



UNIVERSITAS INDONESIA

**IMPLEMENTASI PEMBOBOTAN INVERSE DOCUMENT FREQUENCY
– NORMALIZED DOCUMENT VECTOR PADA SIMPLE-O**

SKRIPSI

VIVI ZULVI FAUZIAH

0606078525

FAKULTAS TEKNIK UNIVERSITAS INDONESIA

TEKNIK ELEKTRO

DEPOK JUNI 2010



UNIVERSITAS INDONESIA

**IMPLEMENTASI PEMBOBOTAN INVERSE DOCUMENT FREQUENCY
– NORMALIZED DOCUMENT VECTOR PADA SIMPLE-O**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

VIVI ZULVI FAUZIAH

0606078525

FAKULTAS TEKNIK UNIVERSITAS INDONESIA

TEKNIK ELEKTRO

DEPOK JUNI 2010

HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.

Nama : Vivi Zulvi Fauziah

NPM : 0606078525

Tanda Tangan :

Tanggal : 15 Juni 2010

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Vivi Zulvi Fauziah
NPM : 0606078525
Program Studi : Teknik Elektro
Judul Skripsi : Implementasi Pembobotan Inverse Document
Frequency – Normalized Document Vector Pada
Simple-O

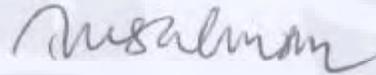
Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

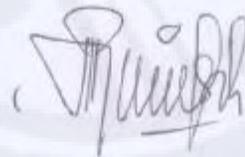
Pembimbing : Dr. Ir. Anak Agung Putri Ratna M.Eng



Penguji : Muhammad Salman ST., MIT



Penguji : Ir. Endang Sriningsih MT, Si



Ditetapkan di : Depok

Tanggal :

UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kehadirat Allah SWT, karena atas segala rahmat dan hidayat-Nya saya dapat menyelesaikan skripsi ini. Saya menyadari bahwa skripsi ini tidak akan terselesaikan tanpa bantuan dari berbagai pihak. Oleh karena itu, saya mengucapkan terima kasih kepada :

1. Ibu Dr. Ir. Anak Agung Putri Ratna, M.Eng, selaku pembimbing skripsi saya, terima kasih atas arahan serta koreksi skripsi saya ini dengan sabar, dan telah memberikan banyak waktu untuk mengarahkan saya hingga selesai.
2. Para peneliti sebelum ini yang juga memberikan sumber bacaan yang banyak bagi saya.
3. Orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral kepada saya.
4. Sahabat, teman satu kelompok (Bram, Boma, Iyus, Mei) serta semua yang telah memberi semangat saya dalam menyelesaikan skripsi ini tepat waktu.
5. Dan seluruh Sivitas akademik Departemen Teknik Elektro yang tidak dapat saya sebutkan satu persatu.

Akhir kata, semoga Allah SWT berkenan membalas kebaikan semua pihak yang telah membantu. Semoga skripsi ini bermanfaat bagi perkembangan ilmu pengetahuan.

Depok, 15 Juni 2010

Vivi Zulvi Fauziah

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademika Universitas Indonesia, saya bertanda tangan di bawah ini :

Nama : Vivi Zulvi Fauziah

NPM : 0606078525

Program studi : Teknik Komputer

Departemen : Teknik Elektro

Fakultas : Teknik

Jenis karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Nonokklusif (*Non-exclusive Royalty Free Right*)** atas karya ilmiah saya yang berjudul :

**IMPLEMENTASI PEMBOBOTAN INVERSE DOCUMENT FREQUENCY
– NORMALIZED DOCUMENT VECTOR PADA SIMPLE-O**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Eksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta sebagai pemegang Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 15 Juni 2010

Yang menyatakan

Vivi Zulvi Fauziah

ABSTRAK

Nama : Vivi Zulvi Fauziah
Program studi : Teknik Komputer
Judul : Implementasi Pembobotan Inverse Document Frequency –
Normalized Document Frequency Pada SIMPLE-O

Metode *Latent Semantic Analysis* (LSA) adalah salah satu metode yang digunakan untuk sistem penilaian essay secara otomatis (*essay grading*). Metode ini merepresentasikan suatu tulisan essay kedalam bentuk matriks dimana pada matriks tersebut dihitung frekuensi kemunculan suatu kata. Matriks tersebut kemudian diolah dengan cara penganalisisan bernama SVD (*Singular Value Decomposition*) yang akan mengkompres matriks besar menjadi matriks yang lebih kecil namun matriks yang lebih kecil tersebut mampu mewakili korelasi antar kata yang lebih baik dibandingkan dengan matriks sebelumnya. Pada skripsi ini akan dibahas mengenai aplikasi SIMPLE-O yang dikembangkan dengan cara memberi pembobotan pada kata-kata kunci pada matriks awal. Pembobotan yang digunakan adalah IDF-NDV (*Inverse Document Frequency – Normalized Document Vector*) yang telah diujikan pada penelitian sebelumnya. Setelah diujikan, pembobotan IDF-NDV masih perlu perbaikan untuk mendekati human raters, dan dibandingkan dengan SIMPLE-O lama, terdapat perbedaan sebesar 11,78%.

Kata kunci : *Essay Grading*, Simple-O, Pembobotan, *Human Raters*, IDF-NDV

ABSTRACT

Name : Vivi Zulvi Fauziah
Study Program: Computer Engineering
Title : Implementation of Inverse Document Frequency- Normalized Document Vector Weighting to SIMPLE-O

Method of Latent Semantic Analysis (LSA) is one of the methods used for automatic essay grading system (grading essays). This method represents an essay written in the matrix form where the matrix is calculate the frequency of occurrence of a word. Matrix is then processed by analyzing method called SVD (Singular Value decomposition), which compresses the large matrix into smaller matrices, but that smaller matrix can represent the correlation between a better word than the previous matrix. In this final project will discuss about the application of SIMPLE-O who developed a way to give weighting to the key words at the matrix. The Weighting method was used is IDF-NDV (Inverse Document Frequency - Normalized Document Vector) that have been tested in previous research. After tested, the IDF-NDV Weighting still need improvement to approach the human raters, and compared with Original SIMPLE-O, there is a difference of 11,78%.

Key words : Essay Grading, Simple-O, Weighting, Human Raters, IDF-NDV

DAFTAR ISI

HALAMAN JUDUL	ii
HALAMAN PERNYATAAN ORISINALITAS	iii
HALAMAN PENGESAHAN	iv
UCAPAN TERIMA KASIH	v
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI	vi
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
DAFTAR RUMUS	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan Penulisan.....	2
1.3 Batasan Masalah.....	2
1.4 Sistematika Penulisan.....	2
BAB II ESSAY GRADING DENGAN PEMBOBOTAN IDF-NDV	4
2.1 Latent Semantic Analysis	4
2.2 Essay Grading Menggunakan LSA.....	6
2.3 Metode Pembobotan LSA Pada Essay Grading.....	7
2.4 Metode Pembobotan IDF-NDV	10
BAB III PERANCANGAN APLIKASI PEMBOBOTAN IDF-NDV PADA SIMPLE-O	13
3.1 Konsep Aplikasi.....	13
3.2 Rancang Bangun Program Utama SIMPLE-O	13
3.3 Proses Pembobotan IDF-NDV.....	23
3.3.1 Algoritma Pembobotan IDF-NDV	23
3.3.2 Aplikasi Pembobotan IDF-NDF Pada Program Utama.....	24
BAB IV PENGUJIAN DAN ANALISIS APLIKASI SIMPLE-O DENGAN PEMBOBOTAN IDF-NDV	28

4.1	Pengujian Aplikasi	28
4.2	Analisis Pengujian.....	30
4.2.1	Metode Penilaian Dengan SIMPLE-O Murni.....	31
4.2.2	Metode Penilaian Dengan SIMPLE-O IDF-NDV	35
4.2.3	Metode Penilaian Manual (<i>Human Raters</i>)	36
4.2.4	Analisis Perbandingan Metode SIMPLE-O Murni, Metode SIMPLE-O IDF-NDV, dan Metode Manual.....	38
4.2.5	Pengaruh Pembobotan Awal SIMPLE-O Dan Pembobotan IDF-NDV Terhadap Waktu Proses.....	43
BAB V KESIMPULAN		47
DAFTAR ACUAN.....		48

DAFTAR TABEL

Tabel 2-1 Pembobotan Lokal (Chisholm & Kolda, 1999)	8
Tabel 2-2 Pembobotan Global [2].....	9
Tabel 4-1 Perolehan Nilai Metode Pertama.....	31
Tabel 4-2 Jawaban Mahasiswa Nilai Tertinggi.....	32
Tabel 4-3 Kunci Jawaban	33
Tabel 4-4 Kata Kunci dan Kata Bobot	34
Tabel 4-5 Perolehan Nilai Metode Kedua	35
Tabel 4-6 Jawaban Mahasiswa Nilai Terendah	36
Tabel 4-7 Penilaian Secara Manual.....	37
Tabel 4-8 Selisih Nilai Kedua Metode	38
Tabel 4-9 Jawaban User 18.....	39
Tabel 4-10 Proses Waktu SIMPLE-O Murni	44
Tabel 4-11 Proses Waktu SIMPLE-O IDF-NDV	45

DAFTAR GAMBAR

Gambar 2-1 Dekomposisi SVD Dengan Faktor k [5]	6
Gambar 2-2 Perbandingan Pembobotan IWF –IDF[6]	10
Gambar 2-3 Perbandingan Pembobotan Normal, dan IWF[6]	11
Gambar 2-4 Perbandingan IWF-NDV Dengan IDF-NDV[6]	12
Gambar 3-1 Flowchart Cara Kerja Aplikasi Simple-O [1].....	14
Gambar 3-2 Tampilan Menu Login	15
Gambar 3-3 Algoritma Login	16
Gambar 3-4 Modul Administrator.....	17
Gambar 3-5 Halaman Utama Dosen	18
Gambar 3-6 Algoritma List Nilai.....	18
Gambar 3-7 Algoritma Soal.....	19
Gambar 3-8 Algoritma Input Soal.....	19
Gambar 3-9 Algoritma Memilih Kata Bobot.....	20
Gambar 3-10 Login Mahasiswa.....	20
Gambar 3-11 Algoritma pengecekan kata dan Pembentukan Matriks.....	21
Gambar 3-12 Algoritma Proses Pemasukkan Jawaban Ujian	22
Gambar 3-13 Algoritma IDF	23
Gambar 3-14 Algoritma NDV	24
Gambar 3-15 Flowchart Penyisipan Pembobotan.....	25
Gambar 3-16 Listing Program Pembobotan IDF-NDV Dalam PHP	26
Gambar 4-1 Halaman Login	28
Gambar 4-2 Halaman Utama Modul Mahasiswa.....	29
Gambar 4-3 Form Isian Jawaban	30
Gambar 4-4 Grafik Perbandingan Metode Penilaian	38
Gambar 4-5 Perbandingan Ketiga Metode	41
Gambar 4-6 Hubungan Antara Panjang Dokumen Dengan Perolehan Nilai	42
Gambar 4-7 Perbandingan Waktu Proses.....	45

DAFTAR RUMUS

Rumus 2.1	5
Rumus 2.2	7
Rumus 2.3	7
Rumus 2.4	10
Rumus 2.5	11
Rumus 2.6	11

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dari perkembangan teknologi, manusia terus mencoba untuk membuat suatu teknologi yang mampu mempermudah pekerjaan dan aktivitas mereka, serta mampu menambah produktivitas dan efisiensi waktu kerja. Pada dewasa ini, hampir semua kebutuhan manusia tidak bisa terlepas dari teknologi komputer dengan berbagai kemampuannya. Dengan adanya teknologi yang semakin berkembang pesat ditemukanlah berbagai aplikasi pada komputer dan periperalnya, seperti teknologi jaringan, web, database, dan lain sebagainya. Teknologi jaringan membuat manusia dapat saling bertukar informasi dengan waktu yang cepat tanpa harus berpindah tempat, dan aplikasi yang sangat berkembang dalam jaringan adalah database yang berbasis web.

Teknologi web yang berkembang dengan cepat membuat web dapat dibentuk menjadi berbagai macam aplikasi yang sangat membantu kegiatan manusia secara online, jadi dimanapun manusia itu berada, asalkan mampu terkoneksi dengan internet, manusia tersebut mampu membuka aplikasi, dokumen, atau pekerjaan apapun tanpa harus membawanya kemana-mana.

Salah satu teknologi yang sedang berkembang adalah pemeriksaan jawaban essay secara online secara otomatis. Metode yang dapat digunakan adalah LSA (*Latent Semantic Analysis*). Metode lainnya yang digunakan untuk essay grading adalah *Project Essay Grade* (PEG), *E-rater*, *Text Categorisation Technique* (TCT), dan *Intelligent Essay Assessor* (IEA) [3], SIMPLE-O[1]. LSA (*Latent Semantic Analysis*) merupakan salah satu metode yang digunakan untuk *essay grading*. Teknologi ini memungkinkan tulisan dapat dianalisis dengan cepat dan dapat dilakukan grading atau penilaian juga dengan cepat berdasarkan kata kunci atau *database* yang telah diberikan [4].

Metode LSA ini terus dikembangkan dengan menerapkan pembobotan pada perhitungan matriksnya. Banyak teknologi pembobotan pada LSA yang telah diterapkan, dan tujuan pengembangan ini adalah untuk mendekati *human raters*

(penilai manusia) sehingga hasil yang akan diperoleh dari penilaian otomatis ini dapat dipertanggungjawabkan.

Salah satu metode pembobotan yang dapat diterapkan adalah metode IDF (Inverse Document Frequency) + NDV (Normalized Document Vector) dan diterapkan pada aplikasi SIMPLE-O kemudian akan dibandingkan dengan metode pembobotan yang sudah ada pada aplikasi SIMPLE-O. Dari perbandingan ini, akan didapat metode yang terbaik untuk penilaian otomatis SIMPLE-O.

