

## BAB 2 LANDASAN TEORI

Bab ini menjabarkan berbagai informasi yang sudah didapatkan dari studi literatur mengenai perkembangan sistem penilai esai otomatis di luar negeri dan di Indonesia, *Vector Space Model* (VSM), *Latent Semantic Analysis* (LSA), perluasan kunci jawaban atau *query expansion*, pemotongan imbuhan kata atau *stemming*, dan pembahasan mengenai apa yang dilakukan dalam penelitian berdasarkan landasan teori tersebut.

### 2.1 Perkembangan Sistem Penilai Esai Otomatis

Salah satu motivasi yang mendasari munculnya penelitian di bidang penilaian esai otomatis adalah pentingnya keterampilan menulis sebagai salah satu cara menyampaikan pendapat dan berkomunikasi sebagai bagian dari proses pembelajaran. Namun, seringkali pengajar tidak memiliki waktu yang cukup untuk memeriksa esai dan memberikan umpan balik yang dapat digunakan siswa untuk memperbaiki esainya. Di sinilah sistem penilai esai otomatis dapat membantu untuk memberikan penilaian dan umpan balik terhadap esai siswa (Calfee, 2000). Selain itu, penilaian esai otomatis dapat mengurangi waktu dan biaya yang digunakan untuk menilai esai (Valenti, Neri, & Cucchiarelli, 2003).

Menurut Page sebagaimana yang disampaikan dalam (Valenti, Neri, & Cucchiarelli, 2003), penilaian esai otomatis dapat dibedakan menjadi dua bagian, yaitu penilaian esai berdasarkan konten tulisan dan berdasarkan gaya atau teknik penulisan. Penilaian berdasarkan konten mengacu pada apa yang disampaikan, dan penilaian berdasarkan gaya atau teknik mengacu pada sintaks kalimat, diksi, dan hal-hal lain yang berkaitan dengan cara menyampaikan suatu hal.

Esai yang dinilai sendiri sebenarnya terbagi dalam dua bagian. Yang pertama adalah esai yang merupakan karangan siswa (bisa berupa eksposisi, deskripsi, argumentasi, atau narasi). Yang kedua adalah jawaban esai pendek yang biasa terdapat pada bagian evaluasi buku teks pelajaran atau ujian. Esai ini tidak berupa karangan yang panjang, melainkan jawaban pendek yang menjawab pertanyaan yang diajukan. Istilah untuk tipe esai ini adalah *content-based essay* atau *short-answer*. Dalam laporan ini digunakan istilah jawaban esai untuk

membedakannya dengan esai karangan.

Subbab berikutnya menjelaskan sistem penilai esai otomatis yang sudah dikembangkan di luar negeri, yang terdiri dari sistem untuk menilai esai berdasarkan gaya/teknik dan konten, serta sistem untuk menilai jawaban ujian esai. Sistem yang dibahas di sini meliputi *Project Essay Grader (PEG)*, *Intelligent Essay Assessor (IEA)*, *Electronic Essay Rater (E-Rater)*, dan *Conceptual Rater (C-Rater)*.

### 2.1.1 *Project Essay Grader (PEG)*

Riset di bidang penilaian esai otomatis atau *Automated Essay Grading (AEG)* diawali oleh Page pada tahun 1966 yang kemudian menghasilkan PEG (*Project Essay Grader*) pada tahun 1994 (Williams, 2001). Page memulainya dengan esai-esai yang sudah dinilai oleh pengajar kemudian bereksperimen dengan fitur-fitur tekstual dalam esai yang bisa diekstrak secara otomatis. Dengan *multiple linear regression*, fitur-fitur ini kemudian dihitung untuk memprediksi nilai dengan menentukan kombinasi yang menghasilkan nilai yang paling mendekati penilaian dari manusia (Kukich, 2000).

Fitur-fitur ini dijelaskan oleh Page dengan dua istilah, *trins* dan *proxes*. “...we coined two explanatory terms: *Trins* were the intrinsic variables of interest - fluency, diction, grammar, punctuation, and many others. We had no direct measures of these, so began with substitutes: *Proxes* were approximations, or possible correlates, of these *trins*. All the computer variables (the actual counts in the essays) were *proxes*. For example, the *trin* of fluency was correlated with the *prox* of the number of words” (Valenti, Neri, & Cucchiarelli, 2003).

*Trins* adalah variabel intrinsik dalam esai dan *proxes* adalah variabel yang digunakan untuk mengaproksimasi nilai *trins* tersebut. Contohnya *trin* untuk kelancaran menulis dinilai dengan *prox* yang berupa jumlah kata pada esai. Dengan menggunakan variabel-variabel ini, maka PEG adalah sistem yang menilai esai berdasarkan kualitas penulisan dan bukan berdasarkan kontennya. (Valenti, Neri, & Cucchiarelli, 2003).

### 2.1.2 *Intelligent Essay Assessor (IEA)*

IEA adalah sistem analisis dan penilaian esai yang disertai dengan

pemberian umpan balik atas esai yang dinilai. IEA dikembangkan dengan teknik *Latent Semantic Analysis* (LSA), yang bisa mensimulasikan berbagai fenomena kognitif manusia seperti kategorisasi kata, pemahaman wacana, dan penilaian kualitas esai. LSA adalah metode untuk menentukan kesamaan makna kata dan dokumen dengan menganalisis korpus teks yang besar (Landauer, Foltz, & Laham 1998).

