

BAB III

ANALISA DAN PERANCANGAN SISTEM

3.1. Analisa Sistem

Permasalahan yang ditimbulkan dalam klasifikasi judul tugas akhir adalah bagaimana cara menggolongkan judul ke dalam setiap topik yang telah ditentukan. Pada tahapan pengumpulan data dilakukan pengambilan judul tugas akhir dalam studi kasus kali ini akan diambil dari <http://monta.if.its.ac.id/> yang merupakan sistem informasi akademik dari program studi Teknik Informatika di Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Setelah itu judul tugas akhir akan diolah kembali untuk diambil beberapa judul yang telah diberi label topik yang nantinya akan disimpan ke dalam *database* untuk dijadikan acuan dalam menentukan topik dari setiap judul tugas akhir yang diinputkan. Tentunya jika kita ingin mengklasifikasikan judul tugas akhir kita harus bisa membedakan dari setiap topik judul tugas akhir, sehingga pada saat melakukan klasifikasi judul tugas akhir tidak terjadi kesalahan.

Sistem akan dibagi dalam 4 tahapan, yakni tahapan yang pertama adalah pengambilan judul tugas akhir yang ke dua adalah *Preprocessing* yang ke tiga adalah *Weighting* (Pembobotan) dan yang ke empat adalah penghitungan hasil dari klasifikasi. Berikut adalah penjelasan singkat tentang proses yang akan dilakukan, tahapan *Preprocessing* di dalam tahapan ini data judul yang dimasukkan akan dijadikan kata dasar dan akan dihitung frekuensi keluarnya kata. Tahapan *Weighting* (Pembobotan), di dalam proses ini metode yang digunakan adalah *Naive bayes*, metode ini akan melakukan pembobotan dari hasil yang dilakukan oleh Tahapan kedua. Setelah hasil dari klasifikasi judul tugas akhir diketahui maka akan dihitung anggota dari tiap-tiap topik sehingga nantinya akan diketahui jumlah minat mahasiswa akan topik dari judul tugas akhir pada setiap tahunnya, hasil dari minat itu nantinya akan diketahui pertumbuhan minat mahasiswa terhadap topik matakuliah yang telah diajarkan pada kurun waktu 3 tahun terakhir.

3.2. Hasil Analisa Sistem

Hasil analisa yang dapat diambil dari uraian diatas adalah membuat sistem yang dapat membantu prodi dalam melihat pertumbuhan dari minat mahasiswa akan topik matakuliah tertentu yang telah diajarkan pada kurun waktu 3 tahun terakhir studi kasus di program studi Teknik Informatika Di Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Pembuatan sistem ini menggunakan metode *Naive bayes*, Dengan menggunakan metode *term frequency* yang menjadi acuan sebuah bobot judul terhadap *query*, melainkan terdapat satu buah variabel baru yaitu topik. Topik dalam hal ini bertindak sebagai penghubung antara *term* dan judul, masing-masing *term* dan judul akan memiliki bobot terhadap suatu topik. Dengan demikian metode ini dapat menggali sebuah makna yang terkandung dalam term maupun judul.

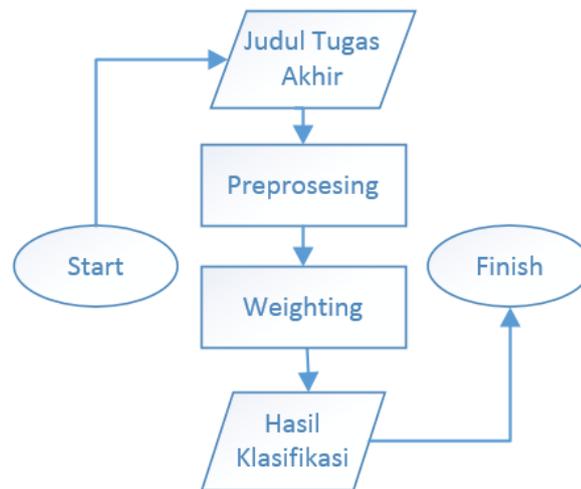
Berdasarkan uraian tersebut maka pada skripsi ini ingin membuat suatu aplikasi dengan judul **“PENGGUNAAN PEMODELAN TOPIK UNTUK MENGETAHUI PERTUMBUHAN MINAT MAHASISWA AKAN TOPIK TUGAS AKHIR MENGGUNAKAN METODE *NAIVE BAYES*”**.

3.3. Perancangan Sistem

Dari analisa yang sudah dilakukan akan dibuat Sistem yang dapat memproses judul Tugas Akhir dan mengelompokkannya berdasar topik yang sudah ada, kemudian dilakukan pengurutan topik Tugas Akhir. Dengan adanya sistem yang akan dibuat ini dapat membantu mahasiswa dalam menentukan judul atau topik Tugas Akhir, untuk lebih jelasnya bisa dilihat pada Gambar 3.1.

1. Tahapan pertama adalah memasukkan judul Tugas Akhir yang nantinya akan diproses lebih lanjut.
2. Tahapan kedua adalah *Preprocessing*, di dalam *preprocessing* data judul yang dimasukkan akan dijadikan kata dasar dan akan dihitung frekuensi keluarnya kata.
3. Tahapan ketiga adalah *Weighting* (Pembobotan), di dalam proses *Weighting* metode yang digunakan adalah *Naive bayes*, metode ini akan melakukan pembobotan dari hasil yang dilakukan oleh Tahapan kedua.

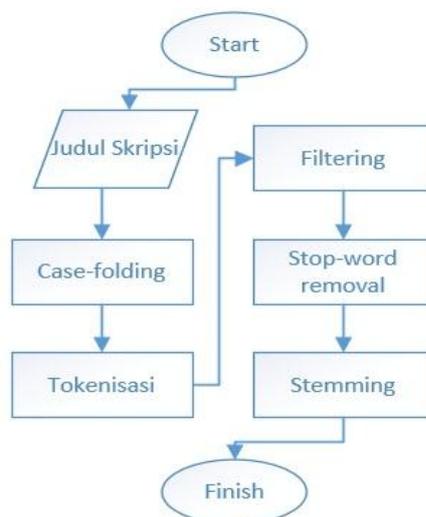
4. Tahapan keempat adalah Hasil Klasifikasi, setelah pembobotan dilakukan terhadap judul ke topik kemudian judul akan diklasifikasikan berdasarkan topik yang sudah ditentukan, sehingga diketahui minat mahasiswa akan topik mata kuliah yang telah diajarkan pada kurun waktu tiga tahun terakhir.



