



SISTEM PENILAIAN UJIAN OTOMATIS UNTUK SOAL ESAI MENGGUNAKAN METODE VECTOR SPACE MODEL

Mi'andri¹, Alda Cendekia Siregar², Putri Yuli Utami³

¹Fakultas Teknik dan Ilmu Komputer, Universitas Muhammadiyah Pontianak

Email: miandri.miandri@unmuhpnk.ac.id¹, alda.siregar@unmuhpnk.ac.id², putriyuli@unmuhpnk.ac.id³

Informasi Artikel	ABSTRACT
<p>Riwayat artikel : Disubmit : 20 Juni 2021 Direvisi : 19 Juli 2021 Diterima : 18 November 2021 Dipublikasi : 31 Desember 2021</p> <p>Keywords: TF-IDF (term frequency - inverse document frequency), vector space model, automated essay scoring</p>	<p><i>Examination is a way to measure a student's ability. Exams are also used as an evaluation tool to assess how far the knowledge has been mastered and the skills that have been acquired. Essay exams are one of the evaluation processes used to determine student abilities. The problem that occurs when the lecturer assesses is the length of time it takes to check student answer sheets, this can affect objectivity in the assessment. To overcome this problem, a system is needed that can make the assessment of essay exam answers faster. This study aims to create an automatic essay examination scoring system that can assist lecturers in assessing student essay exams. The method used is the weighting method TF-IDF (Term Frequency - Inverse Document Frequency) and the Vector Space Model to calculate the similarities of two documents, where the process carried out in the assessment of essay answers is to compare the answers that have been filled in by students with the lecturers' answer keys. Based on the research results, the automatic essay scoring system using the method Vector Space Model has worked well.</i></p>
<p>Kata Kunci: TF-IDF (term frequency - inverse document frequency), vector space model, penilaian esai otomatis</p>	<p style="text-align: center;">ABSTRAK</p> <p>Pelaksanaan ujian merupakan cara untuk mengukur kemampuan seorang mahasiswa. Ujian juga dijadikan sebagai alat evaluasi untuk menilai seberapa jauh pengetahuan yang sudah dikuasai dan keterampilan yang sudah diperoleh. Ujian esai adalah salah satu proses evaluasi yang digunakan untuk mengetahui kemampuan mahasiswa. Masalah yang terjadi pada saat dosen menilai yaitu lamanya waktu yang dibutuhkan untuk memeriksa lembar jawaban mahasiswa, hal ini dapat mempengaruhi objektivitas dalam penilaian. Untuk mengatasi masalah tersebut maka diperlukan sistem yang dapat membuat penilaian jawaban ujian esai lebih cepat. Penelitian ini bertujuan untuk membuat sistem penilaian ujian esai secara otomatis yang dapat membantu dosen dalam menilai ujian esai mahasiswa. Metode yang digunakan yaitu metode pembobotan TF-IDF (Term Frequency - Inverse Document Frequency) dan Vector Space Model untuk menghitung kemiripan dari dua buah dokumen, dimana proses yang dilakukan pada penilaian jawaban esai adalah dengan membandingkan jawaban yang telah diisikan oleh mahasiswa dengan kunci jawaban dosen. Berdasarkan hasil penelitian bahwa sistem penilaian esai otomatis menggunakan metode Vector Space Model telah berjalan dengan baik.</p>



PENDAHULUAN

Pelaksanaan ujian merupakan cara untuk mengukur kemampuan seseorang atau mahasiswa. Ujian juga dijadikan sebagai alat evaluasi untuk menilai seberapa jauh pengetahuan yang sudah dikuasai dan keterampilan yang sudah diperoleh. Ujian esai adalah salah satu proses evaluasi yang digunakan untuk mengetahui kemampuan mahasiswa, dalam pelaksanaan ujian berbasis esai mahasiswa dituntut untuk menjawab soal dengan pemahaman yang mereka miliki. Soal esai akan menghasilkan berbagai kemungkinan jawaban sesuai dengan pemahaman setiap mahasiswa. Mahasiswa tidak bisa menebak jawaban dalam pengisian soal esai, dengan begitu dosen dapat menilai kemampuan mahasiswanya dalam menjawab pertanyaan, sehingga skor nilai untuk jawaban esai bisa beragam. Ada beberapa masalah yang sering terjadi pada saat dosen menilai hasil jawaban ujian esai, yaitu lamanya waktu yang dibutuhkan untuk memeriksa lembar jawaban mahasiswa dan nilai yang diberikan kepada setiap mahasiswa sering tidak konsisten padahal jawaban esai beberapa mahasiswa tersebut sama, hal ini menyebabkan kualitas penilaian menurun dan terkadang penilaian tidak bersifat objektif lagi (Hidayat & Afuan, 2019).