Metode Pembobotan IDF+NDV sebelumnya telah diujikan pada aplikasi LSA yang lain, dan telah terbukti lebih baik jika dibandingkan dengan metode IWF+NDV, yaitu mampu menghasilkan keakuratan penilaian sebesar 99%[6].

1.2 Tujuan Penulisan

Tujuan dari penulisan skripsi ini adalah untuk menganalisis metode pembobotan IDF-NDV sehingga diketahui apakah metode pembobotan tersebut mendekati human raters dan dapat diterapkan secara umum untuk sistem penilaian essay otomatis, dan akan dibandingkan dengan program SIMPLE-O yang telah ada.

1.3 Batasan Masalah

Masalah yang akan dibahas dibatasi pada implementasi pembobotan IDF-NDV pada SIMPLE-O kemudian akan dibandingkan dengan pembobotan yang telah ada pada SIMPLE-O.

1.4 Sistematika Penulisan

Bab I Pendahuluan

Pada bab ini berisi Latar Belakang, Tujuan Penulisan, Batasan Masalah, dan Sistematika Penulisan.

Bab II *Essay Grading* Dengan Pembobotan IDF-NDV

Pada bab ini akan dibahas mengenai dasar teori yang berhubungan dengan LSA, serta metode pembobotan IDF-NDV

Bab III Perancangan Aplikasi Pembobotan IDF-NDV Pada SIMPLE-O

Pada bab ini akan dirancang program essay grading yang ada (SIMPLE-O) dan kemudian akan diberi metode pembobotan IDF-NDV untuk mengetahui apakah pembobotan tersebut lebih mampu mendekati hasilnya ke *human raters* jika dibandingkan dengan metode yang telah diaplikasikan pada SIMPLE-O sebelumnya.

Bab IV Uji Coba Dan Analisis Pengujian Aplikasi

Pada bab ini akan dibahas analisis program yang telah dibuat dan kemudian disimpulkan mengenai performansi dari program tersebut.

Bab IV Kesimpulan

Bab ini merupakan kesimpulan dari penulisan skripsi ini.

BAB II

ESSAY GRADING DENGAN PEMBOBOTAN IDF-NDV

2.1 Latent Semantic Analysis

Latent Semantic Analysis merupakan sebuah metode dan teori untuk mengekstrak dan merepresentasikan penggunaan sebuah kata secara kontekstual dan kesinambungan pengertian dengan menggunakan perhitungan statistic yang dilakukan pada suatu tulisan [4] , dan dengan kata lain, LSA merupakan suatu cara atau metode untuk menghitung tingkat keperluan suatu kata untuk digunakan dengan perhitungan statistic sehingga menghasilkan data-data yang dapat diolah pada sistem komputer biner. Metode LSA sering diaplikasikan pada mesin AI (*Artificial Intelligence*/kecerdasan buatan) untuk mempelajari arti sebuah tulisan.

LSA dipatenkan pada tahun 1988 oleh Scott Deerwester, Susan Dumais, George Furnas, Richard Harshman, Thomas Landauer, Karen Lochbaum, Lynn Streeter[4], dan dimulai penggunaannya untuk aplikasi dengan tujuan seperti:

- Membandingkan dokumen dengan konsep yang sudah ada
- Menemukan kesamaan dari banyak dokumen tanpa kendala bahasa
- Menemukan hubungan dari kata-kata (sinonim, polisemi, dan sebagainya)

Fungsi-fungsi LSA tersebut terus dikembangkan menjadi aplikasi yang membantu dalam pengerjaan hal-hal yang sulit dilakukan dengan cepat seperti pemeriksaan hasil ujian essay, mencari indeks dari suatu teks, mencari keterkaitan satu dokumen dengan dokumen yang lain.

Ide dasar dari LSA adalah mencari frekuensi dari kemunculan suatu kata dan direpresentasikan dalam bentuk matriks. Matriks dibentuk dengan memasukkan data dari suatu bagian tulisan dimana baris-baris pada matriks tersebut merupakan kata-kata unik yang terdapat pada bagian tulisan tersebut, sedangkan kolomnya berisi bagian-bagian tulisan yang saling terpisah pada satu teks utuh. Setiap cell pada matriks tersebut berisi angka frekuensi kemunculan sebuah kata pada satu bagian tulisan.

Hal selanjutnya yang dilakukan adalah dengan mentransformasikan matriks tersebut dengan suatu fungsi aljabar linear yang disebut dengan SVD (*Singular Value Decomposition*). SVD merupakan metode analisis dengan membentuk 3 buah matriks yang apabila dikalikan akan membentuk matriks semula. SVD berfungsi untuk mengestimasi rank dari matriks. Jika diketahui matriks A dengan dimensi $m \times n$, dimana nilai $m \geq n$, dan $\text{rank}(A) = r$ maka nilai *singular value decomposition* dari matriks A dinotasikan sebagai $\text{SVD}(A)$. Maka $\text{SVD}(A)$ dapat ditentukan dengan:

$$A = U\Sigma V^T \dots\dots\dots (2.1)$$

dimana,

$$U^T U = V^T V = I_n$$

$$\Sigma = \text{diag}(s_1, \mathbf{L}, s_n)$$

dimana,

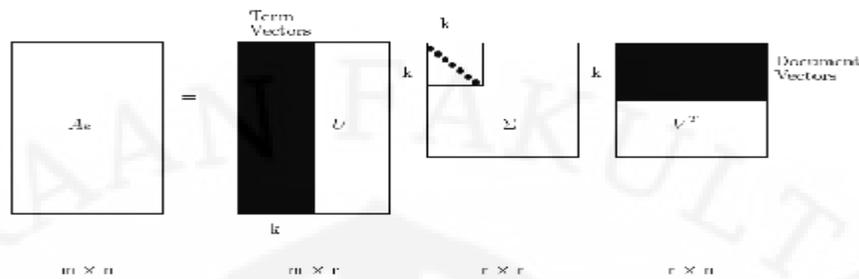
$$s_i > 0 \quad \text{untuk} \quad 1 \leq i \leq r$$

$$s_j = 0 \quad \text{untuk} \quad j \geq r + 1$$

Kolom r pertama dari matriks U dan V mendefinisikan vektor eigen orthonormal yang bersesuaian dengan r nilai vektor eigen tidak-nol dari matriks AA^T dan $A^T A$ berturut-turut. Kolom dari matriks U dan V berisi vektor, masing-masing disebut vektor singular kiri dan kanan.

Nilai singular dari A merupakan elemen diagonal dari matriks Σ , dimana nilai singular didapat dari akar pangkat dua dari nilai absolut dari sejumlah n nilai eigen dari AA^T . Matriks-matriks yang terbentuk dari operasi SVD ini adalah matriks-matriks yang bebas linier yang merepresentasikan vektor-vektor ortogonal pada matriks A .

Setelah ketiga matriks didapatkan, maka yang dilakukan selanjutnya adalah memperkecil dimensi dari matriks diagonal Σ dengan cara mengganti beberapa elemen-elemen diagonal pada matriks Σ dengan nol. Anggaphlah ditentukan faktor reduksi k dan dimensi matriks Σ adalah $n \times n$ dimana $n > k$, maka akan didapatkan matriks Σ yang semula berukuran $n \times n$ menjadi $k \times k$, seperti yang terlihat pada Gambar 2.1.



Gambar 2-1 Dekomposisi SVD Dengan Faktor k [5]

Setelah terbentuk matriks-matriks yang telah direduksi, kemudian perkalian dari ketiga matriks ini akan membentuk matriks A_k , yang dapat mewakili vektor-vektor pada matriks A.

2.2 Essay Grading Menggunakan LSA

Ujian essay pada saat ini tetap menjadi metode pengujian yang menjadi andalan para pengajar untuk memperoleh tingkat pemahaman para anak didiknya atas pengajaran yang selama ini diberikan. Penilaian jawaban essay biasanya tidak objektif karena tidak ada acuan untuk penilaiannya seperti yang ada pada ujian dengan pilihan ganda. Pemeriksaan ujian essay membutuhkan waktu yang lama jika diperiksa secara manual, sehingga dibutuhkan suatu metode untuk memberikan penilaian ujian secara essay dan secara objektif yang salah satu metodenya adalah dengan essay grading menggunakan teknik LSA.

Seperti telah disebutkan diatas, bahwa metode LSA dapat digunakan untuk membentuk matriks dari suatu susunan kata-kata yang terdapat pada ujian essay. Matriks ini adalah matriks yang merepresentasikan frekuensi kemunculan suatu kata yang telah ditentukan sebelumnya (kata kunci). Dengan matriks ini dapat dibentuk suatu vektor representasi dari jawaban essay tersebut.

Untuk dapat memeriksa suatu jawaban essay, diperlukan suatu perbandingan yang selanjutnya akan digunakan untuk bahan perbandingan tingkat kemiripan jawaban essay tersebut, jawaban perbandingan biasanya disebut dengan query, dan jawaban yang akan diperiksa biasanya disebut dokumen. Proses pembentukan matriks query sama seperti pembentukan matriks pada pembentukan matriks dokumen, dimana matriks tersebut dibentuk dengan metode LSA.

Tingkat kemiripan suatu dokumen dengan query dapat ditentukan dengan mencari nilai kosinus dari vektor-vektor yang dibentuk oleh kedua matriks tersebut yaitu matriks query dan matriks dokumen.

$$\cos \alpha = \frac{\bar{q} \cdot \bar{d}}{\|\bar{q}\| \|\bar{d}\|} \quad (2.2)$$

Dimana, \bar{q} merupakan sebuah vektor dari query dan \bar{d} adalah vektor dokumen. α adalah sudut diantara kedua vektor tersebut. Jika \bar{q} dan \bar{d} dinormalisasi, maka *magnitude* dari vektor tersebut adalah 1 dan persamaan diatas dapat disederhanakan menjadi $\cos \alpha = \bar{q} \times \bar{d}$. Jadi, nilai korelasi adalah perhitungan sudut berdasarkan kosinus antara \bar{q} dan \bar{d} . Jika dilakukan pengurutan dari dokumen yang paling dekat ke paling jauh relevansinya, maka dokumen yang paling dekat adalah dokumen yang memiliki sudut α dengan yang paling kecil.

Untuk menjamin relevansi dokumen dengan jawaban aslinya, maka diperlukan uji perbandingan metode tersebut dengan human raters atau penilai manusia, dimana untuk LSA murni, nilai deviasi yang diijinkan adalah sebesar 85% - 91% [5] dan metode SIMPLE-O yang telah ada mampu menghasilkan kedekatan dengan *human raters* sebesar 98% [1].

2.3 Metode Pembobotan LSA Pada Essay Grading

Untuk memperoleh hasil yang lebih baik dan dapat mendekati dengan penilai manusia (*human raters*), maka diperlukan satu metode pembobotan kata kunci dimana untuk kata kunci yang paling berpengaruh pada suatu jawaban akan diberi bobot yang paling tinggi. Metode pembobotan dibagi menjadi tiga bagian, yaitu pembobotan lokal, pembobotan global, dan normalisasi. Setiap elemen pada matriks A akan dibobotkan dengan mengalikan ketiga pembobotan tersebut, sesuai dengan rumus:

$$a_{ij} = L(i, j) \times G(i) \times N(j) \dots \dots \dots (2.3)$$

$L(i,j)$ merupakan bobot lokal untuk kata kunci i dalam dokumen j . $G(i)$ adalah bobot global untuk kata kunci i , dan $N(j)$ adalah faktor normalisasi dokumen j . Bobot lokal adalah fungsi dari berapa banyak setiap kata kunci muncul dalam suatu dokumen. Bobot global adalah fungsi dari berapa banyak setiap kunci muncul dalam semua dokumen. Pembobotan lokal yang bisa digunakan terlihat pada Tabel 2-1:

Tabel 2-1 Pembobotan Lokal (Chisholm & Kolda, 1999)

Formula	Nama Metode	Kependekan
$\begin{cases} 1 & \text{jika } f_{ij} > 0 \\ 0 & \text{jika } f_{ij} = 0 \end{cases}$	Biner	BNRY
f_{ij}	Frekuensi intra-dokumen	FREQ
$\begin{cases} 1 + \log f_{ij} & \text{jika } f_{ij} > 0 \\ 0 & \text{jika } f_{ij} = 0 \end{cases}$	Log	LOGA
$\begin{cases} \frac{1 + \log f_{ij}}{1 + \log a_j} & \text{jika } f_{ij} > 0 \\ 0 & \text{jika } f_{ij} = 0 \end{cases}$	Normalisasi log	LOGN
$\begin{cases} \sqrt{f_{ij} - 0,5} + 1 & \text{jika } f_{ij} > 0 \\ 0 & \text{jika } f_{ij} = 0 \end{cases}$	Akar pangkat dua	SQRT
$\begin{cases} 0,5 + 0,5 \left(\frac{f_{ij}}{x_j} \right) & \text{if } f_{ij} > 0 \\ 0 & \text{if } f_{ij} = 0 \end{cases}$	Normalisasi frekuensi diperlebar	ATF1
$\begin{cases} 0,9 + 0,1 \left(\frac{f_{ij}}{a_j} \right) & \text{jika } f_{ij} > 0 \\ 0 & \text{jika } f_{ij} = 0 \end{cases}$	Normalisasi frekuensi rata-rata diperlebar	ATFA
$\begin{cases} 0,2 + 0,8 \left(\frac{f_{ij}}{x_j} \right) & \text{jika } f_{ij} > 0 \\ 0 & \text{jika } f_{ij} = 0 \end{cases}$	ATF1 dengan perubahan koefisien	ATFC

Pembobotan lokal BNRV merupakan pembobotan yang paling sederhana, karena pembobotan ini hanya mengganti nilai elemen matriks dengan 1 jika frekuensi kemunculannya lebih dari 1, dan 0 jika frekuensi kemunculannya nol. Metode sederhana lainnya adalah metode FREQ dimana, semua nilai matriks tidak diubah. Metode ini digunakan untuk pembobotan pada query dimana frekuensi kemunculan kata kunci tidak banyak, dan metode ini kurang baik jika digunakan pada dokumen karena metode ini tidak membedakan kata kunci yang banyak muncul dengan kata kunci yang jarang muncul.

