

RANCANG BANGUN APLIKASI PENILAIAN UJIAN ESSAY DENGAN MENGGUNAKAN ALGORITMA NAZIEF & ANDRIANI DAN METODE COSINE SIMILARITY

Ade Bastian¹, Harun Sujadi², Pendi Angga Sukmana³

^{1,2,3}Program Studi Informatika, Fakultas Teknik, Universitas Majalengka
Email: ¹adb@ft.unma.ac.id, ²hns@ft.unma.ac.id, ³anggaresol@gmail.com

ABSTRAK

Ujian Essay adalah salah satu proses evaluasi yang sering dilaksanakan oleh setiap bidang pendidikan untuk mengetahui kemampuan siswa terutama mahasiswa. Dengan dilaksanakannya ujian berbasis essay, mahasiswa tidak bisa menebak jawaban dalam pengisian soal essay dan dosen dapat menilai perkembangan kemampuan mahasiswanya dalam menjawab pertanyaan. Namun ada beberapa masalah yang sering terjadi yaitu pada saat dosen menilai hasil jawaban ujian essay, waktu yang dibutuhkan untuk memeriksa jawaban secara manual akan lama dan nilai yang di berikan kepada tiap mahasiswa sering tidak konsisten padahal jawaban essay beberapa mahasiswa tersebut sama. Untuk mengatasi masalah-masalah tersebut diperlukan sebuah aplikasi yang membuat pekerjaan penilaian jawaban ujian essay lebih cepat dari biasanya dan untuk menghitung nilai bisa disamakan dengan kunci jawaban dengan menggunakan metode Cosine Similarity sehingga penilaian jawaban lebih akurat dengan kunci jawaban. Supaya penilaian kunci jawaban lebih akurat diperlukan stemming terlebih dahulu. Yang mana kata yang berimbuhan kembali ke kata dasar. Proses tersebut menggunakan Algoritma Nazief dan Adriani. Setelah metode Cosine Similarity dan algoritma Nazief & Adriani dimasukkan kedalam aplikasi tepatnya dalam proses perhitungan nilai jawabannya, maka waktu penilaian jawaban essay akan cepat serta nilai yang didapatkan mahasiswa akan lebih konsisten.

Kata Kunci: Ujian Essay, Algoritma Nazief & Andriani, Cosine Similarity

1. PENDAHULUAN

1.1. Latar Belakang

Berisi Pendidikan adalah pembelajaran [pengetahuan](#), [keterampilan](#), dan kebiasaan sekelompok orang yang diturunkan dari satu generasi ke generasi berikutnya melalui pengajaran, pelatihan, atau [penelitian](#). Pendidikan sering terjadi di bawah bimbingan orang lain, tetapi juga memungkinkan secara otodidak. Setiap pengalaman yang memiliki efek formatif pada cara orang berpikir, merasa, atau tindakan dapat dianggap pendidikan. Pendidikan umumnya dibagi menjadi tahap seperti prasekolah, sekolah dasar, sekolah menengah dan kemudian perguruan tinggi, universitas atau magang. (Dewey, 1997)

Umumnya didalam setiap dunia pendidikan seperti di perguruan tinggi atau universitas selalu diselenggarakan masa perkuliahan atau penyampaian materi dan praktek. Masa perkuliahan yang terkait dengan mata kuliah yang beragam seperti Metode Numerik, Bahasa Pemrograman, Komputer Grafis, Metodologi Berbasis Objek dan mata kuliah lainnya akan disampaikan oleh dosen kepada setiap mahasiswa.

Agar dapat mengetahui apa yang di dapat oleh mahasiswanya, di tengah dan di akhir pembelajaran atau perkuliahan pasti selalu diadakan yang namanya ujian, jika di tengah masa pembelajaran atau perkuliahan namanya Ujian Tengah Semester (UTS) dan jika di akhir pembelajaran atau perkuliahan namanya Ujian Akhir Semester (UAS). Biasanya yang menjadi soal ujian berupa pilihan ganda dan esai yang harus di isi. Seperti yang kita ketahui, cara

mengisi soal pilihan ganda yaitu dengan memilih salah satu jawaban yang telah di sediakan. Jika jawaban yang kita pilih sesuai maka jawaban benar, jika jawaban yang kita pilih tidak sesuai maka jawaban salah. Pada soal pilihan ganda hanya ada 2 kemungkinan jawaban, yaitu benar atau salah. Jadi seorang dosen tidak memakan waktu lama dalam menilai jawaban yang di dapatkan dari pilihan ganda. Karena tinggal menghitung berapa banyak jawaban yang benar dan berapa banyak jawaban yang salah.

