

PENILAIAN ESAI JAWABAN BAHASA INDONESIA MENGGUNAKAN METODE SVM - LSA DENGAN FITUR GENERIK

Rama Adhitia dan Ayu Purwarianti

Program Studi Teknik Informatika, Institut Teknologi Bandung, Jalan Ganesha 10, Bandung, 40132, Indonesia

E-mail: ramaadhitia@gmail.com

Abstrak

Paper ini mengkaji sebuah solusi untuk permasalahan penilaian jawaban esai secara otomatis dengan menggabungkan *support vector machine* (SVM) sebagai teknik klasifikasi teks otomatis dengan LSA sebagai usaha untuk menangani sinonim dan polisemi antar *index* term. Berbeda dengan sistem penilaian esai yang biasa yakni fitur yang digunakan berupa *index* term, fitur yang digunakan proses penilaian jawaban esai adalah berupa fitur *generic* yang memungkinkan pengujian model penilaian esai untuk berbagai pertanyaan yang berbeda. Dengan menggunakan fitur *generic* ini, seseorang tidak perlu melakukan pelatihan ulang jika orang tersebut akan melakukan penilaian esai jawaban untuk beberapa pertanyaan. Fitur yang dimaksud meliputi persentase kemunculan kata kunci, *similarity* jawaban esai dengan jawaban referensi, persentase kemunculan gagasan kunci, persentase kemunculan gagasan salah, serta persentase kemunculan sinonim kata kunci. Hasil pengujian juga memperlihatkan bahwa metode yang diusulkan mempunyai tingkat akurasi penilaian yang lebih tinggi jika dibandingkan dengan metode lain seperti SVM atau LSA menggunakan *index* term sebagai fitur pembelajaran mesin.

Kata Kunci: *support vector machine, latent semantic analysis, penilaian jawaban esai secara otomatis, fitur generik, jawaban esai berbahasa Indonesia*

Abstract

This paper examines a solution for problems of assessment an essay answers automatically by combining support vector machine (SVM) as automatic text classification techniques and LSA as an attempt to deal with synonyms and the polysemy between index terms. Unlike the usual essay scoring system that used index terms features, the feature used for the essay answers assessment process is a generic feature which allows testing of valuation models essays for a variety of different questions. By using these generic features, one does not need to re training if the person will conduct an assessment essay answers to some questions. The features include percentage of keywords, similarity essay answers with the answer reference, percentage of key ideas, percentage of wrong answer, and percentage of keyword synonyms. The test results also show that the proposed method has a higher valuation accuracy rate compared to other methods such as SVM or LSA, use term index as features in machine learning.

Keywords: *support vector machine, latent semantic analysis, assessment an essay answers automatically, generic feature, essay answer in Bahasa Indonesia*

1. Pendahuluan

Esai dianggap oleh banyak ahli sebagai metode atau alat yang paling tepat untuk menilai hasil dari kegiatan belajar yang kompleks, karena penulisan esai akan melibatkan kemampuan siswa untuk mengingat, mengorganisasikan, mengekspresikan, dan mengintegrasikan gagasan yang dimiliki oleh siswa tersebut [1]. Namun ternyata esai menimbulkan masalah baru yaitu pemeriksaan esai membutuhkan waktu lebih lama dibandingkan waktu yang diperlukan untuk memeriksa jawaban dari pertanyaan bertipe

objektif seperti *multiple choice* dan *short answer*. Selain itu, sifat subjektif dari esai dapat mengakibatkan nilai yang diberikan antara satu penilai manusia dengan penilai manusia yang lain pada esai yang sama mungkin saja berbeda, dan tentu saja hal ini merupakan masalah yang harus dipecahkan.

Berdasarkan penjelasan di atas maka dibutuhkan sebuah sistem penilaian esai otomatis untuk menjawab permasalahan penilaian esai ini. Pada kenyataannya sudah terdapat beberapa sistem penilaian esai secara otomatis baik yang bersifat komersial maupun yang berasal dari

penelitian yang dilakukan oleh para ahli di bidang ini. Beberapa sistem yang dimaksud adalah: PEG (*Project Essay Grade*), *Intelligent Essay Assessor*, *e-Rater*, *Betsy (Bayesian Essay Test Scoring System)*, dan masih ada beberapa sistem lain [1][2]. *E-rater* bahkan telah digunakan untuk melakukan penilaian esai pada pelaksanaan *General Management Aptitude Test (GMAT)* dengan tingkat *similarity* antara nilai yang diberikan *e-rater* dan nilai yang diberikan oleh penilai manusia mencapai 94%. *Similarity* yang terdapat diantara banyak sistem penilaian esai secara otomatis ini ialah bahwa persoalan penilaian esai secara otomatis dianggap sebagai masalah pengelompokan teks secara otomatis yang dapat diselesaikan dengan metode pembelajaran mesin [1].

Pengelompokan teks secara otomatis bertujuan untuk mengklasifikasikan teks-teks (dapat berupa paragraf atau dokumen) ke dalam kategori-kategori yang telah ditentukan sebelumnya [3]. Telah banyak teknik pembelajaran mesin yang dapat diaplikasikan untuk melakukan pengelompokan teks secara otomatis seperti *naive Bayes text classification*, *support vector machines*, dan *K-nearest neighbors classifiers*. Salah satu aspek yang umum terdapat pada pendekatan-pendekatan di atas ialah bahwa masalah pengelompokan teks secara otomatis diperlakukan sebagai masalah pengklasifikasian biasa yang terdiri dari dua proses yaitu: perancangan fitur dan *classification learning*. Dari kedua proses ini, perancangan fitur akan sangat menentukan performa suatu solusi dari masalah pengelompokan teks secara otomatis [4].