Tes uraian yang dikoreksi secara manual memiliki beberapa kelemahan antara lain: (1) Skor dapat berbeda ketika dinilai oleh penilai yang sama pada waktu yang berbeda atau oleh penilai yang berbeda pada waktu yang sama; (2) Penilaian pada tes uraian memerlukan waktu untuk mengoreksi lembar jawaban sangat lama; (3) Jumlah jawaban yang banyak pada setiap kelas, hampir mustahil penilai mempunyai waktu untuk menggunakan tes uraian, kecuali jika mereka mempunyai "mesin penilai"; (4) Biaya yang diperlukan untuk mengoreksi tes uraian adalah cukup besar (Putra & Khusartantya, 2014).

Penelitian pertama dengan judul "Penilaian Ujian Otomatis untuk Soal Bertipe *Essay* pada PJJ APTIKOM menggunakan *Cosine Similarity*" Penelitian mengembangkan sistem penilaian jawaban *essay* secara otomatis pada PJJ APTIKOM dengan menggunakan algoritma similaritas. Terdapat beberapa tahapan yang dilakukan pada penelitian ini antara lain: 1) input data yang dimasukkan oleh pendidik dan peserta didik berupa soal ujian, kunci jawaban dan jawaban; 2) Pemrosesan teks, kunci jawaban dan jawaban dilakukan pemrosesan teks dengan melakukan *tokenisasi*, *stopword removal*, dan *stemming*; 3) Perhitungan bobot, dilakukan dengan menggunakan algoritma TF-IDF; 4) Pengukuran similaritas antara jawaban dan kunci jawaban dengan menggunakan algoritma *cosine similarity*; 5) Penjumlahan nilai similaritas untuk setiap soal. Selanjutnya, nilai similaritas dikonversi menjadi nilai akhir bagi peserta didik (Hidayat & Afuan, 2019).

Penelitian kedua dengan judul "Rancang Bangun Aplikasi Penilaian Ujian Essay Dengan Menggunakan Algoritma Nazief & Andriani Dan Metode *Cosine Similarity*" dengan demikian penelitian yang akan penulis lakukan yaitu mengenai perancangan aplikasi yang akan digunakan untuk



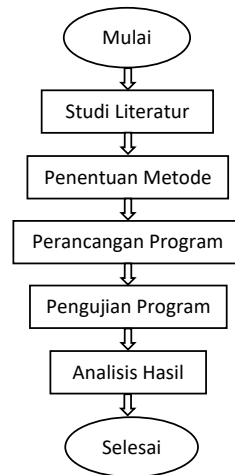
penilaian terhadap jawaban dari soal *essay*. Namun *software* aplikasi yang akan dirancang membutuhkan algoritma-algoritma yang dapat menghitung kemiripan antara jawaban mahasiswa dengan kunci jawaban yang sudah disediakan oleh dosen yang berkaitan. Untuk itu penulis menggunakan Metode *Cosine Similarity* dalam proses penilaian jawaban *essay* mahasiswa. Walaupun dalam penilaian jawaban *essay* dengan menggunakan Metode *Cosine Similarity* saja sudah cukup, namun agar perhitungan kemiripan dokumen jawaban dengan Metode *Cosine Similarity* lebih akurat maka penulis menerapkan algoritma Nazief & Adriani untuk proses *stemming* kata-kata dalam dokumennya (Bastian, Sujadi, & Sukmana, 2018).

Untuk mengatasi masalah tersebut maka diperlukan aplikasi yang dapat membuat pekerjaan penilaian jawaban ujian esai lebih cepat, yaitu aplikasi yang dapat memproses kemiripan teks. Oleh karena itu dalam penelitian ini digunakan metode pembobotan TF-IDF (*Term Frequency - Inverse Document Frequency*) dan *Vector Space Model* untuk menghitung kemiripan dari dua buah dokumen, dimana proses yang dilakukan pada penilaian jawaban esai adalah dengan membandingkan jawaban yang telah diisikan oleh mahasiswa dengan kunci jawaban dosen. Prosesnya, sistem akan melakukan *text preprocessing* terlebih dahulu, setelah proses *text preprocessing*, perhitungan nilai bobot kata akan dilakukan menggunakan pembobotan TF-IDF dan VSM (*Vector Space Model*) akan menghitung nilai similaritas antara *query* dengan dokumen yang didapat dari ekstraksi teks pada dokumen, sehingga didapatkan nilai tingkat kemiripan antara kunci jawaban dosen dengan jawaban mahasiswa.

Berdasarkan latar belakang yang telah diuraikan, pada penelitian ini peneliti membuat sistem yang digunakan untuk penilaian jawaban ujian esai secara otomatis. Sistem ini diharapkan dapat menilai jawaban esai secara efektif dan efisien sehingga dapat mempermudah penilaian yang dilakukan oleh dosen.