Gambar 3.1 Flowchart Perancangan Sistem Pertumbuhan Minat Mahasiswa

3.3.1. Preprocessing

Secara umum sistem ini memiliki beberapa tahapan proses, yang pertama adalah preprocessing. Tahap preprocessing mencakup *case folding*, *filtering*, *stemming*, *Stop-word removal* dan *Stemming*. Gambar 3.2 akan menjelaskan alur proses preprocessing.



Gambar 3.2 Flowchart Preprocessing untuk Mencari Kata Dasar dari Judul

Proses yang terjadi pada gambar 3.2 adalah sebagai berikut:

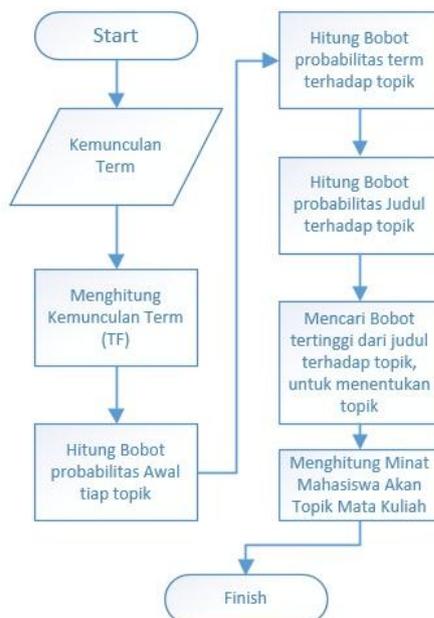
1. Text judul dimasukkan oleh *User* yang nantinya akan dilakukan pemrosesan.
2. Tahapan yang kedua adalah *case-folding*, tahapan ini berfungsi untuk membuat karakter teks menjadi sama (*lowercase* atau *uppercase*).
3. Tahapan ketiga yaitu *Tokenisasi*, *Tokenisasi* adalah proses pemotongan sebuah Judul menjadi bagian-bagian, yang disebut dengan token. Pada saat bersamaan token juga membuang beberapa karakter tertentu yang dianggap sebagai tanda baca.
4. Tahapan yang keempat adalah *filtering*, tahapan ini berfungsi untuk menghilangkan kata atau karakter-karakter yang tidak valid.
5. Tahapan yang kelima adalah *Stop-word removal*, *Stopping* atau *stoplist removal* adalah proses penghilangan kata-kata yang tidak berkontribusi banyak pada Judul Tugas Akhir Kata-kata yang termasuk *stopword* adalah dan, dari, kemudian, yang, di, ke, aku, kamu, dia dan lain-lain.
6. Stemming Nazief dan Andriani, Stemming adalah bagian yang tidak terpisahkan dalam *Information Retrieval* (IR). Algoritma Nazief dan Andriani sebagai algoritma stemming untuk teks berbahasa Indonesia yang memiliki kemampuan prosentase keakuratan (presisi) lebih baik dari algoritma lainnya. Algoritma Stemming Nazief dan Andriani sangat berguna dan menentukan di proses IR terutama dalam pengolahan text bahasa Indonesia. Nazief dan Andriani ini memiliki proses tahap-tahap sebagai berikut:
 - a. Langkah pertama masukkan kata yang akan di proses kemudian kata tersebut akan dibandingkan dengan kata yang sudah ada di dalam *database*. Jika ditemukan maka diasumsikan kata adalah *root word*. Maka proses berhenti.
 - b. Setelah itu dilakukan proses *Inflection Suffixes* (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) dibuang. kemudian Jika berupa *particles* (“-lah”, “-kah”, “-tah” atau “-pun”) maka proses ini diulangi untuk menghapus *Possesive Pronouns* (“-ku”, “-mu”, atau “-nya”).

- c. Langkah selanjutnya adalah menghapus *Derivation Suffixes* (“-i”, “-an” atau “-kan”). Jika kata ditemukan didalam kamus, maka proses akan berhenti sampai disini. Jika tidak maka akan dilanjutkan ke langkah 3a.
 - 1. Jika “-an” sudah dihapus dan huruf terakhir dari kata tersebut adalah “-k”, maka “-k” juga dihapus. Jika kata tersebut ditemukan di kamus maka proses akan berhenti. Jika tidak ditemukan maka berlanjut ke 3b.
 - 2. Langkah selanjutnya adalah menghapus Akhiran (“-i”, “-an” atau “-kan”) dikembalikan, lanjut ke langkah 4.
- d. Setelah itu adalah langkah menghapus *Derivation Prefix*. Jika pada langkah 3 ada sufiks yang dihapus maka dilanjutkan ke langkah 4a, tapi jika tidak ada berlanjut ke langkah 4b.
 - 1. Periksa kata kombinasi awalan dan akhiran yang tidak diperbolehkan. Jika ditemukan maka proses berhenti, jika belum maka berlanjut ke langkah 4b.
 - 2. For $i = 1$ to 3, tentukan jenis awalan kemudian hapus awalan. Jika *root word* masih belum juga ditemukan maka lakukan proses 5, jika sudah maka proses berhenti.
- e. Langkah selanjutnya adalah Melakukan *Recoding*.
- f. Jika semua proses telah selesai tetapi masih belum juga berhasil maka kata awal diasumsikan sebagai *root word*. Proses selesai.

3.3.2. Teorema *Wighting* (Pembobotan)

Teorema *Wighting* (Pembobotan) menggunakan metode *Naive bayes*, *Naive bayes* digunakan dalam penelitian ini untuk menentukan bobot probabilitas yang nantinya akan dicari mana nilai tertinggi pada masing-masing topik. Pada penelitian ini untuk mengetahui suatu judul bagian dari suatu topik maka dilakukan proses klasifikasi. Berikut adalah diagram alir proses perhitungan Naive Bayes pada Gambar 3.3.

Contoh terdapat 11 judul tugas akhir yang sudah melalui tahap *preprocessing* yang memiliki frekuensi sebagaimana tertampil pada tabel 3.1.