Namun beda halnya dengan soal essay yang menuntut mahasiswanya memberikan jawaban dengan pemahaman yang mereka punya. Karena setiap mahasiswa pasti mempunyai pemahaman atau pendapatnya masing-masing. Soal essay yang jawabannya perlu pemahaman atau deskripsi dari setiap mahasiswa pasti akan berbeda-beda dan akan mempunyai penilaian yang beragam. Kemungkinan dari jawaban essay tidak hanya salah dan benar, tapi ada juga kemungkinan hampir benar. Dengan demikian pada dasarnya soal essay setiap soalnya di beri nilai yang beragam. Misalkan nilai jawaban sempurna dari soal A adalah 100, jika salah diberi nilai 30, jika mendekati dikasih nilai 50 dan seterusnya. Dosen juga seorang manusia yang daya ingatnya terbatas serta kemungkinan melakukan kesalahan dalam penilaian. Bisa saja ada dua mahasiswa yang menjawab soal yang sama dengan jawaban yang sama pula dengan jawaban hampir benar, namun dengan nilai yang berbeda dikarenakan kurang telitinya dosen dalam menentukan penilaian. Baik itu disebabkan karena

tulisan mahasiswa yang kurang jelas sehingga mengakibatkan dosen susah dalam membaca jawabannya atau karena lembar jawaban mahasiswa X berada di urutan pertama dan mahasiswa Y berada di urutan terakhir yang mengakibatkan dosen lupa bahwa jawaban mahasiswa X dan mahasiswa Y itu sama.

Tidak hanya masalah dalam penilaian, akan tetapi masalah dengan waktu juga. Soal essay akan membutuhkan waktu yang lebih lama dalam penilaian jawabannya. Itu dikarenakan dalam penilaian jawaban essay perlu di baca dan teliti terlebih dahulu. Apakah jawabannya sama persis, salah, kurang atau bahkan berkembang. Untuk itu dalam penilaian jawaban dari soal essay akan membutuhkan waktu yang lumayan lama dibandingkan dengan penilaian jawaban pilihan ganda.

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 di rancang 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.

1.2. Tinjauan Pustaka

1.2.1 Algoritma Nazief & Adriani

Konjungsi adalah Algoritma *stemming* Nazief dan Adriani dikembangkan berdasarkan aturan morfologi Bahasa Indonesia yang mengelompokkan imbuhan menjadi awalan (*prefix*), sisipan (*infix*), akhiran (*suffix*) dan gabungan awalan akhiran (*confixes*). Algoritma ini menggunakan kamus kata dasar dan mendukung *recoding*, yakni penyusunan kembali kata-kata yang mengalami proses *stemming* berlebih. Aturan morfologi Bahasa Indonesia mengelompokkan imbuhan ke dalam beberapa kategori sebagai berikut (Firdaus, Ernawati, & Vatesia, 2014):

1. *Inflection suffixes* yakni kelompok akhiran yang tidak merubah bentuk kata dasar. Sebagai contoh, kata “duduk” yang diberikan akhiran “-lah” akan menjadi “duduklah”. Kelompok ini dapat dibagi menjadi dua:
 - *Particle (P)* atau partikel yakni termaksud di dalamnya “-lah”, “kah”, “tah” dan “pun”.

- *Possessive pronoun (PP)* atau kata ganti kepunyaan, termaksud di dalamnya “-ku”, “-mu” dan “-nya”.
2. *Derivation suffixes (DS)* yakni kumpulan akhiran asli Bahasa Indonesia yang secara langsung ditambahkan pada kata dasar yaitu akhiran “-i”, “-kan”, dan “-an”.
 3. *Derivation prefixes (DP)* yakni kumpulan awalan yang dapat langsung diberikan pada kata dasar murni, atau pada kata dasar yang sudah mendapatkan penambahan sampai dengan 2 awalan. Termaksud di dalamnya adalah:
 - Awalan yang dapat bermorfologi (“me-“, “be-“, “pe-“ dan “te”).
 - Awalan yang tidak bermorfologi (“di-“, “ke-“ dan “se-“).