Support vector machine (SVM) merupakan sebuah metode pembelajaran mesin yang dapat digunakan untuk melakukan klasifikasi teks secara otomatis. Aplikasi SVM untuk pengklasifikasian teks secara otomatis dirintis oleh Thorsten Joachim dengan menggunakan bobot *index* term sebagai fitur [5]. Penilaian esai menggunakan SVM dapat dilakukan dengan menggunakan sejumlah fitur esai. Fitur esai ini dapat berupa kemunculan suatu frase atau kata tertentu pada esai, kemunculan pasangan kata benda-kata kerja tertentu, bagaimana urutan suatu konsep ditulis pada esai dan kriteria penilaian esai lainnya.

Selain dengan menggunakan metode pembelajaran mesin seperti yang telah dijelaskan sebelumnya, penilaian esai secara otomatis juga dapat dilakukan dengan menggunakan metode *latent semantic analysis (LSA)* yang pada awalnya digunakan untuk melakukan *indexing* pada sistem temu balik informasi [6]. LSA

merupakan teknik matematika yang digunakan untuk mengekstrak dan merepresentasikan *similarity* arti kata dengan cara menganalisis sejumlah besar teks. LSA bekerja dengan cara membentuk matriks term *by document* yang berasal dari sejumlah besar *training essay*. Matriks ini kemudian akan mengalami dekomposisi dan reduksi dimensi, menjadi matriks yang merepresentasikan ruang semantik dari *training essay*. Penilaian esai dengan LSA dilakukan dengan mengukur *similarity* antara dokumen esai yang ingin dinilai dengan sejumlah esai pembanding. *Similarity* antara esai dengan dokumen pembanding diukur dengan cara mengukur *similarity cosine* antara vektor dokumen esai yang ingin dinilai dengan vektor dokumen esai pembanding yang dibentuk berdasarkan matriks ruang semantik [7]. Semakin tinggi tingkat *similarity* esai yang ingin dinilai dengan esai pembanding, semakin tinggi nilai yang diberikan kepada esai tersebut.

Meskipun sistem penilaian esai secara otomatis mempunyai kelemahan yaitu berupa *similarity* antara nilai yang diberikan sistem dengan nilai yang diberikan oleh penilai manusia yang masih sulit dicapai, sistem ini masih akan membantu penilaian dalam kelas yang mempunyai jumlah siswa banyak dalam hal mengurangi waktu yang dibutuhkan untuk melakukan penilaian [2]. Kelebihan lain dari sistem penilaian esai secara otomatis ini ialah konsistensi penilaiannya.

Support vector machine (SVM) merupakan sebuah metode pembelajaran mesin yang dipopulerkan oleh Boser, Guyon, dan Vapnik pada tahun 1992 [8]. Proses pembelajaran pada SVM bertujuan untuk mendapatkan hipotesis berupa bidang pemisah (*hyperplane*) terbaik yang tidak hanya meminimalkan *empirical risk* yaitu rata-rata *error* pada data pelatihan, tetapi juga memiliki generalisasi yang baik.

Generalisasi adalah kemampuan sebuah hipotesis untuk mengklasifikasikan data yang tidak terdapat dalam data pelatihan dengan benar. Untuk menjamin generalisasi ini, SVM bekerja berdasarkan prinsip *structural risk minimization (SRM)* [9]. SRM pada SVM digunakan untuk menjamin batas atas dari generalisasi pada data pengujian dengan cara mengontrol "kapasitas" (fleksibilitas) dari hipotesis hasil pembelajaran.

Masalah pencarian *hyperplane* terbaik pada SVM dapat dirumuskan menjadi sebuah masalah optimasi yang dapat dilihat pada persamaan (1).

$$\min \frac{1}{2} |w|^2 + c \left(\sum_{i=1}^n \varepsilon_i \right) \quad (1)$$

Subject to

$$\begin{aligned} y_i(wx_i + b) &\geq 1 - \varepsilon_i \\ \varepsilon_i &\geq 0 \end{aligned}$$

dengan fungsi keputusan yang dihasilkan dari proses *training* dapat dirumuskan dengan formula seperti pada persamaan (2).

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i K(x_i, x_d) + b \quad (2)$$

Fungsi keputusan pada SVM memanfaatkan sebuah fungsi kernel $K(x_i, x_d)$. Adapun beberapa fungsi kernel yang umum digunakan dapat dilihat pada persamaan (3), (4), (5), dan (6) [10].

Linear:

$$K(X_i, X_j) = X_i^T X_j \quad (3)$$

Polynomial:

$$K(X_i, X_j) = (\gamma X_i^T X_j + r)^d, \gamma > 0 \quad (4)$$

Radial basis function (RBF):

$$K(X_i, X_j) = \exp(-\gamma \|X_i - X_j\|^2), \gamma > 0 \quad (5)$$

Sigmoid:

$$K(X_i, X_j) = \tanh \gamma X_i^T X_j + r \quad (6)$$

SVM pada awalnya hanya dirancang untuk dapat melakukan klasifikasi biner. Namun pada saat ini SVM dapat digunakan untuk melakukan klasifikasi ke dalam banyak kelas yaitu dengan menggunakan *multi class SVM*. Adapun metode-metode yang umum digunakan untuk mengimplementasikan *multiclass SVM* adalah metode *one-against all* dan *one-against one*.