Gambar 3.3 Proses *Naive bayes* Untuk Penentuan Topik Judul Tugas Akhir

1. Kemunculan Term, setelah melakukan *preprocessing* maka akan muncul term dari semua judul yang telah dimasukkan, contoh pada Tabel 3.3.

Tabel 3.1 Topik Tugas Akhir

NO	Topik
1	Rekayasa Perangkat Lunak (RPL)
2	Komputasi Berbasis Jaringan (KBJ)
3	Komputasi Cerdas dan Visi (KCV)
4	Interaksi, Grafik dan Seni (IGS)
5	Dasar dan Terapan Komputasi (DTK)

Tabel 3.2 Contoh Judul Tugas Akhir diambil dari <http://monta.if.its.ac.id/>

No	Judul	Topik
1	Rancang Bangun Sistem Keamanan Penyimpanan Data pada Sistem E-Voting	KBJ
2	Rancang Bangun Aplikasi Komunikasi Audio pada Jaringan wireless Lokal Berbasis Android menggunakan Algoritma Joint Coding Rate Control	KBJ
3	Rancang Bangun Perangkat Lunak Aplikasi Penelusuran Produk pada PT Aneka Tuna Indonesia	RPL
4	Perancangan dan Pengembangan Perangkat Lunak Search Engine Berbasis Pertanyaan dengan menerapkan Metode Semantik	RPL
5	Segmentasi Citra Menggunakan Penghapusan Tekstur dan K-Means dengan Batasan Ruang	KCV

6	Segmentasi dan Klasifikasi Citra Mikroskopik Sel Darah Putih	KCV
7	Permainan First Person Shooter (FPS) Realitas Virtual Perangkat Android Survival Maze	IGS
8	Rancang Bangun disain Game Realitas Virtual Taman Makhluk Purba pada Perangkat Bergerak Berbasis Android	IGS
9	Deteksi Karakteristik Cara Mengemudi Kendaraan Berjalan di Jalan Menggunakan OpenCV	DTK
10	Deteksi jenis kendaraan di jalan menggunakan OpenCV	DTK
11	Rancang Bangun Perangkat Lunak untuk Desain Ruang Berbasis Android	?

Tabel 3.3 Hasil Frekuensi Term Terhadap Judul

No	Kata Dasar	judul									
		1	2	3	4	5	6	7	8	9	10
1	rancang	1	1	1	1	0	0	0	1	0	0
2	bangun	1	1	1	0	0	0	0	1	0	0
3	sistem	2	0	0	0	0	0	0	0	0	0
4	aman	1	0	0	0	0	0	0	0	0	0
5	simpan	1	0	0	0	0	0	0	0	0	0
6	data	1	0	0	0	0	0	0	0	0	0
7	e-voting	1	0	0	0	0	0	0	0	0	0
8	aplikasi	0	1	1	0	0	0	0	0	0	0
9	komunikasi	0	1	0	0	0	0	0	0	0	0
10	audio	0	1	0	0	0	0	0	0	0	0
.....
59	jenis	0	0	0	0	0	0	0	0	0	1

Data lengkap dapat dilihat pada lampiran a

- Menghitung Kemunculan Term, term yang telah didapatkan kemudian akan dihitung frekuensi kemunculannya terhadap topik yang sudah ditentukan, Topik pada Tabel 3.1, sedangkan untuk frekuensi kemunculan term pada Tabel 3.4.

Tabel 3.4 Frekuensi Kemunculan Term Pada judul dan bobot Probabilitas

No	Kata Dasar	Frekuensi Kemunculan Term Terhadap Topik					Naive bayes				
		KBJ	RPL	KCV	IGS	DTK	KBJ	RPL	KCV	IGS	DTK
1	rancang	2	2	0	1	0	3/83	3/77	1/73	2/75	1/70
2	bangun	2	1	0	1	0	3/83	2/77	1/73	2/75	1/70
3	sistem	2	0	0	0	0	3/83	1/77	1/73	1/75	1/70
4	aman	1	0	0	0	0	2/83	1/77	1/73	1/75	1/70

5	simpan	1	0	0	0	0	2/83	1/77	1/73	1/75	1/70
6	data	1	0	0	0	0	2/83	1/77	1/73	1/75	1/70
7	e-voting	1	0	0	0	0	2/83	1/77	1/73	1/75	1/70
8	aplikasi	1	1	0	0	0	2/83	2/77	1/73	1/75	1/70
9	komunikasi	1	0	0	0	0	2/83	1/77	1/73	1/75	1/70
10	audio	1	0	0	0	0	2/83	1/77	1/73	1/75	1/70
...
59	jenis	0	0	0	0	1	1/83	1/77	1/73	1/75	2/70

Data lengkap dapat dilihat pada lampiran b

3. Menghitung Probabilitas Awal Tiap Topik, probabilitas awal adalah peluang munculnya topik yang dihitung dari jumlah judul dari topik dibagi dengan total judul yang diinputkan.
4. Hitung Probabilitas Term Terhadap Topik, menghitung term terhadap semua topik yang nantinya akan dijadikan acuan penghitungan dari data uji, cara perhitungannya terdapat pada rumus (2.4) sedangkan contoh ada pada Tabel 3.4.
5. Hitung Probabilitas Judul Terhadap Topik, setelah probabilitas awal dan probabilitas term terhadap topik diketahui maka langkah selanjutnya adalah menghitung data uji yang telah dimasukkan dengan rumus (2.5) sedangkan contoh seperti dibawah ini.