Sedangkan untuk pembobotan global, metode yang sering digunakan terlihat pada Tabel 2-2.

Tabel 2-2 Pembobotan Global [2]

Formula	Nama Metode	Kependekan
$\log\left(\frac{N}{n_i}\right)$	Invers frekuensi dokumen	IDFB
$\log\left(\frac{N-n_i}{n_i}\right)$	Invers probabilistik	IDFP
$1 + \sum_{j=1}^N \frac{f_{ij}}{F_i} \log\left(\frac{f_{ij}}{F_i}\right) / \log N$	Entropi	ENPY
$\frac{F_i}{n_i}$	Frekuensi global IDF	IGFF
$\sqrt{\frac{F_i}{n_i} - 0,9}$	Akar pangkat dua global IDF	IGFS
1	Tidak ada bobot global	NONE

2.4 Metode Pembobotan IDF-NDV

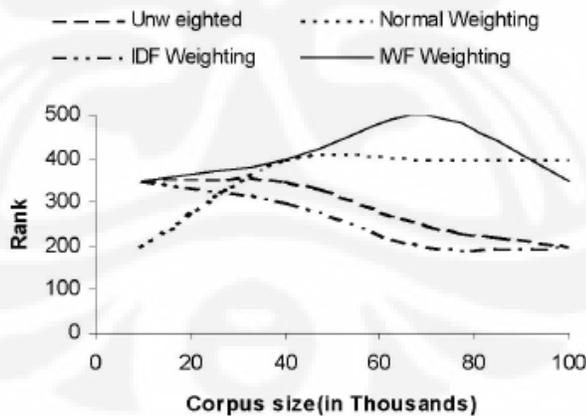
Salah satu metode yang bisa diterapkan adalah metode pembobotan IDF-NDV. IDF merupakan kependekan dari Inverse Document Frequency – Normalized Document Vektor. IDF merupakan pembobotan global dan NDV adalah pembobotan local. Metode pembobotan ini sudah diujikan oleh Sudarus dan kawan-kawan dan membandingkan metode ini dengan metode lainnya seperti IWF-NDV dan NDV. Metode ini terbukti menghasilkan nilai cos yang lebih kecil dibandingkan dengan metode yang lainnya, artinya metode ini mampu mewakili human raters lebih baik[6].

IDF adalah pembobotan global dimana pada pembobotan ini kata kunci yang jarang muncul akan dikuatkan karena tidak semua kata kunci yang sering muncul adalah kata kunci yang penting. Inverse Document Frequency jika dibandingkan dengan Inverse Word Frequency akan menghasilkan nilai cos yang lebih kecil, artinya lebih baik. Rumus untuk IDF terlihat pada Rumus 2.4.

$$n(t_i, d_j) = n(t_i, d_j) * \log \frac{N}{NF(t_j)} \dots\dots\dots(2.4)$$

N : Jumlah total dokumen dalam sebuah tulisan

NF(t_j): Jumlah dokumen dimana kata “i” muncul



Gambar 2-2 Perbandingan Pembobotan IWF –IDF[6]

Jika dibandingkan pada grafik pada Gambar 2.2, maka terlihat metode IWF memiliki rank yang lebih tinggi dibandingkan dengan IDF, artinya pada

metode IDF akan menghasilkan cakupan yang lebih jauh, dan akan menghasilkan vektor yang lebih mewakili keseluruhan isi dokumen.

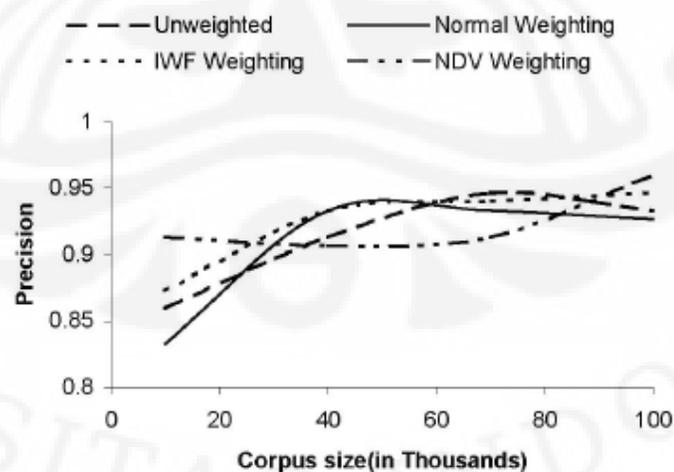
NDV merupakan metode pembobotan lokal hasil modifikasi dari Normal Weighting dimana pembobotan ini menormalisasi semua elemen pada matriks berdasarkan jumlah frekuensi pada seluruh kata kunci di sebuah dokumen. Rumus Normal Weighting terlihat pada Rumus 2.5

$$n(t_i, d_j) = \frac{n(t_i, d_j)}{\sum_{i'=1}^m (n(t_{i'}, d_j))} \dots \dots \dots (2.5)$$

Pembobotan ini berfungsi untuk mengubah nilai setiap elemen menjadi kurang dari 1 agar mencegah adanya nilai terlalu ekstrim muncul. Kemudian, rumus Normal Weighting ini dimodifikasi menjadi nilai Normalized Document Vector :

$$n(t_i, d_j) = \frac{n(t_i, d_j)}{\sqrt{\sum_{i'=1}^m (n(t_{i'}, d_j))^2}} \dots \dots \dots (2.6)$$

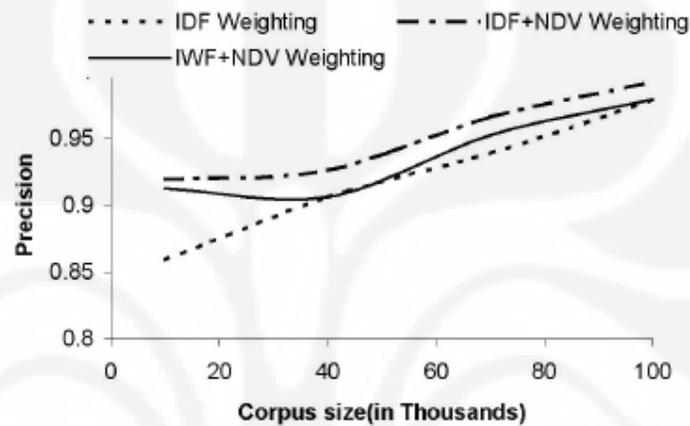
NDV memiliki fungsi yang sama dengan Normal Weighting, yaitu mencegah munculnya nilai ekstrim pada elemen matriks, namun pada NDV, normalisasi dilakukan berdasarkan magnitude dari vektor tersebut. Hasil percobaan NDV pada beberapa jumlah dokumen yang berbeda terlihat pada grafik pada Gambar 2.3



Gambar 2-3 Perbandingan Pembobotan Normal, dan IWF[6]

Pada Gambar 2-3 terlihat bahwa NDV memiliki presisi yang makin tinggi seiring dengan bertambahnya jumlah corpus. Dan secara garis besar performa NDV cukup baik apabila diaplikasikan pada LSA.

Setelah kedua pembobotan ini digabungkan, performanya terlihat pada Gambar 2-4, terlihat pada grafik tersebut secara garis besar performa IDF-NDV memiliki tingkat presisi yang paling tinggi jika dibandingkan dengan pembobotan IWF-NDV atau IDF yang berdiri sendiri.



Gambar 2-4 Perbandingan IWF-NDV Dengan IDF-NDV[6]

BAB III

PERANCANGAN APLIKASI PEMBOBOTAN IDF-NDV PADA SIMPLE-O

3.1 Konsep Aplikasi

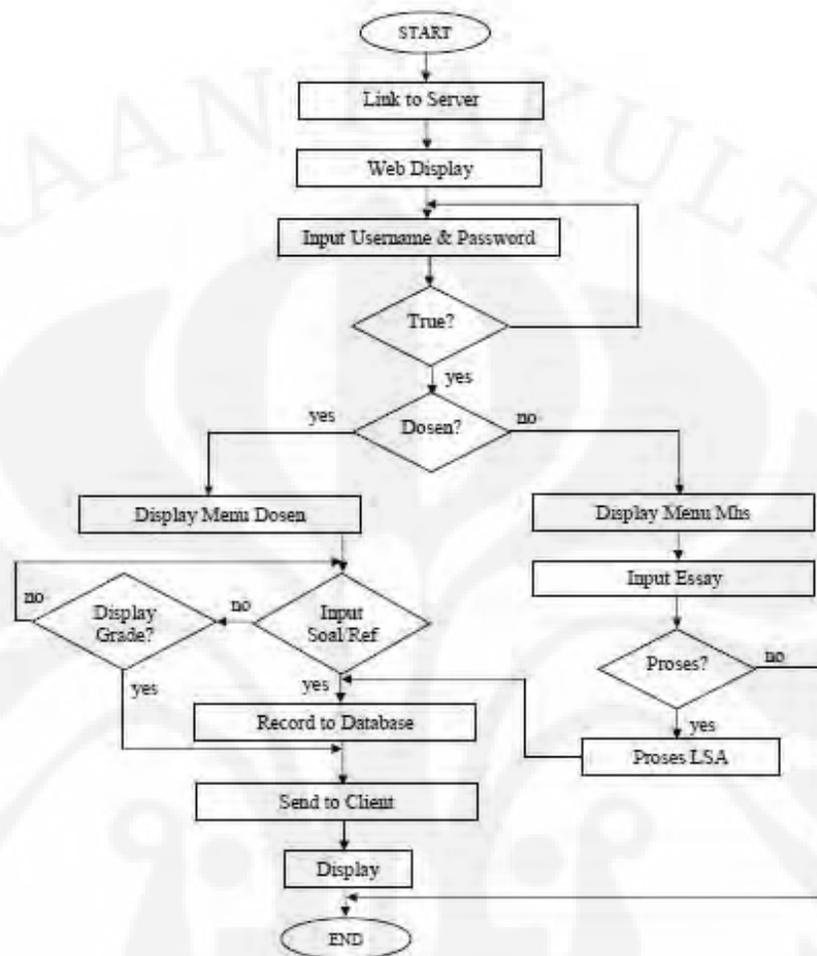
Program SIMPLE-O (Sistem Penilaian Esay Otomatis) merupakan suatu program yang dikembangkan oleh para dosen di Departemen Teknik Elektro Universitas Indonesia yang berfungsi untuk memberikan nilai secara online dan otomatis kepada suatu essay hasil dari ujian. SIMPLE-O tersebut belum terdapat pembobotan secara otomatis, sehingga pada skripsi ini akan dicoba untuk mengaplikasikan pembobotan IDF-NDV pada SIMPLE-O. Pembobotan IDF-NDV berdasarkan suatu ujicoba yang dilakukan oleh Sudarsun, Vankatesh Prabu G, Sathish Kumar V, merupakan pembobotan yang paling mendekati human rater terhadap suatu essay [6]. Program yang akan dibangun dan dianalisis kali ini adalah SIMPLE-O yang telah diberi pembobotan IDF-NDV sehingga menghasilkan suatu kesimpulan apakah pengaplikasian pembobotan IDF-NDV ini bisa membuat program SIMPLE-O menjadi lebih mendekati *human raters*.

Aplikasi SIMPLE-O dibangun berbasis web, sehingga bahasa yang digunakan adalah PHP dengan integrasi My SQL untuk databasenya. Aplikasi ini dibangun dengan menggunakan bantuan aplikasi NetBean.

3.2 Rancang Bangun Program Utama SIMPLE-O

Program ini merupakan program penilaian ujian essay dengan sistem online yang dikembangkan di Departemen Teknik Elektro Universitas Indonesia. Flowchart dari program ini terlihat pada Gambar 3-1 [1].





Gambar 3-1 Flowchart Cara Kerja Aplikasi Simple-O [1]

Flowchart pada Gambar 3.1 di atas menjelaskan mengenai cara kerja aplikasi SIMPLE-O mulai dari *link to server* hingga display akhir dari penilaian yang telah dikirimkan kepada client. Pada awal sistem diberikan input berupa username dan password kemudian diperiksa oleh sistem apakah username dan password yang telah dimasukkan benar atau tidak, jika salah sistem akan kembali meminta memasukkan username dan password namun jika benar akan diteruskan dan diperiksa apakah user dosen atau bukan. Jika user merupakan dosen maka akan tampil menu dosen, sedangkan jika user bukan dosen maka akan tampil menu mahasiswa.

Pada display menu dosen, dosen akan memasukkan input berupa soal atau referensi jawaban yang kemudian akan disimpan dalam database. Jika dosen tidak

memasukkan input maka akan tampil pilihan apakah dosen mau menampilkan grade atau tidak, jika iya maka akan ditampilkan nilai yang diperoleh setelah proses LSA, dan jika tidak maka akan kembali ke menu awal.

Pada display menu mahasiswa, mahasiswa tersebut akan memilih mata kuliah yang diinginkan dan kemudian memasukkan jawaban essay mereka. Setelah itu sistem akan menanyakan jawaban tersebut akan diproses atau tidak. Jika ya maka jawaban tersebut akan diproses menggunakan LSA dan hasilnya akan masuk ke dalam database untuk dibandingkan dengan jawaban dosen. Metode perbandingan yang diaplikasikan adalah mencari nilai cosinus dari vektor yang dibentuk oleh jawaban dosen dan jawaban mahasiswa. semakin besar nilai cosinusnya (mendekati 1) berarti jawaban dosen dan mahasiswa memiliki tingkat kesamaan yang tinggi (nilainya baik) dan sebaliknya.

Secara struktur program, aplikasi yang dibangun memiliki empat modul yaitu:

- 1) Modul login
- 2) Modul Administrator (root)
- 3) Modul dosen
- 4) Modul mahasiswa

Modul login adalah display awal yang akan tampil ketika user membuka aplikasi SIMPLE-O. Pada menu ini terdapat isian username beserta password, dan username yang dimasukkan nantinya akan diperiksa oleh sistem apakah dia sebagai dosen, mahasiswa, atau root.