SVM pada sistem penilaian, Markus Weimer et al telah melakukan eksperimen tentang bagaimana secara otomatis melakukan penilaian kualitas atau memberi peringkat kepada *post* yang ada sebuah forum diskusi tentang perangkat lunak *online* [11]. Terdapat perbedaan mendasar diantara penilaian esai dengan penilaian kualitas *post* yaitu pada penilaian esai biasanya sudah terdapat *guideline* tentang bagaimana cara menilai suatu esai, sedangkan ketika seorang anggota

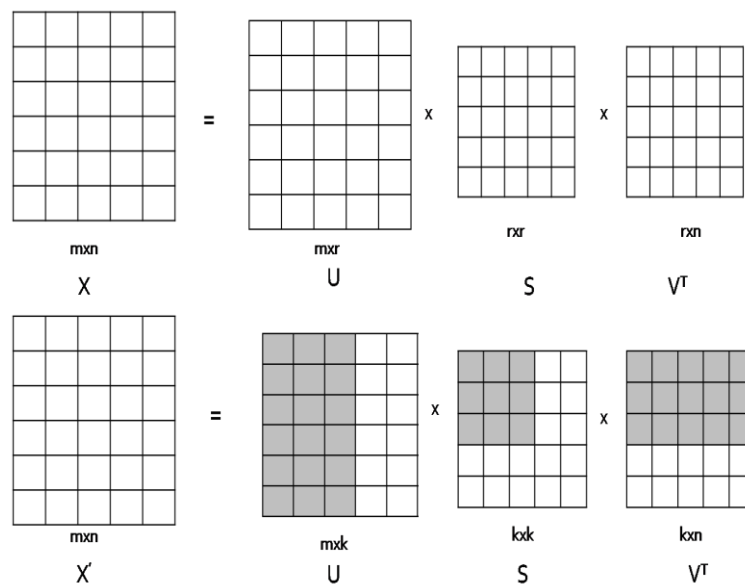
forum ingin memberikan peringkat kepada suatu *post*, setiap anggota akan mempunyai kriteria kualitas yang berbeda-beda.

Pada eksperimen ini digunakan teknik klasifikasi *support vector machine* sebagai *binary classifier* yang akan membagi *post* ke dalam dua kelompok, yaitu kelompok *post* dengan kualitas baik dan kelompok dengan kualitas *post* buruk. Fungsi kernel yang digunakan pada eksperimen ini ialah *radial basis function*. *Feature vector* pada eksperimen ini disusun dari lima buah kelas *feature* yang terdiri dari: (a) *Surface features* yang terdiri dari panjang *post*, jumlah tanda tanya, jumlah tanda seru, jumlah kata yang ditulis dengan huruf kapital; (b) *Lexical features* yang terdiri dari jumlah kesalahan pengejaan kata, jumlah kata umpatan yang terdapat pada *post*; (c) *Syntactic features* yang terdiri dari persentase dari *part of speech tags* sebagaimana yang telah didefinisikan pada *PENN Treebank Tag Set*; (d) *Forum specific features* yang terdiri dari: kandungan *tag html* pada *post*, apakah *post* merupakan hasil salinan dari *mailing list*, jumlah URL, dan *filesystem path*; dan (e) *Similarity Features*: membandingkan tingkat *similarity* antara suatu *post* dengan *topic thread* tempat *post* tersebut berada. Tingkat *similarity* ini diukur dengan menggunakan *cosine similarity* antara *post* dan *topic thread*.

Pengujian terhadap sistem dilakukan dengan menggunakan *test set* yang telah diketahui klasifikasi sebelumnya. Hasil eksperimen memperlihatkan bahwa sistem penilaian *post* yang dibangun mempunyai tingkat akurasi sebesar 89.1 %.

Sistem Penilaian Esai Menggunakan *Latent Semantic Analysis*, Teori LSA. *Latent Semantic Analysis* (LSA) merupakan sebuah metode statistik yang dapat digunakan untuk menentukan dan merepresentasikan *similarity* makna dari kata-kata dan teks dengan cara melakukan analisis terhadap teks dalam jumlah yang besar [12]. LSA pada awalnya digunakan untuk melakukan *indexing* pada bidang sistem temu balik informasi.

LSA menggunakan metode aljabar linear yaitu *singular value decomposition* (SVD). LSA akan membentuk matriks yang merepresentasikan asosiasi antara term-dokumen yang merupakan *semantic space*, yakni kata-kata dan dokumen-dokumen yang berasosiasi dekat akan ditempatkan dekat satu sama lain [6]. Hasil yang didapat ialah kata-kata yang tidak terdapat pada suatu dokumen mungkin saja letaknya pada *semantic space* masih dekat dengan dokumen tersebut apabila hal ini konsisten dengan *major associative pattern* yang terdapat pada data.



Gambar. 1. Proses reduksi dimensi pada LSA.

Teknik *singular value decomposition* yang digunakan pada LSA ialah *reduced SVD*, yaitu akan dilakukan proses pengurangan dimensi (*dimensionality reduction*) pada matriks hasil dekomposisi SVD. *Reduced SVD* pada LSA dapat dijelaskan dengan lebih detail sebagai berikut [13]: Setiap matriks, misalnya matriks yang berukuran $t \times d$, X , seperti matriks term \times dokumen, dapat didekomposisi ke dalam bentuk persamaan 7:

$$X = USV^T \quad (7)$$

U dan V^T merupakan matriks dari vektor singular kanan dan vektor singular kiri. Sedangkan S merupakan matriks diagonal yang berisikan nilai-nilai singular.

