 **JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# IDENTITAS PENGUSUL

Nama : **Ida Ayu Gde Suwiprabayanti Putra**

NRP : **5107 100 001**

Dosen Wali : **F.X. Arunanto, Ir., M.Sc.**

# JUDUL TUGAS AKHIR

***“Implementasi Algoritma Probabilistic Latent Semantic Analysis***

***dalam Pengklasteran Dokumen Berbasis Topik”***

# LATAR BELAKANG

Pengklasteran merupakan sebuah teknik dalam pengelompokkan obyek-obyek kedalam kelas-kelas berdasarkan kesamaan obyek-obyek tersebut. Pengklasteran biasanya digunakan untuk evaluasi hasil dari mesin pencari, navigasi untuk sebuah koleksi dokumen yang besar ataupun untuk peringkasan teks. Dalam melakukan pengklasteran, program komputer pertama-tama harus mendapat masukan berupa apa yang disebut “bag of words” (sekantong kata-kata). Pembuatan “kantong” ini sebenarnya adalah sebuah skema representasi (perwakilan) kata-kata yang ada dalam sekumpulan dokumen. Dengan cara ini sebenarnya sebuah teks diwakilkan sebagai sebuah vektor yang elemen-elemennya merupakan pembobotan berdasarkan frekuensi kemunculan kata-kata di dalam teks bersangkutan yang disebut dengan Ruang Kata (*term space*) [1]. Namun hal itu menyebabkan dibutuhkannya dimensi vektor kata yang besar karena banyaknya frekuensi kata yang harus dihitung dalam suatu dokumen. Untuk mengurangi besarnya dimensi dalam Ruang Kata maka Ruang Kata ditransformasikan menjadi Ruang Konsep (*concept space*).

Dalam Ruang Konsep diasumsikan bahwa kata-kata yang memiliki frekuensi yang sama dalam dokumen yang sama memiliki hubungan semantik sehingga kata-kata tersebut bisa dikelompokkan menjadi topik kata [2]. Salah satu metode yang bisa digunakan dalam Ruang Konsep adalah PLSA (Probabilistic Latent Semantic Analysis). PLSA adalah sebuah metode pendekatan probabilitas untuk dua model (contoh : kata dan dokumen). PLSA biasanya digunakan dalam aplikasi Temu Kembali Informasi, Pengolahan Bahasa Alami, Sistem Pembelajaran dengan Data Teks, dan lain-lain [2]*.* Dalam Tugas Akhir ini, PLSA digunakan untuk mengelompokkan kata-kata ke dalam topik-topik yang belum diketahui (*latent*). Sehingga nantinya dapat dilakukan pengklasteran dokumen berdasarkan topik-topik yang dibahas dalam masing-masing dokumen.

# TUJUAN DAN MANFAAT TUGAS AKHIR

Tugas Akhir ini bertujuan untuk mendesain dan merancang sistem perangkat lunak yang dapat melakukan pengklasteran dokumen berdasrkan Ruang Konsep. Manfaat yang diharapkan nantinya adalah berkurangnya dimensi Ruang Kata karena telah dilakukan pereduksian dimensi vektor menggunakan teknik PLSA berdasarkan Ruang Konsep.

# RUMUSAN MASALAH

Rumusan masalah yang diangkat dalam Tugas Akhir ini adalah sebagai berikut :

1. Bagaimana mengambil kata-kata penting dalam sebuah dokumen sebagai kandidat kata penting untuk topik.
2. Bagaimana menentukan jumlah klaster yang sesuai untuk inisialisasi kelompok topik dan kelompok dokumen.
3. Bagaimana mengelompokkan kata-kata menjadi topik yang latent dalam dokumen.
4. Bagaimana mengelompokkan dokumen berdasarkan topik yang dibahas dan memberikan label pada topik.

# BATASAN MASALAH

Permasalahan yang dibahas dalam Tugas Akhir ini memiliki beberapa batasan, yaitu :

* Untuk perhitungan yang besar seperti perhitungan matriks menggunakan library yang sudah tersedia.
* Untuk mencari sinonim dari kata-kata di dalam topik digunakan kamus WordNet yang sudah tersedia.