METODE PENELITIAN

Tahapan Metode Penelitian penulis uraikan dalam bentuk diagram alir proses penelitian seperti gambar 1 berikut ini :



Gambar 1. Diagram alir proses penelitian

1. Studi Literatur

Metode studi literatur adalah serangkaian kegiatan yang berkenaan dengan metode pengumpulan data pustaka, membaca dan mencatat, serta mengelolah bahan penelitian. Sumber-sumber rujukan (buku, jurnal, artikel) yang diacu dalam penelitian ini adalah yang berkaitan dengan Sistem Penilaian Ujian Otomatis Untuk Soal Esai Menggunakan Vector Space Model. Tujuan dilakukannya studi literatur adalah untuk mendapatkan landasan teori yang dapat mendukung pemecahan masalah yang sedang diteliti.

2. Penentuan Metode

Tahap ini menentukan metode yang tepat sesuai dengan permasalahan. Metode pemecahan masalah yang digunakan dalam penelitian ini yaitu metode Pembobotan TF-IDF (*Term Frequency - Inverse document Frequency*) dan Pengukur Similaritas *Vector Space Model*. Sebelum dihitung pembobotan *Term Frequency Inverse Document Frequency* (TF-IDF) dan menghitung kemiripan antara dokumen dan query harus melalui tahap *text preprocessing*. *Text* pada umumnya memiliki struktur kalimat yang tidak baik dan banyak *noise*. Untuk mendapatkan ekstraksi pada *text* maka harus dilakukan *text preprocessing*. Beberapa proses yang dapat dilakukan pada *text preprocessing* adalah, *tokenizing*, *filtering*, dan *stemming* (Pitriyani & Ardiansyah, 2019).

- 1) *Tokenizing* merupakan proses pemisahan suatu rangkaian karakter berdasarkan karakter spasi, dan memungkinkan pada waktu yang bersamaan dilakukan juga proses penghapusan karakter tertentu, seperti tanda baca.



- 2) *Filtering (stopword removal)* adalah tahap mengambil kata-kata penting dari hasil *tokenizing* dengan *stoplist* (membuang kata sambung atau konjungsi), seperti ‘dan’, ‘yang’, ‘untuk’, ‘dapat’. *stopword* adalah kata-kata yang tidak deskriptif.
- 3) *Stemming* digunakan untuk mengubah *term* menjadi kata dasar. Proses *stemming* dilakukan dengan cara menghilangkan imbuhan baik yang terdiri dari awalan (*prefixes*), maupun akhiran(*suffixes*).

TF merupakan pembobotan yang dimana penting tidaknya sebuah kata diasumsikan sebanding dengan jumlah kemunculan kata tersebut dalam dokumen, sementara IDF merupakan pembobotan yang mengukur seberapa penting sebuah kata dalam dokumen apabila dilihat secara global pada seluruh dokumen. Pada dasarnya TF dan IDF merupakan proses pembobotan suatu kata dalam suatu dokumen dihitung berdasarkan banyaknya kata tersebut muncul (Saputra et al., 2019).

Rumus yang digunakan dalam proses TF-IDF yaitu menggunakan persamaan (1) Berikut (Sanjaya, 2017):

$$W = tf_{ij} \times \log \left(\frac{D}{df_i} \right) \quad (1)$$

dimana

1. W adalah bobot *term* t_j terhadap dokumen d_i
2. tf_{ij} adalah jumlah kemunculan *term* t_j dalam dokumen d_i
3. D adalah jumlah semua dokumen yang ada
4. df_j adalah jumlah dokumen yang mengandung *term* t_j

Kemiripan antara dokumen dan *query* akan dihitung dengan pendekatan *similarity*. Perhitungan *similarity* menggunakan VSM dapat ditulis menggunakan persamaan (2) berikut (Sanjaya, 2017):

$$Sim(d_j, q) = \frac{\sum_{i=1}^t (W_{ij} * W_{iq})}{\sqrt{\sum_{i=1}^t (W_{ij})^2 * \sum_{i=1}^t (W_{iq})^2}} \quad (2)$$

dimana

1. D_j adalah dokumen ke j
 2. Q adalah *query user*
 3. $\sum_{i=1}^t W_{ij}$ adalah jumlah bobot kata i pada dokumen j
 4. $\sum_{i=1}^t W_{iq}$ adalah jumlah bobot kata i pada *query*
3. Perancangan Program



Perancangan program merupakan pengembangan solusi terhadap identifikasi masalah dan menghasilkan serangkaian instruksi yang membangun sebuah program komputer untuk menghasilkan output. Setelah perancangan selesai dilakukan, selanjutnya rancangan akan diimplementasikan pada bahasa pemrograman yang digunakan yaitu PHP dan MySQL sebagai databasenya. Setelah implementasi maka dilakukan pengujian sistem.

4. Pengujian Sistem

Pengujian yang digunakan untuk menguji sistem penilaian esai ini adalah metode pengujian *black box*. *Black Box Testing* atau yang sering dikenal dengan sebutan pengujian fungsional merupakan metode pengujian Perangkat Lunak yang digunakan untuk menguji perangkat lunak tanpa mengetahui struktur internal kode atau Program. Dalam pengujian ini, tester menyadari apa yang harus dilakukan oleh program tetapi tidak memiliki pengetahuan tentang bagaimana melakukannya.

