

Sistem Penilaian Essay Jawaban Berbahasa Indonesia dengan Metode K-Nearest Neighbor (k-NN) Dan Latent Semantic Analysis (LSA)

Agustus Dwi Budi Darsono¹, Sri Hartati Wijono²

^{1,2}Fakultas Sains dan Teknologi Universitas Sanata Dharma, Yogyakarta
agustusdwi.bd@gmail.com, tatik@usd.ac.id

Abstrak — Tes Essay adalah tes yang disusun dalam bentuk pertanyaan terstruktur dan siswa menyusun serta mengorganisasikan sendiri jawaban dengan bahasa sendiri. Kendala tes essay yang banyak dihadapi seperti membutuhkan waktu yang lama mengoreksi, jawaban heterogen, serta subyektifitas penilai sulit dihindari. Maka dengan kendala itu maka perlu dibuatkan sistem penilaian. Metode yang digunakan dalam sistem ini adalah *K-Nearest Neighbor*(K-NN) dan *Latent Semantic Analysis*(LSA). KNN digunakan dalam klasifikasi jawaban siswa dan LSA untuk menghitung nilai. Implementasi sistem menggunakan bahasa Java dan dalam perhitungan SVD dalam LSA menggunakan *Java Library JAMA*. Tingkat akurasi KNN untuk k=1 mencapai 85,13% dan k=3 mencapai 86,67%. Tingkat error sistem Nilai LSA untuk k=1 mencapai 14,02% dan k=3 mencapai 13,11%.

Kata kunci — Tes Essay, KNN, LSA, SVD, error sistem.

I. PENDAHULUAN

A. Latar Belakang

Tes Essay adalah tes yang disusun dalam bentuk pertanyaan terstruktur dan siswa menyusun, mengorganisasikan sendiri jawaban tiap pertanyaan itu dengan bahasa sendiri. Tes essay ini sangat bermanfaat untuk mengembangkan kemampuan dalam menjelaskan atau mengungkapkan suatu pendapat dalam bahasa sendiri. Essay dianggap sebagai oleh banyak ahli sebagai metode atau alat yang paling tepat dalam penilaian dari kegiatan belajar yang kompleks.

Kendala tes essay yang banyak dihadapi seperti membutuhkan waktu yang lama untuk mengoreksi jawaban, jawaban bermacam – macam sehingga sering menyulitkan dalam menilai, subyektifitas penilai sulit dihindari dan semua hal ini merupakan masalah yang harus diselesaikan. Maka dengan kendala itu maka perlu dibuatkan sistem penilaian untuk mengatasi permasalahan tersebut.

Berbagai macam metode telah dikembangkan dalam penilaian essay menggunakan komputer. Metode – metode yang dikembangkan sebagai berikut PEG (*Project Essay Grade*) yang merupakan metode pertama kali yang digunakan dalam penilaian essay otomatis (Hearst, 2000; Page, 1994, 1996). Korelasi dari metode PEG dan human rater yaitu dengan pemeriksaan manual sebesar 0.87 (87%). Lalu metode selanjutnya *Intelligent Essay Assessor* (IEA) yang dikembangkan sekitar akhir tahun 90an [1] dan ini didasarkan pada teknik *Latent Semantic Analysis* (LSA) yang didesain untuk indexing dan text retrieval [2] dan untuk korelasi dari metode IEA dan *human rater* yaitu dengan pemeriksaan manual sebesar 85 % sampai 91%.

Algoritma klasifikasi KNN memprediksi kategori test yang sesuai dengan contoh training K yang merupakan

tetangga terdekat dengan sample uji dan menentukan dengan kemungkinan kategori yang terbesar. Algoritma LSA merupakan salah satu algoritma pengembangan dari bidang ilmu *Information Retrieval*, yang mampu menghimpun sejumlah besar dokumen dalam basis data dan menghubungkan relasi antar dokumen dengan mencocokkan kueri (query) yang diberikan. Fungsi utama LSA ini adalah menghitung kemiripan (similarity) dokumen dengan membandingkan representasi vektor tiap dokumen tersebut.