# RINGKASAN TUGAS AKHIR

Pengklasteran merupakan sebuah teknik dalam pengelompokkan obyek-obyek kedalam kelas-kelas berdasarkan kesamaan obyek-obyek tersebut. Pengklasteran biasanya digunakan untuk evaluasi hasil dari mesin pencari, navigasi untuk sebuah koleksi dokumen yang besar ataupun untuk peringkasan teks. Dalam melakukan pengklasteran dokumen, sebuah teks diwakilkan sebagai sebuah vektor yang elemen-elemennya merupakan pembobotan berdasarkan frekuensi kemunculan kata-kata di dalam teks. Namun hal itu menyebabkan dibutuhkannya dimensi vektor kata yang besar karena banyaknya frekuensi kata yang harus dihitung dalam suatu dokumen. Untuk mengurangi besarnya dimensi tersebut dapat dilakukan alternatif lain yaitu Ruang Konsep.

Dalam Ruang Konsep diasumsikan bahwa kata-kata yang memiliki frekuensi yang sama dalam dokumen yang sama memiliki hubungan semantik sehingga kata-kata tersebut bisa dikelompokkan menjadi topik kata [2]. Salah satu metode yang bisa digunakan dalam Ruang Konsep adalah PLSA (Probabilistic Latent Semantic Analysis). PLSA adalah sebuah metode pendekatan probabilitas untuk dua model (contoh : kata dan dokumen). PLSA biasanya digunakan dalam aplikasi Temu Kembali Informasi, Pengolahan Bahasa Alami, Sistem Pembelajaran dengan Data Teks, dan lain-lain [2]*.* Dalam Tugas Akhir ini, PLSA digunakan untuk mengelompokkan kata-kata ke dalam topik-topik yang belum diketahui (*latent*). Sehingga nantinya dapat dilakukan pengklasteran dokumen berdasarkan topik-topik yang dibahas dalam masing-masing dokumen.

Diagram Alur :

1. Mengambil kata-kata penting dalam dokumen [3,4]



Gambar 1. Alur pengambilan kata-kata penting dalam dokumen

Pada langkah ini, dokumen yang telah terbaca oleh sistem akan dipecah-pecah menjadi kata-kata yang terdapat dalam masing-masing dokumen. Dari semua kata-kata yang ada tersebut kemudian kata-kata yang dianggap tidak penting dihilangkan. Penghilangan kata-kata yang tidak penting ini menggunakan Google Stopword[3]. Kemudian dicari kata dasarnya menggunakan Porter’s Stemming[4].

1. Menentukan jumlah klaster yang sesuai untuk kelompok topik dan kelompok dokumen[5]



Gambar 2. Alur pemilihan jumlah klaster yang sesuai

Pada tahap ini digunakan Hartigan Index untuk mengevaluasi nilai k yang paling tepat. Nilai k tersebut nantinya akan digunakan dalam perhitungan PLSA untuk mencari jumlah topik yang dapat terbentuk dari semua kata-kata pada dokumen tersebut. Nilai k juga dibutuhkan dalam pengelompokan dokumen, sehingga didapatkan berapa jumlah kelompok dokumen yang paling tepat.

1. Melakukan pengidentifikasian terhadap kata-kata yang masuk ke dalam kelompok topik dan kecenderungan topik yang dibahas oleh dokumen [2].



Gambar 3. Alur pengidentifikasian kata-kata, topik dan dokumen

Pada tahap ini dilakukan pengelompokkan kata-kata menjadi topik-topik dan pengidentifikasian topik-topik yang terdapat dalam dokumen dengan menggunakan metode PLSA. Dalam metode PLSA, diawali dengan menentukan jumlah topik yang akan terbentuk. Kemudian diinisialisasi tiga parameter yaitu probabilitas topik, probabilitas kata terdapat dalam topik dan probabilitas dokumen mengandung topik. Ketiga parameter tersebut diperbaharui terus menerus sampai nilai ketiganya saling mendekati atau mencapai batas tresshold tertentu.

