

SIMPLE: Sistem Penilaian Esei Otomatis Berbasis WEB Dengan Metoda Latent Semantic Analysis yang Digunakan Pada Bahasa Indonesia Dengan Penambahan Kata Bobot

Anak Agung Putri Ratna, M Salman, Bagio Budiardjo, Djoko Hartando dan Seinosuke Narita

Departemen Elektro, Fakultas Teknik, Universitas Indonesia Kampus Baru UI Depok, 16424, Indonesia
ratna@eng.ui.ac.id, salman@eng.ui.ac.id, bbudi@eng.ui.ac.id, djoko@eng.ui.ac.id
Faculty of Computer Science, Waseda University
narita@waseda.ac.jp

Abstrak

Penilaian adalah suatu proses untuk mengambil keputusan dengan menggunakan informasi yang diperoleh melalui pengukuran hasil belajar baik yang menggunakan instrumen tes maupun yang tidak [1]. Penilaian dengan esei (essay grading) menjadi pilihan pengajar dalam mengevaluasi tingkat kemampuan dari siswanya walaupun kenyataannya tidak mudah memberikan penilaian yang objektif pada setiap siswa dalam mengevaluasi jawaban dari soal esei. Bentuk esei ini oleh banyak peneliti dianggap alat yang sangat ampuh untuk mengukur hasil pembelajaran, begitu juga untuk mengamati kemahiran berpikir tingkat tinggi seperti sintesis dan analisis. [2]. Pada penelitian ini dibuat suatu sistem penilaian esei otomatis yang disebut SIMPLE (SisteM Penilaian Esei otomatis) dengan menggunakan metode Latent Semantic Analysis (LSA) yang merupakan salah satu metode untuk mengekstrak dan merepresentasikan kalimat dengan perhitungan matematis atau statistik dari teks dengan jumlah yang besar [3]. Perhitungan matematis dilakukan dengan memetakan ada atau tidak adanya kata dari kelompok kata pada matriks. Lebih lanjut penelitian ini dilakukan dengan memasukkan fitur pembobotan kata pada penilaian esei otomatis berbasis web dengan bahasa Indonesia. Ujicoba dilakukan dengan membandingkan hasil yang diperoleh dari sistem dengan tidak menggunakan kata bobot dan sistem dengan menggunakan kata bobot. Ujicoba berhasil mencapai 82.56 – 96.42 % agreement dengan human raters untuk sistem dengan menggunakan kata bobot.

Keywords : SIMPLE, Penilaian Esei Otomatis berbasis WEB, Latent Semantic Analysis, Singular Value Decomposition, eigenvalue, eigenvector dan kesesuaian dengan human raters.

Abstract

Grading is a process for decision making using information from evaluation of learning result whether using a test instrument or not[1]. Grading with essay is an option to evaluate level of knowledge of the students, but essay grading is not giving an objective view to each student. Essay grading by many of researcher is considered a good tools to evaluate result of a learning process and so, to evaluate level of intuition like synthesis and analysis. [2]. This research is intended to create an automatic essay grading which is called SIMPLE (SisteM Penilaian Esei otomatis) using Latent Semantic Analysis (LSA) as one of the method to extract and represent sentence using mathematical calculation or statistic from large amount of text [3]. Mathematical calculation is done by mapping with or without word from matrix group of word. Furthermore, this research is done by implementing weight feature on web based automatic essay grading using Indonesian language. Testing is done by comparing result from system that using weight word and system that not using weight word. Testing has succeeded with 82.56 -96.42 percentage agreement with human raters for system using weight word.

Keywords: SIMPLE, web based automatic essay grading, Latent Semantic Analysis, Singular value decomposition, eigenvalue, eigenvector and agreement with human raters.

1. Pendahuluan

Seiring dengan berkembangnya teknologi telekomunikasi dan komputer, maka pada dunia pendidikan pun terjadi perubahan sistem pengajaran yang cukup signifikan. E-learning merupakan suatu sistem dimana penyampaian materi pembelajaran, pelatihan atau program pendidikan menggunakan peralatan elektronik, termasuk di dalamnya adalah penggunaan komputer, atau peralatan-peralatan elektronik (seperti telepon mobile) dalam tujuan memberikan materi pelatihan, pendidikan atau pembelajaran [4]. E-learning ini berkembang meluas di lingkungan pendidikan rendah dan menengah serta di lingkungan pendidikan perguruan tinggi.

Pada konsep *e-learning*, ujian dapat dilakukan secara *online*, mulai dari menjawab soal ujian, hingga pemberian penilaian. Hal ini memberikan keuntungan bagi pengajar dan siswa karena pelaksanaan ujian dapat menjadi lebih efektif dan efisien. Keuntungan lainnya adalah bahwa sistem evaluasi dengan bantuan komputer dapat memberi penilaian yang lebih cepat dan akurat [2,5]. Selain itu sistem ini dapat menangani kelas dengan jumlah siswa yang besar.

