksi kelas dari data uji $(x, ?)$ sehingga data uji $(x, ?)$ diketahui kelas y yang seharusnya.

Model yang sudah dibangun pada saat pelatihan kemudian dapat digunakan untuk memprediksi label kelas dari data baru yang belum diketahui label kelasnya. Dalam pembangunan model selama proses pelatihan tersebut diperlukan adanya suatu algoritma untuk membangunnya yang disebut sebagai algoritma pelatihan. Ada banyak algoritma pelatihan yang sudah dikembangkan oleh para peneliti seperti *Decision Tree*, *K-Nearest Neighbor*, *Artificial Neural Network* dan sebagainya. Setiap algoritma mempunyai prinsip yang sama yaitu melakukan suatu pelatihan sehingga di akhir pelatihan model dapat memetakan (memprediksi) setiap vektor masukan ke label keluaran yang benar.

2.2 Algoritma Naïve Bayes

Algoritma *Naive Bayes* berbasiskan perhitungan probabilistik dengan asumsi bahwa setiap fitur yang digunakan saling lepas. (Putri, Suparti dan Rahmawati, 2014) Menyatakan *Naive Bayes* merupakan metode klasifikasi teks yang paling populer digunakan. Algoritma ini memiliki kelebihan dari sisi kecepatan pembelajaran dan toleransinya terhadap nilai yang hilang dari fitur. Untuk menangani data numerik, algoritma ini menggunakan *probability density function*, artinya data dianggap mengikuti distribusi normal untuk kemudian dihitung nilai rata-rata dan simpangan bakunya.

Untuk merepresentasikan sebuah kelas, terdapat karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi yang berguna untuk menjelaskan bahwa peluang masuknya sampel karakteristik tertentu kedalam kelas *posterior*. Peluang munculnya suatu kelas (sebelum masuknya sampel tersebut, seringkali disebut *prior*), dikali dengan peluang kemunculan karakteristik sampel secara global disebut juga *evidence*. Nilai *evidence* selalu tetap untuk setiap kelas pada satu sampel. Nilai *posterior* tersebut dibandingkan dengan nilai *posterior* kelas lainnya untuk menentukan ke kelas apa suatu sampel .

Klasifikasi Naive Bayes diasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada hubungannya dengan ciri dari kelas lainnya. Persamaan dari teorema Bayes adalah sebagai berikut:

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)}$$

..... (2-1)

Dengan :

X = Data dengan kelas yang belum diketahui;

H = Hipotesis data X merupakan suatu label kelas tertentu;

$P(H|X)$ = Probabilistik hipotesis H berdasarkan kondisi X (*posteriori probability*); $P(H)$: Probabilistik hipotesis H (*prior probability*);

$P(X|H)$ = Probabilitas X berdasarkan kondisi pada hipotesis H;

$P(X)$ = Probabilistik X;

Untuk menjelaskan teorema *naive bayes*, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut. Karena itu, teorema *bayes* tersebut akan disesuaikan sebagai berikut:

$$P(C|F_1 \dots F_n) = \frac{P(C)P(F_1 \dots F_n|C)}{P(f_1 \dots F_n)}$$

..... (2-2)

Dengan :

C = Sebuah kelas; $F_1 \dots F_n$ = Karakteristik Petunjuk.

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

..... (2-3)

Penjabaran lebih lanjut sebagai Berikut :

$$\begin{aligned}
 P(C|F_1, \dots, F_n) &= P(C) P(F_1, \dots, F_n | C) \\
 &= P(C) P(F_1|C) \\
 &= P(C) P(F_1|C) \\
 &= P(C)P(F_1|C)P(F_2|C,F_1)P(F_3|C,F_1,F_2),P(F_4, \dots, F_n|C,F_1,F_2) \\
 &= P(C)P(F_1|C)P(F_2|C,F_1)P(F_3|C,F_1,F_2), \dots, (F_n|C,F_1,F_2, \dots, F_{n-1})
 \end{aligned}$$