Login	
Userid	<input type="text"/>
Password	<input type="password"/>
<input type="button" value="Reset"/> <input type="button" value="Submit"/>	

Gambar 3-2 Tampilan Menu Login

Algoritma dari modul login terlihat pada Gambar 3-3, dimana algoritma tersebut akan memeriksa apakah username yang dimasukkan ada pada *database*,

jika ada, apakah username tersebut sebagai mahasiswa, dosen atau root (administrator). Menu selanjutnya yang tampil tergantung dari username tersebut, jika user login sebagai mahasiswa, maka selanjutnya akan tampil menu mahasiswa, jika dosen maka akan tampil menu dosen, dan jika sebagai admin, maka akan tampil menu admin.

```

prog());
  [Pengecekan idetifikasi user]

  if idlogin = 1 then           {login dosen}
  (
    [tampilan utama untuk dosen]
    if userName = root then
    (
      [tampilkan menu tambahan untuk root]
    )
    read (pil);
    [ke sub modul sesuai pilihan : listing nilai, tampilkan dan mengisi soal, registrasi atau
logout]
  )
  if idlogin = 2 then         {login mahasiswa}
  (
    [tampilan utama untuk mahasiswa]
    read (pil);
    [ke sub modul sesuai pilihan : listing nilai, Tampilkan dan Menjawab soal, registrasi
atau logout]
  )
Eprog

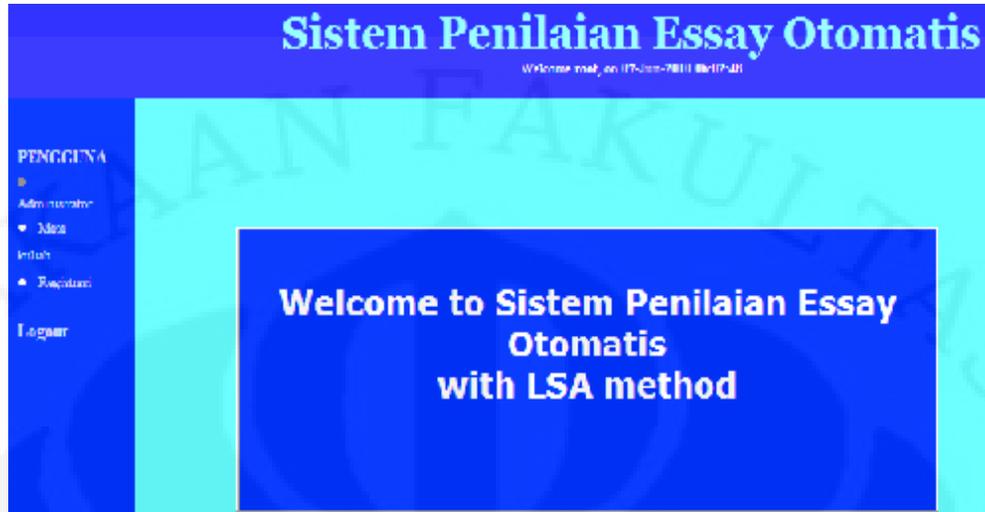
```

Gambar 3-3 Algoritma Login

Apabila user diidentifikasi sebagai Administrator atau root, maka menu yang tampil adalah seperti yang terlihat pada Gambar 3-4, dimana ada 2 buah pilihan yang bisa dipilih, yaitu :

1. Mata Kuliah,
2. Registrasi,

Pilihan mata kuliah berisikan data mata kuliah yang terdaftar pada SIMPLE-O, pada pilihan ini juga bisa menambahkan mata kuliah. Pada pilihan Registrasi, akan tampil user yang sudah terdaftar pada database, dan *privilege* nya apakah sebagai user atau dosen.

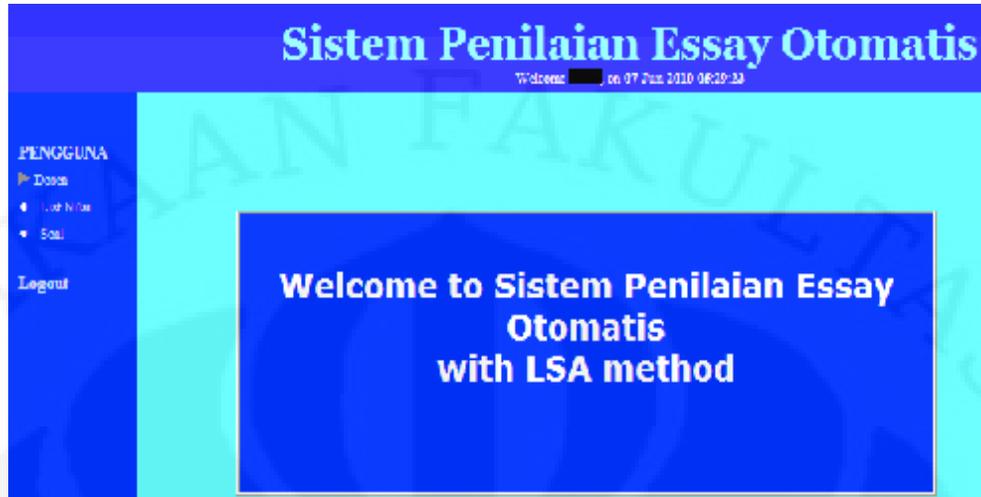


Gambar 3-4 Modul Administrator

Kemudian, ketika user diidentifikasi sebagai dosen maka halaman yang selanjutnya muncul adalah modul dosen yang terlihat pada Gambar 3-5 dimana pada modul ini terdapat 2 buah pilihan.

1. List nilai
2. Soal

List nilai berfungsi untuk melihat nilai dari mahasiswa yang mengambil ujian untuk mata kuliah tertentu. Algoritma program untuk list nilai terlihat pada Gambar 3-5. Pada *database*, dosen juga sudah didaftarkan mata kuliahnya, sehingga dosen bersangkutan hanya dapat menambah soal pada mata kuliah yang telah didaftarkan.



Gambar 3-5 Halaman Utama Dosen

```

Proc listnilai ()
  [load nilai_mhs untuk matkul dari database]
  i=0;
  while(score != EOF)
    i++;
    write ("idmk");
    write ("nama_mhs");
    write ("score");
  ewhile
Eproc

```

Gambar 3-6 Algoritma List Nilai

Pada perintah `while (score!= EOF)` berfungsi untuk melakukan *looping* untuk menampilkan nilai satu persatu nilai yang ada pada database. *Looping* akan terus dilakukan sampai sistem menemukan kata EOF pada file tersebut. EOF menunjukkan akhir dari suatu file (*end of file*).

Pilihan menu soal merupakan halaman untuk memasukan soal yang baru, mengedit, dan menghapus soal yang lama, serta dapat memilih kata bobot. Algoritma untuk soal dapat dilihat pada Gambar 3-4.

```

Proc soal ()
    [load mata kuliah untuk user dari database, dapat di delete, edit dan input soal
    mata kuliah tersebut]
    read (pil);

    if pil = Delete then      {bagian untuk menghapus soal}
    [hapus record dari database]
    else
    if pil = Edit then        {bagian untuk mengedit soal}
    [mengedit soal]
    if pil = Input Soal then  {bagian untuk menambah soal}
    [mengisi soal]
Eproc

```

Gambar 3-7 Algoritma Soal

Pada algoritma yang ditunjukkan pada Gambar 3-7, terlihat bahwa user akan dihadapkan pada pilihan menu, apakah user memilih salah satu dari option yang ada pada menu ini, apakah menghapus, mengedit, atau menginput soal baru.

Untuk meningkatkan kinerja dari algoritma ini, maka pada input soal dibuat pembobotan pada kata kunci tertentu. Kata kunci dipilih dua kali oleh dosen yang bersangkutan. Pilihan yang pertama, dosen memilih kata kunci yang biasa, dari cobaan yang pertama, didapatkan 10 – 20 kata kunci pada satu jawaban mahasiswa. Pilihan yang kedua, dipilih kata kunci berdasarkan pilihan dari tiga orang dosen yang ahli dibidangnya, dan dipilih juga kata bobot atau kata yang paling penting yang harus ada pada jawaban mahasiswa. Dari hasil pemilihan ini, kata kunci biasa dibobotkan dengan dikalikan dengan 1 sedangkan untuk kata bobot dikalikan dengan 2. Algoritma untuk memasukkan soal terlihat pada Gambar 3-8. Dan algoritma untuk pemilihan kata bobot terlihat pada Gambar 3-9.

```

if pil = Input Soal then
    {bagian untuk menambah soal}
    [input soal]
    [input jawaban]
    [input kata kunci]
    [input kata kunci bobot]
    [simpan soal dan kata kunci ke database]

```

Gambar 3-8 Algoritma Input Soal

```

if pil = Pilih Kata Bobot then
    {bagian untuk menambah soal}
    [pilih matkul]
    [pilih soal]
    [input kata kunci bobot]
    [bentuk matriks]
    [Proses SVD]
    [Simpan nilai frobenius/cos Alfa yang sesuai]

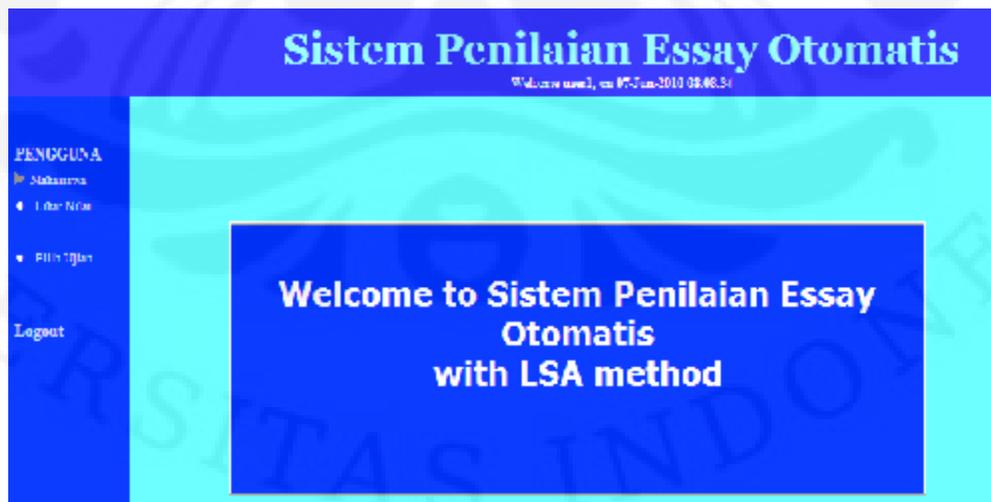
```

Gambar 3-9 Algoritma Memilih Kata Bobot

Kemudian, akan dibahas mengenai pembahasan halaman mahasiswa. Gambar tampilan pada login sebagai mahasiswa terlihat pada Gambar 3-12. Pada halaman mahasiswa ini, hanya terdapat dua pilihan, yaitu untuk melihat nilai dan memasukkan jawaban untuk ujiannya.

Pada pilihan melihat nilai, akan tampil halaman untuk melihat nilai saja hasil ujiannya. Algoritma yang digunakan untuk mengubah jawaban essay dari mahasiswa menjadi bentuk LSAny terlihat pada Gambar 3-13.

Kemudian, pada pilihan untuk memasukkan jawaban, mahasiswa dapat memilih mata kuliah yang akan diikuti ujiannya, dan setelah itu diminta untuk memasukkan jawabannya. Mahasiswa hanya diperbolehkan menjawab 1 kali untuk setiap pertanyaan. Algoritma untuk memasukkan jawaban dan mengkonversinya ke bentuk LSA terlihat pada Gambar 3-14.



Gambar 3-10 Login Mahasiswa

```

Proc ujian ()
  [load matkul untuk user dari database]
  while matkul != EOF
    [tampilkan mata kuliah yang dipilih]
  ewhile
    read (pil);
    if pil = Back then
      goto Halaman_Muka;
    else

      if pil = Lihat Soal then
        write ("Soal", soal[ ]);
        write ("Jawaban");
        read jawab_mhs[ ];
        if submit = true then
          [cek persamaan kata-kata jawaban mahasiswa]
          [bentuk matriks dari jawaban mahasiswa];
          [bandingkan matriks jawaban dengan referensi];
          [kirim nilai ke tabel database];

        else
          ();
      else
        ();
  Eproc

```

Gambar 3-11 Algoritma pengecekan kata dan Pembentukan Matriks

```

Proc ujian ()
  [load matkul untuk user dari database]
  i = 0;
  write ("Mata Kuliah");
  while matkul != EOF
    i++;
    write ["matkul"];
    write ("Lihat Soal");
    write ("Back");

  ewhile
  read (pil);
  if pil = Back then
    goto Halaman_Muka;
  else

  if pil = Lihat Soal then
    write ("Soal", soal[ ]);
    write ("Jawaban");
    read jawab_mhs[ ];
    if submit = true then
      [cek persamaan kata-kata jawaban mahasiswa]
      [pengecekan urutan SPO]
      [bentuk matriks dari jawaban mahasiswa dengan pembobotan
pada kata kunci bobot];
      [bandingkan matriks jawaban dengan referensi, dengan
membandingkan nilai normalisasi frobenius dan nilai kosinus alfa];
      [kirim nilai ke tabel database];
    else
      ();
  else
    ();
Eproc