Bentuk evaluasi/soal ujian baik dengan pembelajaran konvensional maupun dengan *e-learning* berdasarkan bentuknya dapat dibedakan menjadi dua bentuk utama [1], yaitu soal test bentuk objektif dan soal tes bentuk *essay*. Soal test bentuk objektif merupakan soal dengan pilihan jawaban yang sudah disiapkan. Contoh produk komersial yang telah digunakan secara luas untuk *non-essay* adalah seperti WebCT [6] dan Cisco *online Assessment Sistem*. Bentuk *online assessment* yang pernah dikembangkan di Departemen Elektro Fakultas Teknik Universitas Indonesia juga berbasis *non-essay*, seperti *Yes/No, Multiple Choice, matching questions*, serta mengisi satu kata pada bagian yang kosong. [7 - 10].

Soal esei merupakan bentuk evaluasi dimana pilihan jawaban tidak disediakan, dan siswa harus menjawab

dengan kalimat, sehingga jawaban dapat sangat bervariasi sesuai dengan pemikiran masing-masing peserta ujian.

Sampai saat ini sudah banyak sistem penilaian esei yang diperkenalkan, seperti PEG (*Project Essay Grading*) [3, 5], *E-rater* [3, 5, 11, 12], *Bayesian Scoring System* [2, 3, 5, 13] dan IEA (*Intelligent Essay Assessor*) [3, 5, 13-17].

Penerapan Penilaian Esei Otomatis yang telah disebutkan dalam hal ini telah dilakukan dengan basis bahasa Inggris. Bahasa basis yang digunakan disini sangat mempengaruhi proses dan hasil yang dilakukan pada proses penilaian terhadap jawaban siswa karena karakteristik dari bahasa yang berbeda.

Salah satu metode yang dapat digunakan untuk menilai esei secara otomatis adalah metode yang disebut dengan *Latent Semantic Analysis*. Metode ini adalah metode yang digunakan pada IEA (*intelligent essay assessor*) yang mempunyai kesesuaian (*agreement*) dengan *human raters* 85% - 91%, merupakan metode yang mengekstrak dan merepresentasikan kalimat dengan perhitungan matematis atau statistik pada text dengan jumlah yang besar [3]. Perhitungan matematis dilakukan dengan memetakan ada atau tidak adanya kata dari kelompok kata pada matriks.

Pada penelitian ini akan dibuat sistem yang diberi nama SIMPLE dengan menggunakan metode LSA berbasis bahasa Indonesia dan menambah pembobotan pada kata-kata yang dianggap penting dari kata kunci yang dipilih. Nilai bobot diberikan dengan mengalikan nilai tersebut dengan bobot dua.

2. Teknik Latent Semantic Analysis

Sistem penilaian esei otomatis dengan metode LSA, menerapkan teknik analisis faktor pada matriks, yaitu teknik aljabar linier *Singular Value Decomposition* (SVD). SVD adalah salah satu metode untuk memecahkan masalah-masalah

matematik linier [18]. Metode SVD berdasarkan pada teori aljabar linier yang menyatakan bahwa suatu matriks A yang berukuran $m \times n$ mempunyai nilai singular yang merupakan akar pangkat dua eigenvalue $A^T A$.

Pada SVD, matriks akan didekomposisi menjadi tiga komponen matriks. Komponen matriks pertama mendeskripsikan entitas baris sebagai vektor orthogonal matriks. Komponen matriks kedua mendeskripsikan matriks diagonal yang berisi nilai skalar dan yang ketiga adalah matriks entitas kolom sebagai vektor orthogonal matriks.

Setelah memperoleh 3 matriks dari proses SVD, proses berikutnya yang perlu dilakukan adalah mereduksi dimensi dari matriks dengan cara mengurangi dimensi dari matriks kedua yang berupa matriks diagonal. Pengurangan dimensi matriks diagonal ini dilakukan dengan cara mengeset semua nilai diagonal matriks ke dua menjadi nol kecuali diagonal dimensi yang dipilih. Dan jika ketiga komponen matriks tersebut dikalikan maka menghasilkan matriks rekonstruksi yang lain dengan tujuan untuk nilai korelasi yang diinginkan.

Untuk sembarang matriks $A_{m \times n}$ dimana $m \geq n$, matriks tersebut dapat difaktorisasi seperti pada persamaan (1).

$$A_{m \times n} = U_{m \times n} \cdot S_{n \times n} \cdot V_{n \times n}^T \dots \dots \dots (1)$$

U : matriks orthogonal berukuran $m \times n$,
 S : matriks diagonal berukuran $n \times n$,
 dengan elemen matriks positif atau nol, dan
 V : matriks orthogonal berukuran $n \times n$.