Berdasarkan pengklasifikasi imbuhan-imbuhan di atas, maka bentuk kata berimbuhan dalam Bahasa Indonesia dapat dimodelkan sebagai berikut (Firdaus, Ernawati, & Vatesia, 2014):

$$[DP+ [DP+ [DP+]]] \text{ Kata Dasar } [[+DS] [+PP]$$

Gambar Error! No text of specified style in document..1Model Kata Berimbuhan dalam Bahasa Indonesia

Keterangan :

DP :*Derivation prefixes*

DS :*Derivation suffixes*

PP :*Possessive pronoun*

Dengan model bahasa Indonesia di atas serta aturan-aturan dasar morfologi Bahasa Indonesia, aturan yang digunakan dalam proses algoritma Nazief & Adriani sebagai berikut (Firdaus, Ernawati, & Vatesia, 2014):

1. Tidak semua kombinasi awalan dan akhiran diperbolehkan. Kombinasi-kombinasi imbuhan yang tidak diperbolehkan, yaitu “be-i”, “ke-i”, “ke-kan”, “me-an”, “se-i”, “se-kan” dan “te-an”.
2. Penggunaan imbuhan yang sama secara berulang tidak diperkenankan.
3. Jika suatu kata hanya terdiri dari satu atau dua huruf, maka proses tidak dilakukan.
4. Penambahan suatu awalan tertentu dapat mengubah bentuk asli kata dasar, ataupun awalan yang telah diberikan sebelumnya pada kata dasar bersangkutan. Sebagai contoh, awalan “me-“ dapat berubah menjadi “meng-“, “men-“, “meny-“, dan “mem-“. Oleh karena itu diperlukan suatu aturan yang mampu mengatasi masalah morfologi ini.

Algoritma Nazief & Adriani memiliki tahap-tahap sebagai berikut (Firdaus, Ernawati, & Vatesia, 2014):

1. Cari kata dalam kamus jika ditemukan maka diasumsikan bahwa kata tersebut adalah kata dasar. Algoritma berhenti. Jika tidak ditemukan maka lakukan langkah 2.

2. Hilangkan *inflectional suffixes* bila ada. Dimulai dari *inflectional particle* (“-lah”, “-kah”, “-tah” dan “-pun”), kemudian *possessive pronoun* (“-ku”, “-mu” dan “-nya”). Cari kata pada kamus jika ditemukan algoritma berhenti, jika kata tidak ditemukan dalam kamus lakukan langkah 3.
3. Hilangkan *derivation suffixes* (“-an”, “-i” dan “-kan”). Jika akhiran “-an” dihapus dan ditemukan akhiran “-k”, maka akhiran “-k” dihapus. Cari kata pada kamus jika ditemukan algoritma berhenti, jika kata tidak ditemukan maka lakukan langkah 4.

4. Pada langkah 4 terdapat tiga iterasi.

a. Iterasi berhenti jika :

- 1) Ditemukannya kombinasi awalan yang tidak diizinkan berdasarkan awalan
Tabel Error! No text of specified style in document..1 Kombinasi Awalan Akhiran yang Tidak Diizinkan

Awalan	Akhiran yang tidak diizinkan
be-	-i
di-	-an
ke-	-i, -kan
me-	-an
se-	-i, kan

- 2) Awalan yang dideteksi saat ini sama dengan awalan yang dihilangkan sebelumnya.
- 3) Tiga awalan telah dihilangkan.

b. Identifikasikan tipe awalan dan hilangkan. Awalan terdiri dari dua tipe:

- 1) Standar (“di-“, “ke-“, “se-“) yang dapat langsung dihilangkan dari kata.
- 2) Kompleks (“me-“, “be-“, “pe-“, “te-“) adalah tipe-tipe awalan yang dapat bermorfologi sesuai kata dasar yang mengikutinya. Oleh karena itu dibutuhkan aturan pada tabel 2.2 untuk mendapatkan hasil pemenggalan yang tepat.