Penelitian yang pernah dilakukan oleh Rizqi Bayu Aji P,dkk. dari Institut Teknologi Telkom Bandung dengan judul “*Automatic Essay Grading Sistem Menggunakan Metode Latent Semantic Analysis*” menyimpulkan bahwa dengan algoritma LSA dapat digunakan untuk menilai jawaban siswa dan menghasilkan output sebuah nilai. Korelasi penilaian *human raters* dengan sistem adalah 45.03% dan 50.55% [6]. Penelitian lain adalah penilaian jawaban esai dengan menggabungkan *support vector machine* (SVM) sebagai teknik klasifikasi teks otomatis dengan LSA [5] dengan hasil pengujian memperlihatkan bahwa dengan metode yang diusulkan mempunyai tingkat akurasi penilaian yang lebih tinggi jika dibandingkan dengan metode lain seperti SVM atau LSA dengan index term. Maka dari penelitian yang telah dilakukan [5] dan [6] akan peneliti akan menguji hasil dari metode KNN sebagai metode pengklasifikasian jawaban digabungkan LSA untuk membandingkan hasil dengan hasil penilaian manual.

B. Rumusan Masalah

1. Bagaimana membangun sebuah sistem penilaian pada jawaban ujian berbentuk essay menggunakan metode LSA (*Latent Semantic Analysis*) dan *k-Nearest Neighbor* (k-NN)?
2. Bagaimana perbandingan hasil metode LSA (*Latent Semantic Analysis*) dan *k-Nearest Neighbor* (k-NN) dalam penilaian essay dengan penilaian manual (*human rater*)?

C. Tujuan

Tujuan untuk menganalisa, mendesain dan mengimplementasikan algoritma metode LSA (*Latent Semantic Analysis*) dan *k-Nearest Neighbor* (k-NN) untuk penilaian essay jawaban berbahasa Indonesia.

II. DASAR TEORI

A. Information Retrieval

Information retrieval (IR) adalah proses menemukan bahan (biasanya dokumen) dari bersifat tidak terstruktur (biasanya teks) yang memenuhi kebutuhan informasi dari dalam koleksi besar (biasanya disimpan di komputer).[8]

Sistem Temu-Balik Informasi (*Information Retrieval*) digunakan untuk menemukan kembali informasi-informasi yang relevan terhadap kebutuhan pengguna dari suatu kumpulan informasi secara otomatis. Salah satu aplikasi umum dari sistem temu kembali informasi adalah *search-engine* atau mesin pencarian yang terdapat pada jaringan internet. Pengguna dapat mencari halaman-halaman Web yang dibutuhkannya melalui mesin tersebut.

B. K-Nearest Neighbor (KNN) Teks Classification

KNN merupakan merupakan salah satu algoritma non-parameter yang sangat penting dalam bidang pengenalan pola[3] dan ini adalah algoritma klasifikasi yang *supervised*. Aturan dari klasifikasi KNN dihasilkan dengan contoh training data dengan datanya sendiri tanpa data tambahan. Algoritma klasifikasi KNN memprediksi kategori test yang sesuai dengan contoh training K yang merupakan tetangga terdekat dengan sample uji dan menentukan dengan kemungkinan kategori yang terbesar. Proses dari algoritma KNN untuk klasifikasi dokumen [10]:

Misalkan ada *j* kategori *training* dan jumlah dari *training* adalah *i*. Setelah preprocessing untuk tiap dokumen, maka akan menjadi *m*-dimensi vector fitur.

1. Membuat dokumen X menjadi sama *text feature vector* bentuk (X_1, X_2, \dots, X_m) seperti semua di *training sample*.
2. Hitung similarity diantara *training sample* dan dokumen X. Ambil dokumen ke-*i* sebagai contoh $d_i(d_{i1}, d_{i2}, \dots, d_{im})$ maka similarity $SIM(X, d_i)$ sebagai berikut :

$$SIM(X, d_i) = \frac{\sum_{j=1}^m X_j \cdot d_{ij}}{\sqrt{\sum_{j=1}^m X_j^2} \cdot \sqrt{\sum_{j=1}^m d_{ij}^2}} \dots (1)$$