Langkah selanjutnya dari SVD ialah membentuk aproksimasi dari matriks X , X' dengan melakukan pengurangan dimensi. Jika nilai-nilai singular dari matriks S diurutkan berdasarkan nilainya, maka k nilai terbesar dapat tetap disimpan dalam matriks tersebut, dan nilai-nilai lain yang lebih kecil dapat diset menjadi nol. Jika baris dan kolom yang berkaitan pada matriks U dan V^T juga di set menjadi 0, maka hasil kali dari ketiga matriks ini akan membentuk matriks X' yang merupakan matriks *least square approximation* dari matriks X . Gambar 1 merupakan ilustrasi pembentukan matriks X' dengan menggunakan dimensi sebesar k .

Dari matriks-matriks hasil dekomposisi SVD terdapat tiga operasi perbandingan yang dapat dilakukan yaitu [6]:

Membandingkan seberapa besar similarity diantara dua term yang berbeda. Koordinat dari suatu term pada *semantic space* direpresentasikan oleh vektor baris yang dari matriks $U \times S$ yang bersesuaian dengan term tersebut. Oleh karena itu *similarity* antara dua term yang berbeda dapat diperoleh dari *cosine similarity* antara koordinat-koordinat dari kedua term tersebut.

Membandingkan seberapa besar similarity diantara dua dokumen yang berbeda. Koordinat suatu dokumen pada *semantic space* direpresentasikan oleh vektor baris dari matriks $V \times S$ yang bersesuaian dengan dokumen tersebut. Oleh karena itu *similarity* antara dua dokumen yang berbeda dapat diperoleh dari *cosine similarity* antara koordinat-koordinat dari kedua dokumen tersebut.

Mengetahui seberapa besar suatu term tertentu berasosiasi dengan suatu dokumen. Berbeda dari dua operasi sebelumnya yang memerlukan penghitungan *cosine similarity*, seberapa besar asosiasi antara suatu kata i dengan suatu dokumen j , dapat diketahui dari nilai *cell* i, j dari matriks aproksimasi *term-document* yang dihasilkan oleh SVD.

Dokumen yang tidak muncul pada matriks *term-document*, juga dapat direpresentasikan sebagai sebuah *pseudodocument* dalam *semantic space*. Untuk dapat merepresentasikan dokumen eksternal ini sebagai *pseudodocument* pada *semantic space*, maka dapat dilakukan dengan menggunakan persamaan 8.

$$\hat{D} = DUS^{-1} \quad (8)$$

dimana \hat{D} adalah representasi *pseudodocument* pada *semantic space*, dan D adalah vektor bobot term dari dokumen.

LSA pada sistem penilaian esai, anak Agung Putri Ratna et al telah mengembangkan sebuah sistem *grading* esai otomatis dengan memberikan pembobotan kata pada kata kunci yang dianggap penting. Sistem ini diujicobakan pada ujian *online* dengan kelas kecil dan kelas menengah dengan menggunakan metode *latent semantic analysis* (LSA). Sistem penilaian esai ini diberi nama Sistem Penilaian Esai Otomatis (SIMPLE) [14].

SIMPLE terdiri dari dua buah modul yaitu modul untuk pengajar dan modul untuk siswa. Modul untuk pengajar berfungsi menanyakan kata kunci kepada pengajar. Modul siswa berfungsi untuk menerima jawaban dari esai dan kemudian memberikan nilai kepada esai siswa. Jawaban dari siswa akan diperiksa apakah ada kata kunci atau kata bobot yang digunakan, yang kemudian kalimat jawaban diterjemahkan ke dalam bentuk matriks. Matriks yang dibentuk dari jawaban siswa dan jawaban referensi akan dicari nilai normalisasinya. Nilai siswa tersebut didapat dengan membandingkan nilai normalisasi matriks dari jawaban siswa dan matriks dari jawaban referensi.

Pada hasil pengujian yang telah dilakukan, telah diambil kesimpulan bahwa SIMPLE mempunyai performa yang baik dalam penilaian esai dalam Bahasa Indonesia. Pada percobaan yang dilakukan untuk menguji SIMPLE, tingkat akurasi untuk kelas kecil berkisar antara 69.8% sampai dengan 94.64%. Sedangkan untuk kelas menengah tingkat akurasi berkisar antara 77.18% sampai dengan 98.42%.

Thomas Landauer et al juga telah melakukan studi tentang penggunaan LSA untuk melakukan penilaian esai. Pada studi yang dilakukan oleh Landauer et al ini, *semantic space* yang digunakan dibangun dari beberapa buku pegangan yang wajib dibaca siswa, serta dari jawaban esai siswa. Terdapat tiga buah metode yang telah diuji coba untuk melakukan penilaian esai yaitu yang pertama, sekumpulan contoh jawaban esai dinilai terlebih dahulu (*pre-graded essay*) oleh penilai. Kemudian dilanjutkan dengan penghitungan *cosine similarity* antara jawaban esai yang akan dinilai dan sekumpulan jawaban esai yang tadi telah dinilai terlebih dahulu. Jawaban esai siswa akan diberi nilai berupa nilai rata-rata dari *pre-graded essay* yang mempunyai tingkat kesamaan