```

Gambar 3-12 Algoritma Proses Pemasukkan Jawaban Ujian

3.3 Proses Pembobotan IDF-NDV

3.3.1 Algoritma Pembobotan IDF-NDV

Seperti yang telah diuraikan pada BAB II tentang pembobotan, untuk mendapatkan hasil yang lebih baik dan akurat, maka perlu diterapkan pembobotan untuk program SIMPLE-O yang telah dibangun sebelumnya. Sesuai penjelasan pada BAB II, pembobotan yang dipilih adalah pembobotan IDF-NDV yang memiliki rumus sebagai berikut :

$$n(t_i, d_j) = n(t_i, d_j) * \log \frac{N}{NF(t_j)}$$

$$n(t_i, d_j) = \frac{n(t_i, d_j)}{\sqrt{\sum_{i=1}^m (n(t_i, d_j))^2}}$$

Rumus 3.1 adalah rumus untuk pembobotan IDF. Pembobotan ini mengubah elemen setiap matriks Term Frequency ke bentuk Inversenya, dengan cara dikalikan dengan logaritma hasil pembagian jumlah dokumen dengan jumlah dokumen dimana terdapat kata kunci. Jika diterjemahkan dalam bentuk bahasa pemrograman (pseudocode), maka akan seperti yang terlihat pada Gambar 3-15.

```
function IDF()
  for (i=1, i<baris+1, i++)
    for (j=1, j<kolom+1, j++)
      nilai_log=log(N/banyak_term_dokumen[j])
      X[i,j]=X[i,j]*nilai_log
    endfor
  endfor
endfunction
```

Gambar 3-13 Algoritma IDF

Rumus 3.2 adalah rumus untuk pembobotan NDV. Pada pembobotan ini setiap elemen matriks TF dinormalisasi dengan cara membaginya dengan akar dari jumlah kuadrat semua elemen matriks. Tujuan dari normalisasi adalah untuk menghilangkan dominasi dari elemen yang bernilai besar yang akan berpengaruh

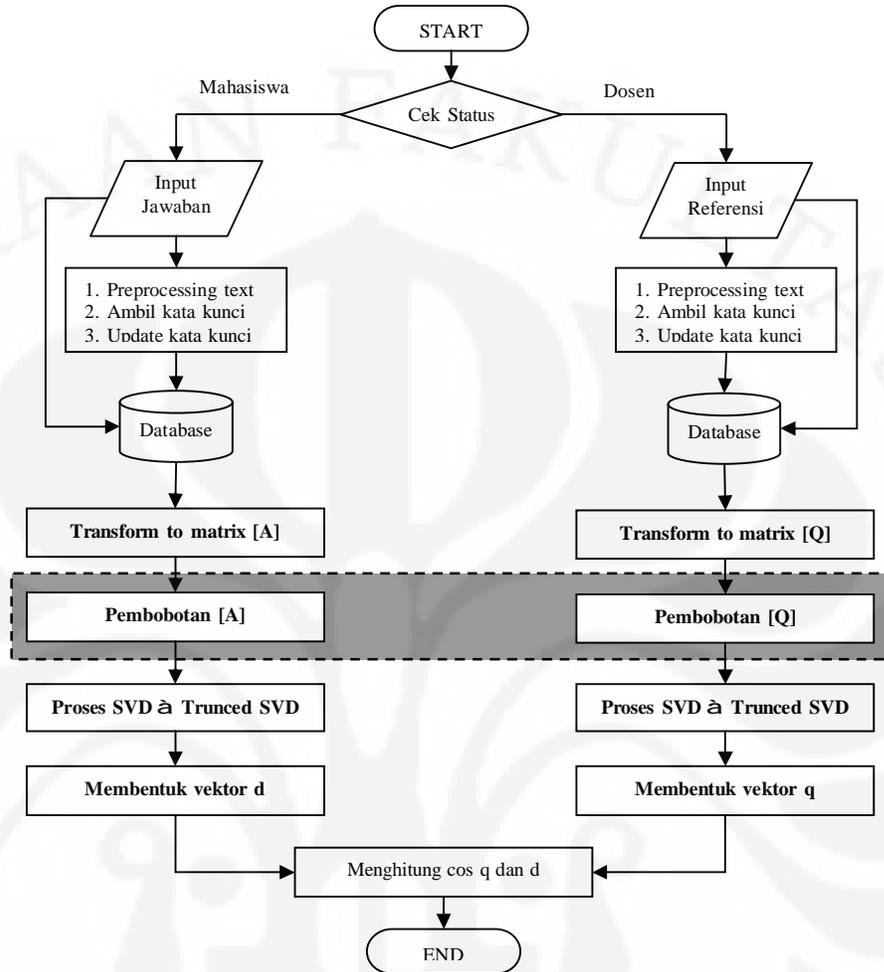
pada hasil SVDnya. Jika diterjemahkan dalam bentuk algoritmanya maka akan terlihat seperti Gambar 3-16.

```
function NDV()
  for (i=1,i<baris+1,i++)
    for (j=1,j<kolom+1,j++)
      jumlah=jumlah+X[i,j]^2
    endfor
  endfor
  for (i=1,i<baris+1,i++)
    for (j=1,j<kolom+1,j++)
      jumlah=sqrt(jumlah)
      X[i,j]=X[i,j]/jumlah
    endfor
  endfor
endfunction
```

Gambar 3-14 Algoritma NDV

3.3.2 Aplikasi Pembobotan IDF-NDF Pada Program Utama

Pemberian pembobotan pada program utama dilakukan setelah semua program telah selesai dibuat. Penyisipan dilakukan pada program dimana matriks telah terbentuk, baik pada jawaban dosen maupun jawaban mahasiswa. Matriks yang terbentuk kemudian ditransformasi ke bentuk IDF-NDVnya, baru kemudian diproses lebih lanjut. Flowchart dari pembobotan ini terlihat pada Gambar 3-17.



Gambar 3-15 Flowchart Penyisipan Pembobotan

Penyisipan program pembobotan ini dilakukan pada program utama dengan menambahkan sebuah fungsi objek khusus pembobotan. Program yang disisipkan berbentuk bahasa php yang dapat dilihat pada Gambar 3-18.

```

function idf_ndv(){
$term=0;
$term2=0;
$k=0;
for($i=1;$i<=$this->m;$i++)
{
for($j=1;$j<=$this->n;$j++)
{
if ($this->n<2)
$term=1;
else
{
for ($k=1;$k<=$this->n;$k++)
{
if ($this->A[$k][$j]!=0)
{
$term2++;
}
}
$term=$term2;
if ($term==0)
$term=1;
}
$stemp=$this->n/$term;
$nilai_log=log10($stemp);
$this->A[$i][$j]=$this->A[$i][$j]*$nilai_log;
}
}
for($i=1;$i<=$this->m;$i++)
{
for($j=1;$j<=$this->n;$j++)
{
$total=$total+($this->A[$i][$j]*$this->A[$i][$j]);
}
}
for($i=1;$i<=$this->m;$i++)
{
for($j=1;$j<=$this->n;$j++)
{
$total1=sqrt($total);
$this->A[$i][$j]=$this->A[$i][$j]/$total1;
}
}
return $this->A;
}

```

Gambar 3-16 Listing Program Pembobotan IDF-NDV Dalam PHP

Pada program tersebut terdapat prosedur untuk mencegah nilai pembagi bernilai 0, sehingga apabila nilai $NF(i)$ bernilai 0, maka angka 0 akan diubah menjadi 1. Nilai $NF(i)$ akan bernilai 0 saat dokumen yang diperiksa sangat kecil, dan tidak ada kata kunci yang muncul pada dokumen tersebut.

Program pembobotan IDF-NDV dibuat pada file `Matrix.php`, dimana pada file tersebut berisikan semua perhitungan dalam bentuk matriks, proses SVD juga terjadi pada file tersebut.

BAB IV

PENGUJIAN DAN ANALISIS APLIKASI SIMPLE-O DENGAN PEMBOBOTAN IDF-NDV

4.1 Pengujian Aplikasi

SIMPLE-O merupakan program yang dirancang dengan menggunakan program berbasis web yang diintegrasikan ke dalam server database, sehingga untuk menjalankannya dibutuhkan database My SQL dan PHP engine yang terinstall pada komputer.

Ujicoba dilakukan secara offline dengan cara mengambil jawaban-jawaban dari mahasiswa yang telah tersimpan pada database hasil ujian online sebelumnya. Ujian online yang diambil adalah ujian Pengantar Teknik Komputer pada tanggal 11 Mei 2010, dengan jumlah peserta yang diambil adalah 30 orang, dengan menggunakan program SIMPLE-O yang belum diberikan pembobotan. Setiap mahasiswa melakukan ujian secara bersamaan dengan menggunakan username yang telah dibuatkan sehingga setiap mahasiswa memiliki login yang berbeda. Halaman login terlihat pada Gambar 4-1.

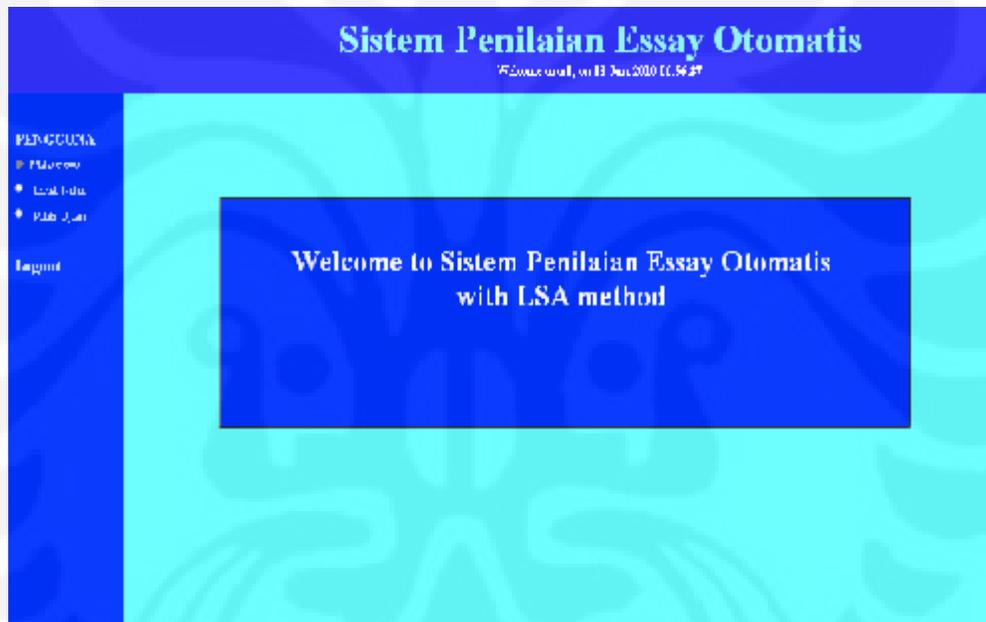


Login	
User:	<input type="text"/>
Password:	<input type="password"/>
<input type="button" value="Reset"/> <input type="button" value="Login"/>	

Gambar 4-1 Halaman Login

Sebelum mahasiswa melakukan ujian, terlebih dahulu dosen membuat pertanyaan beserta kunci jawabannya dan memilih kata kunci dan kata bobot. Kata kunci yang dipilih adalah kata kunci yang mengandung inti dari jawaban tersebut, dimana pada setiap jawaban mahasiswa yang nanti akan diperiksa adalah kata kuncinya. Kata bobot dipilih dari kata-kata kunci yang memiliki tingkat kepentingan dan bobot yang lebih dibandingkan kata kunci yang lainnya pada satu jawaban.

Setelah login, mahasiswa kemudian masuk ke halaman utama, dimana terdapat dua pilihan, yaitu melihat nilai dan atau memilih mata kuliah yang diujikan seperti yang terlihat pada Gambar 4-2.



Gambar 4-2 Halaman Utama Modul Mahasiswa

Ujian yang dipilih kali ini adalah ujian Pengantar Teknik Komputer, dimana ujian dan didalamnya terdapat 8 soal yang telah dipersiapkan seperti yang terlihat pada Gambar 4-3. Setiap mahasiswa dihadapkan pada pertanyaan dan hanya diperbolehkan menjawab satu kali saja.

No	Form Jawaban
1. Terdapat dua jawaban yang sama dan salah satu jawaban tersebut adalah jawaban yang benar.	Jawaban soal nomor 1...
2. Terdapat jawaban yang benar dan salah satu jawaban tersebut adalah jawaban yang benar.	Jawaban soal nomor 2...

Gambar 4-3 Form Isian Jawaban

Setelah mahasiswa mengisi seluruh jawaban dan mensubmit jawaban tersebut, maka secara otomatis seluruh jawaban akan diperiksa secara otomatis dengan menggunakan SIMPLE-O, dan hasilnya akan masuk ke dalam database

Selanjutnya adalah menguji jawaban-jawaban tersebut dengan menggunakan program yang telah dibobotkan. Proses memasukkan jawaban mahasiswa dilakukan secara manual dengan memasukkan jawaban-jawaban tersebut ke dalam form. Hasilnya kemudian dicatat.

Pemeriksaan juga dilakukan secara manual oleh dosen yang bersangkutan, yang kemudian disebut dengan *human raters*. Setelah semua data terkumpul, data-data tersebut akan dibandingkan dan dianalisis perbedaannya. Hasil pengujian dan analisis akan dibahas pada subbab selanjutnya.

4.2 Analisis Pengujian

Setelah dilakukan ujian online, penilaian dilakukan dengan tiga metode yang berbeda. Metode pertama adalah dengan menilai ujian dengan menggunakan program SIMPLE-O yang lama. Metode kedua adalah penilaian dengan menggunakan aplikasi SIMPLE-O yang telah dibobotkan dengan pembobotan IDF-NDV. Metode yang terakhir adalah penilaian secara manual.

4.2.1 Metode Penilaian Dengan SIMPLE-O Murni

Pengujian pertama dilakukan tanpa mengubah program awal dari SIMPLE-O. Perubahan yang dilakukan hanya perbaikan proses perhitungan SVD yang belum sempurna pada program sebelumnya. Hasil yang diperoleh terlihat pada Tabel 4-1.