1. Melakukan pengelompokkan dokumen dan pemberian label pada kelompok topik[6,7]



Gambar 4. Alur pemberian label pada topik Gambar 5. Alur pengelompokan dokumen

Pada tahap ini dilakukan pemberian label pada topik yang terbentuk dan pengelompokkan dokumen. Pengelompokkan dokumen dilakukan dengan metode K-Means dan pemberian label pada topik dilakukan dengan menggunakan kamus WordNet. Dalam kamus WordNet diketahui sinonim dari setiap kata-kata. Sehingga nantinya dapat diketahui suatu kata umum dari semua kata-kata yang dicari.

Penjelasan Diagram Alur :

1. Tahap Pra-Pemrosesan (Gambar 1)
2. Pengambilan kata-kata dalam semua dokumen

Pada langkah ini, dokumen yang telah terbaca oleh sistem akan dipecah-pecah menjadi kata-kata yang terdapat dalam masing-masing dokumen.

1. Penghilangan Stopwords

Dari semua kata-kata yang ada tersebut kemudian kata-kata yang dianggap tidak penting dihilangkan. Penghilangan stopwords ini menggunakan Google Stopword [3].

1. Penerapan Stemming

Stemming adalah teknik dalam mengambil kata-kata dasar dari masing-masing kata yang ada. Penerapannya menggunakan metode Porter’s stemming [4].

1. Tahap penentuan jumlah klaster yang sesuai (Gambar 2)

Pada tahap ini digunakan Hartigan Index untuk mengevaluasi nilai k yang paling tepat.

Algoritmanya adalah sebagai berikut :

* Menghitung jarak antara data dengan centroid

Dimana :

N = jumlah data

k = jumlah klaster

= data yang akan dihitung

= data centroid

* Menghitung nilai hartigan index

Dimana :

n = jumlah data

k= jumlah klaster

Kemudian dipilih nilai hartigan index yang paling tinggi yang merupakan k klaster yang paling tepat.

Contoh :

Dimisalkan terdapat 6 data :

|  |  |  |
| --- | --- | --- |
|  | **X** | **Y** |
| **Data1** | 1 | 3 |
| **Data2** | 1 | 0 |
| **Data3** | 2 | 0 |
| **Data4** | 4 | 5 |
| **Data5** | 5 | 6 |
| **Data6** | 6 | 6 |

Tabel 1. Contoh data untuk penghitungan hartigan index

Maka akan dihitung jarak antara data dengan centroid untuk masing-masing k. Pada data ini maka terdapat 4 k yaitu saat k=2, k=3, k=4 dan k=5

Saat k = 2

Centroid

|  |  |  |
| --- | --- | --- |
|  | **X** | **Y** |
| **Centroid 1** | 1,333333 | 1 |
| **Centroid 2** | 5 | 5,666667 |

Tabel 2. Contoh data centroid untuk penghitungan hartigan index

err(k) =

err(2) =(d2(X1,C1)+ d2(X2,C1)+ d2(X3,C1))+( d2(X4,C2)+ d2(X5,C2)+ d2(X6,C2))

err(2 )=(4,111111111+1,111111111+1,444444444)+( 1,444444444+0,111111111+1,111111111)

err(2) = 6,666667 + 2,666667

err(2) = 9,333333

Perhitungan nilai Hartigan

H(k) = (n – k – 1)

H(2) = (6 – 2 – 1)

H(2) = 3 ()

H(2) = 5,85

Perhitungan yang sama dilakukan untuk k=3 dan seterusnya. Kemudian dilihat nilai H(k) yang paling tinggi, maka itulah nilai k yang paling tepat.

1. Tahap pengidentifikasian kata-kata yang terdapat di dalam topik dan topik-topik yang terdapat di dalam dokumen (Gambar 3)

Pada tahap ini digunakan algoritma PLSA (Probabilistic Latent Semantic Analysis).