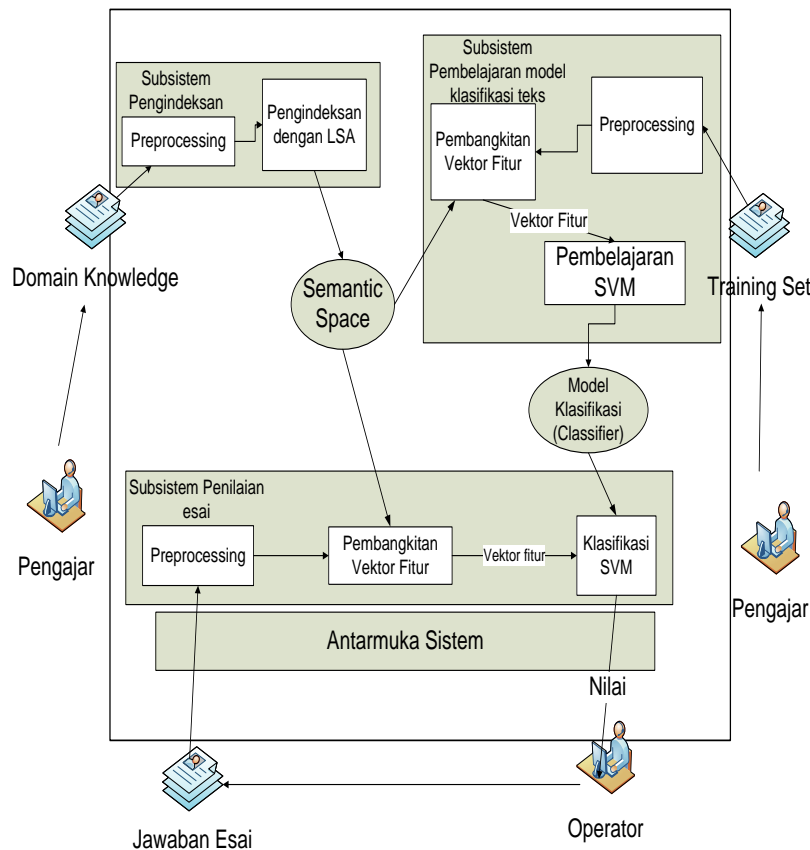
relatif tinggi dengan dengan jawaban yang akan dinilai.

Kedua, dibutuhkan sebuah *golden essay* yang ditulis oleh pengajar dan dijadikan sebagai acuan jawaban esai yang benar. Nilai dari jawaban esai siswa diberikan berupa nilai *cosine similarity* antara jawaban esai siswa tersebut dengan *golden essay*. Ketiga, dilakukan penghitungan *cosine similarity* antara setiap kalimat dari dokumen yang berisikan materi pelajaran, dengan kalimat yang berasal dari jawaban esai siswa. Nilai akhir jawaban esai siswa akan dihitung dari jumlah kumulatif nilai *cosine similarity*. Varian lain dari metode ketiga ini ialah dengan menghitung nilai *cosine similarity* maksimum antara setiap kalimat dari sekumpulan kalimat yang menurut pengajar adalah penting, dengan setiap kalimat pada jawaban esai siswa.

Menggabungkan SVM dan LSA pada Sistem Penilaian Esai. Masalah penilaian esai secara otomatis dapat dilihat sebagai suatu masalah pengklasifikasian teks secara otomatis. Jika dipandang dari sudut pandang pengklasifikasian teks secara otomatis, masalah ini dapat didefinisikan sebagai masalah pengklasifikasian suatu esai ke dalam beberapa kategori nilai.

Karakteristik khusus yang terdapat pada masalah penilaian esai adalah jawaban siswa tidak dapat dinilai dengan cara membandingkan apakah jawaban siswa tersebut persis sama dengan jawaban referensi (*golden answer*) yang telah didefinisikan sebelumnya. Sebuah pertanyaan bertipe esai dapat dijawab benar dengan banyak kemungkinan jawaban, asalkan jawaban yang diberikan siswa mengandung gagasan-gagasan yang sama dengan gagasan-gagasan yang terkandung pada jawaban referensi. Variasi jawaban ini didapat karena adanya faktor *synonymity* dan *polysemy* sehingga memungkinkan siswa untuk mengungkapkan gagasan yang sama dengan gagasan yang terdapat pada jawaban referensi namun dengan menggunakan kata-kata yang berbeda.

Melihat berbagai pertimbangan di atas maka ide utama yang akan digunakan untuk menyelesaikan masalah penilaian jawaban esai secara otomatis adalah dengan menggabungkan metode SVM dengan metode LSA. SVM akan digunakan untuk melakukan proses *training* dan proses pengelompokan jawaban esai ke dalam label nilai tertentu. LSA digunakan sebagai solusi terhadap masalah *polysemy* dan *synonymity*, dimana LSA akan diaplikasikan pada proses ekstraksi fitur dari jawaban esai.



Gambar 2. Arsitektur sistem penilaian esai menggunakan SVM dan LSA.

Berikut adalah gambaran lengkap mengenai arsitektur sistem penilaian esai otomatis dengan menggunakan SVM dan LSA. Sebagaimana yang terlihat pada gambar 2, sistem penilaian esai otomatis yang diusulkan terdiri dari tiga buah subsistem yaitu: (1) Subsistem pengindeksan, yang akan menghasilkan *file* indeks yang berisikan matriks *semantic space*; (2) Subsistem pembelajaran model klasifikasi teks, yang melakukan proses pembelajaran *learning* terhadap *training set* untuk kemudian menghasilkan model klasifikasi (*classifier*); dan (3) Subsistem penilaian esai, yang akan melakukan penilaian terhadap jawaban esai siswa dan kemudian memberikan nilai kepada siswa.

Pada sistem penilaian esai otomatis ini juga dideskripsikan dua buah role pengguna yaitu: Pengajar. Pengguna dengan role pengajar akan menggunakan sistem penilaian esai otomatis ini khususnya pada subsistem pengindeksan dan subsistem pembelajaran model klasifikasi teks. Alasannya adalah karena pada subsistem pengindeksan diperlukan pengetahuan tentang materi yang diujikan pada pertanyaan esai untuk penentuan dokumen-dokumen yang akan menjadi domain knowledge dan juga karena diperlukan pendefinisian kelompok kategori nilai serta

penentuan training set yang akan digunakan pada subsistem pembelajaran model klasifikasi teks. *Operator*. Pengguna dengan role operator hanya dapat menggunakan sistem penilaian esai otomatis ini pada subsistem penilaian esai. Alasannya adalah karena pada subsistem penilaian esai tidak dibutuhkan pengetahuan sebagaimana yang dibutuhkan pada dua subsistem lainnya.