Sebelum menilai esai, IEA dilatih terlebih dahulu dengan menggunakan korpus yang terdiri dari teks yang berhubungan dengan domain dari esai yang akan dinilai. Kata-kata dan dokumen *training* direpresentasikan sebagai vektor dalam ruang semantik berdimensi tinggi. Kemudian esai siswa dan kata-kata di dalamnya juga direpresentasikan sebagai vektor. Gabungan vektor ini merupakan matriks yang kemudian didekomposisi berdasarkan teknik *Singular Value Decomposition* (SVD) dan dikurangi dimensinya sehingga hubungan mendalam antara kata dan dokumen dapat disimpulkan. Nilai siswa ditentukan dari jarak antara vektor esai siswa dengan vektor esai yang sudah dinilai sebelumnya atau esai standar yang sudah diketahui kualitasnya (Landauer, Foltz, & Laham 1998).

IEA merupakan sistem yang digunakan untuk menilai esai berdasarkan kontennya. IEA dapat memberikan umpan balik terhadap sebuah esai yang terdiri dari pengorganisasian esai, penggunaan bahasa, dan lain-lain. Sebuah contoh eksperimen dengan IEA dilakukan pada New Mexico State University untuk kuliah Psikolinguistik. Untuk mengembangkan sistem ini, LSA dilatih dengan empat bab dari buku yang digunakan di kelas tersebut. Kemudian digunakan empat puluh esai yang sudah dinilai dari kelas tahun sebelumnya dan esai yang akan dinilai diberi nilai berdasarkan kedekatannya dengan empat puluh esai tersebut. Pada percobaan ini, korelasi nilai IEA dengan nilai manusia mencapai 0.80 (Landauer, Foltz, & Laham, 1999). Eksperimen lain untuk esai GMAT (*Graduate Management Achievement Test*) menghasilkan persetujuan dengan manusia sebesar 85%-91% (Valenti, Neri, & Cucchiarelli, 2003).

### **2.1.3 Electronic Essay Rater (E-Rater)**

E-Rater dikembangkan menggunakan kombinasi teknik statistika dan *Natural Language Processing* (NLP) untuk mengekstrak fitur linguistik dari esai

yang akan dinilai. Fitur-fitur ini terdiri dari analisis struktur wacana, analisis struktur sintaks kalimat, dan penggunaan *vocabulary* sesuai domain yang diujikan. E-Rater mengekstrak fitur-fitur tersebut dari contoh esai yang sudah dinilai untuk membangun model dan kemudian digunakan teknik *multiple linear regression* untuk memprediksi nilai esai yang akan dinilai berdasarkan fitur-fitur tersebut (Williams, 2001).

Untuk membangun model, E-Rater dilatih dengan 270 esai yang sudah dinilai manusia. Esai yang mengandung topik sesuai pertanyaan, memiliki struktur argumen yang terorganisasi dengan baik dan koheren, serta menunjukkan variasi penggunaan kata dan struktur penulisan akan mendapatkan nilai tinggi dari penilaian berskala enam. Sistem ini sangat kompleks dan membutuhkan korpus latihan yang besar. Namun hasilnya sangat baik, dengan persetujuan antara nilai sistem dengan nilai manusia mencapai 87%-94% (Valenti, Neri, & Cucchiarelli, 2003).

#### **2.1.4 Conceptual Rater (C-Rater)**

Berbeda dengan tiga sistem yang sudah dijelaskan sebelumnya, C-Rater dikembangkan untuk menilai jawaban esai atau *short answer* dari pertanyaan berbasis konten seperti yang terdapat pada ujian atau bagian evaluasi buku teks pelajaran (Valenti, Neri, & Cucchiarelli, 2003).

C-Rater banyak mengambil teknik dan program NLP yang digunakan E-Rater. Berbeda dengan E-Rater yang menilai esai dari keterampilan penulisan dan bukan konten, C-Rater bertujuan menilai sebuah jawaban untuk ditentukan benar atau salahnya. Hal ini dilakukan dengan memeriksa apakah sebuah jawaban mengandung konsep yang benar sesuai pertanyaan atau tidak tanpa melihat teknik penulisannya.

C-Rater menghasilkan analisis dari relasi logika antara komponen sintaksis yang muncul pada setiap kalimat yang muncul pada jawaban untuk menangkap konsep yang disampaikan. Sistem ini tidak membutuhkan banyak jawaban yang sudah dinilai untuk *training*, melainkan cukup menggunakan satu jawaban yang benar (kunci jawaban) untuk dibandingkan dengan jawaban siswa. Persetujuan antara C-Rater dan penilai manusia mencapai 80% (Valenti, Neri, & Cucchiarelli,

2003).

## 2.2 Perkembangan Sistem Penilai Esai Otomatis untuk Bahasa Indonesia

Sistem penilai esai otomatis untuk Bahasa Indonesia sudah dikembangkan melalui penelitian di Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia. Semua sistem ini dikembangkan dengan teknik LSA. Esai yang dinilai adalah jawaban ujian bentuk esai.

Sistem ini merepresentasikan dokumen jawaban siswa sebagai matriks  $A$  pada ruang semantik LSA, dan kunci jawaban sebagai vektor  $q$ . Kemudian dilakukan proses SVD terhadap matriks  $A$  dengan mereduksi dimensinya, sehingga matriks  $A$  didekomposisi menjadi  $A = U_k \cdot \Sigma_k \cdot V_k$ . Matriks  $A_k$  hasil pengurangan dimensi ini adalah aproksimasi terhadap matriks  $A$  dengan hanya menggunakan  $k$  dimensi. Kesamaan antara jawaban siswa dan kunci jawaban kemudian dihitung dari jarak vektornya.