Algoritmanya adalah sebagai berikut :

1. Membuat matrix jumlah kata-kata dalam masing-masing dokumen

|  |  |  |
| --- | --- | --- |
|  | Dokumen 1 | Dokumen 2 |
| Kata 1 | Jumlah kata 1 dalam dokumen 1 | Jumlah kata 1 dalam dokumen 2 |
| Kata 2 | Jumlah kata 2 dalam dokumen 1 | Jumlah kata 2 dalam dokumen 1 |
| Kata 3 | Jumlah kata 3 dalam dokumen 1 | Jumlah kata 3 dalam dokumen 1 |

Tabel 3. Contoh jumlah kata dalam dokumen untuk teknik PLSA

Dimana jumlah kata dalam dokumen direpresentasikan dengan :

🡪 = dokumen, = kata

1. Menentukan jumlah topik (z)
2. Menginisialisasi 3 parameter yaitu .

Dimana :

= probabilitas topik

= probabilitas dokumen mengandung topik

= probabilitas kata terdapat dalam topik

1. Menghitung probabilitas untuk masing-masing parameter menggunakan Expectation Maximization dimana terdapat 2 langkah yaitu Step E dan Step M.

Step E :

Menghitung probabilitas topik di dalam dokumen dan kata dengan rumus :

Step M :

Memperbaharui nilai parameter dengan rumus :

P(w|z)

P(d|z)

P(z)

Kedua langkah tersebut diulangi terus menerus hingga nilai dari ketiga parameter saling mendekati atau berdasarkan tresshold tertentu.

Contoh :

Dimisalkan terdapat 9 dokumen, 5 dokumen mengenai interaksi manusia dan komputer, 4 dokumen mengenai teknik graph matematika

|  |  |
| --- | --- |
| d1 | Human machine interface for ABC computer application |
| d2 | A survey of user opinion of computer system response time |
| d3 | The EPS user interface management system |
| d4 | System and human system engineering testing of EPS |
| d5 | Relation of user perceived response time to error measurement |

|  |  |
| --- | --- |
| d6 | The generation of random, binary, ordered trees |
| d7 | The intersection graph of paths in trees |
| d8 | Graph minors IV : Widhts of trees and well-quasi-ordering |
| d9 | Graph minors: A survey |

Tabel 4. Contoh dokumen untuk penghitungan algoritma PLSA

Kemudian dilakukan perhitungan kemunculan kata dalam dokumen :

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | d1 | d2 | d3 | d4 | d5 | d6 | d7 | d8 | d9 |
| human | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| interface | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| computer | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| user | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| system | 0 | 1 | 1 | 2 | 0 | 0 | 0 | 0 | 0 |
| response | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| time | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| EPS | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| survey | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| trees | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| graph | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| minors | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

Tabel 5. Contoh matrix frekuensi kata dalam dokumen

* + - Membuat matrix jumlah kata-kata dalam masing-masing dokumen yaitu n(d,w)
    - Menentukan jumlah topik (z)

Pada tahap ini jumlah topik didapatkan dari metode Hartigan Index.

Dimisalkan topik = 3

* + - Menginisialisasi 3 parameter yaitu yang diinisialisasi secara random

Contoh :

p(w|z)

0.3000 0.2000 0.5000 🡪 dirandom, dan jika dijumlah, jumlahnya sama dengan 1

p(d|z)

0.1800 0.0300 0.0900 0.1400 0.1500 0.0100 0.1100 0.1500 0.1300

dirandom, dan jika dijumlah, jumlahnya sama dengan 1

p(z)

0.2400 0 0

0 0.6700 0

0 0 0.0900

dirandom, dan jika dijumlah, jumlahnya sama dengan 1

Menghitung probabilitas untuk masing-masing parameter menggunakan Expectation Maximization dimana terdapat 2 langkah yaitu Step E dan Step M.