$$\begin{aligned}
 p(\text{"RPL"} | \text{"judul 11"}) &= p(\text{"RPL"}) \times p(\text{"rancang"} | \text{"RPL"}) \times p(\text{"bangun"} | \text{"RPL"}) \\
 &\quad \times p(\text{"angkat"} | \text{"RPL"}) \times p(\text{"lunak"} | \text{"RPL"}) \times p(\text{"Desain"} \\
 &\quad | \text{"RPL"}) \times p(\text{"ruang"} | \text{"RPL"}) \times p(\text{"basis"} | \text{"RPL"}) \times \\
 &\quad p(\text{"android"} | \text{"RPL"}) \\
 &= 1/5 \times 3/77 \times 2/77 \times 3/77 \times 3/77 \times 1/77 \times 1/77 \times 2/77 \times 1/77 = \\
 &0,000000000000001748
 \end{aligned}$$

$$\begin{aligned}
 p(\text{"KBJ"} | \text{"judul 11"}) &= p(\text{"KBJ"}) \times p(\text{"rancang"} | \text{"KBJ"}) \times \\
 &\quad p(\text{"bangun"} | \text{"KBJ"}) \times p(\text{"angkat"} | \text{"KBJ"}) \times \\
 &\quad p(\text{"lunak"} | \text{"KBJ"}) \times p(\text{"Desain"} | \text{"KBJ"}) \times
 \end{aligned}$$

$$\begin{aligned}
& p(\text{"ruang"} | \text{"KBJ"}) \times p(\text{"basis"} | \text{"KBJ"}) \times \\
& p(\text{"android"} | \text{"KBJ"}) \\
& = 1/5 \times 3/83 \times 3/83 \times 1/83 \times 1/83 \times 1/83 \times 1/83 \times 2/83 \times 2/83 = \\
& 0,000000000000000320
\end{aligned}$$

$$\begin{aligned}
p(\text{"KCV"} | \text{"judul 11"}) &= p(\text{"KCV"}) \times p(\text{"rancang"} | \text{"KCV"}) \times \\
& p(\text{"bangun"} | \text{"KCV"}) \times p(\text{"angkat"} | \text{"KCV"}) \times \\
& p(\text{"lunak"} | \text{"KCV"}) \times p(\text{"Desain"} | \text{"KCV"}) \times \\
& p(\text{"ruang"} | \text{"KCV"}) \times p(\text{"basis"} | \text{"KCV"}) \times \\
& p(\text{"android"} | \text{"KCV"}) \\
& = 1/5 \times 1/73 \times 1/73 \times 1/73 \times 1/73 \times 1/73 \times 2/73 \times 1/73 \times 1/73 = \\
& 0,000000000000000050
\end{aligned}$$

$$\begin{aligned}
p(\text{"IGS"} | \text{"judul 11"}) &= p(\text{"IGS"}) \times p(\text{"rancang"} | \text{"IGS"}) \times p(\text{"bangun"} | \text{"IGS"}) \\
& \times p(\text{"angkat"} | \text{"IGS"}) \times p(\text{"lunak"} | \text{"IGS"}) \times p(\text{"Desain"} \\
& | \text{"IGS"}) \times p(\text{"ruang"} | \text{"IGS"}) \times p(\text{"basis"} | \text{"IGS"}) \times \\
& p(\text{"android"} | \text{"IGS"}) \\
& = 1/5 \times 2/75 \times 2/75 \times 1/75 \times 1/75 \times 2/75 \times 1/75 \times 1/75 \times 3/75 = \\
& 0,000000000000000479
\end{aligned}$$

$$\begin{aligned}
p(\text{"DTK"} | \text{"judul 11"}) &= p(\text{"DTK"}) \times p(\text{"rancang"} | \text{"DTK"}) \times p(\text{"bangun"} | \text{"DTK"}) \\
& \times p(\text{"angkat"} | \text{"DTK"}) \times p(\text{"lunak"} | \text{"DTK"}) \times \\
& p(\text{"Desain"} | \text{"DTK"}) \times p(\text{"ruang"} | \text{"DTK"}) \times p(\text{"basis"} | \text{"DTK"}) \\
& \times p(\text{"android"} | \text{"DTK"}) \\
& = 1/5 \times 1/70 = \\
& 0,0000000000000000
\end{aligned}$$

6. Mencari Bobot Tertinggi dari Judul Terhadap Topik, ambil nilai tertinggi dari proses penghitungan yang telah dilakukan dari judul terhadap topik, contoh Hasil dari perhitungan *naive bayes* ini maka dapat diambil kesimpulan bahwa judul 11 termasuk ke dalam golongan topik RPL seperti terlihat pada Tabel

3.5, karena nilai dari RPL lebih besar dibandingkan dengan nilai topik yang lain.

Tabel 3.5 Tabel Hasil Pemrosesan *Naive bayes*

NO	Topik	Hasil
1	Rekayasa Perangkat Lunak (RPL)	0,000000000000001748
2	Komputasi Berbasis Jaringan (KBJ)	0,00000000000000320
3	Komputasi Cerdas dan Visi (KCV)	0,00000000000000050
4	Interaksi, Grafik dan Seni (IGS)	0,00000000000000479
5	Dasar dan Terapan Komputasi (DTK)	0,00000000000000035

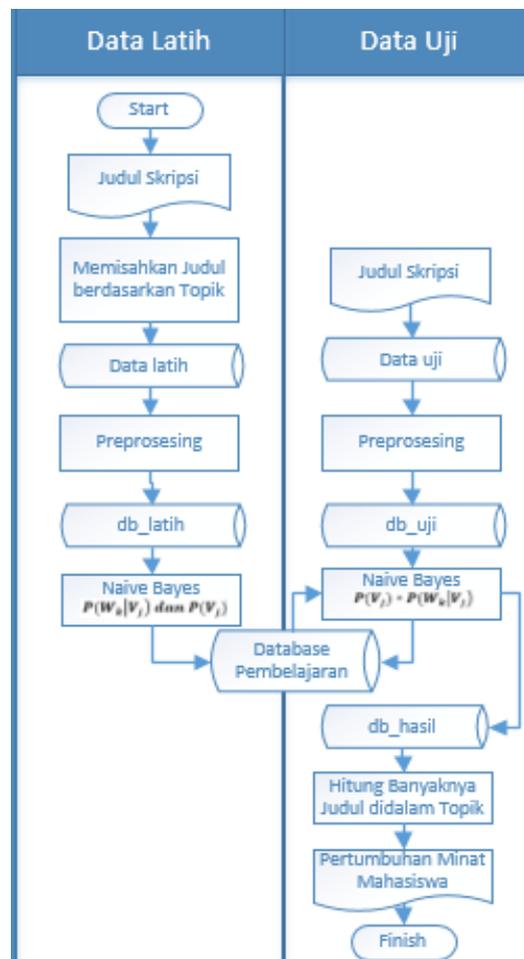
- Menghitung pertumbuhan minat mahasiswa, setelah semua judul latih sudah terdeteksi dalam Topik maka langkah selanjutnya adalah lanjutkan penghitungan terhadap semua judul latih seperti pada rumus (2.6), ketika semua judul latih sudah terdeteksi topiknya, maka langkah terakhir adalah menghitung anggota topik dari hasil yang sudah didapatkan dengan cara menjumlahkan semua hasil dari klasifikasi yang sudah didapatkan dilangkah 6, berdasarkan topik yang sudah ditentukan, maka akan diketahui mana topik yang paling banyak diminati oleh mahasiswa pada setiap tahunnya.