Vektor kunci jawaban direpresentasikan sebagai vektor dalam dimensi  $k$  dengan rumus sebagai berikut:

$$\bar{q} = q^T U_k \Sigma_k^{-1}$$

Pada vektor  $d$  yang merepresentasikan jawaban siswa dan merupakan kolom matriks  $A$  juga diterapkan hal yang sama sehingga menghasilkan:

$$\bar{d} = d^T U_k \Sigma_k^{-1}$$

Jawaban siswa dinilai dengan menghitung kedekatan antara vektor jawabannya dengan vektor kunci jawaban menggunakan rumus *cosine similarity* sebagai berikut:

$$\cos \alpha = \frac{\bar{q} \cdot \bar{d}}{\|\bar{q}\| \|\bar{d}\|}$$

Publikasi pertama mengenai sistem penilaian jawaban esai dengan LSA yang penulis temukan adalah (Krisnanda, 2005) yang melakukan percobaan dengan variasi banyaknya kata kunci jawaban. Kesimpulan yang didapatkan adalah, semakin banyak kata kunci yang digunakan, maka nilai ujian yang

diberikan LSA akan semakin mendekati nilai ujian mahasiswa yang diberikan penilai manusia. Nilai korelasi dengan penilai manusia juga mengalami peningkatan. Korelasi yang dihasilkan mencapai 0.86-0.96. Kata kunci yang digunakan terdiri dari 10, 20, 30, dan 40 kata kunci. Makin banyak kata kunci yang digunakan, maka korelasi akan makin tinggi.

Tahun 2007 terdapat penelitian mengenai sistem penilaian esai yang serupa dengan Krisnanda, tapi sudah disempurnakan dengan pembobotan kata, urutan kata, dan persamaan kata. Sistem ini disebut (**SIstiM PeniLaian Esei Otomatis**) (Ratna, Budiarjo, & Hartanto, 2007). Jawaban siswa dan kunci jawaban akan direpresentasikan dalam bentuk matriks kemudian dihitung nilai normalisasinya dengan normalisasi Frobenius. Rumus normalisasi Frobenius untuk matriks A adalah sebagai berikut:

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

Nilai normalisasi ini kemudian dibandingkan untuk menghasilkan nilai jawaban siswa. Hasil uji coba SIMPLE untuk lima soal jawaban esai untuk kelas kecil (lima mahasiswa) mencapai kesamaan nilai dengan nilai manusia sebesar 69.80%-94.64% dan untuk kelas menengah (sepuluh mahasiswa) sebesar 77.18%-98.42%.

Pengembangan selanjutnya dilakukan dengan implementasi pembobotan yang berbeda pada SIMPLE dilakukan oleh Octaria, Hermawandi, dan Harisma pada tahun 2008. Pembobotan adalah nilai kepentingan suatu kata pada suatu dokumen, yang terdiri dari bobot lokal, global, dan normalisasi.

Octaria menerapkan empat kombinasi pembobotan untuk jawaban siswa dan kunci jawaban. Nilai korelasi sistem dengan manusia yang paling tinggi adalah sebesar 0.39 yang didapatkan saat menerapkan pembobotan lokal SQRT, pembobotan global NORMAL, dan normalisasi COSINUS untuk jawaban siswa dan pembobotan lokal BINARY dan pembobotan global NORMAL untuk kunci jawaban (Octaria, 2008).

Hermawandi menerapkan skema pembobotan yang disingkat dengan SICBI yang merupakan akronim dari SQRT-IGFF-COSN-BNRY-IDFB. Secara berurutan, jenis pembobotan ini adalah pembobotan yang diterapkan untuk bobot

lokal, global, dan normalisasi jawaban, serta bobot lokal dan global untuk kunci jawaban. Percobaan dilakukan dengan 10, 15, dan 20 siswa dengan rata-rata selisih nilai sistem dengan penilai manusia untuk tiap percobaan mencapai 13.98, 17.84, dan 10.90. Dengan percobaan ini tidak bisa ditarik kesimpulan bahwa makin banyak jumlah siswa maka makin baik performa sistem. Penelitian selanjutnya dapat dilakukan untuk jumlah mahasiswa yang lebih banyak (Hermawandi, 2008).

Harisma menerapkan skema pembobotan yang berbeda dengan kedua penelitian sebelumnya, yaitu dengan tiga tingkatan bobot kata kunci yang dapat dipilih pengguna, yang masing-masing bernilai 1, 2, dan 3. Uji coba dilakukan dengan 10 soal esai untuk 23 mahasiswa dan korelasi nilai sistem dengan penilai manusia mencapai 0.77 dengan rata-rata selisih nilai 17.36 (Harisma, 2008).

Penelitian dengan sistem yang sama pada tahun 2005 sampai dengan 2008 memberikan hasil yang bervariasi. Penelitian (Krisnanda, 2005) dan (Ratna, Budiarto, & Hartanto, 2007) memberikan hasil korelasi dan persetujuan yang tinggi. Namun penelitian-penelitian berikutnya yang dilakukan pada tahun 2008 memberikan hasil korelasi dan persetujuan dengan penilai manusia yang lebih rendah dari penelitian sebelumnya. Hal ini mungkin disebabkan oleh perbedaan pengukuran kesamaan jawaban esai. Penelitian (Krisnanda, 2005) menggunakan perbandingan normalisasi Frobenius antara matriks jawaban dengan kunci jawaban sedangkan (Octaria, 2008), (Hermawandi, 2008), dan (Harisma, 2008) menggunakan *cosine similarity*.