5. Analisis Hasil

Analisis hasil merupakan pembahasan yang dilakukan dengan menginput data pertanyaan soal, jawaban mahasiswa, dan kunci jawaban. setelah itu teks inputan dilakukan tahap *text preprocessing* yaitu *tokenizing*, *stopword*, dan *stemming*. Tahapan berikutnya dilakukan pembobotan TF-IDF dan perhitungan kesamaan dokumen dengan VSM, setelah didapat hasil kemiripannya kemudian mengkonversikannya menjadi nilai ujian esai berdasarkan tabel rentang penilaian manusia.

A. Analisis Kebutuhan Sistem

Pada tahap ini mengumpulkan data awal yang berhubungan dengan pelaksanaan ujian. Untuk pengumpulan data awal maka perlu dilakukan analisa terhadap sistem penilaian ujian esai otomatis.

1. Kebutuhan Fungsional Sistem

- 1) Terdapat form login untuk mahasiswa yang akan mengikuti ujian.
- 2) Sistem dapat menampilkan soal ujian berbentuk form isian.
- 3) Mahasiswa tidak dapat mengisi kembali jawaban ujian yang sudah diisi sebelumnya.
- 4) Sistem dapat menampilkan nilai per soal.
- 5) Sistem dapat menampilkan total nilai dari seluruh soal yang ada.

2. Kebutuhan Sistem

- 1) Aplikasi yang dibangun berbasis web.
- 2) Perangkat lunak yang digunakan yaitu, OS Windows 7 atau lebih, OS Android, Internet Browser, Web Server yang mendukung PHP, MySQL.



- 3) Untuk pengguna perangkat keras yang digunakan berupa seperangkat komputer atau laptop dengan spesifikasi minimum Processor Intel dual core, Ram minimal 1GB, Kapasitas Harddisk minimal : 2GB atau lebih.
- 4) Untuk pengguna Smartphone, Ram minimal 1GB.

B. Analisis Kebutuhan Pengguna

Pengguna yang akan menggunakan sistem ini dibagi menjadi 3 level yaitu mahasiswa, dosen, dan administrator.

1. Kebutuhan *Input*

- Mahasiswa : Login, input jawaban.
- Dosen : Login, input soal dan kunci jawaban setiap soal untuk menghitung bobot dan kemiripan dengan jawaban mahasiswa.
- Administrator : Login, master data (data stopword, data kata dasar, data mahasiswa).

2. Kebutuhan Proses

- Mahasiswa : Autentikasi login, simpan jawaban ke database
- Dosen : Autentikasi login, proses data soal, proses data jawaban, proses text preprocessing kunci jawaban dan jawaban mahasiswa, proses periksa ujian, perhitungan bobot TF-IDF dan kemiripan vector space model.
- Administrator : Autentikasi login, proses data mahasiswa, proses data stopword, proses data kata dasar.

3. Kebutuhan Output

- Mahasiswa : Hasil verifikasi *login* tampil halaman beranda, tampilan *alert* berhasil jika sudah mengisi jawaban dan *alert* gagal jika mengisi jawaban yang sudah diisi sebelumnya.
- Dosen : Hasil verifikasi *login* tampil halaman beranda, daftar soal, daftar jawaban, daftar kalimat text preprocessing, daftar kemunculan kata, hasil perhitungan bobot TF-IDF, hasil perhitungan kemiripan *vector space model*, daftar nilai mahasiswa.
- Administrator : Hasil verifikasi *login* tampil halaman beranda, daftar kata *stopword*, daftar kata dasar, daftar mahasiswa.

HASIL DAN PEMBAHASAN

Setelah sistem didesain, maka selanjutnya akan dilakukan tahap implementasi. Tujuan implementasi adalah untuk mengkonfirmasi modul program perancangan pada para pelaku sistem sehingga pengguna dapat memberi masukan kepada pembangun sistem.

1. Halaman Login



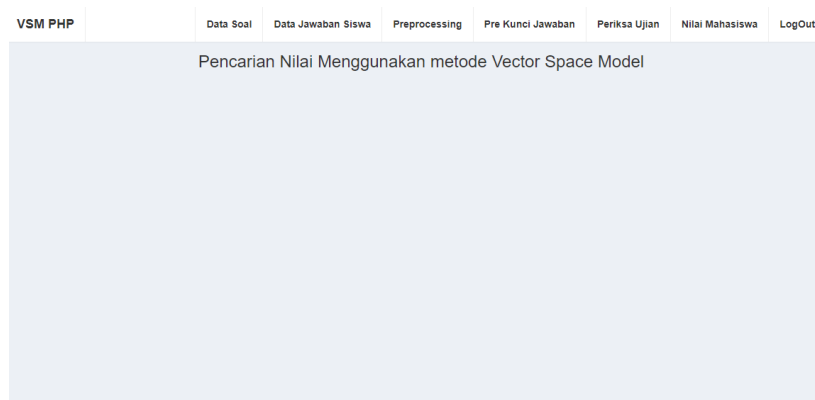
Form login adalah bentuk autentikasi *user* untuk masuk kedalam sistem, seperti terlihat pada gambar 3.