Matriks U merupakan matriks orthogonal yang kolom-kolomnya terdiri atas n *eigenvector* yang dinormalkan (*normalized eigenvector*) dari n *eigenvalue* matriks AA^T . Matriks S merupakan matriks diagonal yang elemen-elemennya adalah nilai singular matriks A . Matriks V merupakan matriks orthogonal yang kolom-kolomnya terdiri atas n *eigenvector* yang dinormalisasikan (*normalized eigenvector*) dari n *eigenvalue* matriks $A^T A$. Pembentukan matriks A sebagai perkalian matriks disebut sebagai metode SVD.

Apabila matriks A merupakan matriks bujursangkar, misalnya berukuran $n \times n$; maka matriks U , S , dan V juga akan merupakan matriks bujursangkar berukuran $n \times n$.

Untuk memperoleh *inverse* matriks A dapat digunakan persamaan (2):

$$A^{-1}_{n \times n} = \left. \begin{aligned} & (U_{n \times n} \cdot S_{n \times n} \cdot V_{n \times n}^T)^{-1} \\ & = (V_{n \times n}^T)^{-1} \cdot S^{-1}_{n \times n} \cdot U^{-1} \end{aligned} \right\} (2)$$

karena matriks U dan V merupakan matriks orthogonal, maka berlaku persamaan (3):

$$U^T U = V^T V = I \quad (3)$$

sehingga persamaannya menjadi seperti persamaan (4).

$$U^T = U^{-1} \text{ dan } V^T = V^{-1} \quad (4)$$

Dengan demikian *inverse* matriks A akan diperoleh seperti persamaan (5).

$$A^{-1} = V \cdot S^{-1} \cdot U^T \quad (5)$$

Langkah-langkah metode SVD dalam pembentukan matriks komponen U , S , dan V dari matriks A , dapat diuraikan sebagai berikut:

- 1) hitung matriks $A^T A$,
- 2) hitung *eigenvalue* dari matriks $A^T A$,
- 3) hitung *eigenvector* dari matriks $A^T A$,
- 4) membentuk komponen V dengan menormalisasi *eigenvector* matriks $A^T A$,
- 5) membentuk diagonal S dengan menghitung nilai singular matriks A dari akar kuadrat *eigenvalue* matriks $A^T A$,
- 6) hitung matriks AA^T ,
- 7) hitung *eigenvalue* dari matriks AA^T ,
- 8) hitung *eigenvector* dari matriks AA^T ,
- 9) membentuk matriks komponen U dengan menormalisasi *eigenvector* matriks AA^T .

Eigenvalue dan Eigenvector

Jika dinyatakan suatu matriks kuadrat A dengan dimensi $n \times n$, mempunyai persamaan vektor seperti Persamaan (6)

$$Ax = \lambda x \Rightarrow (A - \lambda I)x = 0 \rightarrow x \neq 0 \quad (6)$$

dimana λ adalah suatu nilai skalar dan akan dicari semua nilai λ dan vektor tidak nol

dari x dari solusi persamaan (6), maka semua nilai λ yang memenuhi persamaan (6) disebut *eigenvalue* dari matriks A . sedangkan *eigenvector* merupakan *eigenvalue* matriks A yang berkorespondensi dengan vektor x yang bersesuaian. Beberapa hal yang berkaitan dengan *eigenvalue* dan *eigenvector* diantaranya adalah; *eigenvalue* dan *eigenvector* hanya didefinisikan oleh matriks kuadrat, solusi mencari x yang tidak nol, *eigenvalue* bisa memiliki harga kompleks untuk matriks bernilai riil.

Tiga langkah untuk menentukan *eigenvalue* dan *eigenvector* diuraikan di bawah ini:

- 1) Tentukan determinan matriks A adalah 0, seperti ditunjukkan pada persamaan (7).

$$\det [A - \lambda I] = 0 \quad (7)$$

- 2) Tentukan *eigenvalue* dengan mencari semua n akar persamaan (7) dan beri notasi $\lambda_1, \lambda_2, \dots, \lambda_n$.

- 3) Tentukan *eigenvector* dengan mencari x_1, x_2, \dots, x_n yang berkorespondensi dengan Persamaan (8).

$$(A - \lambda_i I) x_i = 0, \quad i = 1, 2, \dots, n \quad (8)$$

Normalisasi Matriks

Normalisasi *Frobenius* [19] dari suatu matriks adalah sama dengan normalisasi dari matriks diagonalnya, seperti yang ditunjukkan pada persamaan (9).

$$\|A\|_F \equiv \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2} \quad (9)$$

Eigenvector dapat dinormalisasi dengan membagi ke bentuk normalnya masing-masing. Jika x_1, x_2, \dots, x_n adalah *eigenvector* dari matriks U , maka normalisasinya seperti Persamaan (10).