### **2.3 Vector Space Model (VSM)**

VSM adalah representasi kumpulan dokumen sebagai vektor dalam sebuah ruang vektor. VSM merupakan teknik dasar dalam perolehan informasi yang dapat digunakan untuk penilaian relevansi dokumen terhadap kata kunci pencarian (*query*) pada mesin pencari, klasifikasi dokumen, dan pengelompokan dokumen (Manning, Raghavan, & Schütze, 2008).

Kumpulan kata-kata dan dokumen direpresentasikan dalam bentuk matriks kata-dokumen. Baris matriks mewakili kata-kata dan kolomnya mewakili dokumen. Contoh kumpulan dokumen dan representasi matriks kata-dokumennya

adalah sebagai berikut:

D1: Saya mahasiswa Ilmu Komputer.  
 D2: Saya menimba ilmu di Fakultas Ilmu Komputer.  
 D3: mahasiswa Fakultas Ilmu Komputer banyak.



	D1	D2	D3
<b>Banyak</b>	0	0	1
<b>Di</b>	0	1	0
<b>Fakultas</b>	0	1	1
<b>Ilmu</b>	1	2	1
<b>Komputer</b>	1	1	1
<b>Mahasiswa</b>	1	0	1
<b>Menimba</b>	0	1	0
<b>Saya</b>	1	1	0

**Gambar 2.1: Contoh Dokumen dan Matriks Kata-dokumen**

D1, D2, dan D3 masing-masing dianggap sebagai satu dokumen yang berisi satu baris teks. Dokumen-dokumen ini kemudian direpresentasikan dalam matriks kata-dokumen. Setiap elemen matriks  $a_{ij}$  merepresentasikan nilai kemunculan kata  $i$  dalam dokumen  $j$ . Nilai  $a_{ij}$  dihasilkan dari skema pembobotan untuk menunjukkan seberapa penting sebuah kata dalam suatu dokumen. Pada contoh di atas, pembobotan yang digunakan adalah *term frequency*, yaitu jumlah kemunculan kata dalam dokumen. Misalnya pada baris keempat dan kolom kedua matriks ( $a_{42}$ ) yang berisi nilai 2, menunjukkan bahwa kata “ilmu” muncul sebanyak 2 kali pada dokumen D2. Pada contoh ini diasumsikan pengolahan teks *case-insensitive*, yaitu tidak memperhatikan penggunaan huruf besar atau kecil (dianggap sama saja).

Terdapat tiga bentuk pembobotan yang dapat diberikan pada suatu kata dalam matriks yaitu pembobotan lokal, pembobotan global, dan normalisasi. Pembobotan lokal diberikan pada suatu kata berdasarkan jumlah kemunculannya pada satu dokumen. Pembobotan global diberikan pada suatu kata atas kemunculannya di semua dokumen. Sedangkan normalisasi adalah sebuah cara untuk menormalkan panjang vektor dokumen sehingga vektor tersebut independen terhadap panjangnya. Setiap kata mendapatkan bobot dari tiga bentuk pembobotan ini sehingga bobot totalnya adalah perkalian dari tiga pembobotan

tersebut. Beberapa contoh pembobotan untuk ketiga bentuk pembobotan ini dapat dilihat pada Tabel 2.1.

Tabel 2.1: Contoh Skema Pembobotan

Pembobotan Lokal	Pembobotan Global	Normalisasi
Natural/ <i>Term frequency</i> (TF) $f_{ij}$	<i>Inverse Document Frequency</i> (IDF) $\log\left(\frac{N}{n_i}\right)$	Normalisasi kosinus $\frac{1}{\sqrt{w_1^2 + w_2^2 + \dots + w_M^2}}$
Logaritma $1 + \log f_{ij}$ jika $f_{ij} > 0$ 0 jika $f_{ij} = 0$	<i>Global Frequency IDF</i> (GfIDF) $\frac{F_i}{n_i}$	
Biner 1 jika $f_{ij} > 0$ 0 jika $f_{ij} = 0$	<i>Probabilistic IDF</i> $\log\left(\frac{N - n_i}{n_i}\right)$	

**Keterangan:**

$f_{ij}$  = jumlah kemunculan kata  $i$  pada dokumen  $j$   
 $F_i$  = jumlah kemunculan kata  $i$  pada seluruh dokumen  
 $N$  = jumlah dokumen  
 $n_i$  = jumlah kemunculan kata  $i$  pada suatu dokumen  
 $w$  = bobot kata

Jenis pembobotan yang sering digunakan dalam mesin pencari adalah TF IDF, yaitu kombinasi antara *Term Frequency* (TF) dengan *Inverse Document Frequency* (IDF). Secara umum pada sebuah kumpulan dokumen terdapat kata-kata yang sangat sering muncul pada sebuah dokumen dan pada banyak dokumen lainnya namun tidak penting untuk menjelaskan karakteristik atau isi dokumen. Kata-kata ini disebut dengan *stopwords*. Dengan TF IDF, bobot untuk kata-kata ini dapat menjadi kecil karena jumlah kemunculannya pada sebuah dokumen dikalikan dengan nilai logaritma dari jumlah dokumen dibagi jumlah kemunculannya pada dokumen tersebut. Sebaliknya, kata-kata yang muncul dengan frekuensi sedang dan tidak terdapat pada banyak dokumen biasanya penting dan merepresentasikan isi dokumen. Dengan TF IDF, bobot kata-kata ini menjadi besar.