Gambar 3. Halaman *login*

2. Halaman Dashboard

Halaman *dashboard* akan muncul setelah dosen *login* ke dalam sistem, seperti terlihat pada gambar

4.



Gambar 4. Halaman *dashboard*

3. Halaman Perhitungan Vector Space Model

Halaman ini berfungsi untuk menghitung kesamaan antara *query* atau kunci jawaban dengan jawaban mahasiswa.



VSM PHP Data Soal Data Jawaban Siswa Preprocessing Pre Kunci Jawaban Periksa Ujian Nilai Mahasiswa LogOut

Perhitungan Vector Space Model

Pilih Soal dan Masukan Jawaban

Pilih Soal *
 Jelaskan apa yang dimaksud sistem pakar ?

Masukan Jawaban *
 sistem pakar program komputer pakar rancang memodelkan penyelesaian masalah sistem pakar model prosedur kait domain tingkat ahli banding ahli pakar komputer menyelesaikan masalah sistem pakar sistem komputer meniru pakar sistem pakar sistem komputer meniru pakar pakar maksud punya ahli khusus menyelesaikan masalah diselesaikan awam sistem pakar sistem usaha adopsi pengetahuan manusia komputer komputer menyelesaikan masalah laku ahli

Proses VSM

Kemunculan term dalam kalimat

Gambar 5. Halaman perhitungan *vector space model*

4. Tahapan Perhitungan Sistem Penilaian Esai

1) Teks Preprocessing

Setelah data pertanyaan soal, jawaban mahasiswa, dan kunci jawaban diinput, tahap selanjutnya yaitu dilakukan text preprocessing. Tahap text preprocessing terdiri dari tokenizing, stopword removal, dan stemming, seperti terlihat pada tabel 1.

Tabel 1. Teks *query* dan jawaban mahasiswa

<i>Query</i>	Kecerdasan buatan merupakan bagian dari ilmu pengetahuan komputer yang khusus ditujukan dalam perancangan otomatisasi tingkah laku cerdas dalam sistem kecerdasan komputer. kecerdasan buatan adalah kawasan penelitian, aplikasi dan instruksi yang terkait dengan pemrograman komputer untuk melakukan hal yang dalam pandangan manusia adalah cerdas. kecerdasan buatan merupakan sebuah studi tentang bagaimana membuat computer melakukan hal hal yang pada saat ini dapat dilakukan lebih baik oleh manusia. kecerdasan buatan adalah kecerdasan yang ditunjukkan oleh suatu entitas buatan. Kecerdasan diciptakan dan dimasukkan ke dalam suatu mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan manusia. kecerdasan buatan merupakan studi bagaimana membuat agar komputer dapat melakukan sesuatu sebaik yang dilakukan manusia.
Dokumen 1	Kecerdasan buatan merupakan bagian dari ilmu pengetahuan komputer yang khusus ditujukan dalam perancangan otomatisasi tingkah laku cerdas dalam sistem kecerdasan komputer.
Dokumen 2	kecerdasan buatan adalah kawasan penelitian, aplikasi dan instruksi yang terkait dengan pemrograman komputer untuk melakukan hal yang dalam pandangan manusia adalah cerdas.
Dokumen 3	kecerdasan buatan merupakan sebuah studi tentang bagaimana membuat computer melakukan hal hal yang pada saat ini dapat dilakukan lebih baik oleh manusia.
Dokumen 4	kecerdasan buatan adalah kecerdasan yang ditunjukkan oleh suatu entitas buatan. Kecerdasan diciptakan dan dimasukkan ke dalam suatu mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan manusia.
Dokumen 5	kecerdasan buatan merupakan studi bagaimana membuat agar komputer dapat melakukan sesuatu sebaik yang dilakukan manusia.



2) Menghitung TF-IDF Pembobotan Kata Dan Vector Space Model

Tahapan selanjutnya adalah menghitung frekuensi kata berdasarkan jumlah kata didalam dokumen yang telah didapatkan dari hasil preprocessing, kemudian hitung dfi nya yaitu berapa jumlah dokumen yang mengandung kata kemunculan, setelah itu membagi total dokumen(D) dengan dfi, contoh kata 'rupa' ($5:3 = 1,667$), selanjutnya mendapatkan nilai IDF yaitu dengan rumus $\text{LOG}(D/\text{dfi})$. Hasil perhitungan dari TF-IDF dapat dilihat pada tabel 2.