$$e_1 = \frac{x_1}{|x_1|}, \quad e_2 = \frac{x_2}{|x_2|}, \quad \dots, \quad e_n = \frac{x_n}{|x_n|} \Rightarrow |e_i| = 1 \quad (10)$$

bentuk normal vektornya didefinisikan seperti Persamaan (11).

$$|x| = \sqrt{x^T x} = \sqrt{\begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}} = \sqrt{|x_1|^2 + |x_2|^2 + \dots + |x_n|^2} \quad (11)$$

sehingga normalisasi *eigenvector* matriks U seperti Persamaan (12).

$$U = [e_1 \ e_2 \ \dots \ e_n] \quad (12)$$

3. Algoritma SIMPLE

SIMPLE terdiri dari beberapa modul. Modul-modul tersebut adalah sebagai berikut[20]:

1. Modul login
2. Modul dosen
3. Modul mahasiswa

Pada penelitian ini telah dibuat algoritma SIMPLE berbasis web. Proses yang dilakukan adalah, pertama memilih kata-kata kuncinya. Bila ada kata-kata kunci yang dipilih sebagai kata bobot, maka diberi bobot dua kali kata kunci yang biasa. Setelah itu kata kunci serta kata bobot dipetakan dalam matriks dan kemudian dilakukan proses SVD, sehingga terbentuk 3 matriks, yaitu 2 matriks orthogonal dan 1 matriks diagonal.

Setelah mengadakan penyederhanaan pada matriks diagonalnya, dicari normalisasi Frobenius untuk masing-masing matriks, baik matriks dari jawaban referensi maupun matriks jawaban siswa. Nilai siswa didapatkan dengan membandingkan nilai normalisasi matriks jawaban siswa dengan nilai normalisasi matriks jawaban referensi. Algoritma proses ujian mahasiswa dengan memasukkan unsur pembobotan kata ditunjukkan pada Gambar 1.

Gambar 2 merupakan *activity diagram* untuk pembobotan kata serta konversi matriks dari jawaban referensi. Dengan ini menunjukkan bagaimana proses konversi ke matriks dibentuk, termasuk proses pembobotan. Kalimat jawaban referensi yang dimasukkan kemudian dipilih kata kunci serta kata bobotnya. Kemudian dicari kata kunci dan kata bobot dari setiap kalimat pada jawaban untuk membentuk kolom matriks. Untuk kata kunci diberi nilai 1 untuk setiap pemunculan dan untuk kata bobot akan dikalikan dengan nilai bobot dua. Proses ini dilakukan sampai semua kalimat telah diperiksa dan dikonversi ke dalam kolom matriks.

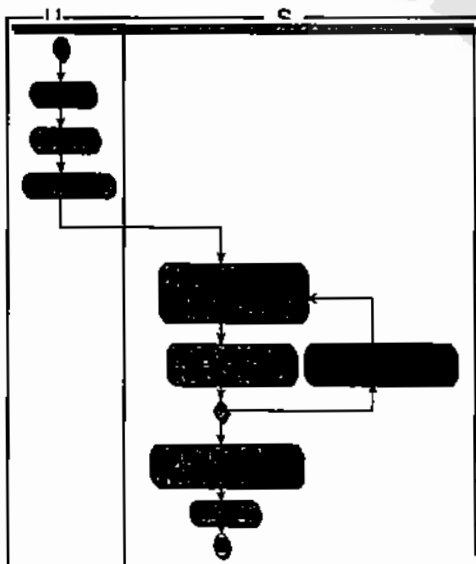
```

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
[bentuk matriks dari jawaban mahasiswa dengan pembobotan pada kata kunci bobot];
[bandingkan matriks jawaban dengan referensi, dengan membandingkan nilai normalisasi frobenius];
[kirim nilai ke tabel database];
else
    ();
else
    ();

Eproc
    
```

Gambar 1. Algoritma Proses Mengikuti Ujian Oleh Mahasiswa Dengan Memasukkan Unsur Pembobotan Kata



Gambar 2. Activity Diagram Pembobotan Kata dan Konversi Matriks (U = user, S = sistem)

4. Uji Coba

Perangkat keras yang digunakan untuk keperluan pengembangan sistem digunakan sebuah komputer dengan spesifikasi Pentium M Centrino sebagai server. Untuk kliennya ada beberapa komputer yang tergabung dalam jaringan LAN Departemen Elektro FTUI, mengakses server dengan Wireless LAN.

Sedangkan untuk perangkat lunak, pada server digunakan Apache HTTP Server 2.0.49, MySQL Server Clients 4.0.20d, yang berfungsi sebagai pusat pemrosesan database dan Matlab 5.3, untuk melakukan perhitungan matematis aljabar. Untuk pemrograman digunakan bahasa pemrograman PHP [21, 22] Pada bagian client digunakan Web browser dalam hal ini Microsoft Internet Explorer, walaupun bisa digunakan beberapa browser yang lain seperti Opera, Mozilla, Netscape.