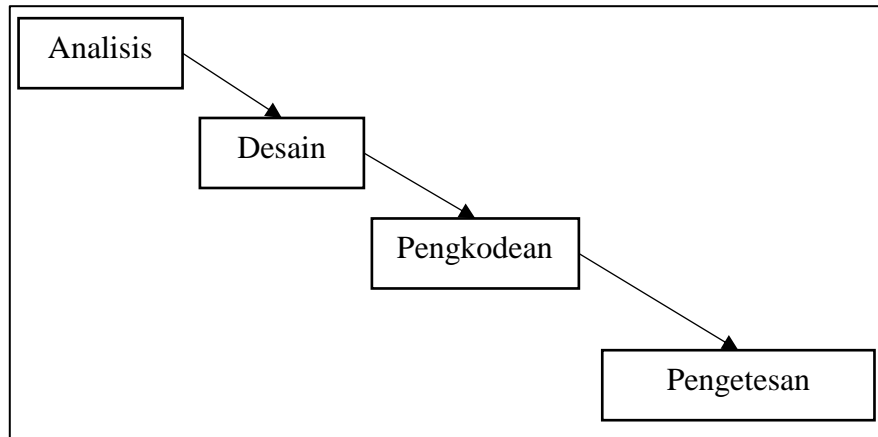
Adapun langkah – langkah pada *naïve bayes* yang akan dilakakukan menggunakan rumus di atas sebagai berikut (Suyanto, 2017) :

1. Penentuan atribut yang akan digunakan .
2. Penentuan Data Set Training.
3. Lakukan Perhitungan Jumlah Kasus Pada Atribut .
4. Tentukan Kasus Baru yang akan di klasifikasikan menggunakan metode naive bayes.
5. Klasifikasikan Kasus baru berdasarkan kasus yang sama dengan kasus yang lama.
6. Kalikan semua Kelas Variable untuk mendapatkan nilai dari masing masing kelas.
7. Bandingkan Hasil perkalian dari masing masing kelas makan akan di ketahui konsentrasi skripsi intelegent system atau multimedia berdasarkan hasil nilai terbesar dari masing-masing kelas.
8. Lakukan Rekomendasi Bahasa Pemrograman berdasarkan hasil perbandingan antara konsentrasi skripsi intelegent system dan multimedia.

2.3 Metode Pengembangan Waterfall

SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*clasic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau berturut dimulai dari

analisis, desain, pengkodean, pengujian dan tahap pendukung (*support*) (Ananto dan Fitria, 2017). Berikut ini adalah gambar model air terjun :



Gambar 2.2 Metode *Waterfall*.

1) Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2) Desain

Desain perangkat lunak proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengkodean. Tahap ini mentranslasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan (Fitria dan YP, 2014).

3) Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Perancangan sistem yang akan dibuat adalah untuk menghasilkan suatu program yang kemudian dapat diimplementasikan pada tahap implementasi sistem (Irianto dan Fitria, 2016).

4) Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahn (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan. Tahap ini merupakan tahapan untuk menerjemahkan data atau pemecahan masalah yang telah dirancang kedalam bahasa pemrograman komputer yang telah ditentukan. Semua tahapan ini desain perangkat lunak sebagai sebuah program lengkap (Fitria dan Arfida, 2015).

2.4 Basis Data

Sebuah Basis Data adalah sebuah struktur yang umumnya dikategorikan dalam hal Sebuah Basis Data flat dan sebuah Basis Data relasional. Basis data relasional lebih disukai karena lebih masuk akal dibandingkan Basis data flat Ada tabel-tabel yang menyimpan data. Setiap tabel terdiri dari kolom dan baris. Sebuah kolom mendefinisikan jenis informasi apa yang akan disimpan. Diperlukan kolom khusus untuk setiap jenis informasi yang ingin di simpan (Ananto dan Fitria, 2017).

Kalau kolom mendefinisikan jenis informasi apa yang akan disimpan, maka sebuah baris adalah data aktual yang disimpan. Setiap baris dari tabel adalah masukan dari tabel tersebut dan berisi nilai-nilai untuk setiap kolom tabel tersebut.

2.4.1 Data Base Management System (DBMS)

DBMS (*Database Management System*) adalah suatu sistem aplikasi yang digunakan untuk meyimpan, mengelola, dan menampilkan data.