Tabel 4-1 Perolehan Nilai Metode Pertama

No	Username	NPM	Nilai
1	user1	0906632530	42,61
2	user2	0906632316	49,15
3	user3	0906489006	50,50
4	user4	0906512406	33,75
5	user5	0906638484	45,27
6	user6	0906632562	32,84
7	user7	0906517161	43,29
8	user8	0906512665	47,76
9	user9	0906489826	36,31
10	user10	0906638452	47,50
11	user11	0906638401	36,77
12	user12	0906632436	28,35
13	user13	0906632726	40,99
14	user14	0906512753	40,89
15	user15	0906556244	44,43
16	user16	0906632474	60,20
17	user17	0906553324	55,58
18	user18	0906517325	49,37
19	user19	0906638250	33,08
20	user20	0906638345	55,61
21	user21	0906556181	46,38
22	user22	0906632745	64,35
23	user23	0906632606	54,33
24	user24	0906488943	55,25
25	user25	0906557511	51,02
26	user26	0906632354	49,61
27	user27	0906632341	40,02
28	user28	0906632713	63,97
29	user29	0906489800	44,73
30	user30	0906632663	38,03

Pada Tabel 4-1 terlihat bahwa nilai yang diperoleh paling besar adalah 64,35 dengan nomor NPM 0906632745. Jika kita buka detail dari nilai yang diperoleh terlihat pada Tabel 4-2. Sedangkan nilai yang paling kecil adalah 28,35 yang diperoleh oleh user12.

Tabel 4-2 Jawaban Mahasiswa Nilai Tertinggi

Soal	Jawaban	Nilai
1	simplex adalah mode transmisi satu arah dalam mode ini data hanya diperbolehkan mengalir satu arah saja contohnya sinyal dari stasiun TV	37,57
2	protokol artinya adalah aturan aturan yang ada dalam hal ini yang di maksud dengan internet protokol adalah satu set aturan untuk komunikasi dalam dan antara jaringan yang satu dengan jaringan lain internet protokol berfungsi untuk mengidentifikasi error menentukan bagian mana dari data yang akan dikirim dan mengkompres data	94,39
3	HUB adalah jamak dari sebuah repeater yaitu yang menghubungkan jaringan yang identik hub ini melayani sebagai pusat titik koneksi untuk LAN	48,04
4	UTP (unshielded twisted pair) adalah salah satu dari macam twisted pair yang mempunyai dua atau empat pasang kabel dan mempunyai jangkauan 100 m UTP memiliki efek pembatalan UTP adalah macam kabel yang umum digunakan untuk pengkabelan jaringan	55,05
5	LAN (local area network) adalah suatu grup komputer yang terinterkoneksi di bawah kontrol administrasi yang sama dapat berupa jaringan lokal kecil contohnya di perkantoran	62,22
6	Topologi fisik adalah layout fisik dari komponen pada suatu jaringan topologi fisik terdiri dari BUS RING STAR dan MESH sedangkan topologi logik menentukan bagaimana suatu host untuk berkomunikasi dalam jaringan	92,58
7	jaringan peer to peer adalah hubungan antar komputer (jaringan) yang masing masing komputernya mempunyai tanggung jawab yang sama tidak ada titik pusat kontrol pada jaringan selain tanpa administrasi jaringan peer to peer juga tidak memiliki sekuriti jaringan yang terpusat sehingga masing masing komputer harus menggunakan sekuriti yang terpisah untuk mengoreksi data selain itu jaringan peer to peer ini lebih kompleks dan susah bila ada komputer yang di tambahkan	85,37
8	Carrier Sense Multiple Access with Collision Detection atau sering disingkat menjadi CSMA/CD adalah sebuah metode media access control (MAC) yang digunakan oleh teknologi jaringan Ethernet Dengan metode ini sebuah node jaringan yang akan mengirim data ke node tujuan pertama-tama akan memastikan bahwa jaringan sedang tidak dipakai untuk transfer dari dan oleh node lainnya Jika pada tahap pengecekan ditemukan transmisi data lain dan terjadi tabrakan (collision) maka node tersebut diharuskan mengulang permohonan (request) pengiriman pada selang waktu berikutnya yang dilakukan secara acak (random) Dengan demikian maka jaringan efektif bisa digunakan secara bergantian	39,59

Perolehan nilai yang diperoleh sangat dipengaruhi oleh pemilihan kata bobot. Jika dibandingkan dengan kunci jawaban yang diberikan oleh dosen pada Tabel 4-3 dan Tabel 4-4, ada pada jawaban-jawaban dengan perolehan nilai yang tinggi yaitu pada soal nomor 2, dikarenakan banyak kata kunci dari jawaban dosen terdapat di jawaban tersebut. Makin banyak kata kunci yang muncul, nilai yang

diperoleh makin besar, dan ini berarti juga hasil dipengaruhi oleh panjang-pendeknya jawaban. Jika jawaban makin panjang artinya kata kunci yang dipilih makin banyak dan makin sering muncul, sehingga bobot kata kunci semakin berkurang. Jika salah satu kunci tidak ada atau sedikit jumlahnya maka tidak akan terlalu banyak berpengaruh pada nilainya.

Tabel 4-3 Kunci Jawaban

Soal	Jawaban
1	Transmisi Simplex atau disebut dengan transmisi satu arah adalah transmisi tunggal, transmisi yang dilakukan pada hanya satu arah saja
2	Internet protocol adalah satu set aturan atau rule untuk komunikasi dalam dan antara komputer di jaringan. Banyak protokol terdiri atas suite (atau group) protokol stack pada layer. Layer ini tergantung pada operasi dari layer yang lainnya dalam suit, supaya berfungsi dengan benar
3	Hub dapat mengembangkan cakupan sinyal dengan menerima kemudian meregenerasinya dan mengirim keluar ke port yang lain. Trafik akan dikirim keluar oleh seluruh port dari hub. Memungkinkan banyak collisions pada segmen jaringan dan sering bukan merupakan solusi yang baik. Juga disebut concentrators karena mereka melayani sebagai pusat titik koneksi untuk LAN.
4	Pada kabel UTP terdapat dua atau empat pasang kabel. Kabel jenis ini mempunyai efek pembatalan (cancellation effect) untuk pengurangan interferensi yang disebabkan oleh electromagnetic interface (EMI) dan radio frequency interference (RFI). Sangat umum untuk pengkabelan jaringan. Mempunyai jangkauan dari 328 ft (100 meter)
5	Suatu group / kumpulan komputer yang terinerkoneksi dibawah control administrasi yang sama. Dapat berupa jaringan lokal kecil yang di-instal di perkantoran. Dapat terdiri dari jaringan lokal yang terhubung yang terdiri dari beberapa ratus host, di-instal pada beberapa gedung dan lokasi. LAN ini mempunyai daerah cakupan yang sempit, paling jauh sekitar 1 km. Kampus termasuk LAN.
6	Topologi Fisik adalah layout fisik dari komponen yang ada di jaringan. Sedangkan Topologi logic menentukan bagaimana host mengakses ke medium untuk berkomunikasi lintas jaringan.
7	Jaringan peer to peer adalah jaringan komputer yang dapat berbagi files, mengirim pesan, mencetak pada shared printer. Pada jaringan tersebut, masing-masing komputer mempunyai kemampuan dan tanggung jawab yang sama. Setiap pengguna memutuskan data mana dan devais mana yang akan di-share. Pada jaringan ini tidak ada titik pusat kontrol pada jaringan. Jumlah anggota jaringan, paling baik bila terdiri dari 10 atau kurang dari 10 komputer.
8	Seluruh host atau komputer yang terhubung, mendengarkan jaringan saat ingin mengirimkan data. Bila komputer tidak mendeteksi tidak ada host yang mengirimkan data, maka host tersebut mencoba untuk mengirimkan data. Bila tidak ada host lain yang mengirimkan data, maka transmisi tersebut berhasil dan data akan sampai pada tujuan dengan sukses. Jika ada host atau komputer lain mengirimkan atau mentransmisikan data pada saat yang bersamaan, tabrakan atau collision akan terjadi pada media jaringan. Host atau komputer pertama yang mendeteksi tabrakan atau collision tersebut, maka akan mengirim sinyal yang menyatakan bahwa terjadi tabrakan atau collision dan memberitahukan ke seluruh host untuk menghentikan pengiriman dan untuk menjalankan algoritma backoff. Seluruh komputer yang tergabung pada jaringan tersebut akan menghentikan program dan mencoba kembali proses pengirimannya setelah periode waktu tertentu. (random)

Tabel 4-4 Kata Kunci dan Kata Bobot

No Soal	Kata Kunci	Kata Bobot
1	Transmisi, transmisi, satu, arah, tunggal,	Transmisi transmisi satu arah tunggal
2	set, aturan, rule, komunikasi, komputer, jaringan, protokol, suite, group, stack, layer, Layer, operasi, suit, berfungsi, benar,	komputer jaringan protokol group stack operasi berfungsi komunikasi suit
3	mengembangkan, cakupan, sinyal, menerima, meregenerasinya, mengirim, keluar, port, lain, Memungkinkan, banyak, collisions, segmen, jaringan, sering, solusi, concentrators, melayani, pusat, titik, koneksi, LAN,	mengembangkan cakupan sinyal menerima meregenerasinya mengirim keluar port lain Memungkinkan banyak collisions segmen jaringan sering solusi concentrators melayani pusat titik koneksi LAN
4	kabel, UTP, dua, empat, pasang, Kabel, mempunyai, efek, pembatalan, cancellation, effect, pengurangan, interferensi, electromagnetic, interface, EMI, radio, frequency, interference, RFI, umum, pengkabelan, jaringan, jangkauan, 328, ft, 100, meter,	kabel UTP dua empat pasang Kabel mempunyai efek pembatalan cancellation effect pengurangan interferensi electromagnetic interface EMI radio frequency interference RFI umum pengkabelan jaringan jangkauan 328 ft 100 meter
5	group, kumpulan, komputer, terinerkoneksi, dibawah, kontrol, administrasi, jaringan, lokal, kecil, di-instal, di, perkantoran, terhubung, beberapa, ratus, host, gedung, lokasi, daerah, cakupan, sempit, 1, km, Kampus,	group kumpulan komputer terinerkoneksi dibawah kontrol administrasi jaringan lokal kecil di-instal di perkantoran terhubung beberapa ratus host gedung lokasi daerah cakupan sempit 1 km Kampus
6	layout, fisik, komponen, jaringan, bagaimana, host, mengakses, medium, berkomunikasi, lintas,	layout fisik komponen jaringan bagaimana host mengakses medium berkomunikasi lintas
7	Jaringan, jaringan, komputer, yang, dapat, berbagi, files, mengirim, pesan, mencetak, shared, printer, kemampuan, dan, tanggung jawab, sama, pengguna, memutuskan, data, devais, di-share, tidak, ada, titik, pusat, kontrol, anggota, terdiri, dari, 10, atau, kurang,	Jaringan jaringan komputer yang dapat berbagi files mengirim pesan mencetak shared printer kemampuan dan tanggung jawab sama pengguna memutuskan data devais di-share tidak ada titik pusat kontrol anggota terdiri dari 10 atau kurang
8	host, komputer, terhubung, mendengarkan, jaringan, saat, mengirimkan, data, tidak, mendeteksi, mencoba, berhasil, akan, sampai, pada, tujuan, sukses, mentransmisikan, bersamaan, tabrakan, collision, terjadi, media, Host, mengirim, sinyal, menyatakan, memberitahukan, menghentikan, pengiriman, menjalankan, algoritma, backoff, tergabung, program, kembali, proses, pengirimannya, periode, waktu, tertentu, random,	host komputer terhubung mendengarkan jaringan saat mengirimkan data tidak mendeteksi mencoba berhasil akan sampai pada tujuan sukses mentransmisikan bersamaan tabrakan collision terjadi media Host mengirim sinyal menyatakan memberitahukan menghentikan pengiriman menjalankan algoritma backoff tergabung program kembali proses pengirimannya periode waktu tertentu random

4.2.2 Metode Penilaian Dengan SIMPLE-O IDF-NDV

Penilaian metode yang kedua adalah dengan menggunakan SIMPLE-O yang telah diberikan pembobotan IDF-NDV, dan pembobotan yang terdapat pada SIMPLE-O Murni dihilangkan, yaitu dengan menghilangkan proses kata bobot lebih. Perubahan yang dilakukan pada pembobotannya saja, baik di jawaban mahasiswa maupun di jawaban dari dosen. Metode ini menghasilkan nilai yang terlihat pada Tabel 4-5.

Pada Tabel 4-5 terlihat bahwa nilai yang paling besar yaitu 56,99 dan yang memperoleh masih sama, yaitu user 22. Sedangkan nilai terkecil adalah 26,21 yang diperoleh oleh user12 yang berarti sama dengan metode sebelumnya.

Jika dilihat pada nilai terkecil, jawabannya terlihat pada Tabel 4-6, dan kunci jawaban masih menggunakan kunci jawaban pada Tabel 4-3.

Tabel 4-5 Perolehan Nilai Metode Kedua

No	Username	NPM	Nilai
1	user1	0906632530	39,4
2	user2	0906632316	39,55
3	user3	0906489006	40,72
4	user4	0906512406	31,97
5	user5	0906638484	37,93
6	user6	0906632562	28,49
7	user7	0906517161	42,07
8	user8	0906512665	41,95
9	user9	0906489826	34,28
10	user10	0906638452	41,95
11	user11	0906638401	33,38
12	user12	0906632436	26,21
13	user13	0906632726	35,52
14	user14	0906512753	34,32
15	user15	0906556244	41,82
16	user16	0906632474	50,35
17	user17	0906553324	42,17
18	user18	0906517325	35,35
19	user19	0906638250	31,55
20	user20	0906638345	51,37
21	user21	0906556181	39,35
22	user22	0906632745	56,99
23	user23	0906632606	51,3
24	user24	0906488943	54,16
25	user25	0906557511	43,54
26	user26	0906632354	41,58

Table 4-5 Perolehan Nilai Metode Kedua (lanjutan)

No	Username	NPM	Nilai
27	user27	0906632341	34,15
28	user28	0906632713	56,23
29	user29	0906489800	36,86
30	user30	0906632663	32,87

Tabel 4-6 Jawaban Mahasiswa Nilai Terendah

Soal	Jawaban	Nilai
1	metode yang menggunakan metode transmisi berupa satu arah misalnya radio yang kita dengar	41,5
2	internet protocols merupakan suatu aturan yang ditetapkan bagi para pengguna internet	20,59
3	HUB merupakan suatu media transmisi untuk menghubungkan beberapa komputer ke beberapa komputer lainnya serta setiap komputer tersebut dapat saling berkomunikasi atau berinteraksi	0
4	UTP (Unshielded Twisted Pair) merupakan suatu media transmisi berupa kabel yang biasa di hubungkan dengan menggunakan RJ45	36,15
5	LAN (Local Area Network) merupakan suatu jaringan komputer dalam skala geografis kecil biasanya diterapkan atau ditemukan didalam suatu universitas	31,39
6	topologi fisik adalah topologi yang kita rancang atau kita gunakan sedangkan topologi logik merupakan suatu logik atau logika didalam topologi tersebut	28,57
7	jaringan peer to peer adalah suatu jaringan dimana ada dua komputer atau lebih yang saling berinteraksi satu sama lainnya secara langsung	50,76
8	CSMA atau CD bekerja berdasarkan ethernet network serta didalam suatu tranmisi data CSMA atau CD tersebut berperan sebagai penghubung antara tujuh lapisan layer (layer OSI)	10,69

Dari jawaban tersebut, ada nilai yang bernilai 0, yaitu pada jawaban nomor 3. Jika dibandingkan dengan kunci jawabannya, dan kata kuncinya, pada jawaban tersebut tidak ada kata kunci yang muncul, sehingga nilainya 0.