Setelah topik dari setiap judul pada setiap tahunnya diketahui contoh pada tabel 3.5 maka dapat dilihat topik mana yang sedang menjadi *trending* pada setiap tahunnya. Dari data tersebut maka akan diketahui perkembangan topik pada setiap tahunnya apakah mengalami penurunan atau kenaikan dari banyaknya judul yang telah diambil oleh mahasiswa.

3.3.3. Flowchart Sistem

Untuk menggambarkan proses dari Sistem Penggunaan Metode *Text Mining Classification* Sebagai Penunjang Mahasiswa Menentukan Topik Tugas Akhir Pada Program Studi Teknik Informatika Di Teknologi Sepuluh Nopember (ITS) Surabaya, dibuatlah desainnya menggunakan *Data Flow Sistem*, dengan ini maka akan diketahui keterkaitan item di dalam sistem yang telah dibuat.

Gambar 3.4 merupakan gambaran umum dari sistem Penggunaan Metode *Text Mining Classification* Sebagai Penunjang Mahasiswa Menentukan Topik Tugas Akhir Pada Program Studi Teknik Informatika Di Institut Teknologi Sepuluh Nopember (ITS) Surabaya dimulai dari input judul Tugas Akhir yang sudah dikelompokkan berdasarkan topik yang sudah ada, data ini dijadikan sebagai data uji. Setelah itu data judul yang sudah dimasukkan ke dalam *database* akan di proses lagi dengan diambil kata dasarnya yang sesuai dengan kata-kata dasar di dalam *database*, kemudian judul yang sudah terpisah menjadi kata dasar disimpan kembali ke dalam *database*.



Gambar 3.4 Flowchart Sistem Penghitungan Minat Mahasiswa

Judul yang sudah dimasukkan ke dalam *database* kemudian akan mengalami proses pemisahan menjadi kata dasar atau *term*, kata yang sudah mengalami *preprosesing* kemudian akan dilakukan perhitungan menggunakan *term*

frequency (TF), setelah nilai TF sudah diketahui maka selanjutnya akan diproses dengan menggunakan metode Naïve Bayes untuk mengetahui probabilitas setiap judul terhadap topik sehingga diketahui topik dari setiap judul yang telah diinputkan, langkah terakhir adalah menghitung minat mahasiswa terhadap topik sehingga diketahui perkembangan minat dari mahasiswa pada kurun waktu 3 tahun terakhir.

3.3.4. Klasifikasi Kebutuhan

Input sistem atau data-data yang dibutuhkan dalam proses Penunjang Mahasiswa Menentukan Topik Tugas Akhir Pada Program Studi Teknik Informatika Di Institut Teknologi Sepuluh Nopember (ITS) Surabaya yaitu data judul atau topik Tugas Akhir yang sudah diambil oleh mahasiswa sebelumnya dengan format pertahun.

3.3.5. Kebutuhan Keluaran Data

Output sistem yang dihasilkan dari penelitian ini adalah sebuah sistem yang dapat menampilkan informasi yang bisa memberikan laporan mengenai minat mahasiswa terhadap topik mata kuliah tertentu yang telah diajarkan di Program Studi Teknik Informatika Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Dari data judul Tugas Akhir yang sudah dimasukkan per tahun. Sistem ini akan melakukan pengurutan terhadap judul Tugas Akhir yang paling banyak diambil oleh mahasiswa berdasarkan topiknya.

3.4. Spesifikasi Pembuatan Sistem

3.4.1 Kebutuhan Perangkat Lunak

Perangkat Lunak (*Software*) adalah semua program-program yang terdapat di dalam komputer, *software* merupakan ruh dari perangkat keras. dibutuhkan perangkat lunak untuk agar bisa mendukung dalam pembuatan aplikasinya.

Perangkat lunak yang dibutuhkan untuk pembuatan *sistem* ini diantaranya:

1. Sistem Operasi Windows 7
2. Xampp 5.6.30
3. Adobe Dreamweaver CS5

4. Browser (Mozilla, Chrome)

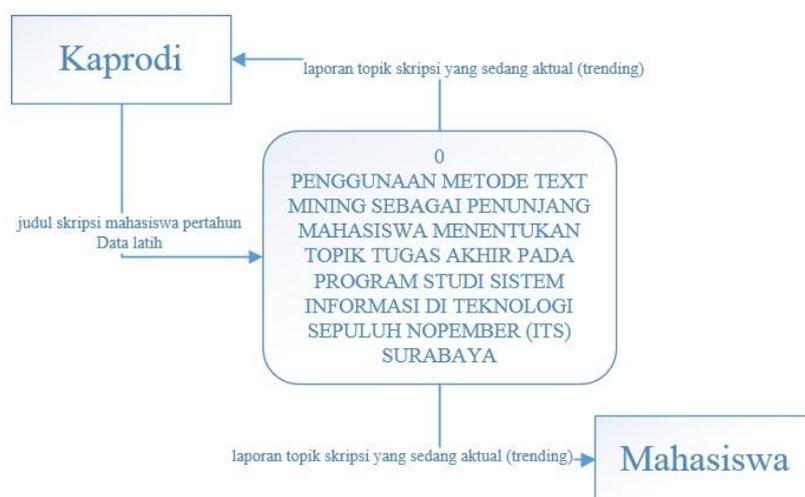
3.4.2 Kebutuhan Perangkat Keras

Perangkat Keras (*Hardware*) adalah perangkat komputer atau komponen-komponen komputer pendukung yang digunakan. Spesifikasi perangkat keras yang digunakan adalah Laptop TOSHIBA Satelit L735 dengan Prosesor Intel(R) Core(TM) i3-2350 CPU @ 2.30GHz, dan RAM 2GB DDR 3.