VSM merupakan model perolehan informasi yang sederhana dan dapat memberikan berbagai bobot kata untuk membedakan kata-kata yang penting dengan yang tidak. Kelemahan VSM antara lain adalah tidak dapat menangkap persamaan makna kata. Dalam mesin pencari, kelemahan ini diatasi dengan cara memperbarui *query*. Terdapat dua metode dalam memperbarui *query* ini, yaitu metode global dan metode lokal. Contoh metode global adalah *query expansion* atau perluasan kunci jawaban dan contoh metode lokal adalah *relevance feedback* atau umpan balik dari pengguna terhadap relevansi hasil pencarian. Penelitian ini menggunakan *query expansion* untuk mengatasi kekurangan VSM. Penjelasan mengenai metode ini terdapat pada Subbab 2.5.

#### 2.4 Latent Semantic Analysis (LSA)

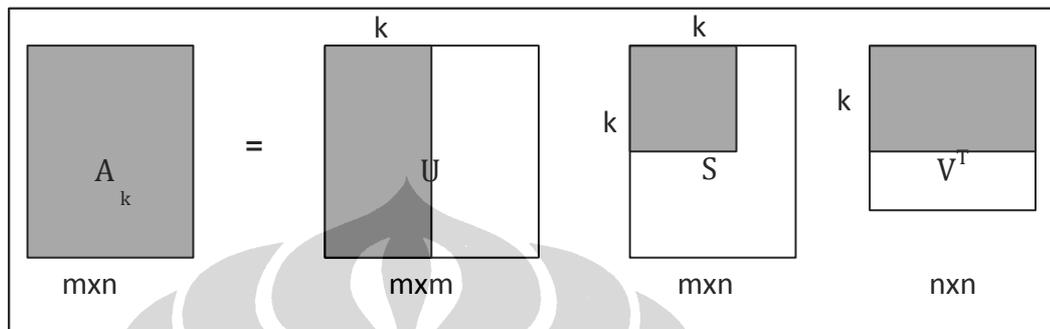
LSA adalah metode untuk menentukan kesamaan makna kata dan dokumen dengan menganalisis korpus teks yang besar (Landauer, Foltz, & Laham 1998). LSA tidak menggunakan pemrosesan NLP atau program kecerdasan buatan, melainkan murni metode matematika/statistika yang dapat mengekstrak dan menyimpulkan hubungan antara kata pada dokumen sesuai penggunaan kontekstualnya.

Proses LSA dimulai dengan membuat matriks kata-dokumen, di mana baris matriks merepresentasikan kata-kata dan kolom matriks merepresentasikan dokumen-dokumen. Setiap elemen matriks  $a_{ij}$  berisi nilai kemunculan kata  $i$  pada dokumen  $j$ . Kemudian LSA menerapkan *Singular Value Decomposition* (SVD) terhadap matriks ini. SVD menghasilkan tiga komponen matriks, yaitu satu matriks ortogonal baris, satu matriks ortogonal kolom, dan satu matriks diagonal. Matriks diagonal yang dihasilkan berisi elemen nonnegatif, dan elemen yang bukan nol disebut nilai singular dari  $A$ . Banyaknya nilai singular  $A$  merupakan *rank* dari  $A$  yang besarnya  $r \leq \min(m,n)$ . Jika matriks  $A$  berukuran  $m \times n$ , maka penerapan SVD terhadap matriks  $A$  menghasilkan tiga komponen matriks sebagai berikut.

$$A_{m \times n} = U_{m \times m} S_{m \times n} V_{n \times n}^T$$

Dari dekomposisi matriks ini, dimensi matriks  $A$  dapat dikurangi sampai sebesar  $k$  dan matriks  $A$  dapat direkonstruksi hanya dengan menggunakan  $k$

dimensi untuk mengaproksimasi matriks  $A$ . Rekonstruksi dengan pengurangan dimensi ini menghasilkan matriks  $A_k$  yang disebut dengan *low rank approximation* (Manning, Raghavan, & Schutze, 2008). Rekonstruksi hanya dengan  $k$  dimensi dilakukan dengan mengambil hanya  $k$  dimensi dari komponen matriks  $U$ ,  $S$ , dan  $V$  sehingga  $A_k = U_k S_k V_k^T$ .



**Gambar 2.2: Proses SVD dengan Pengurangan Dimensi**

*Low rank approximation* dari  $A$  menghasilkan representasi baru untuk setiap dokumen, di mana kesamaan antara kata-kata dan dokumen pada matriks  $A$  dapat ditemukan. *Rank* yang digunakan dalam SVD akan berbeda untuk setiap kasus. Jika *rank* yang digunakan terlalu kecil maka estimasi kesamaan antar kata/dokumen akan terlalu besar. Sebaliknya, jika *rank* terlalu besar maka kesamaan antar kata/dokumen tidak bisa ditangkap. Pemilihan *rank* dapat dilakukan dengan mencoba beberapa *rank* yang sering digunakan dan dipilih yang memberikan hasil optimal.

Berikut ini akan diberikan contoh pengaruh *low rank approximation* terhadap matriks kata-dokumen. Contoh ini diambil dari (Landauer, Foltz, & Laham, 1998).

c1: *Human machine interface for ABC computer applications*  
 c2: *A survey of user opinion of computer system response time*  
 c3: *The EPS user interface management system*  
 c4: *System and human system engineering testing of EPS*  
 c5: *Relation of user perceived response time to error measurement*

m1: *The generation of random, binary, ordered trees*  
 m2: *The intersection graph of paths in trees*  
 m3: *Graph minors IV: Widths of trees and well-quasi-ordering*  
 m4: *Graph minors: A survey*

**Gambar 2.3: Contoh Teks untuk LSA**

Sumber: (Landauer, Foltz, & Laham, 1998)

Pada Gambar 2.3 terdapat sembilan judul teks yang masing-masing

dianggap sebagai satu dokumen (c1-c5 dan m1-m5). Dokumen yang namanya diawali huruf “c” berisi judul dalam topik interaksi manusia-komputer sedangkan dokumen yang namanya diawali huruf “m” berisi judul dalam topik teori graf.