Sistem penilaian esai otomatis menggunakan metode yang diusulkan menerima tiga jenis dokumen masukan yaitu yang pertama adalah *Domain knowledge*, yang merupakan komponen pembangun dari *semantic space*. Dari *domain knowledge* LSA dapat menentukan tingkat kepentingan kata serta hubungan *higher order occurrence* yang terdapat pada suatu domain permasalahan. *Domain knowledge* akan diambil dari dokumen-dokumen yang berisikan domain pengetahuan dari jawaban dan pertanyaan esai. Misalnya seorang pengajar akan mengujikan sebuah pertanyaan tentang masalah sistem pakar, maka *domain knowledge* dapat berupa isi buku-buku, artikel-artikel, ataupun jurnal tentang masalah sistem pakar.

Kedua, *training data*, berguna untuk membangun model klasifikasi menggunakan metode SVM. Pada sistem penilaian esai otomatis

ini pendefinisian kategori nilai akan ditentukan oleh *training set* yang akan digunakan. Bila pada *training set* yang digunakan terdapat contoh-contoh jawaban esai untuk n kategori nilai, maka subsistem penilaian esai akan melakukan penilaian esai ke dalam salah satu nilai dari n kategori nilai yang ada pada *training set*. Oleh karena itu jika pada subsistem penilaian esai diinginkan jawaban esai dinilai dengan menggunakan n kategori nilai, maka pengajar harus dapat menyediakan *training set* yang didalamnya terdapat contoh-contoh jawaban esai dalam n kategori nilai tersebut. Ketiga, jawaban esai yang akan dinilai yang berasal dari jawaban ujian siswa. Jawaban ini nantinya akan diproses oleh subsistem penilaian.

Pada gambar juga terlihat proses pembangkitan fitur yang mana memegang peran penting bagi kinerja sistem secara keseluruhan. Berbeda dengan pendekatan umum dimana vektor fitur disusun dari term yang terdapat pada esai, pada penelitian ini vektor fitur terdiri dari beberapa informasi yang mendeskripsikan hubungan antara jawaban esai siswa dengan jawaban esai referensi (*golden essay*). Informasi-informasi yang terkandung pada vektor fitur dapat dijelaskan secara lebih rinci sebagai berikut:

Persentase kemunculan kata kunci

Fitur persentase kemunculan kata kunci ini adalah jumlah kemunculan kata kunci pada jawaban esai per jumlah kata kunci yang terdefinisi. Semakin banyak kemunculan kata kunci pada jawaban, berarti semakin besar peluang jawaban esai ini untuk mendapatkan nilai tinggi.

Sebagai contoh dengan jika terdapat kata kunci terdefinisi *working, memory, dinamis, basis, pengetahuan, statik*, dan misalkan terdapat jawaban esai yaitu *isi working memory akan berubah terus menerus selama proses inferensi berlangsung, sedangkan isi basis pengetahuan hanya dapat diubah oleh knowledge engineer*; maka persentase kata kunci adalah sebesar 0.6.

Similarity antara jawaban esai siswa dengan jawaban referensi (golden answer)

Fitur *similarity* dengan jawaban referensi ini ialah fitur yang menggambarkan seberapa besar kesamaan semantik jawaban esai siswa dengan semantik jawaban referensi yang telah didefinisikan sebelumnya. *Similarity* antara jawaban siswa dengan jawaban referensi dapat diketahui dengan menghitung sudut antara vektor bobot jawaban esai siswa dengan vektor bobot jawaban referensi pada *semantic space* yang dihasilkan oleh LSA.

Persentase kemunculan gagasan kunci pada jawaban esai

Fitur persentase kemunculan gagasan kunci ini ialah fitur yang menggambarkan kemunculan gagasan kunci yang seharusnya muncul pada jawaban siswa dibagi jumlah gagasan kunci yang terdefinisi. Gagasan kunci ini adalah gagasan penting yang menjadi inti dari jawaban yang benar (jawaban referensi).

Dalam pemeriksaan kemunculan gagasan kunci diperlukan sekumpulan kalimat gagasan kunci yang sebelumnya telah didefinisikan oleh penguji. Penghitungan fitur dimulai dengan mengubah setiap gagasan kunci dan setiap kalimat pada jawaban ke dalam vektor-vektor pada *semantic space*. Untuk setiap gagasan kunci yang terdefinisi, hitung besar *cosine similarity* gagasan kunci tersebut dengan setiap kalimat pada jawaban. Jika terdapat besar *cosine similarity* yang melebihi suatu *threshold* maka diasumsikan gagasan kunci tersebut muncul pada jawaban. Nilai fitur adalah jumlah kemunculan gagasan kunci dibagi dengan jumlah gagasan kunci yang terdefinisi.

Persentase kemunculan gagasan yang salah pada jawaban esai

Seringkali pada kegiatan belajar mengajar, para siswa mempunyai pemahaman yang salah terhadap materi yang diajarkan. Tentu saja apabila pemahaman yang salah ini muncul pada jawaban esai, seharusnya akan mengurangi nilai dari jawaban esai. Oleh karena itu persentase kemunculan gagasan yang salah akan digunakan sebagai fitur menggambarkan apakah terdapat gagasan yang salah pada jawaban esai dan seberapa banyak gagasan yang salah terdapat pada jawaban. Adapun cara penghitungan nilai fitur ini sama dengan cara penghitungan fitur sebelumnya, namun menggunakan gagasan salah yang terdefinisi.