3.4.3 Diagram Konteks

Pada gambar *Context Diagram* dibawah menggambarkan dan menjelaskan proses umum yang terjadi pada Sistem Penggunaan Metode *Text Mining Classification* Sebagai Penunjang Mahasiswa Menentukan Topik Tugas Akhir Pada Program Studi Teknik Informatika Di Teknologi Sepuluh Nopember (ITS) Surabaya.

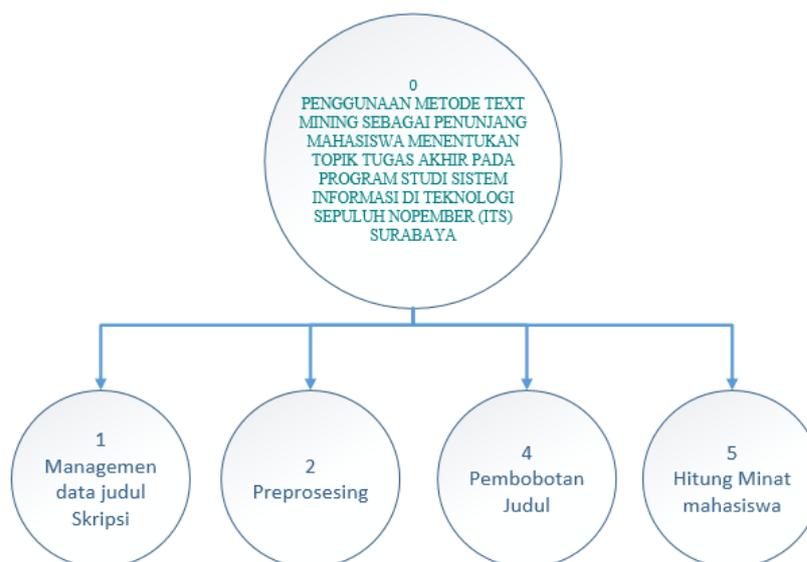
Gambar 3.5 *Diagram Konteks* menjelaskan secara singkat tentang sistem yang akan dibuat, diagram ini menggambarkan *ouput* dan *input* di dalam sistem dan juga menjelaskan tentang pelaku yang berperan dalam sistem. Adapun identitas atau pelaku yang terlibat di dalam Sistem Penggunaan Metode *Text Mining Classification* Sebagai Penunjang Mahasiswa Menentukan Topik Tugas Akhir Pada Program Studi Teknik Informatika Di Teknologi Sepuluh Nopember (ITS) Surabaya, adalah kaprodi dan Mahasiswa. *Input* yang terdapat di sistem tersebut adalah Data latih, Data judul Tugas Akhir dan juga topik yang sedang banyak diambil oleh mahasiswa sedangkan keluaran dari sistem.



Gambar 3.5 *Diagram Konteks* Sistem Menghitung Perkembangan Minat Mahasiswa

3.4.5 Diagram Berjenjang

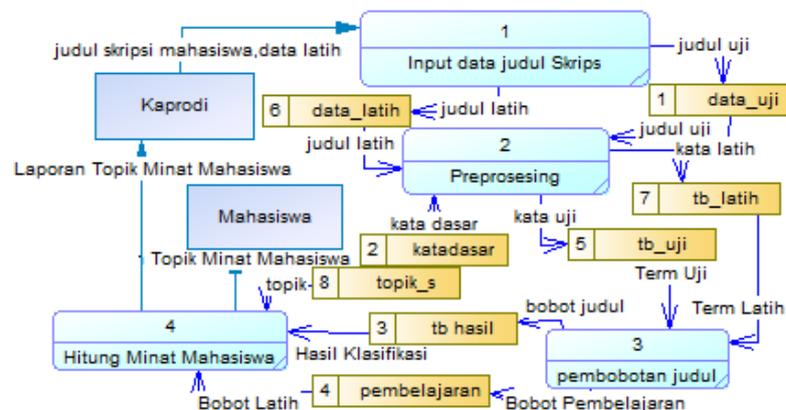
Gambar 3.6 Diagram Berjenjang ini menggambarkan pembagian dari proses yang terdapat di Sistem Penggunaan Metode *Text Mining Classification* Sebagai Penunjang Mahasiswa Menentukan Topik Tugas Akhir Pada Program Studi Teknik Informatika Di Teknologi Sepuluh Nopember (ITS) Surabaya. yang dibagi menjadi beberapa bagian untuk mempermudah dalam pemahamannya dan juga bisa menjadi lebih spesifik, hasil dari proses pembagian sistem yang digambarkan meliputi *Managemen* judul Tugas Akhir, *stemming stopword* dan *tokenizer*, Pengindekan Judul, dibagi menjadi 4 proses yaitu menghitung *probabilitas* menggunakan *naive bayes*, menghitung iterasi, mengambil nilai bobot baru, menentukan hasil dengan *naive bayes* dan menentukan topik dengan nilai tertinggi.



Gambar 3.6 Diagram Berjenjang Menghitung Perkembangan Minat Mahasiswa

3.4.6 DFD Level 1

Pada Gambar 3.7 dapat dilihat DFD level 0 dari Sistem Penggunaan Metode *Text Mining Classification* Sebagai Penunjang Mahasiswa Menentukan Topik Tugas Akhir Pada Program Studi Teknik Informatika Di Teknologi Sepuluh Nopember (ITS) Surabaya:



Gambar 3.7 DFD Level 0 Sistem Menghitung Perkembangan Minat Mahasiswa

1. Input data judul tugas akhir, langkah pertama adalah memasukkan judul tugas akhir yang diinputkan oleh prodi baik itu data latih ataupun data uji yang diambil dari <http://monta.if.its.ac.id/>. Data yang sudah diinputkan akan disimpan kedalam *database*, data latih masuk kedalam tabel *data_latih* sedangkan untuk data uji dimasukkan kedalam tabel *data_uji*.
2. *Preprosesing*, judul yang sudah berada didalam *database* kemudian akan diproses kembali untuk dilakukan proses *case-folding*, *Tokenisasi*, *Filtering*, *Stop-word* dan *Stemming* sehingga akan diketahui term yang menjadikan ciri dari setiap judul tugas akhir, setelah itu dilakukan penghitungan dari kemunculan term dengan menggunakan *term frequency* (TF), Judul yang sudah mengalami *Preprosesing* kemudian akan di simpan kedalam tabel *tb_latih* untuk hasil dari data latih dan *tb_uji* untuk hasil dari data uji.
3. Pembobotan Judul, judul yang sudah diketahui frekuensi kemunculan dari setiap termnya kemudian dicari bobot terhadap topik dengan menggunakan metode *naive bayes*, sehingga diketahui bobot latih untuk data latih dan diketahui topik dari judul untuk data uji.
4. Hitung minat mahasiswa, setelah diketahui topik dari setiap judul uji kemudian judul tugas akhir dijumlahkan berdasarkan topiknya, sehingga diketahui jumlah dari setiap topik yang sudah diambil oleh mahasiswa.