Kata yang dimasukkan pada matriks hanya kata-kata yang dicetak miring, yaitu kata yang setidaknya muncul dua kali pada dokumen yang berbeda. Hasil representasi matriks ini adalah matriks A sebagai berikut.

Tabel 2.2: Matriks Kata-dokumen

	C1	C2	C3	C4	C5	M1	M2	M3	M4
<b>Human</b>	1	0	0	1	0	0	0	0	0
<b>Interface</b>	1	0	1	0	0	0	0	0	0
<b>Computer</b>	1	1	0	0	0	0	0	0	0
<b>User</b>	0	1	1	0	1	0	0	0	0
<b>System</b>	0	1	1	2	0	0	0	0	0
<b>Response</b>	0	1	0	0	1	0	0	0	0
<b>Time</b>	0	1	0	0	1	0	0	0	0
<b>EPS</b>	0	0	1	1	0	0	0	0	0
<b>Survey</b>	0	1	0	0	0	0	0	0	1
<b>Trees</b>	0	0	0	0	0	1	0	1	0
<b>Graph</b>	0	0	0	0	0	0	1	1	1
<b>Minors</b>	0	0	0	0	0	0	1	1	1

Sumber: (Landauer, Foltz, & Laham, 1998)

Dari matriks tersebut, dapat dihitung korelasi antara kata “human” dengan “user”, yaitu  $r(\text{human.user})$  yaitu sebesar  $-0.38$  dan  $r(\text{human.minors})$  sebesar  $-0.29$ . Nilai kemunculan kata “survey” pada dokumen m4 adalah 1 dan “trees” pada dokumen m4 adalah 0. Keempat hal ini akan diamati untuk dilihat perbandingannya setelah dilakukan SVD.

SVD dilakukan dengan mengurangi dimensi matriks A menjadi 2 ( $k=2$ ) sehingga menghasilkan tiga komponen matriks  $U_k S_k V_k$  sebagai berikut:

U =		S =		V =	
-0.2214	0.1132	3.3409	0	-0.1974	0.0559
-0.1976	0.0721	0	2.5417	-0.6060	-0.1656
-0.2405	-0.0432			-0.4629	0.1273
-0.4036	-0.0571			-0.5421	0.2318
-0.6445	0.1673			-0.2795	-0.1068
-0.2650	-0.1072			-0.0038	-0.1928
-0.2650	-0.1072			-0.0146	-0.4379
-0.3008	0.1413			-0.0241	-0.6151
-0.2059	-0.2736			-0.0820	-0.5299
-0.0127	-0.4902				
-0.0361	-0.6228				
-0.0318	-0.4505				

Gambar 2.4: Tiga Komponen Matriks Hasil SVD

*Low rank approximation*  $A_k$  atau matriks rekonstruksi dari  $A$  hanya dengan 2 dimensi dibangun dengan mengalikan  $U_k S_k V_k^T$  menghasilkan matriks berikut:

Tabel 2.3: Matriks  $A_k$  dengan  $k = 2$

	C1	C2	C3	C4	C5	M1	M2	M3	M4
<b>Human</b>	0.16	0.4	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
<b>Interface</b>	0.14	0.37	0.33	0.4	0.16	-0.03	-0.07	-0.1	-0.04
<b>Computer</b>	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
<b>User</b>	0.26	0.84	0.61	0.7	0.39	0.03	0.08	0.12	0.19
<b>System</b>	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
<b>Response</b>	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
<b>Time</b>	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
<b>EPS</b>	0.22	0.55	0.51	0.63	0.24	-0.07	-0.14	-0.2	-0.11
<b>Survey</b>	0.1	0.53	0.23	0.21	0.27	0.14	0.31	0.44	0.42
<b>Trees</b>	-0.06	0.23	-0.14	-0.27	0.14	0.24	0.55	0.77	0.66
<b>Graph</b>	-0.06	0.34	-0.15	-0.3	0.2	0.31	0.69	0.98	0.85
<b>Minors</b>	-0.04	0.25	-0.1	-0.21	0.15	0.22	0.5	0.71	0.62

Sumber: (Landauer, Foltz, & Laham, 1998)

Hasil rekonstruksi ini menunjukkan bahwa korelasi “human” dengan “user” atau  $r(\text{human.user})$  meningkat menjadi 0.94 dibandingkan korelasi pada matriks asli yang hanya -0.38. Sebaliknya,  $r(\text{human.minor})$  turun menjadi -0.83 dari korelasi pada matriks asli yaitu sebesar -0.29. Pada matriks asli, “human” tidak pernah muncul pada dokumen yang sama dengan “user” maupun “minor”. Karena itu korelasinya sama-sama rendah. Tetapi pada matriks  $A_k$ , “human” berkorelasi kuat dengan “user” sedangkan korelasi dengan “minor” menjadi lebih rendah. Hal ini terjadi karena “human” dan “user” muncul pada konteks dengan makna yang sama, meskipun bukan pada dokumen yang sama. Pengurangan dimensi merepresentasikan keduanya memiliki kesamaan, sebagaimana hal yang sebaliknya terjadi pada “human” dan “minor” (tidak memiliki kesamaan karena tidak pernah muncul pada konteks dengan makna yang sama).