Step E :

Menghitung

0 0 0 0.1993 0 0.7996 0 0 0 0 0 0.0007

0 0 0 0.3510 0 0.4036 0 0 0 0 0 0.2477

0 0 0 -0.3522 0 0.6085 0 0 0 0 0 0.7471

0 0 0 0.9986 0 0.0039 0 0 0 0 0 -0.0011

0 0 0 0.1011 0 0.3940 0 0 0 0 0 0.5021

0 0 0 0.0015 0 0.9950 0 0 0 0 0 0.0017

0 0 0 0.5979 0 0.4044 0 0 0 0 0 -0.0008

0 0 0 -0.3011 0 0.8022 0 0 0 0 0 0.4997

0 0 0 -0.3982 0 0.4001 0 0 0 0 0 0.9989

0 0 0 -0.0519 0 0.8117 0 0 0 0 0 0.2453

0 0 0 0.2496 0 0.0067 0 0 0 0 0 0.7470

0 0 0 0.0007 0 0.0006 0 0 0 0 0 0.9991

Step M :

Memperbaharui nilai parameter dengan rumus :

P(w|z)

P(d|z)

P(z)

Hasil matrix P(w|z) :

|  |  |  |  |
| --- | --- | --- | --- |
|  | Topik1 | Topik2 | Topik3 |
| human | 1.5243 | -0.5071 | 0.6065 |
| interface | 0.9973 | 0.9984 | 0.0023 |
| computer | 0.2872 | 0.7793 | 0.5877 |
| user | -0.7095 | 1.7957 | 1.5730 |
| system | 0.3431 | 0.7879 | 1.7794 |
| response | -0.7093 | 0.7912 | 1.5766 |
| time | -0.7093 | 0.7912 | 1.5766 |
| EPS | 0.5266 | 0.5035 | 0.5970 |
| survey | -0.6125 | 1.7165 | 0.7733 |
| trees | 0.9768 | 0.5891 | 0.2761 |
| graph | 1.4190 | 1.1783 | 0.1918 |
| minors | 0.9374 | 0.8658 | 0.2243 |

Tabel 6. Bobot masing-masing kata di dalam topik

Melihat dari bobot masing-masing kata di dalam masing-masing topik maka bisa diambil kesimpulan :

Topik 1 : human, trees, graph, minors

Topik 2 : interface, computer, user, system, survey

Topik 3 : response, time, EPS

Hasil matrix P(d|z) :

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | d1 | d2 | d3 | d4 | d5 | d6 | d7 | d8 | d9 |
| topik1 | 0.1981 | 0.9487 | 1.1496 | 0.1004 | 1.5980 | -0.0519 | 0.1977 | 0.1984 | -0.1479 |
| topik2 | 1.8117 | 2.8059 | 1.6037 | 2.3898 | 1.4033 | 0.8117 | 0.8184 | 0.8190 | 0.4074 |
| topik3 | 0.9955 | 2.2479 | 1.2484 | 1.5046 | -0.0002 | 0.2453 | 0.9923 | 1.9914 | 2.7450 |

Tabel 7. Bobot masing-masing topik di dalam dokumen

Melihat dari bobot masing-masing kata di dalam masing-masing topik maka bisa diambil kesimpulan :

Topik 1 paling banyak dibicarakan dalam dokumen 3

Topik 2 paling banyak dibicarakan dalam dokumen 2

Topik 3 paling banyak dibicarakan dalam dokumen 9

1. Tahap pengelompokkan dokumen dan pemberian label pada kelompok topik (Gambar 4)

* Pemberian label pada kelompok topik

Dalam hasil PLSA, hanya diketahui kata apa saja yang masuk ke dalam kelompok topik, namun tidak bisa diketahui topik tersebut membahas apa. Maka, digunakan Wordnet untuk mengetahui hal tersebut. WordNet adalah database Bahasa Inggris, yang dikembangkan di bawah arahan George A. Miller (Emeritus). Kata benda, kata kerja, kata sifat dan kata keterangan dikelompokkan ke dalam sinonim kognitif, masing-masing menyatakan sebuah konsep yang berbeda[6]. Dalam WordNet disediakan Hiponim dan Hipernim dari setiap kata-kata. Hipernim adalah kata-kata yang mewakili banyak kata lain. Kata Hipernim dapat menjadi kata umum dari penyebutan kata-kata lainnya. Sedangkan Hiponim adalah kata-kata yang terwakili artinya oleh kata Hipernim. Umumnya kata-kata Hipernim adalah suatu kategori dan Hiponim merupakan anggota dari kata Hipernim.