3. Setelah itu memilih *k samples* yang lebih besar dari *N similarity* dari $SIM(X, d_i)$ dan perlakukan itu seperti sebuah koleksi KNN dari X. lalu hitung *probability* dari X untuk tiap kategori dengan rumus :

$$P(X, C_j) = \frac{\sum_{i=1}^k SIM(X, d_i) \cdot y_i(d, C_j)}{\sum_{i=1}^k SIM(X, d_i)} \dots (2)$$

Dimana $y_i(d, C_j)$ adalah fungsi atribut kategori yang memenuhi:

$$y_i(d, C_j) = \begin{cases} 1, & d \in C_j \\ 0, & d \notin C_j \end{cases} \dots (3)$$

4. Tentukan kategori dokumen X yang mempunyai nilai $P(X, C_j)$ terbesar.

C. Latent Semantic Analysis (LSA)

Algoritma LSA (*Latent Semantic Analysis*) merupakan sebuah teori dan metode untuk mengekstraksi dan merepresentasikan makna konseptual kata dengan menggunakan perhitungan statistic yang diterapkan pada korpus besar teks. Ide dasarnya bahwa keseluruhan informasi tentang semua konteks kata dimana kata muncul dan tidak muncul menyediakan suatu kendala yang sangat menentukan kesamaan kata – kata dan mengatur kata satu dengan kata lain [2].

LSA mengasumsikan bahwa kata-kata yang dekat dalam arti akan terjadi pada potongan-potongan serupa teks. Sebuah matriks yang mengandung jumlah kata per paragraf (baris mewakili kata-kata unik dan kolom mewakili setiap paragraf) dibangun dari sepotong besar teks dan teknik matematika yang disebut dekomposisi nilai singular (SVD) digunakan untuk mengurangi jumlah baris sambil menjaga struktur kesamaan di antara kolom. Kata-kata tersebut kemudian dibandingkan dengan mengambil cosinus dari sudut antara dua vektor yang dibentuk oleh dua baris. Nilai mendekati 1 mewakili kata-kata yang sangat mirip sementara nilai mendekati 0 mewakili kata-kata yang sangat berbeda.[4]

Teknik SVD yang digunakan pada LSA adalah *reduced SVD*, dimana akan dilakukan proses pengurangan dimensi (*dimensionality reduction*) pada matriks hasil dekomposisi SVD. Pengurangan dimensi ini sebagai *noise reduction* process. [5]

D. Singular Value Decomposition (SVD)

SVD adalah sebuah metode untuk mengidentifikasi dan mengurutkan dimensi yang menunjukkan data mana yang menunjukkan variasi yang paling banyak. Berkaitan dengan hal itu, SVD dapat mengidentifikasi dimana variasi muncul paling banyak, sehingga hal ini memungkinkan untuk mencari pendekatan yang terbaik pada data asli menggunakan dimensi yang lebih kecil. Oleh karena itu SVD dapat dilihat sebagai metode pengurangan data[6].

SVD didasarkan pada teorema aljabar linier yang mengatakan bahwa sebuah matriks A dapat dipecah menjadi 3 buah matriks yaitu Matriks *orthogonal* U, Matriks diagonal S dan *transpose* matriks *orthogonal* V [7] Teorema ini sebagai berikut :

$$A_{mm} = U_{mm} S_{mm} V^T \dots (4)$$

keterangan :

- A = matriks berdimensi mxn
- U = matriks vektor singular kiri berdimensi mxm
- S = matriks nilai singular berdimensi mxn
- V^T = matriks transpose vektor singular kanan berdimensi nxn

E. Cosine similarity

Cosine similarity adalah ukuran kesamaan antara dua vektor dari sebuah ruang hasil kali dalam yang mengukur cosinus dari sudut tersebut. Cosinus dari 0 ° adalah 1, dan kurang dari 1 untuk setiap sudut lainnya. *Cosine similarity* adalah pada keputusan orientasi dan tidak besarnya: dua vektor dengan orientasi yang sama memiliki kesamaan *Cosine* dari 1, dua vektor pada 90 ° memiliki kesamaan 0, dan dua vektor bertentangan memiliki kesamaan -1, independen dari besarnya sudut. *Cosine similarity* digunakan dalam ruang positif, di mana hasilnya dibatasi dalam [0,1]. [8]