Pengambilan data dilakukan dengan memilih 8 soal yang diambil dari mata kuliah Infrastruktur Informasi. Soal dan jawaban diambil dari buku Tanenbaum edisi ke-4 [23]. Pada Ujicoba SIMPLE dilakukan dengan menentukan 2 tahapan kata kunci. Tahapan pertama adalah kata kunci pertama yang dalam hal ini mengambil 20 kata kunci. Kemudian diambil lagi kata kunci kedua, yang disebut dengan kata bobot. Dengan kata bobot ini dilakukan 4 kali pengambilan data. Ujicoba pertama adalah dengan mengambil kata kunci bobot dengan jumlah 5 kata kunci bobot, yang kedua dengan 6 kata kunci bobot, yang ketiga dengan 7 kata kunci bobot dan yang keempat dengan 8 kunci bobot.

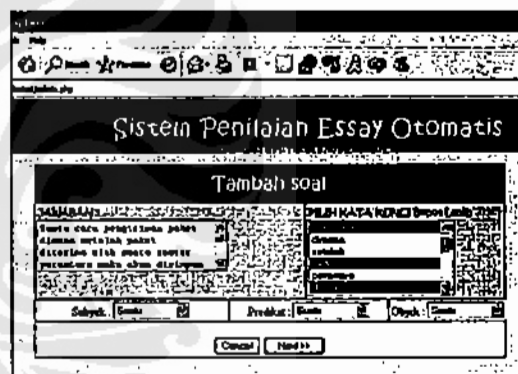
Pada percobaan pertama, dilakukan dengan menjawab salah satu soal. Disini dipilih kata kunci biasa sejumlah 20 kata dan kata kunci bobot 5 – 8 kata.

Gambar 3, 4, 5, 6 dan 7 menunjukkan tampilan pada sistem SIMPLE. Gambar 3 adalah gambar yang menunjukkan jendela login, Gambar 4 menunjukkan jendela untuk menambah soal. Gambar 5 adalah gambar yang menunjukkan bagaimana pengajar memilih kata kunci dan kata bobot. Proses pemilihan kata kunci dilakukan setelah soal dimasukkan ke sistem.

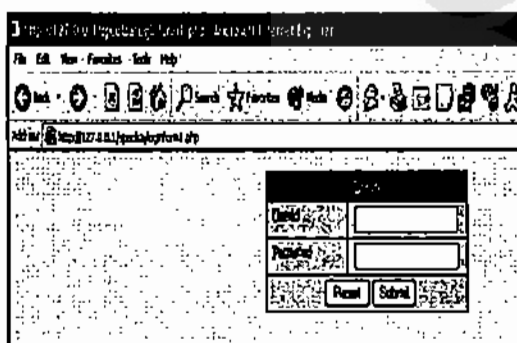
menunjukkan jendela penilaian untuk pengajar. Pengajar tersebut dapat melihat semua nilai dari siswa yang mengikuti ujian mata pelajaran tersebut.



Gambar 4.
Menambah Soal



Gambar 5.
Memilih Kata Kunci

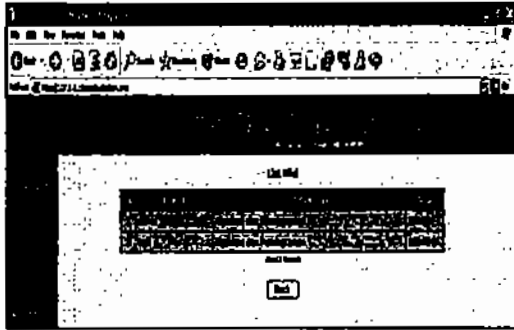


Gambar 3.
Jendela Login

Untuk siswa, setelah mereka log-on pada sistem, mereka dapat memilih ujian yang akan mereka ambil. Gambar 6 menunjukkan halaman menjawab ujian untuk siswa, sedangkan Gambar 7



Gambar 6.
Jendela Jawaban Siswa

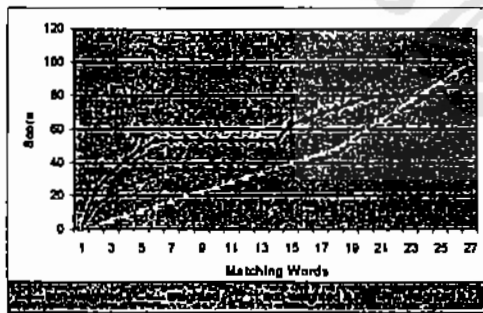


Gambar 7.
Jendela Penilaian

5. Hasil Ujicoba

Ujicoba kinerja telah dilakukan dengan membandingkan nilai tanpa bobot dengan *human raters*, dan nilai bobot dengan *human raters*.