Contoh :

Kata-kata yang terdapat pada satu topik : human, trees, graph, minors.

Maka kita bisa membentuk suatu diagram dari keempat kata tersebut dengan mencari Hiponim atau Hipernimnya dan mengetahui hubungan dari keempat kata tersebut.

Hasil dari WordNet :



Gambar 6. Hasil WordNet untuk sebuah topik

Berdasarkan hasil dari WordNet (Gambar 6) maka dapat diambil kesimpulan bahwa pada topik tersebut yang dibicarakan adalah mengenai Organism dan Abstraction.

* Pengelompokan dokumen

Pengelompokan dokumen dilakukan dengan menggunakan algoritma K-Means. K-Means merupakan salah satu algoritma dalam pengklasteran data. Dimana pengelompokan dilakukan berdasarkan kedekatan topik-topik yang dibahas oleh masing-masing dokumen.

Algoritmanya adalah sebagai berikut :

1. Menginisialisasi jumlah dan letak k (centroid)
2. Menghitung kedekatan antara data dengan k
3. Mengelompokan data-data yang dekat dengan masing-masing k

Dimana nantinya akan didapatkan kelompok-kelompok dokumen berdasarkan kedekatan dengan topik yang dibahas.

# METODOLOGI

Metodologi yang akan dilakukan dalam Tugas Akhir ini memiliki beberapa tahapan, diantaranya sebagai berikut :

1. Desain Algoritma

Pada tahap ini dilakukan pencarian, pengumpulan, penyaringan, pembelajaran dan pemahaman yang berhubungan dengan Ruang Konsep, Expectation Maximization, Probabilistic Latent Semantic Analysis, Hypergraph Partitioning, K-Way Spectral Clustering, K-Means. Literatur yang digunakan dalam pengerjaan Tugas Akhir ini sebagian besar berasal dari internet berupa makalah ilmiah, tesis, artikel, materi kuliah, serta beberapa buku referensi.

1. Implementasi

Implementasi merupakan tahap untuk membangun sistem tersebut berupa aplikasi *desktop* menggunakan bahasa pemrograman JAVA. Pada tahap ini dilakukan pendesainan interface dan pengimplementasian algoritma.

1. Pengujian dan Evaluasi

Pada tahap ini dilakukan uji coba terhadap sistem yang telah dibuat dengan menggunakan 3 dataset yaitu :

* Reuters-21578

Dataset ini merupakan kumpulan dokumen yang muncul dalam Reuters Newswire pada tahun 1987. Dalam dataset ini terdapat 22 file teks, dimana dalam masing-masing file teks terdapat 1000 dokumen, kecuali pada dokumen terakhir terdapat 578 dokumen, jadi total terdapat 21578 dokumen. Masing-masing dokumen memiliki fitur date, topics, places, people, orgs, exchanges, companies, unknown, text, title, dan dateline [8].

* 20Newsgroups

Dataset ini merupakan kumpulan email yang awalnya dikumpulkan oleh Ken Lang. Dataset ini mengandung 20.000 dokumen yang dipisah dalam 20 topik yang berbeda, masing-masing topik memiliki 1000 dokumen. Dalam masing-masing dokumen terdapat fitur path, from, newsgroups, subject, date, organization, lines, message-id, references, expires, nntp-posting-host, dan sender [8].

* WebKB

Dataset ini merupakan kumpulan dari berbagai halaman web yang dikumpulkan dari Departemen Ilmu Komputer dari berbagai Perguruan Tinggi (Cornell, Texas Washington, Wisconsin dan beberapa Perguruan Tinggi lainnya) pada Januari 1997. Dalam dataset ini terdapat 8282 halaman yang terbagi dalam 7 kelas yaitu Mahasiswa sebanyak 1641 halaman, Fakultas sebanyak 1124 halaman, Staf sebanyak 137 halaman, Departemen sebanyak 182 halaman, Kuliah sebanyak 930 halaman, Proyek sebanyak 504 halaman, dan Lainnya sebanyak 3764 halaman. Dalam dataset ini terdapat beberapa fitur yaitu server, content-type, last-modified, content-length, head, title, dan body [8].

yang kemudian dievaluasi dengan melihat nilai :

* Precision

Precision merupakan ukuran untuk melihat kualitas dari dokumen yang sudah terklaster dengan benar terhadap dokumen dalam klaster tersebut.