Tabel 2. Hasil perhitungan TF-IDF

Term (t)	TF						dfi	D/df	IDF $\text{LOG}(D/\text{df})$
	Query	D1	D2	D3	D4	D5			
cerdas	10	3	2	1	3	1	5	1	0,000
rupa	3	1	0	1	0	1	3	1,667	0,511
ilmu	1	1	0	0	0	0	1	5	1,608
pengetahuan	1	1	0	0	0	0	1	5	1,608
komputer	5	2	1	0	1	1	4	1,250	0,223
khusus	1	1	0	0	0	0	1	5	1,608
tuju	1	1	0	0	0	0	1	5	1,608
ancang	1	1	0	0	0	0	1	5	1,608
otomatisasi	1	1	0	0	0	0	1	5	1,608
tingkah	1	1	0	0	0	0	1	5	1,608
laku	8	1	1	2	2	2	5	1	0,000
sistem	1	1	0	0	0	0	1	5	1,608
kawasan	1	0	1	0	0	0	1	5	1,608
teliti	1	0	1	0	0	0	1	5	1,608
aplikasi	1	0	1	0	0	0	1	5	1,608
instruksi	1	0	1	0	0	0	1	5	1,608
kait	1	0	1	0	0	0	1	5	1,608
pemrograman	1	0	1	0	0	0	1	5	1,608
pandang	1	0	1	0	0	0	1	5	1,608
manusia	4	0	1	1	1	1	4	1,250	0,223
buah	1	0	0	1	0	0	1	5	1,608
studi	2	0	0	1	0	1	2	2,5	0,916
computer	1	0	0	1	0	0	1	5	1,608
saat	1	0	0	1	0	0	1	5	1,608
baik	2	0	0	1	0	1	2	2,500	0,916
entitas	1	0	0	0	1	0	1	5	1,608
cipta	1	0	0	0	1	0	1	5	1,608
masuk	1	0	0	0	1	0	1	5	1,608
mesin	1	0	0	0	1	0	1	5	1,608
kerja	1	0	0	0	1	0	1	5	1,608



Tahapan selanjutnya setiap kata yang ditemukan dihitung bobotnya, yaitu TF kemunculan kata pada dokumen dikalikan dengan nilai IDF.

Tabel 3. Pembobotan TF-IDF

Query	W				
	D1	D2	D3	D4	D5
0,000	0,000	0,000	0,000	0,000	0,000
1,532	0,511	0,000	0,511	0,000	0,511
1,608	1,608	0,000	0,000	0,000	0,000
1,608	1,608	0,000	0,000	0,000	0,000
1,115	0,446	0,223	0,000	0,223	0,223
1,608	1,608	0,000	0,000	0,000	0,000
1,608	1,608	0,000	0,000	0,000	0,000
1,608	1,608	0,000	0,000	0,000	0,000
1,608	1,608	0,000	0,000	0,000	0,000
1,608	1,608	0,000	0,000	0,000	0,000
0,000	0,000	0,000	0,000	0,000	0,000
1,608	1,608	0,000	0,000	0,000	0,000
1,608	0,000	1,608	0,000	0,000	0,000
1,608	0,000	1,608	0,000	0,000	0,000
1,608	0,000	1,608	0,000	0,000	0,000
1,608	0,000	1,608	0,000	0,000	0,000
1,608	0,000	1,608	0,000	0,000	0,000
1,608	0,000	1,608	0,000	0,000	0,000
1,608	0,000	1,608	0,000	0,000	0,000
0,892	0,000	0,223	0,223	0,223	0,223
1,608	0,000	0,000	1,608	0,000	0,000
1,831	0,000	0,000	0,916	0,000	0,916
1,608	0,000	0,000	1,608	0,000	0,000
1,608	0,000	0,000	1,608	0,000	0,000
1,831	0,000	0,000	0,916	0,000	0,916
1,608	0,000	0,000	0,000	1,608	0,000
1,608	0,000	0,000	0,000	1,608	0,000
1,608	0,000	0,000	0,000	1,608	0,000
1,608	0,000	0,000	0,000	1,608	0,000
1,608	0,000	0,000	0,000	1,608	0,000