Pada ujicoba ini diharuskan menjawab 8 soal yang telah disediakan dan membandingkan hasil dari sistem dengan 20 kata kunci, dengan 5 kata bobot akan dibandingkan dengan 20 kata kunci dengan tanpa kata bobot. Masing-masing nilai tersebut yang didapatkan dari sistem akan dibandingkan juga dengan nilai *human raters*nya masing-masing. Hasilnya dapat dilihat pada Gambar 8.

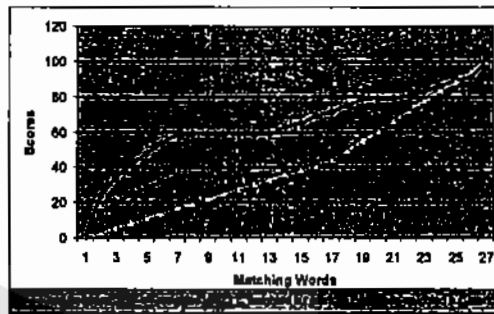


Gambar 8.
Hasil Perbandingan Nilai Dari *Human Raters* Dengan Nilai Dari Sistem LSA Dalam Bentuk Grafik Dengan 5 Kata Bobot Penting

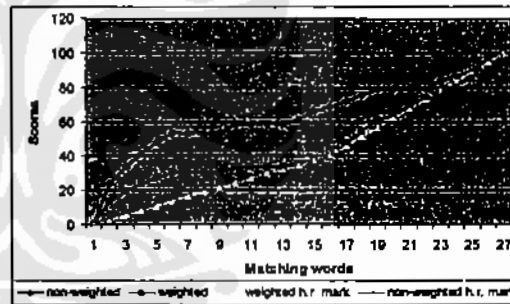
Pada Gambar 8 terlihat bahwa bila dibandingkan antara nilai *human raters* dengan nilai yang didapat dengan sistem dengan kata bobot lebih sesuai dengan nilai *human raters* dengan tanpa kata bobot. Nilai kesesuaian rata-rata dengan *human raters*, untuk nilai sistem dengan tanpa kata

bobot adalah 88.13 dan nilai *human raters* dengan kata bobot adalah 95.88.

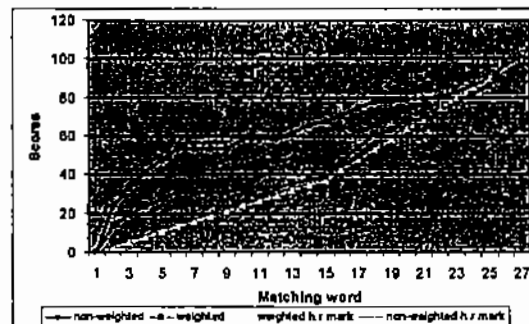
Gambar 9, 10 dan 11 menunjukkan perbandingan nilai dari sistem dengan pemilihan 20 kata kunci dengan 6, 7 dan 8 kata bobot. Untuk nilai kesesuaian rata-rata dengan *human raters* untuk 5, 6, 7 dan 8 kata bobot ditunjukkan pada Tabel 1.



Gambar 9.
Hasil Perbandingan Nilai Dari *Human Raters* Dengan Nilai Dari Sistem LSA Dalam Bentuk Grafik Dengan 6 Kata Bobot Penting



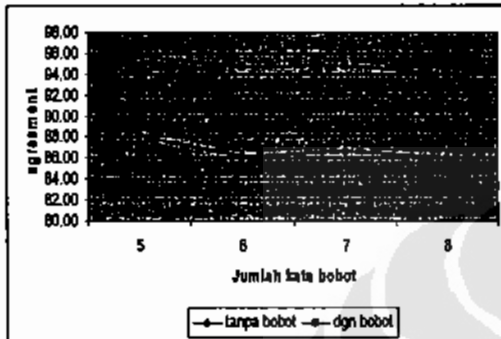
Gambar 10.
Hasil Perbandingan Nilai Dari *Human Raters* Dengan Nilai Dari Sistem LSA Dalam Bentuk Grafik Dengan 7 Kata Bobot Penting



Gambar 11.
Hasil Perbandingan Nilai Dari *Human Raters* Dengan Nilai Dari Sistem LSA Dalam Bentuk Grafik Dengan 8 Kata Bobot Penting

Tabel 1.
Perbandingan Nilai Pada Jumlah Kata Bobot yang Berbeda Dari 5, 6, 7 dan 8.