Untuk mengimbangi efek panjang dokumen, cara standar untuk mengukur kesamaan antara 2 dokumen d_1 dan d_2

dengan menghitung *Cosine similarity* dari (d_1) dan (d_2) . [8] dalam persamaan sebagai berikut :

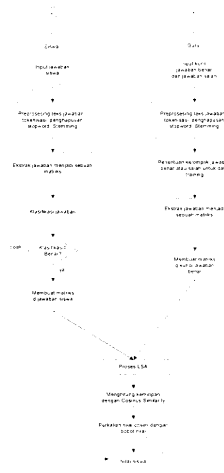
$$\text{sim}(d_1, d_2) = \frac{\vec{V}(d_1) \cdot \vec{V}(d_2)}{\|\vec{V}(d_1)\| \|\vec{V}(d_2)\|} \quad \dots(5)$$

III. METODE PENELITIAN

A. Gambaran Sistem

Sistem koreksi essay jawaban otomatis ini dikembangkan untuk membantu pengguna dalam memberikan nilai secara otomatis pada pada jawaban siswa sehingga sistem ini akan mempercepat proses penilaian. Kumpulan jawaban baik jawaban siswa maupun kunci jawaban berupa teks file dengan ekstensi txt. Sistem akan melakukan *preprocessing* (melakukan tokenisasi, menghilangkan *stopword* dan melakukan stemming) teks jawaban. Dengan melakukan preprocessing tersebut maka akan dihasilkan sebuah kata yang unik sehingga dapat diproses untuk langkah selanjutnya.

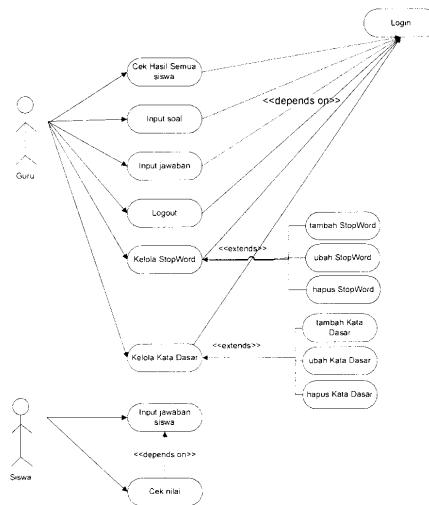
Untuk arsitektur sistem ini ditunjukkan pada gambar 3.1 . Dokumen yang telah melewati proses preprocessing maka akan dilakukan klasifikasi menggunakan algoritma kNN (*k-Nearest Neighbours*) teks clustering. Dalam klastring ini sebelumnya sudah ada data training untuk jawaban benar dan salah yang telah dilakukan oleh admin. Selanjutnya dokumen jawaban siswa sebagai data testing akan di clustering untuk masuk ke jawaban salah atau benar. Lalu setelah clustering, jika jawaban salah maka tidak akan masuk proses LSA karena sudah masuk nilai minimal dan jika jawabannya benar maka akan masuk ke algoritma LSA (*Lantent Semantic Analysis*). LSA menggunakan proses SVD (*Singular Value Decomposition*) maka akan menghasilkan sebuah vector jawaban siswa dan vector jawaban kunci. Dari vector jawaban tersebut akan dilihat berapa tingkat kemiripan dengan *cosine similarity*. Untuk mendapatkan nilai maka hasil *cosinus similarity* tersebut dikalikan bobot nilai.



gambar 1. Gambaran arsitektur sistem yang dikembangkan

B. Use Case

Diagram use case merupakan gambaran fungsionalitas dari suatu sistem, sehingga pengguna sistem mengerti mengenai kegunaan sistem yang akan dibangun.



gambar 2. Use Case

C. Perhitungan Akurasi

Dalam penghitungan akurasi ini dengan menghitung jumlah record uji yang diprediksi secara tepat.

$$\text{Akurasi} = \frac{\text{Jumlah prediksi tepat}}{\text{jumlah total prediksi}} \times 100\% \quad \dots(6)$$

D. Prosentase Kesalahan Sistem

Untuk menghitung prosentase kesalahan sistem maka harus dihitung berapa besar tingkat kesalahannya. Prosentase kesalahan sistem merupakan nilai maksimal jawaban dikurangi hasil bagi nilai sistem dengan nilai manual dikalikan 100%. Lalu dengan prosentase kesalahan sistem tersebut dapat diukur berapa persen tingkat keberhasilannya. Berikut adalah detail perhitungan manual prosentase kesalahan sistem [9].