(Rosa dan Salahuddin, 2011) Suatu sistem aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut :

- a. Menyediakan fasilitas untuk mengelola akses data
- b. Mampu menangani integritas data
- c. Mampu menangani akses data
- d. Mampu menangani backup data

Karena pentingnya data bagi suatu organisasi atau perusahaan, maka hampir sebagian besar perusahaan memanfaatkan DBMS dalam mengelola data yang mereka miliki. Pengelola DBMS sendiri biasanya ditangani oleh tenaga ahli yang spesialis menangani DBMS yang disebut sebagai DBA (Database Administrator).

Berikut ini adalah 4 macam DBMS versi komersial yang paling banyak digunakan di dunia saat ini, yaitu :

- a. Oracle
- b. Microsoft SQL Server
- c. IBM DB2
- d. Microsoft Access

Sedangkan DBMS versi *open source* yang cukup berkembang dan paling banyak digunakan saat ini adalah MySQL dan Sqlite. Hampir semua DBMS mengadopsi SQL sebagai bahasa untuk mengelola data pada DBMS.

2.4.2 Structured Query Language (SQL)

Menurut (Arief M Rudianto, 2011) SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada DBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus. MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (DBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang

bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya yakni SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (Fitria dan Arfida, 2015).

2.4.3 Conceptual Data Model (CDM)

CDM adalah model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas serta hubungan antara entitas-entitas itu. Biasanya CDM direpresentasikan dalam bentuk Entity Relationship Diagram. Adapun manfaat penggunaan CDM dalam perancangan database dapat memberikan gambaran yang lengkap mengenai struktur basis data yaitu arti, hubungan, dan batasan-batasan dalam memodelkan struktur logis dari keseluruhan aplikasi data, tidak tergantung pada software atau pertimbangan model struktur data serta menggambarkan secara detail struktur basis data dalam bentuk logik (Jogiyanto, 2005).

Jenis – jenis objek dalam CDM

1. Entity
2. Relationship
3. Inheritance


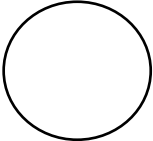

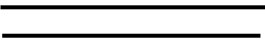
2.4.4 Data Flow Diagram (DFD)

Data Flow Diagram adalah suatu model logika data atau proses yang dibuat lebih mendetail dibanding diagram konteks yang diperbolehkan, bisa dicapai dengan mengembangkan diagram (Mustika dan Fitria, 2014). Sisa diagram asli dikembangkan ke dalam gambaran yang lebih terperinci yang melibatkan tiga sampai sembilan proses dan menunjukkan penyimpanan data dan aliran data baru pada level yang lebih rendah.

DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level yang lebih detail untuk merepresentasikan 16 aliran informasi lebih detail. DFD menyediakan mekanisme untuk pemodelan

fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi pada perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur dengan fungsi dan prosedur.

Tabel 2.1 Simbol-simbol DFD

Komponen	Keterangan
	Entitas adalah Kesatuan di lingkungan luar sistem yang dapat berupa orang atau organisasi atau sistem lainnya yang berada di lingkungan luarnya yang memberikan input atau menerima output dari sistem
	Proses adalah kegiatan atau aktivitas yang dilakukan oleh mesin atau komputer dari suatu masukan yang akan masuk ke proses dan menghasilkan keluaran proses.
	Garis atau data flow menyimbolkan aliran arus data
	Penyimpanan data dalam database, biasanya berupa tabel

2.5 Aplikasi Website

Menurut (Arief M Rudianto, 2011) Website adalah "kumpulan dari halaman web yang sudah dipublikasikan di jaringan internet dan memiliki domain/URL (Uniform Resource Locator) yang dapat diakses semua pengguna internet dengan cara mengetikkan alamatnya". Hal ini dimungkinkan dengan adanya teknologi World Wide Web (WWW) fasilitas hypertext guna menampilkan data berupa teks, gambar, animasi, suara dan multimedia lainnya data tersebut dapat saling pada web server untuk

dapat di akses melalui jaringan internet. Agar data pada web dapat di baca kita harus menggunakan web server terlebih dahulu seperti Mozilla Firefox, Internet Explorer, Opera Mini atau yang lainnya.