3.4.8 Desain Database

Desain tabel pada sistem peretingan judul Tugas Akhir berdasarkan topik ini adalah sebagai berikut :

1. Tabel data_uji

Tabel 3.6 digunakan untuk menampung semua judul Tugas Akhir yang akan diproses dalam sistem.

Tabel 3.6 Tabel data_uji

File Name	Type	Size	Keterangan
id_uji	Int	5	Primary Key
no_induk	Text		
tgl_entr	Text		
Judul	Text		
Pengarang	Text		
Penerbit	Text		
Tahun	Text		
no_kelas	Text		
no_panggil	Int	5	
subyek	Text		

2. Tabel data_latih

Tabel 3.7 dibawah ini digunakan untuk menampung semua judul yang nantinya akan dijadikan data latih.

Tabel 3.7 Tabel data_latih

File Name	Type	Size	Keterangan
id_latih	Int	5	Primary Key
no_induk	Text		
tgl_entr	Text		
Judul	Text		
Pengarang	Text		
Penerbit	Text		
Tahun	Text		
no_kelas	Text		
no_panggil	Int	5	
subyek	Text		

3. Tabel td_latih

Tabel 3.8 dibawah ini berfungsi untuk menampung semua hasil dari *stimming*, *stopword* dan juga *tokenizing* dari data latih.

Tabel 3.8 Tabel td_latih

File Name	Type	Size	Keterangan
Id_stem_latih	Int	5	Primary Key
Id_latih	Int	5	
Kata_dasar	Text		
J_kata	Text		

4. Tabel td_uji

Tabel 3.9 dibawah ini berfungsi untuk menampung semua hasil dari *stimming*, *stopword* dan juga *tokenizing* dari data uji.

Tabel 3.9 Tabel td_uji

File Name	Type	Size	Keterangan
Id_stem_uji	int	5	Primary Key
Id_latih	int	3	
Kata_dasar	text		
J_kata	Int	2	
Tahun	int	4	

5. Tabel katadasar

Tabel 3.10 dibawah ini berfungsi untuk menampung semua kata dasar yang nantinya akan dijadikan pembanding di proses dan *stimming*, *stopword*.

Tabel 3.10 Tabel katadasar

File Name	Type	Size	Keterangan
Id_word	Int	10	Primary Key
Word	text		
stopword	text		
katagory	text		

6. Tabel topik_s

Tabel 3.11 dibawah ini berfungsi untuk berikan topik dari data yang sudah diproses.

Tabel 3.11 Tabel topik_s

File Name	Type	Size	Keterangan
Id_topik	Int	2	Primary Key
S_topik	Text		
Topik	Text		

7. Tabel td_hasil

Tabel 3.12 berfungsi untuk menampung hasil dari seluruh proses yang ada, sehingga nantinya pengambilan laporan tidak perlu harus memproses data lagi.

Tabel 3.12 Tabel td_hasil

File Name	Type	Size	Keterangan
Id_hasil	int	10	Primary Key
Id_topik	int	3	
Tahun_a	int	4	
Tahun_b	int	4	
jumlah	text		

Tabel 3.13 Tabel pembelajaran

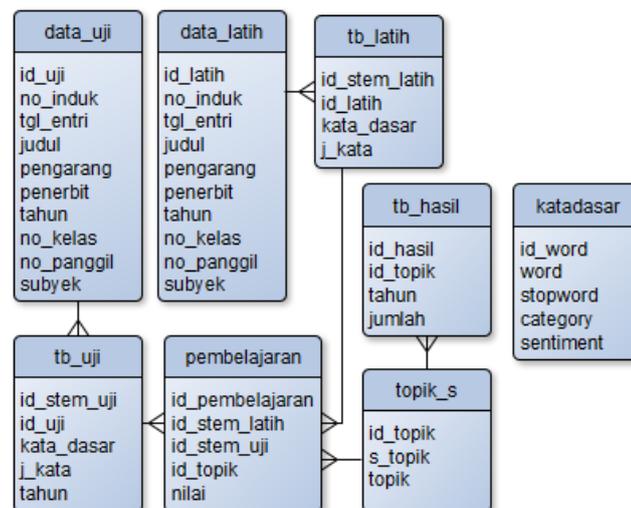
File Name	Type	Size	Keterangan
Id_pembelajaran	int	10	Primary Key
Id_stem_latih	int	5	
Id_stem_uji	int	5	
Nilai	int	5	

8. Tabel pembelajaran

Tabel 3.13 dibawah ini berfungsi untukmenampung hasil dari data uji yang sudah di proses.

3.4.9. Entity Relationship Diagram

Gambar 3.8 Menggambarkan tentang *Entity Relationship Diagram* (ERD) pada Sistem yang akan dibuat, ERD ini membentuk suatu hubungan antar tabel yang menjadi suatu relasi, berikut ini adalah relasi yang terdapat dalam tabel diatas:



Gambar 3.8 ERD Sistem Menghitung Perkembangan Minat Mahasiswa

1. Tabel *data_uji*

Tabel *data_uji* berisikan semua judul uji, memiliki relasi *one-to-many* ke tabel *tb_uji*, yaitu *field* *id_uji* dari tabel *data_uji* direlasikan dengan *field* *id_uji* yang ada di tabel *tb_uji*.

2. Tabel *data_latih*

Tabel *data_latih* berisikan semua judul latihan, memiliki relasi *one-to-many* ke tabel *tb_latih*, yaitu *field* *nim* dari tabel *data_latih* direlasikan dengan *field* *id_latih* yang ada di tabel *tb_latih*.

3. Tabel *tb_uji*

Tabel *tb_uji* berisikan hasil dari semua kata dasar dari judul uji, memiliki relasi *one-to-many* dengan tabel pembelajaran, yaitu *field* *id_stem_uji* dari tabel *tb_uji* dengan tabel *pembelajaran*.