Persentase kemunculan sinonim kata kunci

Fitur ini dihitung dengan membandingkan kata kunci terdefinisi yang tidak terdapat jawaban esai dengan kata-kata pada jawaban. Perbandingan ini dilakukan dengan mengukur besar *cosine similarity* antara vektor *semantic space* kata kunci dengan vektor *semantic space* kata yang terdapat pada jawaban.

3. Analisis dan Pembahasan

Pengujian dilakukan dengan menggunakan sebuah perangkat lunak sistem penilaian esai otomatis yang dibangun dengan menggunakan .NET *framework*, pustaka SVM.NET untuk mengimplementasikan metode *support vector machine* serta pustaka ALGLIB untuk mengimplementasikan metode *latent semantic*

analysis.

Data Pengujian. *Data set* yang digunakan untuk eksperimen berasal dari jawaban pada ujian tengah semester tahun ajaran 2008/2009 mata kuliah Inteligencia Buatan pada program studi Teknik Informatika, Institut Teknologi Bandung. *Data set* ini kemudian diketik ulang ke dalam format file ANSI text file (*.txt). *Data set* terdiri dari jawaban-jawaban atas dua pertanyaan yaitu: (1) "Jelaskan mengapa *working memory* dikatakan bersifat dinamis dan basis pengetahuan bersifat statis!". (2) "Jelaskan perbedaan antara *forward chaining* dengan *backward chaining*".

Jawaban dinilai dengan nilai diskrit dengan rentang nilai 1 – 3 yang berarti terdapat tiga kelas klasifikasi. Jumlah jawaban untuk masing-masing pertanyaan dapat dilihat pada tabel I.

TABEL I
STATISTIK DARI DATA SET

Question Id	Number of Answers		
	Grade 1	Grade 2	Grade 3
1	4	21	75
2	6	30	54

Data set dari pertanyaan kedua digunakan sebagai *training data*, sedangkan data set dari pertanyaan pertama digunakan sebagai data uji. Sedangkan domain *knowledge* dibentuk dari 100 dokumen tentang sistem pakar yang dikumpulkan secara manual dari artikel-artikel internet dan buku.

Hasil pengujian, terdapat empat skenario pengujian yang dilakukan pada penelitian ini yaitu *pertama*, pengujian sistem penilaian esai otomatis dengan menggunakan metode SVM & LSA dengan *the proposed method of SVM & LSA* dengan menggunakan fitur-fitur yang telah dijelaskan sebelumnya. Terdapat beberapa parameter yang digunakan, seperti dimensi dari *semantic space* dan *threshold* dari *cosine similarity* untuk menghitung fitur persentase kemunculan sinonim kata kunci, persentase kemunculan gagasan kunci, dan persentase kemunculan gagasan salah. *Kedua*, pengujian sistem penilaian esai menggunakan metode yang digunakan oleh Landauer yaitu dengan menghitung nilai *cosine similarity* antara jawaban esai dengan jawaban referensi (*golden answer*). Terdapat empat buah *threshold*, dua buah untuk masing-masing pertanyaan.

Ketiga, pengujian sistem penilaian esai menggunakan SVM & LSA dengan *index term* sebagai vektor fitur. Sistem ini menggunakan fitur yang umum digunakan pada penilaian esai yaitu *index term* sebagai vektor fitur. Pengujian ini dilakukan untuk melihat dampak penggunaan vektor fitur baru pada skenario pengujian pertama. Keempat, pengujian sistem penilaian esai

menggunakan SVM dengan *index term* sebagai vektor fitur. Pengujian ini dilakukan untuk melihat dampak penggunaan LSA pada penilaian esai.

Hasil pengujian dari skenario pengujian pertama dapat dilihat pada tabel II. Tingkat akurasi dari metode yang diusulkan adalah sekitar 77% dengan tingkat akurasi tertinggi mencapai 78%. Hasil pengujian untuk skenario pengujian kedua dapat dilihat pada tabel III. Tabel III memperlihatkan bahwa akurasi sistem penilaian esai yang menggunakan metode LSA seperti yang digunakan oleh Landauer lebih rendah dari tingkat akurasi metode yang diusulkan. Hasil pengujian untuk skenario ketiga dapat dilihat pada tabel IV.

TABEL II
HASIL EKSPERIMEN SISTEM PENILAIAN ESAI DENGAN
MENGUNAKAN METODE YANG DIUSULKAN

Dimension	LSA Threshold of Cosine Similarity	SVM Parameter		Accuracy (%)	Error Number		
		C	γ		3	2	1
25	0.5	8	64	78	0	19	3
	0.6	8	64	76	0	21	3
	0.7	8	128	76	0	21	3
50	0.5	8	64	77	1	19	3
	0.6	8	128	78	1	18	3
	0.7	8	512	76	0	21	3
75	0.5	8	64	77	2	18	3
	0.6	8	64	76	1	20	3
	0.7	8	32	77	2	18	3

TABEL III
HASIL EKSPERIMEN SISTEM PENILAIAN ESAI MENGGUNAKAN
LSA DENGAN INDEX TERM SEBAGAI VEKTOR FITUR

Dimension	LSA	Accuracy (%)	Error Number		
			3	2	1
<i>For 1st question</i>					
25		67	15	15	3
50		69	15	14	2
75		68	14	16	2
<i>For 2nd question</i>					
25		61.11	0	30	5
50		55.55	7	28	5
75		55.55	11	26	3