Tabel 4. Perhitungan VSM

Term (t)	Query ²	D1 ²	D2 ²	D3 ²	D4 ²	D5 ²
cerdas	0,000	0,000	0,000	0,000	0,000	0,000
rupa	2,346	0,261	0,000	0,261	0,000	0,261
ilmu	2,587	2,587	0,000	0,000	0,000	0,000
pengetahuan	2,587	2,587	0,000	0,000	0,000	0,000
komputer	1,243	0,199	0,050	0,000	0,050	0,050
khusus	2,587	2,587	0,000	0,000	0,000	0,000
tuju	2,587	2,587	0,000	0,000	0,000	0,000
ancang	2,587	2,587	0,000	0,000	0,000	0,000
otomatisasi	2,587	2,587	0,000	0,000	0,000	0,000
tingkah	2,587	2,587	0,000	0,000	0,000	0,000
laku	0,000	0,000	0,000	0,000	0,000	0,000
sistem	2,587	2,587	0,000	0,000	0,000	0,000
kawasan	2,587	0,000	2,587	0,000	0,000	0,000
teliti	2,587	0,000	2,587	0,000	0,000	0,000
aplikasi	2,587	0,000	2,587	0,000	0,000	0,000
instruksi	2,587	0,000	2,587	0,000	0,000	0,000
kait	2,587	0,000	2,587	0,000	0,000	0,000
pemrograman	2,587	0,000	2,587	0,000	0,000	0,000
pandang	2,587	0,000	2,587	0,000	0,000	0,000
manusia	0,796	0,000	0,050	0,050	0,050	0,050
buah	2,587	0,000	0,000	2,587	0,000	0,000
studi	3,354	0,000	0,000	0,839	0,000	0,839
computer	2,587	0,000	0,000	2,587	0,000	0,000
saat	2,587	0,000	0,000	2,587	0,000	0,000
baik	3,354	0,000	0,000	0,839	0,000	0,839
entitas	2,587	0,000	0,000	0,000	2,587	0,000
cipta	2,587	0,000	0,000	0,000	2,587	0,000
masuk	2,587	0,000	0,000	0,000	2,587	0,000
mesin	2,587	0,000	0,000	0,000	2,587	0,000
kerja	2,587	0,000	0,000	0,000	2,587	0,000
SUM	70,594	21,156	18,209	9,748	13,035	2,037
SQRT	8,402	4,600	4,267	3,122	3,610	1,427

Tabel 5. Perhitungan VSM

Term (t)	Q ² * D1	Q ² * D2	Q ² * D3	Q ² * D4	Q ² * D5
cerdas	0,000	0,000	0,000	0,000	0,000
rupa	2,346	0,000	2,346	0,000	2,346
ilmu	2,587	0,000	0,000	0,000	0,000
pengetahuan	2,587	0,000	0,000	0,000	0,000
komputer	2,487	1,243	0,000	1,243	1,243
khusus	2,587	0,000	0,000	0,000	0,000



tuju	2,587	0,000	0,000	0,000	0,000
ancang	2,587	0,000	0,000	0,000	0,000
otomatisasi	2,587	0,000	0,000	0,000	0,000
tingkah	2,587	0,000	0,000	0,000	0,000
laku	0,000	0,000	0,000	0,000	0,000
sistem	2,587	0,000	0,000	0,000	0,000
kawasan	0,000	2,587	0,000	0,000	0,000
teliti	0,000	2,587	0,000	0,000	0,000
aplikasi	0,000	2,587	0,000	0,000	0,000
instruksi	0,000	2,587	0,000	0,000	0,000
kait	0,000	2,587	0,000	0,000	0,000
pemrograman	0,000	2,587	0,000	0,000	0,000
pandang	0,000	2,587	0,000	0,000	0,000
manusia	0,000	0,796	0,796	0,796	0,796
buah	0,000	0,000	2,587	0,000	0,000
studi	0,000	0,000	3,354	0,000	3,354
computer	0,000	0,000	2,587	0,000	0,000
saat	0,000	0,000	2,587	0,000	0,000
baik	0,000	0,000	3,354	0,000	3,354
entitas	0,000	0,000	0,000	2,587	0,000
cipta	0,000	0,000	0,000	2,587	0,000
masuk	0,000	0,000	0,000	2,587	0,000
mesin	0,000	0,000	0,000	2,587	0,000
kerja	0,000	0,000	0,000	2,587	0,000
SUM	25,528	20,148	17,610	14,974	11,093

Tabel 6. Perhitungan Kemiripan

Kemiripan	Nilai kemiripan
Sim(D1,Q)	$25,528/(8,402*4,600) = 0,661$
Sim(D2,Q)	$20,148/(8,402*4,267) = 0,562$
Sim(D3,Q)	$17,610/(8,402*3,122) = 0,671$
Sim(D4,Q)	$14,974/(8,402*3,610) = 0,494$
Sim(D5,Q)	$11,093/(8,402*1,427) = 0,925$

3) Konversi Nilai Kemiripan Menjadi Nilai Ujian Esai

Nilai kemiripan yang dihasilkan sebelumnya dikonversi menjadi nilai jawaban ujian esai mahasiswa berdasarkan rentang nilai versi penilaian manusia (*human rates*). Adapun rentang nilai tersebut adalah seperti yang ditunjukkan oleh tabel 7 (Fuat, 2010).



Tabel 7. Rentang nilai jawaban mahasiswa

Perbandingan Rentang Nilai	
Nilai Kemiripan	Nilai <i>Human Rates</i>
0.01 - 0.10	10
0.11 - 0.20	20
0.21 - 0.30	30
0.31 - 0.40	40
0.41 - 0.50	50
0.51 - 0.60	60
0.61 - 0.70	70
0.71 - 0.80	80
0.81 - 0.90	90
0.91 - 1	100

Maka konversi nilai dari dokumen jawaban mahasiswa yang sudah dihitung sebelumnya dapat dilihat pada tabel 8.