2.5.1 HTML

Menurut (Kadir, 2009) HTML (*Hypertext Markup Language*) adalah bahasa format khusus yang programmer gunakan untuk memformat dokumen untuk ditampilkan di web. Lalu menurut (Adi Nugroho, 2006) HTML atau *Hyper Text Markup language* adalah halaman standar web presentasi dengan menggunakan format untuk menentukan pesan dan tata letak halaman web. Berdasarkan pendapat para ahli di atas, penulis menyimpulkan bahwa HTML atau *Hyper Text Markup Language* adalah bahasa pemrograman yang digunakan untuk memformat dokumen untuk ditampilkan di web.

2.5.2 Javascript

Menurut (Kadir, 2009) menyatakan bahwa Javascript adalah bahasa skrip (bahasa yang kodenya ditulis menggunakan teks biasa) yang ditempelkan pada dokumen HTML dan diproses pada sisi klien. Dengan adanya bahasa ini, kemampuan dokumen HTML menjadi semakin luas. Sebagai contoh, dengan menggunakan JavaScript dimungkinkan untuk memvalidasi masukan – masukan pada formulir sebelum formulir dikirim ke *server*. Selain itu, dengan menggunakan JavaScript juga dimungkinkan untuk mengimplementasikan tugas yang bersifat interaktif tanpa berhubungan dengan *server*. Beberapa contoh yang bisa dilakukan melalui JavaScript :

1. Menampilkan jam lokal pada halaman web.
2. Mengatur warna latar belakang halaman web.
3. Mengganti gambar ketika pemakai menempatkan petunjuk *mouse* ke suatu gambar.
4. Memvalidasi keabsahan data yang dimasukan oleh pemakai.

2.5.3 PHP

PHP adalah Bahasa server-side –scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis (Arief M Rudianto, 2011) . Karena PHP merupakan server-side-scripting maka sintaks dan perintah-perintah PHP akan dieksekusi diserver kemudian hasilnya akan dikirimkan ke browser dengan format HTML.

Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman web lebih terjamin. PHP dirancang untuk membuat halaman web yang dinamis, yaitu halaman web yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman web.

PHP juga dapat berjalan pada berbagai web server seperti IIS (Internet Information Server), PWS (Personal Web Server), Apache, Xitami. PHP juga mampu berjalan di banyak sistem operasi yang beredar saat ini, diantaranya : Sistem Operasi Microsoft Windows (semua versi), Linux, Mac Os, Solaris. PHP dapat dibangun sebagai modul web server Apache dan sebagai binary yang dapat berjalan sebagai CGI (Common Gateway Interface). PHP dapat mengirim HTTP header, dapat mengatur cookies , mengatur authentication dan redirect user.

Salah satu keunggulan yang dimiliki PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam software sistem manajemen basis data atau Database Management Sistem (DBMS), sehingga dapat menciptakan suatu halaman web dinamis. PHP mempunyai koneksitas yang baik dengan beberapa DBMS seperti Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server, Solid, PostgreSQL, Adabas, FilePro, Velocis, dBase, Unix dbm, dan tidak terkecuali semua database ber-interface ODBC.

2.5.4 CSS

Casading Style Sheet (CSS) adalah suatu teknologi yang digunakan untuk memperindah halaman *website* (Arief M Rudianto, 2011). CSS mempunyai 2 bagian utama yaitu selectors dan deklarasi. Yang dimaksud selectrors biasanya element HTML yang ingin diubah, sedangkan deklarasi biasanya terdiri dari properti dan nilai. Properti sendiri adalah atribut style yang ingin diubah, dan setiap properti memiliki nilai. *Cascading Style Sheet* (CSS) merupakan aturan untuk mengatur bebrapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam.

2.5.5 MySQL

MySQL adalah sebuah program *databaseserver* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, serta menggunakan perintah standar SQL (*Structured Query Language*). *MySQL* juga telah mendukung bahasa pemrograman berfitur API seperti Java sehingga memudahkan para programmer java untuk berkoneksi dengan menggunakan *MySQL* (Arief M Rudianto, 2011). Berikut beberapa keistimewaan yang dimiliki *MySQL* antara lain:

1. *Portabilitas* yaitu *MySQL* dapat berjalan secara stabil pada berbagai macam sistem operasi.
2. *Open Source* yaitu *MySQL* di distribusikan secara gratis.
3. *Multi User* yaitu *MySQL* dapat digunakan dengan banyak *user* tanpa memiliki masalah.
4. *Performance Turning* yaitu *MySQL* memliki tipe kolom yang sangat kompleks.
5. Perintah dan fungsi yaitu *MySQL* operator dan fungsi yang secara penuh perintah *Select* dan *Where* dalam *query*.
6. Keamanan yaitu *MySQL* memiliki beberapa lapisan keamanan dengan sistem perizinan yang detail serta sandi yang terenkripsi.
7. Skalabilitas dan pembatasan yaitu *MySQL* mampu menangani *database* berskala besar dengan jumlah *record* lebih dari 50 juta, tabel lebih dari 60 ribu dan baris lebih dari 5 milyar.