$$\text{Prosentase kesalahan} = 100 - \left\{ \frac{NS}{NM} * 100 \right\} \quad \dots(7)$$

dimana : NS = Nilai sistem
 NM = Nilai manual

IV. HASIL DAN PEMBAHASAN

A.Skenario Pengujian

Pengujian dilakukan dengan perangkat lunak sistem penilaian yang dibangun dengan bahasa pemrograman JAVA dan menggunakan pustaka library JAMA dalam perhitungan proses SVD. Untuk pengujian dilakukan dengan 2 skenario dengan membandingkan nilai k (jumlah tetangga terdekat) yaitu yang pertama dengan nilai k=1 dan yang kedua dengan nilai k=3.

Data pengujian. Data set yang digunakan berasal dari jawaban ujian semester kelas VI SD tahun pelajaran 2013/2014 mata pelajaran Pendidikan Kewarganegaraan sebanyak 39 siswa. Jawaban pertanyaan data set ini diketik ulang dalam format teks (.txt) dengan pertanyaan sebanyak 5 soal yaitu (1) Sebutkan 4 (empat) contoh organisasi yang berada di lingkungan masyarakat !, (2) Sebutkan 3 (tiga) contoh tugas ketua dalam sebuah organisasi !, (3) Pada kegiatan musyawarah termasuk pengamalan Pancasila sila ke berapa dan bagaimana bunyinya ? .(4) Sebutkan 3 (tiga) ciri musyawarah untuk mufakat !.(5)Sebutkan 3 (tiga) upaya untuk menciptakan NKRI yang lebih aman dan damai !

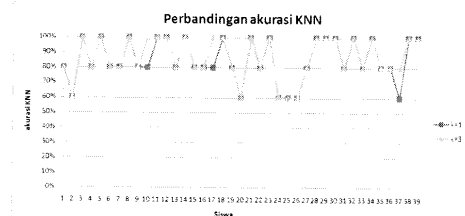
B.Hasil Pengujian

Dari hasil pengujian dapat dianalisis hasilnya seperti pada uraian dibawah ini :

Berdasarkan hasil pengujian dengan nilai k=1 diperoleh akurasi KNN untuk penilaian manual dan sistem dengan akurasi tertinggi 100% dan terendah sebesar 60%. Rerata akurasi KNN secara keseluruhan mencapai 85.13%. Untuk penilaian LSA, prosentase kesalahan tertinggi untuk koreksi Nilai LSA 33.221% dan untuk prosentase terendah adalah 0.0037%. Rerata prosentase kesalahan Nilai LSA keseluruhan mencapai 14.02%.

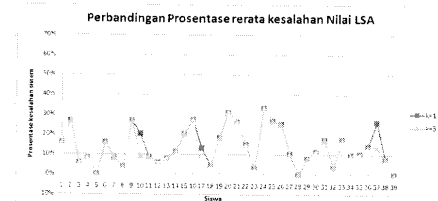
Berdasarkan hasil pengujian dengan nilai k=3 yang diperoleh, akurasi KNN untuk penilaian manual dan sistem dengan akurasi tertinggi 100% dan terendah sebesar 60%. Rerata akurasi KNN secara keseluruhan mencapai 86.67%. Untuk penilaian LSA, prosentase kesalahan tertinggi antara koreksi nilai manual dengan koreksi Nilai LSA adalah 33.221 % dan untuk prosentase terendah adalah 0.0037 %. Rerata prosentase kesalahan Nilai LSA keseluruhan mencapai 13.1132 %. Pengujian dengan data uji yang sama telah dilakukan untuk menguji nilai LSA tanpa melalui klasifikasi KNN menunjukkan error kesalahan sistem sebesar 36.643 %. Untuk KNN mempunyai kelemahan jika jawaban siswa mengandung jawaban salah yang lebih besar dari jawaban benar yang ada maka akan disimpulkan bahwa jawaban siswa tersebut salah.