Precision dapat dihitung dengan rumus :

Semakin tinggi nilai precision maka semakin baik kualitasnya[9].

* Akurasi

Akurasi merupakan ukuran untuk melihat kualitas dari dokumen yang sudah terklaster dengan benar terhadap semua dokumen dalam dataset.

Akurasi dapat dihitung dengan rumus :

Semakin tinggi nilai akurasi maka semakin baik kualitasnya[9].

* Normalized Mutual Information (NMI)

NMI adalah suatu cara untuk memperkirakan kualitas pengklasteran dengan melihat benar atau tidaknya suatu kelas dari sebuah dataset.

(7)

Keterangan :

* adalah total jumlah dokumen.
* adalah jumlah klaster atau kelas.
* dan adalah total dokumen dalam masing-masing klaster and klaster
* adalah jumlah umum di kedua klaster and klaster

Rentang nilai NMI adalah dalam 0 sampai 1 dan semakin mendekati 1 maka kualitas klaster semakin lebih baik [10].

1. Penyusunan buku Tugas Akhir

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi perangkat lunak yang telah dibuat.

# JADWAL KEGIATAN TUGAS AKHIR

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No. | Kegiatan | Bulan | | | | | | | |
| Maret | | April | | Mei | | Juni | |
| 1. | Desain Algoritma |  |  |  |  |  |  |  |  |
| 2. | Implementasi |  |  |  |  |  |  |  |  |
| 3. | Pengujian dan Evaluasi |  |  |  |  |  |  |  |  |
| 4. | Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |

# DAFTAR PUSTAKA

1. M. Matteucci (2008). *A Tutorial on Clustering Algorithms*. <http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/> diakses tanggal 3 Mei 2011
2. T. Hofmann (1999). *Probabilistic Latent Semantic Analysis*. In Proc. of Uncertainty in Artificial Intelligence, UAI’99, 1999, pp. 289--296.
3. Online SEO Guides : Google Stop Words (2011). *Words Ignored By Search Engines*. <http://www.link-assistant.com/seo-stop-words.html> diakses pada tanggal 24 Maret 2011
4. M.F. Porter. (1980). *An algorithm for suffix stripping*. Program, 14(3), 130-137.
5. Li, X., R. Ramachandran, S. Movva, S. Graves, B. Plale, and N. Vijayakumar (2008). *Storm Clustering for Data-driven Weather Forecasting*. 88th AMS Annual Meeting, New Orleans, LA.
6. Wordnet : A lexical database for English (2011). *About WordNet*. <http://wordnet.princeton.edu/> diakses pada tanggal 3 Mei 2011.
7. Tan, Steinbach, Kumar (2006). *The K-Means Algorithm*. ICDM: Top Ten Data Mining Algorithms
8. S. F. Hussain (2010). *Datasets (Machine Learning/Data Mining)*. <http://sites.google.com/site/fawadsyed/datasets> diakses pada tanggal 3 Mei 2011
9. M. Junker, R. Hoch, A. Dengel (1999). *On the Evaluation of Document Analysis Components by Recall, Precision, and Accuracy*. ICDAR '99 Proceedings of the Fifth International Conference on Document Analysis and Recognition.
10. A. Strehl (2002). *Normalized Symmetric Mutual Information*. <http://www.lans.ece.utexas.edu/~strehl/diss/node107.html> diakses pada tanggal 3 Mei 2011.

**LEMBAR PENGESAHAN**

###### **Surabaya, Mei 2011**

Menyetujui,

Dosen Pembimbing I

(Diana Purwitasari, S.Kom., M.Sc.)

( NIP.19780410 200312 2 001)

Dosen Pembimbing II

(Daniel Oranova Siahaan, S.Kom, M.Sc., PD.Eng.)

( NIP. 19741123 200604 1 001)