TABEL IV
HASIL EKSPERIMEN SISTEM PENILAIAN ESAI MENGGUNAKAN
SVM & LSA DENGAN INDEX TERM SEBAGAI VEKTOR FITUR
(10 FOLD CROSS VALIDATION)

Dimension	Average Accuracy (%)
25	72.6%
50	73%
75	73%

TABEL V
HASIL EKSPERIMEN SISTEM PENILAIAN ESAI MENGGUNAKAN
SVM DENGAN INDEX TERM SEBAGAI VEKTOR FITUR (10 FOLD
CROSS VALIDATION)

Question ID	Average Accuracy (%)
1	73%
2	69.4%

Tabel IV memperlihatkan bahwa metode yang diusulkan (tabel II) dengan vektor fitur baru

mencapai tingkat akurasi yang lebih tinggi dibandingkan *index* term sebagai vektor fitur. Sedangkan untuk skenario pengujian keempat (tabel V), tingkat akurasi tertinggi yang didapat mencapai 73%. Hal ini menunjukkan bahwa LSA tidak memberikan dampak peningkatan performansi yang signifikan untuk sistem penilaian esai menggunakan SVM dengan *index* term sebagai vektor fitur. Untuk melihat tingkat kepentingan fitur, untuk skenario pengujian pertama juga dilakukan pengujian dengan mengurangi jenis fitur yang digunakan. Hasil pengujiannya dapat dilihat pada tabel VI.

Tabel VI memperlihatkan bahwa SVM & LSA dengan menggunakan fitur *similarity* dengan jawaban referensi, persentase kemunculan kata kunci, dan persentase kemunculan gagasan salah dapat mencapai tingkat akurasi sebesar 79%.

TABEL VI
HASIL EKSPERIMEN PENGURANGAN FITUR PADA SISTEM
PENILAIAN ESAI DENGAN MENGGUNAKAN METODE YANG
DIUSULKAN

Reduced Features	Accuracy (%)	Error Number		
		3	2	1
<i>Similarity with golden essay</i>	76	1	20	3
<i>Key idea</i>	76	3	18	3
<i>Incorrect idea</i>	76	3	18	3
<i>Keyword</i>	72	5	20	3
<i>Synonym occurrence</i>	77	1	19	3
<i>Synonym occurrence, incorrect idea</i>	65	20	12	3
<i>Synonym occurrence, key idea</i>	79	6	13	2
<i>Synonym occurrence, keyword occurrence</i>	72	11	15	2
<i>Synonym occurrence, similarity with golden essay</i>	73	4	20	3

4. Kesimpulan

Hasil pengujian memperlihatkan bahwa dengan sistem penilaian esai dengan menggunakan metode yang diusulkan dapat memberikan tingkat akurasi penilaian yang lebih tinggi dibandingkan dengan skenario pengujian sistem penilaian esai otomatis lainnya yaitu dengan menggunakan metode Landauer, SVM & LSA dengan *index* term sebagai vektor fitur, SVM dengan *index* term sebagai vektor fitur. Selain itu dengan menggunakan metode yang diusulkan, tidak diperlukan adanya pengulangan proses *training* pada pertanyaan yang berbeda untuk melakukan penilaian esai.

Referensi

[1] V. Salvatore, N. Fransesca, & A. Cucchiarelli, "An Overview of Current Research on Automated Essay Grading," *Journal of*

- Information Technology Education*, vol. 2, 2003.
- [2] F. Wild, C. Stahl, G. Stermsek, & G. Neumann, Gustaf, "Parameters Driving Effectiveness of Automated Essay Scoring with LSA" *In Proceedings of the 9th CAA Conference*, 2005.
- [3] Peng, Fuchun., Schuurmans, Dale, Wang, & Shaoujun, "Language and Task Independent Text Categorization with Simple Language Models" *In Proceeding of HLT-NAACL*, 2003.
- [4] S. Scott & S. Matwin, "Feature Engineering for Text Classification" *In Proceeding of the Sixteenth International Conference on Machine Learning*, 1999.
- [5] Joachims & Thorsten, Text Categorization with Support Vector Machine: Learning with Many Relevant Features, Technical Report 23, Universitas Dortmund, LS VIII, 1997.
- [6] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, & R. Harshman, "Indexing by Latent Semantic Analysis," *Journal of the American Society for Information Science*, vol. 41(6), pp. 391-407, 1990.
- [7] Miller & Tristan, "Essay Assesment with Latent Semantic Analysis," *Journal of Educational Computing Research*, vol. 28, 2003.
- [8] Paramita, "Penerapan Support Vector Machine untuk Ekstraksi Informasi dari Dokumen Teks," Laporan Tugas Akhir, Program Studi Teknik Informatika, STEI Institut Teknologi Bandung, Indonesia, 2008.
- [9] Sembiring and Krisantus, Tutorial SVM Berbahasa Indonesia, 2007.
- [10] W.H. Chih, C.C. Chih, & J.L. Chih, A Practical Guide to Support Vector Classification, 2003.
- [11] M. Weimer, I. Gurevych, & M. Muhlhauser, Automatically Assessing the Post Quality in Online Discussions on Software, 2007.
- [12] T.K. Landauer, P.W. Foltz, & D. Laham, An Introduction to Latent Semantic Analysis, Discourse Processes, vol. 25, pp. 259-284, 1998.
- [13] M. Berry, S.T. Dumais, & G.W. O'Brien, Using Linear Algebra for Intelligent Information Retrieval, SIAM Review, 1994.
- [14] A.A.P. Ratna, B. Budiardjo, & D. Hartanto "SIMPLE: Sistem Penilai Esei Otomatis Untuk Menilai Ujian dalam Bahasa Indonesia," *Jurnal Makara Teknologi*, vol. 11, 2007.