4.2.3 Metode Penilaian Manual (*Human Raters*)

Metode yang ketiga adalah metode penilaian manual, yang dilakukan oleh dosen. Penilaian secara manual menghasilkan nilai yang terlihat pada Tabel 4-7.

Untuk penilaian secara manual, nilai yang dibuat adalah berdasarkan penilaian seperti biasa yang sudah dilakukan untuk pemeriksaan jawaban essay atau isian. Penilaian ini bersifat subjektif, artinya setiap pemeriksa akan menghasilkan nilai yang belum tentu sama, dan sangat tergantung dari penilai.

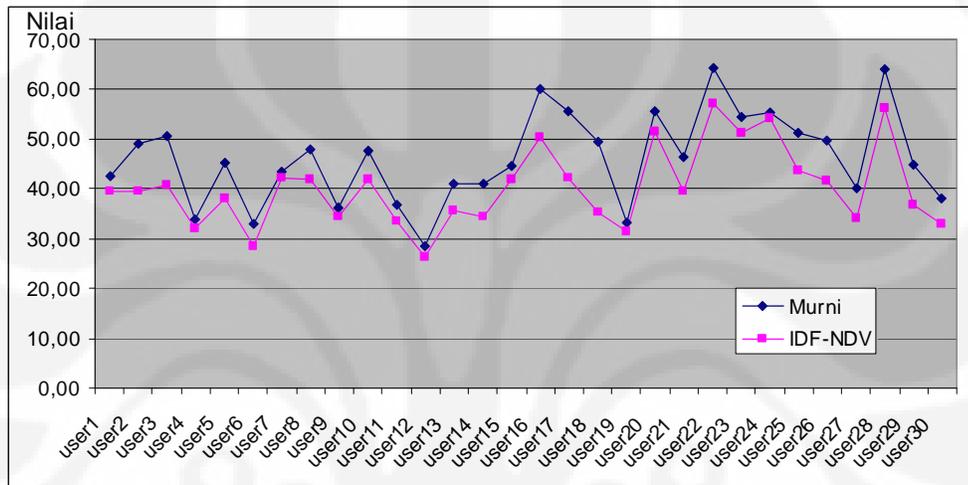
Penilai manusia biasanya menilai dari kemiripan dengan kunci jawaban dan dicocokkan dengan pengetahuan yang diketahui. Apabila pada jawaban yang diberikan memiliki kesamaan maksud ataupun kesamaan dengan kunci jawaban, meskipun kalimat-kalimat pada jawaban tersebut memiliki unsur kiasan, penilai manusia mampu menganalisis kemiripan jawaban tersebut, inilah yang tidak dimiliki oleh sistem penilaian otomatis SIMPLE-O.

Tabel 4-7 Penilaian Secara Manual

No	Username	NPM	Nilai
1	user1	0906632530	70
2	user2	0906632316	41
3	user3	0906489006	67
4	user4	0906512406	55
5	user5	0906638484	61
6	user6	0906632562	59
7	user7	0906517161	66
8	user8	0906512665	53
9	user9	0906489826	45
10	user10	0906638452	63
11	user11	0906638401	55
12	user12	0906632436	60
13	user13	0906632726	62
14	user14	0906512753	60
15	user15	0906556244	69
16	user16	0906632474	65
17	user17	0906553324	65
18	user18	0906517325	65
19	user19	0906638250	52
20	user20	0906638345	69
21	user21	0906556181	60
22	user22	0906632745	70
23	user23	0906632606	70
24	user24	0906488943	70
25	user25	0906557511	70
26	user26	0906632354	52,5
27	user27	0906632341	48
28	user28	0906632713	70
29	user29	0906489800	70
30	user30	0906632663	54

4.2.4 Analisis Perbandingan Metode SIMPLE-O Murni, Metode SIMPLE-O IDF-NDV, dan Metode Manual

Ketiga metode yang sudah diujikan, memiliki karakteristik dan hasil yang berbeda. Bila dilihat pada Gambar 4-4, untuk metode SIMPLE-O Murni, dihasilkan grafik nilai yang lebih tinggi dibandingkan dengan metode SIMPLE-O yang sudah dibobotkan. Selisih yang dihasilkan bervariasi, mulai dari 1,09 hingga 14,02, yang bisa dilihat pada Tabel 4-8.



Gambar 4-4 Grafik Perbandingan Metode Penilaian

Tabel 4-8 Selisih Nilai Kedua Metode

No	Username	NPM	Nilai-1	Nilai-2	Selisih
1	user1	0906632530	42,61	39,4	3,21
2	user2	0906632316	49,15	39,55	9,60
3	user3	0906489006	50,50	40,72	9,78
4	user4	0906512406	33,75	31,97	1,78
5	user5	0906638484	45,27	37,93	7,34
6	user6	0906632562	32,84	28,49	4,35
7	user7	0906517161	43,29	42,07	1,22
8	user8	0906512665	47,76	41,95	5,81
9	user9	0906489826	36,31	34,28	2,03
10	user10	0906638452	47,49	41,95	5,54
11	user11	0906638401	36,77	33,38	3,39
12	user12	0906632436	28,35	26,21	2,14
13	user13	0906632726	40,99	35,52	5,47
14	user14	0906512753	40,89	34,32	6,57
15	user15	0906556244	44,43	41,82	2,61
16	user16	0906632474	60,20	50,35	9,85
17	user17	0906553324	55,58	42,17	13,41
18	user18	0906517325	49,37	35,35	14,02

Tabel 4-8 Selisih Nilai Kedua Metode (Lanjutan)

No	Username	NPM	Nilai-1	Nilai-2	Selisih
19	user19	0906638250	33,08	31,55	1,53
20	user20	0906638345	55,61	51,37	4,24
21	user21	0906556181	46,38	39,35	7,03
22	user22	0906632745	64,35	56,99	7,36
23	user23	0906632606	54,33	51,3	3,03
24	user24	0906488943	55,25	54,16	1,09
25	user25	0906557511	51,02	43,54	7,48
26	user26	0906632354	49,61	41,58	8,03
27	user27	0906632341	40,02	34,15	5,87
28	user28	0906632713	63,97	56,23	7,74
29	user29	0906489800	44,73	36,86	7,87
30	user30	0906632663	38,03	32,87	5,16

Variasi selisih yang dihasilkan disebabkan oleh faktor banyak atau sedikitnya kata kunci yang ada pada jawaban tersebut. Variasi tertinggi adalah pada user18, dimana user 18 memiliki nilai dengan metode awal adalah 49,37. Variasi selisih yang dihasilkan disebabkan oleh faktor banyak atau sedikitnya kata kunci yang muncul pada jawaban tersebut. Pembobotan IDF yang diimplementasikan pada program ini membuat elemen-elemen matriks menjadi seimbang, dan mengurangi dominasi. Jawaban yang terdapat pada user18 banyak terdapat kata-kata kunci yang sangat dominan, dapat dilihat pada Tabel 4-9. Pada tabel tersebut terlihat kata-kata kunci disebutkan berulang-ulang, sehingga membuat ada dominasi di salah satu kata kunci. Pada program SIMPLE-O murni, semakin banyak kata kunci disebut, maka nilai akan semakin tinggi, namun, setelah dibobotkan, nilai yang diberikan tidak terlalu tinggi karena efek dominasi yang berkurang.

Tabel 4-9 Jawaban User 18

Soal	Jawaban
1	mode transmisi simplex adalah salah satu mode transmisi yang hanya mendukung koneksi satu arah sinyal tidak dapat dilakukan dengan arah yang berbeda baik di saat yang bersamaan maupun di saat yang berbeda contoh penggunaannya adalah pada sinyal televisi atau pemancar radio AM
2	IP (Internet Protocols) adalah deretan angka yang digunakan untuk menentukan posisi sebuah komputer dalam sebuah jaringan Nilai IP merupakan deretan angka 32 bit yang dibagi menjadi 4 masing masing 8 bit Kelas masing masing subnet berbeda tergantung pada penggunaan komputer dalam jaringan

Tabel 4-9 Jawaban User 18 (Lanjutan)

Soal	Jawaban
3	hub merupakan penghubung komputer dalam jaringan Keunggulan dari hub adalah setiap komputer langsung terkoneksi pada hub tanpa melalui perantara dan setiap komputer memiliki koneksi masing masing pada hub Eksekusi setiap komputer dilakukan di saat yang bersamaan
4	Unshield Twisted Pair (UTP) adalah kabel jenis kabel yang mendukung adanya interrupt bila terjadi electric interacy Biasa digunakan dalam jumlah dua atau empat
5	LAN (local area network) adalah jaringan komputer jarak pendek sebenarnya LAN terdiri atas 3 jenis jangkauan yaitu PAN (local area network) CAN (campus area network) dan LAN itu sendiri jangkauan LAN berkisar antara 200 meter sampai 50 kilometer Pada beberapa jaringan masih menggunakan kabel sebagai connector meskipun ada juga yang menggunakan gelombang elektromagnetik dengan alasan sumber daya biasa digunakan dalam wilayah kampus satu lokasi atau wilayah perkantoran
6	topologi fisik adalah topologi yang didasarkan pada letak komputer itu dalam jaringan Karena itu biasanya topologi fisik biasanya dilambangkan dengan penomoran masing masing komputer yang dapat berubah topologi logic adalah topologi yang didasarkan pada cara komputer masuk dalam jaringan Karena itu biasanya topologi logic dilambangkan dengan nomor Internet Protocol address tiap tiap komputer
7	koneksi dimana setiap host (komputer) dapat berfungsi baik sebagai client maupun sebagai server Setiap komputer dapat melakukan komunikasi tanpa melalui server Sistem jaringan ini mempunyai kelemahan yaitu sistem keamanan harus dilakukan oleh masing masing komputer dan sistem distribusi data tidak dapat diketahui secara pasti
8	CSMA CD merupakan cara pengaturan client server dimana setiap kegiatan client berhubungan dengan server Client mengirimkan sinyal pada server server mengolahnya lalu kemudian meneruskannya Metode ini sudah sangat jarang dipakai karena membutuhkan komputer dengan spesifikasi tinggi sebagai server

Kemudian, jika dilihat perbandingan antara hasil pengujian otomatis dibandingkan dengan penilaian manual, terlihat perbedaan yang cukup signifikan. Grafik perbandingan dapat dilihat pada Gambar 4-5.

Dari grafik terlihat bahwa nilai yang diberikan oleh *human raters* tidak memiliki sifat yang sama dengan hasil penilaian otomatis, jika dihitung persentase perbedaan dari ketiga metode ini adalah sebagai berikut:

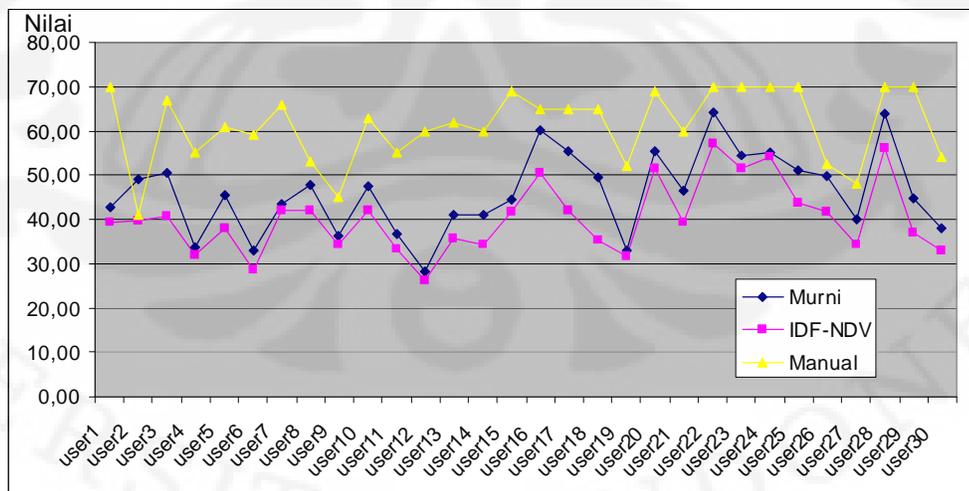
1. SIMPLE-O Murni ~ SIMPLE-O IDF-NDV = 11,78%
2. SIMPLE-O Murni ~ Manual = 24,75%
3. SIMPLE-O IDF-NDV ~ Manual = 34,25%

Dari hasil perbandingan ketiga metode, didapatkan bahwa untuk SIMPLE-O Murni dengan SIMPLE-O IDF-NDV memiliki perbedaan sebesar 11,78%, dan

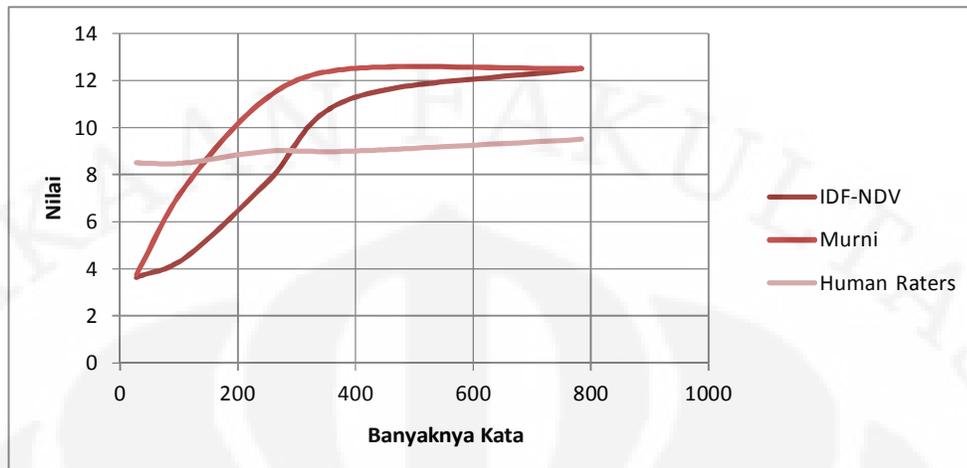
memiliki perbedaan dengan metode manual sebesar 34,25%, artinya setelah diberikan pembobotan IDF-NDV, tidak ada peningkatan untuk mendekati human raters, dan justru menjauhi. Hal ini disebabkan oleh metode pembobotan IDF-NDV adalah metode pembobotan yang bersifat menetralsisir sifat-sifat dominasi pada satu kata kunci, dan untuk itu hanya efektif pada essay-essay dengan jumlah kata yang banyak.