8. *Konektivitas* yaitu *MySQL* dapat melakukan koneksi dengan klien menggunakan *TCP/IP, UNIX dan Named Pipes*.
9. *Lokalisasi* yaitu *MySQL* dapat mendeteksi kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa meskipun bahasa Indonesia belum masuk di dalamnya.
10. *Antarmuka* yaitu *MySQL* memiliki antarmuka yang mendukung berbagai bahasa pemrograman yang menggunakan fungsi API.
11. *Klien dan Peralatan* yaitu *MySQL* dilengkapi dengan berbagai *tool* yang dapat digunakan secara *online*.

2.6 Pengujian Kotak Hitam (*Black Box Testing*)

Black-box testing yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program (Pressman, 2010). Pengujian dimaksudkan untuk mengetahui apakah fungsi – fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misal untuk proses Mesin ATM maka kasus uji yang dibuat adalah

1. Jika Pin yang di masukkan benar maka akan masuk ke menu ATM
2. Jika Pin yang di masukkan salah maka akan meminta Pin Kembali

2.7 Penelitian Terkait

Penelitian yang terkait sebelumnya menggunakan beberapa jurnal yang terkait dengan penelitian ini yang dapat dilihat pada tabel 2.2.

Tabel 2.2 Penelitian Terkait

No	Judul	Nama	Tahun	Kelebihan	Kekurangan
1	Penentuan Peminatan Tugas Akhir Mahasiswa Teknik Informatika Unsika	Sofi Defiyanti	2015	Dengan Sisem pakar , Sistema dapat Menentukan Peminatan Tugas Akhir Mahasiswa Teknik Informatika	Peneltian ini hanya berdasarkan tiga matakuliah yang di ambil dari semester awal sampai dengan semester enam.

Tabel 2.2 (Lanjutan)

No	Judul	Nama	Tahun	Kelebihan	Kekurangan
2	Sistem Rekomendasi Penentuan Judul Skripsi menggunakan Algoritma Decision Tree	A Sofalul Khazari & Fitri Marisa	2017	Sistem Dapat Menentukan Topik Skripsi Menggunakan Algoritma ID3	Penelitian ini hanya berdasarkan pengelompokan data sampel yang ada jika ada nya penambahan data maka hal sistem tidak mendukung.
3	Sistem Pakar Pemilihan Topik Judul Skripsi dengan Metode Forward Chining	Ricky Tri Utomo & Yo Ceng Giap	2017	Mahasiswa dapat menentukan topik judul skripsi dengan menggunakan program sistem pakar pemilihan topik judul skripsi berbasis website	penelitian ini hanya dilakukan berdasarkan sebuah pertanyaan yang sudah tersedia oleh sistem.
4	Implementasi Sistem Penentu Skripsi Mahasiswa STIKOM Jambi dengan Em-Clustering	Errissya Rasywir	2017	Menentukan Topik Skripsi dan Rekomendasi Judul Skripsi yang telah ada	Penelitian ini hanya dilakukan berdasarkan beberapa referensi topik yang ada, dan kurangnya evaluasi sistem.
5	Penggunaan Metode Fuzzy Inference System (FIS) Mamdani dalam pemilihan Peminatan Mahasiswa untuk Tugas Akhir	Yulmaini	2015	Dari hasil pengujian menggunakan metode fuzzy mamdani dapat menghasilkan sebuah system yang digunakan untuk menentukan peminatan dalam pemilihan tugas akhir.	penelitian ini dilakukan dengan program sederhana yang diharapkan untuk dikembangkan menggunakan program yang lain untuk mencapai suatu keakuratan data sehingga dapat menjaga dan memelihara keakuratan data