4. Tabel *tb_latih*

Tabel *tb_latih* berisikan hasil dari semua kata dasar dari judul latihan, memiliki relasi *one-to-many* dengan tabel pembelajaran, yaitu *field* *id_stem_latih* dari tabel *tb_latih* dengan tabel *pembelajaran*.

5. Tabel *pembelajaran*

Tabel *pembelajaran* berisikan nilai probabilitas dari judul tugas akhir, memiliki relasi *many-to-one* dengan tabel *topik_s*, yaitu *field* *id_topik* dari tabel *topik_s* direlasikan dengan tabel *pembelajaran*.

6. Tabel *topik_s*

Tabel *topik_s* berisikan data dari topik matakuliah, miliki relasi *one-to-many* dengan tabel *tb_hasil*, yaitu *field* *id_topik* dari tabel *topik_s* direlasikan dengan tabel *tb_hasil*.

7. Tabel *tb_hasil*

Tabel *tb_hasil* berisikan hasil dari klasifikasi yang telah dilakukan.

8. Tabel *katadasar*

Tabel *katadasar* berisikan semua kata dasar yang sesuai dengan kamus besar bahasa indonesia.

3.4.10 Desain Interface

Desain *interface* adalah bagian yang bertugas untuk menghubungkan antara perangkat lunak/aplikasi dengan pengguna. *Interface* yang akan dibuat ini adalah dengan menggunakan bahasa pemrograman *PHP*, *CSS*, *HTML* dan juga *Java Script* dan *database MySQL*.

1. Halaman *Login*

Halaman *login* seperti di **Gambar 3.9** merupakan tampilan yang pertama akan tampil pada saat mengakses sistem ini, halaman *login* juga berfungsi sebagai *security* karena semua yang bisa melewati halaman ini berarti sudah terdaftar di dalam sistem, untuk masuk ke sistem harus mengisi *username* dan *password*.



Gambar 3.9 Form Login

2. Halaman Utama

Gambar 3.10 Dibawah ini merupakan halaman utama yang akan menampilkan grafik *reting* dari topik yang paling banyak diambil oleh mahasiswa, sehingga bisa terlihat informasi yang menampilkan minat mahasiswa terhadap topik mata kuliah tertentu yang telah diajarkan.

ITS	
<div style="border: 1px solid black; padding: 5px; text-align: center;">User</div> Menu	Dashboard DASHBOARD <div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; width: 45%; height: 100px;"></div> <div style="border: 1px solid black; width: 45%; height: 100px;"></div> </div>

Gambar 3.10 Desain Halaman Utama

3. Halaman *Import* data

Gambar 3.11 merupakan halaman *Import* data yang nantinya akan diolah di dalam sistem, sehingga Kaprodi tidak perlu melakukan input satu per satu judul ke dalam *database*. Halaman ini hanya bisa diakses oleh Kaprodi saja.

ITS	
User	Home > Import Data
Menu	<p>Import Data</p> <p>Import Data</p> <p><input type="text"/> <input type="button" value="Upload"/> <input type="button" value="Browse"/></p> <hr/> <p>Pilih Data</p> <p><input type="text"/> <input type="text"/></p> <p><input type="text"/> <input type="text"/></p> <p><input type="button" value="Submit"/></p>

Gambar 3.11 *Desain* Halaman *Import* Data

4. Halaman *Input* data

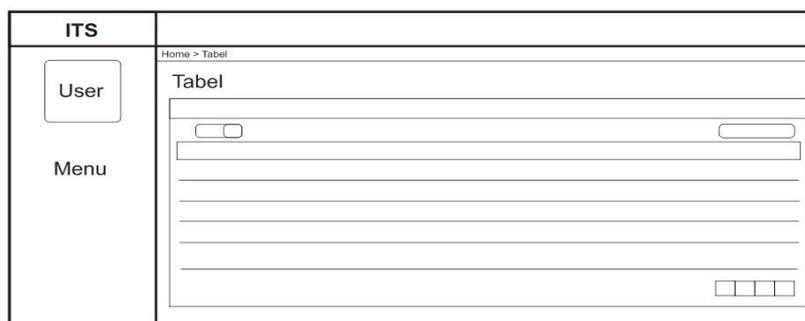
Gambar 3.12 Dibawa ini merupakan halaman *input* data yang fungsinya untuk melakukan input kategori ke dalam *database* yang nantinya akan dijadikan lebel dari data yang telah diolah. Halaman ini hanya bisa diakses oleh Kaprodi saja.

ITS	
User	Home > Input
Menu	<p>Form Input</p> <p>Input A <input type="text"/></p> <p>Input B <input type="text"/></p> <p><input type="text"/></p>

Gambar 3.12 *Desain* Halaman *Input* Data

5. Halaman *View* data

Gambar 3.13 Dibawa ini merupakan halaman untuk melihat semua data yang telah di masukkan kedalam *database*. Didalam halaman ini juga terdapat menu untuk menghapus ataupun mengupdate data yang sudah ada. Halaman ini hanya bisa diakses oleh Kaprodi saja.



Gambar 3.13 Desain Halaman View Data

3.5 Skenario Pengujian

Pada penelitian ini, untuk mengukur evaluasi kinerja sistem temu kembali informasi digunakan pengujian *precision* dan *accuracy*. *Precision* adalah rasio judul relevan yang berhasil ditemukembalikan dari seluruh judul yang berhasil ditemukembalikan.

Tabel 3.14 Parameter menghitung *precision* dan *accuracy*

Keterangan	Relevan	Tidak Relevan
Terambil	True positive (tp)	False positive (fp)
Tidak terambil	False negative (fn)	True negative (tn)

Nilai *precision*, dan *accuracy* dinyatakan dalam persen. Semakin tinggi ketiga nilai tersebut menunjukkan semakin baiknya kinerja aplikasi. Evaluasi yang akan dilakukan dalam penelitian ini adalah menghitung nilai dari *precision* dan *accuracy* berdasarkan judul yang berhasil ditemukan kembali oleh sistem aplikasi yang dibuat. Sedangkan untuk menentukan nilai dari *precision* dan *accuracy* harus didapatkan jumlah judul yang relevan terhadap suatu topik judul.