Pada essay dengan jumlah kata yang banyak, jumlah kata kunci yang muncul bersifat tidak merata, ada kata kunci yang sering muncul dan ada juga yang jarang muncul, hal ini diantisipasi oleh metode IDF untuk membuat kata kunci yang sering muncul tidak memiliki pengaruh besar terhadap hasil nilai yang akan diperoleh, karena banyaknya kata kunci tersebut yang muncul tidak mencerminkan jawaban essay yang semakin benar dan baik.

Pada essay dengan jumlah kata sedikit, kata kunci yang muncul juga sedikit dan lebih merata, sehingga teknik IDF akan justru mengurangi nilai. Namun, pada jawaban essay yang pendek, penilai manual akan melihat jawaban hanya pada kesamaan dengan kunci jawaban saja, tidak memperhatikan banyak-sedikitnya kata kunci yang muncul, sedangkan penilai otomatis akan melihat kata kunci yang muncul, bahkan mungkin pada jawaban essay yang pendek ada beberapa kata kunci yang tidak muncul sehingga nilai yang diperoleh kecil.



Gambar 4-5 Perbandingan Ketiga Metode



Gambar 4-6 Hubungan Antara Panjang Dokumen Dengan Perolehan Nilai

Pada Gambar 4-6 terlihat grafik hubungan antara panjangnya dokumen dengan perolehan nilai. Grafik tersebut merupakan grafik hasil ujicoba pada 1 soal saja, sehingga nilai maksimum yang dapat diperoleh untuk 1 soal tersebut adalah 12,5 yaitu didapat dari 100 dibagi dengan 8 soal. Ujicoba dilakukan dengan cara memasukkan salah satu jawaban mahasiswa di satu nomor ujian, dan untuk mendapatkan banyak kata yang bervariasi (dari 1-800 kata), jawaban tersebut dibuat berulang-ulang sehingga akan didapatkan kata kunci yang disebut berulang-ulang, dan beberapa kata kunci akan bernilai nol (karena jawaban tersebut bukan jawaban sempurna, sehingga ada beberapa kata kunci yang tidak muncul). Untuk kurva *human raters*, didapatkan dengan menilai jawaban secara manual. Hasilnya, pada grafik tersebut menunjukkan bahwa pada pembobotan murni yang ada pada SIMPLE-O memiliki kurva yang lebih tinggi dibandingkan dengan pembobotan IDF-NDV, dan seperti yang telah dijelaskan sebelumnya bahwa pembobotan IDF-NDV merupakan pembobotan terbalik, yang berfungsi untuk mencegah dominasi dari kata-kata kunci yang sering muncul, sehingga yang terjadi pada grafik adalah bahwa seringkali kata kunci yang muncul akan dikurangi efeknya, sehingga grafik terlihat lebih landai dibandingkan dengan pembobotan murni SIMPLE-O.

Dengan bentuk grafik seperti ini juga terlihat bahwa pembobotan IDF-NDV akan lebih baik digunakan untuk jawaban-jawaban yang relatif panjang, dibandingkan dengan jawaban yang pendek, karena pada jawaban yang panjang,

terdapat kata kunci yang berulang muncul, sedangkan jawaban yang pendek akan memiliki kata kunci yang sedikit muncul. Pada jawaban yang panjang, banyaknya kata kunci yang muncul belum tentu mencerminkan jawaban yang baik, sehingga nilainya pun tidak terlalu besar, namun, pada jawaban yang pendek, nilai kata-kata kunci kecil sehingga nilai yang dihasilkan pada pembobotan IDF-NDV juga akan lebih kecil dibandingkan dengan pembobotan murni pada SIMPLE-O, padahal kata kunci yang sedikit muncul mungkin bisa lebih mewakili jawaban yang benar dan bernilai lebih tinggi jika dinilai oleh *human raters*. Terlihat juga pada grafik bahwa selisih antara penilaian oleh *human raters* dengan pembobotan IDF-NDV lebih kecil untuk dokumen yang lebih besar.

Pembobotan murni SIMPLE-O merupakan pembobotan manual dimana dosen memilih kata bobot dari beberapa kata kunci secara manual, 2 dosen akan memilih kata bobot tersebut dan kata bobot akan memiliki bobot 2 kali dari kata kunci biasa. Pembobotan seperti ini berdampak pada nilai dari sebuah jawaban. Apabila terdapat kata bobot yang banyak pada jawaban tersebut, maka nilainya akan semakin besar meskipun pada jawaban tersebut tidak semua kata kunci muncul. Sedangkan pada pembobotan IDF-NDV, semua kata kunci bernilai sama, yang membedakan adalah ketika kata kunci tersebut jarang muncul, maka efeknya akan dikuatkan, sebaliknya jika kata kunci tersebut sering muncul, efeknya terhadap nilai dikurangi.

4.2.5 Pengaruh Pembobotan Awal SIMPLE-O Dan Pembobotan IDF-NDV Terhadap Waktu Proses

Pembobotan pada SIMPLE-O Murni merupakan pembobotan manual dimana pembobotan ini diaplikasikan dengan memilih kata bobot yang bernilai lebih, pembobotan ini memiliki waktu proses seperti yang terlihat pada Tabel 4-10.

Tabel 4-10 Proses Waktu SIMPLE-O Murni

Username	Waktu (detik)	Username	Waktu (detik)
user1	5,07758	user16	5,62009
user2	4,22495	user17	4,81067
user3	4,17805	user18	4,63533
user4	3,10441	user19	2,52094
user5	2,58849	user20	5,04886
user6	1,94377	user21	4,99224
user7	4,8512	user22	6,49134
user8	3,53558	user23	5,92229
user9	3,01208	user24	6,11568
user10	3,36488	user25	5,12725
user11	2,51903	user26	3,72496
user12	2,37255	user27	3,39943
user13	2,89222	user28	5,53244
user14	3,64956	user29	6,99791
user15	4,13401	user30	4,40797

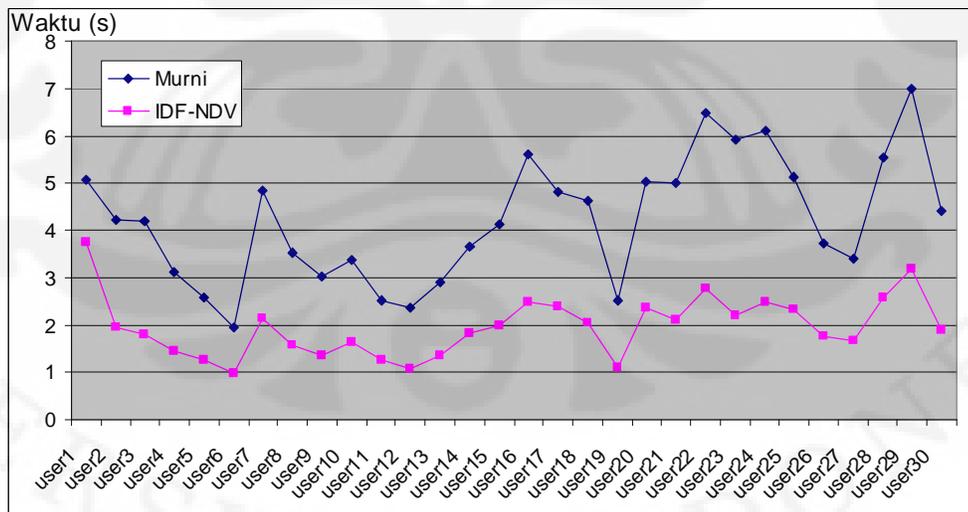
Rata-rata dari waktu proses tersebut adalah 4,22 detik, waktu ini diperoleh selama proses penghitungan mulai dari submit data ke server, pemrosesan matrix, hingga menghasilkan nilai. Makin besar matrix atau makin panjang jawaban, proses yang dibutuhkan makin lama.

Untuk proses waktu SIMPLE-O IDF-NDV, yang terlihat pada Tabel 4-11. Terlihat bahwa rata-rata prosesnya adalah 1,95 detik. Proses ini terbilang cukup cepat bila dibandingkan dengan proses SIMPLE-O Murni, dengan perbedaan sebesar 2,26 detik sehingga lebih cepat 2,16 kali. Dimana peningkatan didapatkan dari rata-rata waktu lama dibagi dengan rata-rata waktu baru.

Proses yang lebih cepat ini disebabkan karena pembobotan IDF-NDV merupakan pembobotan secara otomatis dan tidak membutuhkan banyak proses pada pengerjaannya.

Tabel 4-11 Proses Waktu SIMPLE-O IDF-NDV

Username	Waktu (Detik)	Username	Waktu (Detik)
User1	3,74074	user16	2,48962
User2	1,96099	user17	2,38955
User3	1,7953	user18	2,03243
User4	1,44153	user19	1,1137
User5	1,25295	user20	2,36303
User6	0,973417	user21	2,11701
User7	2,13696	user22	2,76071
User8	1,57818	user23	2,19672
User9	1,34834	user24	2,48195
user10	1,62633	user25	2,33993
user11	1,25227	user26	1,76484
user12	1,06472	user27	1,66316
user13	1,35839	user28	2,59297
user14	1,83952	user29	3,17713
user15	1,97659	user30	1,89121



Gambar 4-7 Perbandingan Waktu Proses

Pada Gambar 4-6 terlihat grafik perbandingan antara waktu yang dibutuhkan untuk proses pembobotan yang telah ada di program SIMPLE-O dan pembobotan IDF-NDV. Pada grafik terlihat secara garis besar waktu yang dibutuhkan untuk proses pembobotan IDF-NDV relatif lebih singkat dibandingkan dengan pembobotan yang telah ada pada SIMPLE-O.

Waktu proses pembobotan IDF-NDV lebih singkat dikarenakan pada pembobotan yang telah ada, proses perulangan untuk mengecek kata bobot lebih lama dibandingkan dengan pembobotan IDF-NDV yang menerapkan pembobotan pada semua kata kunci, tanpa memeriksa kembali kata bobotnya.

BAB V

KESIMPULAN

Pada skripsi ini, beberapa kesimpulan yang dapat diambil dari hasil percobaan adalah sebagai berikut:

1. LSA dapat dikembangkan dengan metode pembobotan IDF-NDV yang telah berhasil diimplementasikan pada SIMPLE-O.
2. Pembobotan IDF-NDV yang diimplementasikan pada SIMPLE-O memiliki kesamaan dengan metode lainnya sebagai berikut:
 - a. SIMPLE-O Murni ~ SIMPLE-O IDF-NDV = 11,78%
 - b. SIMPLE-O Murni ~ Manual = 24,75%
 - c. SIMPLE-O IDF-NDV ~ Manual = 34,25%
3. Pembobotan IDF-NDV kurang sesuai digunakan untuk jawaban-jawaban singkat.
4. Waktu proses SIMPLE-O IDF-NDV secara rata-rata 2,16 kali lebih cepat bila dibandingkan dengan waktu proses SIMPLE-O Murni.

DAFTAR ACUAN

- [1] Anak Agung Putri Ratna, Bagio Budiarto, Ade W. Astato, dan Djoko Hartanto.(2007). SIMPLE-O:Web Based Automated Essay Grading System Using Latent Semantic Analysis Method for Indonesian Language Considering weight word, and word synonym. The 10th International Conference on Quality and Research Faculty of Engineering University of Indonesia, 4-6 December. Depok, Indonesia.
- [2] Erica Chisholm & Tamara G. Kolda. 1999. New Term Weighting Formulas for the Vector Space Method in Information Retrieval. US : Oak Ridge National Laboratory.
- [3] Laurence Rudner & Phill Gagne. 2001. An overview of three approaches to scoring written essays by computer. Practical assessment, Research, and Evaluation, A-peer reviewed electronic journal, from:
[http://ericae_met/pare/getvn.asp? v=7&n=26](http://ericae.met.pare.getvn.asp?v=7&n=26)
- [4] Launder, T. K., Foltz, P. W., Laham, D. 1998. An Introduction to Latent Semantic Analysis. Discourse Processes, 25, 259-284.
- [5] Palmer, J., Williams, R., Dreher, H. 2002. Automated Essay Grading System Applied to a First Year University Subject – How Can We do it Better?”, InSITE – “Where Parallels Intersect”. Pp 1221-1229, Informing Science, Perth, Australia.
- [6] Sudarsun, Venkatesh Prabhu, and Satish Kumar. 2010. Role of Weighting on TDM in Improving Performance of LSA on Text Data. Nanyang Technological University from IEEE Explore. Singapore.