Begitu juga dengan nilai kemunculan “survey” dan “trees” pada dokumen m4. Kata “trees” tidak pernah muncul pada dokumen m4 ini, tetapi dengan pengurangan dimensi, nilai kemunculannya naik menjadi 0.66. Hal ini terjadi karena dokumen m3 yang sama-sama mengandung kata “graph” dan “minors” sebagaimana m4, mengandung kata “trees”. Sehingga meskipun pada m4 tidak terdapat kata “trees”, nilai kemunculan kata ini tinggi setelah pengurangan

dimensi. Nilai kemunculan 0.66 bisa diartikan sebagai estimasi berapa kali kata tersebut akan muncul pada dokumen yang juga mengandung kata “graph” dan “minors”. Berbeda dengan “survey” pada m4 yang nilai awalnya 1 berubah menjadi 0.42. Hal ini terjadi karena “survey” tidak diharapkan pada konteks ini dan dianggap tidak penting dalam menjelaskan karakteristik dokumen.

LSA tidak hanya menggunakan kemunculan kata pada dokumen yang sama sebagai informasi, tetapi juga menggunakan kata-kata yang tidak muncul dengan mengamati pola detail dari kemunculan kata-kata tersebut pada keseluruhan dokumen. LSA merepresentasikan makna kata sebagai semacam rata-rata dari semua dokumen di mana kata tersebut muncul, dan merepresentasikan makna dokumen sebagai semacam rata-rata dari semua makna kata yang terkandung di dalamnya (Landauer, Foltz, & Laham, 1998).

Representasi makna kata dan dokumen yg dibuat oleh LSA mampu mensimulasikan berbagai fenomena kognitif manusia seperti kategorisasi kata, pemahaman wacana, dan penilaian kualitas esai. Tetapi, untuk melakukan semua hal ini diperlukan korpus yang besar agar LSA bisa membangun ruang semantiknya.

Contohnya untuk kategorisasi kata, LSA diujicobakan untuk menjawab tes mencocokkan persamaan makna kata pada tes TOEFL (*Test of English as a Foreign Language*). Setiap soal terdiri dari satu kata atau frase dan peserta tes diminta untuk memilih satu dari empat pilihan kata/frase lain yang paling mirip maknanya. LSA menjawab benar sebanyak 64% dari 80 soal, sama dengan rata-rata hasil yang didapatkan oleh sampel peserta yang berasal dari negara yang tidak berbahasa Inggris. Sebelumnya, LSA dilatih dengan lima juta kata yang berasal dari ensiklopedia (Landauer, Foltz, & Laham, 1998).

Contoh lain untuk penilaian esai, pada penelitian mengenai penilaian esai tentang fungsi jantung yang diadakan di University of Colorado, diperlukan 27 artikel yang berkaitan dengan jantung dan sistem sirkulasi sebagai data pelatihan sistem. Begitu juga pada penelitian lain yang menilai esai tentang Psikologi, sistem dilatih dengan buku teks yang digunakan pada kuliah Psikologi (Landauer, et.al, 1997).

## 2.5 Perluasan Kata Kunci Pencarian (*Query Expansion*)

*Query expansion* adalah sebuah teknik yang digunakan pada mesin pencari di mana pengguna memberikan masukan kata-kata tambahan pada kata kunci pencarian (*query*). Hal ini dilakukan untuk meningkatkan *recall*, yaitu perbandingan antara jumlah dokumen relevan yang keluar dengan jumlah seluruh dokumen yang relevan (Manning, Raghavan, & Schutze, 2008).

Pertanyaan mendasar dalam melakukan *query expansion* adalah bagaimana cara membuat *query* baru yang sudah di-*expand*. Cara pertama adalah pengguna dapat memberikan tambahan kata-kata sendiri. Cara kedua adalah dengan analisis global menggunakan thesaurus, yaitu dengan menambahkan sinonim atau kata-kata yang berhubungan dengan kata-kata pada *query*. Thesaurus dapat diartikan sebagai kumpulan *term* atau istilah pada domain tertentu dan relasi antara istilah-istilah tersebut. Terdapat thesaurus untuk domain kesehatan, matematika, ilmu komputer, dan lain-lain (Chevallet, 2002). Relasi yang paling sering digunakan adalah sinonim.

Cara membuat thesaurus untuk *query expansion* antara lain sebagai berikut:

- a) Thesaurus manual. Editor manusia membuat daftar sinonim untuk setiap konsep secara manual.
- b) Thesaurus yang diturunkan secara otomatis. Thesaurus dibangun secara otomatis berdasarkan data statistik dari kemunculan kata pada dokumen dalam domain tertentu.

Selain itu, tentunya dapat digunakan thesaurus yang sudah ada. *Query expansion* berdasarkan thesaurus memiliki keuntungan, yaitu tidak memerlukan masukan dari pengguna.

## 2.6 Pemotongan Imbuhan (*Stemming*)

Berbagai bentuk kata seringkali digunakan untuk menyatakan suatu makna. Terdapat beberapa bentuk kata berbeda untuk makna yang serupa, misalnya demokrasi, demokratis, demokratisasi. Dalam beberapa situasi, akan sangat menguntungkan jika pencarian dokumen dengan satu kata kunci menghasilkan dokumen-dokumen yang mengandung kata tersebut dalam bentuk yang berbeda (Manning, Raghavan, & Schutze, 2008). Hal ini dapat dilakukan

dengan pemotongan imbuhan.