No	Jumlah kata bobot	Agreement dengan human rater tanpa bobot	Agreement dengan human raters dengan bobot
1.	5	88.13	95.88
2.	6	86.20	94.74
3.	7	86.66	94.80
4.	8	86.00	93.94



Gambar 12.
Nilai Agreement Dengan Human Raters Dengan Jumlah Keyword 20 dan Jumlah Kata Bobot Bervariasi Dari 5, 6, 7 dan 8

Pada Tabel 1 dapat dilihat bahwa nilai kesesuaian dengan *human raters* lebih baik dengan menggunakan kata bobot dibandingkan dengan kesesuaian dengan *human raters* tanpa bobot. Untuk ujicoba yang ditampilkan pada makalah ini, didapat hasil dengan kisaran 86.00 % – 88.13 % untuk nilai kesesuaian dengan human rater tanpa bobot dan kisaran 93.94 % – 95.88 % untuk kesesuaian dengan *human raters* dengan menggunakan kata bobot.

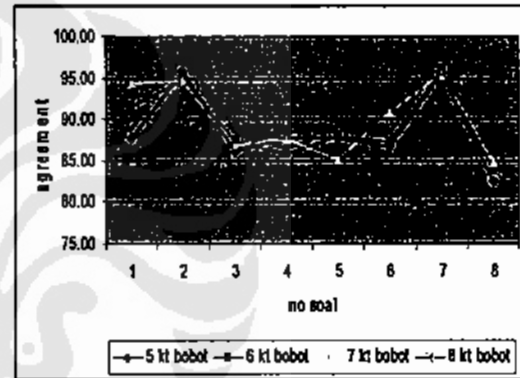
Pada hasil ujicoba ini juga ditunjukkan adanya penurunan nilai kesesuaian dengan semakin banyak kata bobot yang digunakan. Perbandingan ini dapat dilihat pada Gambar 12.

Untuk keseluruhan ujicoba yang dilakukan terhadap 8 soal yang dipilih, didapat hasil kesesuaian dengan *human raters* dengan tanpa kata bobot adalah 74.86 % - 88.13 %. Dan dengan kata bobot : 82.56 % - 96.42 %. Tabel kesesuaian

dengan *human raters* untuk keseluruhan ujicoba dengan bobot dapat dilihat pada Tabel 2

Tabel 2.
Nilai Kesesuaian Dari ke 8 Soal Dengan Kata Bobot Dari 5 – 8

	5 kt bobot	6 kt bobot	7 kt bobot	8 kt bobot
Nilai Agreement	82.56	83.69	84.85	86.14
	86.20	85.78	85.19	86.79
	86.74	87.32	86.89	88.85
	87.72	87.38	87.66	89.76
	87.77	87.94	90.91	93.94
	87.84	91.64	94.33	95.57
	95.71	94.74	94.80	96.00
	95.88	96.04	95.48	96.42



Gambar 13.
Kesesuaian Dari ke 8 Soal Dengan Kata Bobot 5, 6, 7 dan 8

Sedangkan grafik yang menunjukkan kesesuaian dengan *human raters* dari ke 8 soal dengan kata bobot 5, 6, 7 dan 8 dapat ditunjukkan pada Gambar 13. Dari Gambar 13 dapat dilihat bahwa soal no. 2 dan soal no. 7 mempunyai kesesuaian dengan *human raters* lebih baik.

6. Kesimpulan

1. Sistem SIMPLE dapat bekerja dengan baik dengan menggunakan bahasa Indonesia.
2. Hasil ujicoba dengan menggunakan pembobotan dengan mengalikan dengan nilai dua menunjukkan:
 - a. Hasil Ujicoba dengan tanpa bobot, kesesuaian: 74.86 % - 88.13 %

b. Hasil . Ujicoba dengan bobot, kesesuaian: 82.56 % - 96.42 %

Hal ini menunjukkan bahwa sistem dengan bobot lebih baik nilai kesesuaiannya dibandingkan dengan sistem tanpa bobot.

3. Dari 8 soal yang diujicoba didapat hasil dimana kesesuaian dengan *human raters* dapat tergantung dari soal yang diberikan, penentuan kata kunci serta jumlah kata bobot yang digunakan.