Tabel 8. Hasil konversi nilai

Dokumen	Nilai Kemiripan	Nilai <i>Human Rates</i>
D1	0,661	70
D2	0,562	60
D3	0,671	70
D4	0,494	50
D5	0,925	100

SIMPULAN

Penelitian ini telah berhasil mengembangkan sistem penilaian ujian otomatis untuk soal esai menggunakan metode *vector space model*. Sistem yang dikembangkan telah melalui proses pengujian blackbox dengan hasil pengujian sesuai untuk semua skenario pengujian. Berdasarkan hasil penelitian yang dilakukan menggunakan data sampel jawaban mahasiswa untuk satu soal yang telah dihitung kemiripannya berdasarkan kata kunci jawaban, maka jawaban mahasiswa D1 menghasilkan nilai kemiripan 0,661 dan dikonversikan berdasarkan rentang penilaian manusia yaitu mendapatkan nilai 70. Nilai kemiripan mahasiswa D2 adalah 0,562 dikonversi menjadi 60. Nilai kemiripan mahasiswa D3 adalah 0,671 dikonversi menjadi 70. Nilai kemiripan mahasiswa D4 adalah 0,494 dikonversi menjadi 50. Nilai kemiripan mahasiswa D5 adalah 0,925 dikonversi menjadi 100.

DAFTAR RUJUKAN

- Bastian, A., Sujadi, H., & Sukmana, P. A. (1861). *Rancang Bangun Aplikasi Penilaian Ujian Essay Dengan Menggunakan Algoritma Nazief & Andriani Dan Metode Cosine Similarity*. 62–68.
- Febriyanto, F. (2019). Sistem Penilaian Otomatis Jawaban Esai Dengan Menggunakan Metode Vector



- Space Model Pada Beberapa Perkuliahan Di Stmik Indonesia Banjarmasin. *Jurnal Teknologi Informasi*, XIV(1), 53–68.
- Fuat, R. (2010). Sistem Nilai Essay Otomatis Pada Elearning menggunakan Metode Cosine Similarity. *Surabaya: Buku Tugas Akhir Mahasiswa Jurusan Sistem Informasi Fakultas Teknologi Informasi Institut Teknologi Sepuluh November*.
- Hidayat, N., & Afuan, L. (2019). Penilaian Ujian Otomatis untuk Soal Bertipe Essay pada PJJ APTIKOM menggunakan Cosine Similarity. *Seminar Nasional APTIKOM (SEMNASTIK)*, 259–271.
- Ismi, D. P., & Ardianto, F. (2020). Peringkasan Ekstraktif Teks Bahasa Indonesia dengan Pendekatan Unsupervised Menggunakan Metode Clustering. *Cybernetics*, 3(02), 90. <https://doi.org/10.29406/cbn.v3i02.2290>
- Pitriyani, E., & Ardiansyah. (2019). *Rekomendasi Pembimbing dan Penguji Tugas Akhir Menggunakan Vector Space Model*. 2019(November), 317–323.
- Putra, R. E., & Khusartantya, K. (2014). *Aplikasi Penilaian Tes Esai Berbahasa Indonesia Menggunakan Vector Space Model (VSM)*. Universitas Diponegoro.
- Rohayati, Ratnawati, & Kunda, A. (2018). Implementasi Sistem Ujian Berbasis Online Pada Ujian Esai Bahasa Indonesia. *Inspiration: Jurnal Teknologi Informasi Dan Komunikasi*, 8(2), 81–88. <https://doi.org/10.35585/inspir.v8i2.2463>
- Sanjaya, F. (2017). Pemanfaatan Sistem Temu Kembali Informasi dalam Pencarian Dokumen Menggunakan Metode Vector Space Model. *J-INTECH (Journal Information and Technology)*, 05(02), 1689–1699.
- Saputra, E. P., Studi, P., Informasi, S., Bina, U., Informatika, S., Cengkareng, K., & Barat, K. J. (2019). *Implementasi Information Retrieval System Menggunakan Teknik Vector Space Models (VSM) Untuk Sistem Pengujian Komputer Berbasis Teks*. 11(2), 9–18.
- Tria, A., & Arifin, J. (2017). Perancangan Sistem Koreksi Otomatis Ujian Online Multiple Choice Dan Essay Pada Kuliah Mikroprosesor Berbasis Tf-Idf Dan Vektor Space Model Di Stmik Asia Malang. *Jouticla*, 3(2), 81–90.
- Yustiana, D. (2015). Penilaian Otomatis Terhadap Jawaban Esai Pada Soal Berbahasa Indonesia Menggunakan Latent Semantic Analysis. *Seminar Nasional "Inovasi Dalam Desain Dan Teknologi" - IDeaTech 2015*.
- Zeniarta, J., Salam, A., & Achsanu, I. (2020). Sistem Koreksi Jawaban Esai Otomatis (E-Valuation) dengan Vector Space Model pada Computer Based Test (CBT). *Seminar Nasional Dinamika Informatika*.