Pemotongan imbuhan atau *stemming* adalah teknik yang umum dilakukan pada proses perolehan informasi untuk menghilangkan variasi morfologi. Biasanya pemotong imbuhan terdiri dari sekumpulan aturan dan kamus. Penggunaan pemotong imbuhan dapat menimbulkan beberapa masalah antara lain sebagai berikut:

- a) Menyebabkan tidak diperolehnya dokumen yang relevan karena kurangnya konteks dari kata yang dipotong imbuhan.
- b) Terkadang terlalu banyak melakukan pemotongan. Contohnya "organization" menjadi "organ".
- c) Adanya pemotongan yang tidak dilakukan, misalnya "machinery" tidak dipotong menjadi "machine".
- d) Hasil pemotongan kadang sukar untuk diinterpretasikan karena bukan merupakan kata. Contohnya "general" menjadi "gener" (Adriani, 2008).

Pengaruh pemotong imbuhan terhadap unjuk kerja sistem perolehan informasi sangat beragam. Pada mesin pencari, penggunaan pemotong imbuhan ada yang dapat membantu meningkatkan jumlah dokumen pada suatu koleksi, tapi ada juga yang menurunkan jumlah dokumen yang diperoleh. (Adriani, 2008).

Untuk sistem penilai esai otomatis, salah satu penelitian menunjukkan bahwa pengaruh pemotong imbuhan saja dapat mengurangi korelasi nilai sistem dengan nilai dari manusia. Tetapi, kombinasi pemotong imbuhan dengan penghilangan kata-kata tidak penting (*stopword*) sedikit meningkatkan efektivitas sistem dibandingkan dengan hanya menghilangkan *stopword* saja (Wild, et.al, 2005). Penelitian lain oleh Nakov sebagaimana disebutkan dalam (Wild, et.al, 2005) menunjukkan bahwa penghilangan *stopword* meningkatkan efektivitas sistem secara keseluruhan sedangkan penggunaan pemotong imbuhan menyebabkan peningkatan signifikan hanya terdapat pada satu kasus saja.

## 2.7 Pembahasan

Penilaian jawaban esai pada penelitian ini menggunakan model perolehan informasi *Vector Space Model* yang biasanya digunakan untuk mesin pencarian

dan *Latent Semantic Analysis* yang bisa digunakan untuk mesin pencarian (*Latent Semantic Indexing*) dan juga sudah digunakan untuk menilai kualitas esai (IEA) dan menilai jawaban ujian esai (Ratna, Budiardjo, & Hartanto, 2007).

Penilaian jawaban esai yang dilakukan dalam penelitian ini adalah berdasarkan pencocokkan kata kunci sebagaimana yang dilakukan pada sistem penilaian jawaban esai otomatis untuk Bahasa Indonesia menggunakan LSA yang dilakukan (Hermawandi, 2008), (Octaria, 2008), dan (Harisma, 2008), yaitu mencocokkan kata-kata yang muncul pada kunci jawaban sebagai vektor *query* dan kumpulan jawaban esai siswa yang akan dinilai sebagai vektor dokumen. Kedekatan antara kedua vektor tersebut dihitung dengan *cosine similarity*. Proses ini dilakukan untuk model VSM dan juga LSA.

Sistem penilai jawaban esai berbasis LSA yang dikembangkan dalam penelitian ini direkonstruksi seperti sistem yang dikembangkan pada penelitian sebelumnya, sebagaimana yang dijelaskan dalam publikasi penelitiannya. Selain menggunakan cara yang sama dengan yang sudah dilakukan sebelumnya, penilaian LSA pada penelitian ini juga ditambah dengan dua skema penilaian LSA yang menggunakan data *training* yang berasal dari teks sesuai domain ujian esai. Rincian penambahan ini dijelaskan dalam Subbab 3.6.

Sistem penilai jawaban esai berbasis LSA dan VSM ini dilengkapi dengan perluasan kunci jawaban dan akan diujicobakan dengan dokumen yang mengalami proses pemotongan imbuhan. Perluasan kunci jawaban pada sistem ini menggunakan teknik *query expansion* pada mesin pencari. Hal ini diperlukan untuk mengakomodasi kemungkinan penggunaan kata yang berbeda untuk satu makna yang sama.

Perluasan kunci jawaban dilakukan dengan melihat ke daftar persamaan kata dan mencocokkan semua kata pada kunci jawaban dengan isi daftar ini. Jika persamaan kata terdapat pada daftar, maka kata ini ditambahkan ke kunci jawaban. Persamaan kata-kata pada kunci jawaban ini disusun dengan cara mengambil semua kata pada kunci jawaban dan mencari persamaan katanya dari WordNet Bahasa Indonesia yang sudah memiliki 1544 makna kata (Darma Putra, 2008).

Program pemotong imbuhan tidak diintegrasikan dengan sistem, tetapi

digunakan di luar sistem untuk memproses dokumen. Dokumen hasil pemotong imbuhan kemudian dimasukkan ke dalam sistem. Program pemotong imbuhan yang digunakan adalah pemotong imbuhan untuk Bahasa Indonesia yang dibuat oleh Ichsan dan Adriani (Ichsan & Adriani, 2008) dan pemotong imbuhan untuk Bahasa Inggris yaitu *Porter Stemmer*. Pemotong imbuhan Bahasa Inggris digunakan untuk mengatasi penggunaan istilah asing berbahasa Inggris dalam dokumen jawaban esai.