Grafik perbandingan hasil akurasi KNN untuk k=1 dan k=3 ditunjukkan pada gambar 3.



gambar 3. Grafik perbandingan akurasi KNN

Untuk grafik perbandingan prosentase rerata kesalahan nilai LSA untuk k=1 dan k=3 ditunjukkan pada gambar 4



gambar 4. Grafik perbandingan prosentase rerata kesalahan nilai LSA.

V. KESIMPULAN DAN SARAN

A.Kesimpulan

Melalui penelitian yang telah dilakukan dapat disimpulkan bahwa :

1. Penerapan algoritma K-Nearest Neighbor (K-NN) Dan Latent Semantic Analysis (LSA) dalam penilaian essay ini menghasilkan :
 - a. Rata – rata akurasi untuk K-Nearest Neighbor (K-NN) untuk k=1 sebesar 85.13% dan untuk k=3 sebesar 86.67%. Perbedaan selisih untuk rata-rata akurasi KNN untuk k=1 dan k=3 sebesar 1.54%.
 - b. Prosentase rata-rata tingkat kesalahan nilai LSA untuk k=1 sebesar 14.02 % dan untuk k=3 sebesar 13.11 %. Perbedaan selisih untuk prosentase rata-rata tingkat kesalahan nilai LSA untuk k=1 dan k=3 sebesar 0.91%.
2. Untuk beberapa siswa dengan perbedaan nilai yang sangat mencolok antara nilai sistem dan koreksi manual disebabkan karena permasalahan makna kata ataupun sinonim maupun ada tidaknya term kunci pada jawaban. Sehingga kata-kata kunci dalam jawaban sangat berpengaruh pada nilai yang didapatkan..

3. Untuk KNN mempunyai kelemahan jika jawaban siswa mengandung jawaban salah yang lebih besar dari jawaban benar yang ada maka akan disimpulkan bahwa jawaban siswa tersebut salah.

B.Saran

Melalui penelitian yang telah dilakukan dapat disimpulkan bahwa :

1. Sistem dapat dikembangkan untuk jenis soal essay yang lain.
2. Sistem perlu penambahan proses untuk mendeteksi sinonim kata atau penambahan kata kunci sehingga nilai akurasi akan lebih tinggi dan nilai kesalahan sistem lebih kecil.

DAFTAR PUSTAKA

- [1] Hearst, M. "The debate on automated essay grading." *IEEE Intelligent Systems*. 15(5). pp.22-37. March 2000
- [2] Landauer, T. K., Foltz, P. W., & Laham, D. "Introduction to Latent Semantic Analysis". *Discourse Processes*. 25. 259-284. October 1998
- [3] Belur V. Dasarathy. "Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques". *McGraw-Hill Computer Science Series, IEEE Computer Society Press*. Las Alamitos, California. July 1991.
- [4] Susan T. Dumais. "Latent Semantic Analysis". *Annual Review of Information Science and Technology* 38: 188. doi:10.1002/aris. 1440380105. September 2005
- [5] Adhitia, Rama dan Ayu Purvianti. "Penilaian esai jawaban bahasa Indonesia menggunakan metode SVM-LSA dengan fitur generik". *Jurnal Sistem Informasi MTI UI*. Volume 5. Nomor 1: hal. 1412 – 8896. April 2009
- [6] Rizqi Bayu Aji P.ZK, Abdurrahman Baizal, S.Si., M.Kom, Yanuar Firdaus, S.T., M.T. "Automatic Essay Grading Sistem Menggunakan Metode Latent Semantic Analysis" *Seminar Nasional Aplikasi Teknologi Informasi 2011 (SNATI 2011)*. Juni 2011
- [7] Baker, Kirk. "Singular Value Decomposition Tutorial". March 2005
- [8] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press. 2008
- [9] Muhammad Zakiya Nafik, Indriati, ST., M.Kom., Achmad Ridok, M.Kom. "Sistem Penilaian Otomatis Jawaban Esai Menggunakan Algoritma Levenshtein Distance". *Journal at Information Technology And Computer Science Departement, Universitas Brawijaya Malang*. Volume 3, Nomor 4. Juni 2014
- [10] Yong Zhou, Li Youwen and Xia Shixiong. 2009. "An Improved KNN Text Classification Algorithm Based on Clustering" *Journal Of Computers*. VOL. 4. NO. 3. March 2009