Daftar Acuan

- [1]. Asmawi Zainul, Noebi Nasution, "Penilaian Hasil Belajar", PAU untuk peningkatan dan Pengembangan Aktivitas Instruksional Direktorat Jendral Pendidikan Tinggi Departemen Pendidikan Nasional, Jakarta, 2001
- [2]. Lawrence M Rudner, Tahung Liang, "Automated Essay Scoring Using Bayes' Theorem", *The Journal of Technology, Learning and Assessment*, volume 1, Number 2, June 2002
- [3]. Salvatore Valenti, Francesca Neri, Alessandro Cucchiarelli, "An Overview of Current Research on Automated Essay Grading", *Journal of Information Technology Education*, Volume 2, 2003.
- [4]. Derek Stokley, "E-learning Definition and Explanation (Elearning, Online Training, Online Learning)", <http://derekstockley.com.au/elearning-definition.html>, Australia, last up date : 26 February 2006
- [5]. Laurence Rudner, Phill Gagne, "an Overview of Three Approaches to Scoring Written Essays by Computer", *Practical Assessment, Research & Evaluation*, A peer-reviewed electronic Journal, 2001 from: <http://ericae.net/pare/getvn.asp?v=7&n=26>
- [6]. Linlin Irene Chen, Ruth Robbins, "On Line Testing for Business Programming Class", *Proceeding of 8th Annual International Distance Education Conference* January 23-26, 2001
- [7]. Anak Agung Putri Ratna, Luhur Bayuaji and Muhammad Salman, "Design And Implementation of Distance Learning in Basic Computer Course", *IMSA'01 IASTED Proceeding*, pp. 51-55, Hawaii, 2001
- [8]. Anak Agung Putri Ratna, Adjie Pamungkas, Natalia Evianti and Muhammad Salman, "Daskom On-Line: User Management Implementation on Web Based Learning Application", *Web Net Proceedings*, pp 1041-1042, Orlando 2001
- [9]. Anak Agung Putri Ratna, Astha Ekadiyanto, Djoko Hartanto, Seinosuke Narita, "Daskom On-Line: Implementation Distance Learning On Basic of Computer Course", *Web Net Proceedings*, pp 1600-1602, Denver 2002
- [10]. Anak Agung Putri Ratna, Patar P Raymont, Natalia Evianti, Djoko Hartanto, Seinosuke Narita, "DistanceE-Learning Implementation and Analysis on Jarkom On-line Evaluation Sistem", *IMSA'03 IASTED Proceeding*, Hawaii, 2003
- [11]. Jill Burstein, Magdalena Wolska, "Toward Evaluation of Writing Style: Finding Overlay Repetitive Word Use in Student Essays", 10th Conference of the European Chapter of Association for Computational Linguistics, April 12- 17, 2003, Budapest, Hungary.
- [12]. Burstein, J., Leacock, C. Chodorow, M. , "*CriterionSM : Online essay evaluation : An Application for automated evaluation of student essays*" *Proceedings of the Fifteenth Annual Conference on Innovative Applications of Artificial Intelligence*, Acapulco, Mexico, August 2003. (This paper has recieved a AAAI Deployed Application Award.)
- [13]. Marti A Hearst, "The Debate on Automated Essay Grading", *IEEE Intelligent Systems*, September/ Oktober 2000.
- [14]. Peter W Foltz, Darrel Laham, Thomas K Landauer, "The Intelligent

- Essay Assessor Application to Educational Technology”, Interactive Multimedia Electronic Journal of Computer Enhanced Learning, Wake Forest University, 1999.
- [15]. Peter W Foltz, Darrel Laham, Thomas K Landauer, “Automated Essay Scoring: Application to Educational Technology”, Proceeding of World Conference on Educational Multimedia, Hypermedia and Telecommunications (EDMEDIA) Volume 1999, Issue 1, 1999, Seattle, US
- [16]. Knowledge Analysis Technologies, “Intelligent Essay Assessor”, Download dari site: <http://www.knowledge-technologies.com/IEA.shtml>, akses tgl 9 September 2004
- [17]. Knowledge Analysis Technologies, “Intelligent Essay Assessor”, Download dari site: <http://www.knowledge-technologies.com/IEA.shtml>, akses tgl 9 September 2004
- [18]. Cambridge University Press, Program Copyright © 1988-1992. Numerical Recipes in C: The Art of Computing. <http://www.nr.com>.
- [19]. Eric W Weisstein, “Frobenius Norm”, Mathword, A Wolfram Web Resource, Wolfram Research Inc., 2004, <http://www.mathword.wolfram.com/FrobeniusNorm.html>, akses tgl 9 September 2004.
- [20]. Anak Agung Putri Ratna, Adhe Widi A, Siti Andella, Bagio Budiardjo, Seinosuke Narita, “Web Based Automated Essay Grading System Using Latent Semantic Analysis Method for Indonesian Language with Most Important Words”, *ED-Media Proceedings*, Orlando 2006
- [21]. David A. Frisbie, Kristie K. Waltman, “Developing a Personal Grading Plan”, Educational Measurement: Issues and Practice, Fall 1992, UW’s Faculty Resource On Grading, Iowa, <http://depts.washington.edu/grading/plan/frisbie1.htm>, akses pada Maret 2006.
- [22]. Christian M. Reiner, Timothy W. Bothell, Richard R. Sudweeks, Bud Wood. “Preparing Effective Essay Questions”, new Forums Press, 2002, <http://testing.byu.edu/info/handbooks/WritingEffectiveEssayQuestions.pdf>, akses pada Maret 2006
- [23]. Andrew S Tanenbaum, “Computer Networks”, fourth Edition, Prentice Hall PTR International Edition, New Jersey, 2003