Tabel Error! No text of specified style in document..2 Aturan Pemenggalan Awalan

Aturan	Format Kata	Pemenggalan
1	berV...	ber-V ... be-rV
2	berCAP...	ber-CAP... dimana C != 'r' & P != 'er'
3	berCAerV	ber-CaerV... dimana C != 'r'

Aturan	Format Kata	Pemenggalan
4	Belajar	bel-ajar
5	berC1erC2...	be-C1erC2... dimana C1 != 'r' 'l'
6	terV...	ter-V... te-rV...
7	terCerV...	ter-CerV dimana C != 'r'
8	terCP...	Ter-CP... dimana C != 'r' dan P != 'er'
9	teC1erC2...	Te-C1erC2... dimana C1 != 'r'
10	me{l r w y}V ...	me - {l r w y} V...
11	mem{b f v}...	mem- {b f v}...
12	Mempe	mem-pe...
13	mem{rV V}...	me-m{rV V}... me-p{rV V}
14	men{c d j s z} ...	men- {c d j s z}...
15	menV...	me-nV... me-tV
16	meng{g h q k} ...	meng- {g h q k}...
17	mengV...	meng-V... meng-kV... mengV-... jika V='e'
18	menyV...	meny-sV....
19	mempA...	mem-pA... dimana A != 'e'
20	pe{w y}V...	pe- {w y}V...
21	perV...	per-V... pe-rV...
23	perCAP...	per-CAP... dimana C != 'r' dan P != 'er'
24	perCAerV...	per-CAerV... dimana C != 'r'
25	pem{b f V}...	pem- {b f V}...
26	pem{rV V}...	pe-m{rV V}... pe-p{rV V}...
27	pen{c d j z}...	pen- {c d j z}...
28	penV...	pe-nV... pe-tV...
29	pengC...	peng-C...

Aturan	Format Kata	Pemenggalan
30	pengV...	peng-V... peng-kV... pengV-... jika V='e'
31	penyV...	peny-sV...
32	pelV...	pe-IV... kecuali "pelajar" yang menghasilkan "ajar"
33	peCerV...	Per-erV ... dimana C!={r w y l m n}
34	peCP Pe-CP...	dimana C!={r w y l m n} dan P!='er'
35	terC1erC2...	ter-C1erC2... dimana C1!='r'
36	peC1erC2...	pe-C1erC2... dimana C1!={r w y l m n}

Keterangan simbol huruf
 C : huruf konsonan
 V : huruf vocal
 A : huruf vocal atau konsonan
 P : partikel atau fragmen dari setiap kata, misalnya "er"

- c. Cari kata yang telah dihilangkan awalnya. Apabila tidak ditemukan, maka langkah 4 diulang kembali. Apabila ditemukan, maka algoritma berhenti.
5. Apabila setelah langkah 4 kata dasar masih belum ditemukan, maka proses *recording* dilakukan dengan mengacu pada aturan tabel 2.2. *Recording* dilakukan dengan menambahkan karakter *recording* di awal kata yang dipenggal. Pada tabel 2.2, karakter *recording* adalah huruf kecil setelah tanda hubung ('-') dan terkadang berada sebelum tanda kurung. Sebagai contoh, kata "menangkap" (aturan 15) pada tabel 2.2, setelah dipenggal menjadi "nangkap". Karena tidak valid, maka *recording* dilakukan dan menghasilkan kata "tangkap".
6. Jika semua langkah gagal, maka input kata yang diuji pada algoritma ini di anggap sebagai kata dasar.

1.2.2 Metode Cosine Similarity

Cosine similarity adalah ukuran kesamaan yang lebih umum digunakan dalam *information retrieval* dan merupakan ukuran sudut antara vektor dokumen D_a (titik (a_x, b_x)) dan D_b (titik (a_y, b_y)). Tiap vektor tersebut merepresentasikan setiap kata dalam setiap dokumen (teks) yang dibandingkan dan

membentuk sebuah segitiga, sehingga dapat diterapkan hukum kosinus untuk menyatakan bahwa

$$\cos(C) a^2 + b^2 - c^2 / 2ab \tag{2.1}$$

dimana

$$a^2 = a_x^2 + a_y^2, b^2 = b_x^2 + b_y^2 \tag{2.2}$$

dan

$$c^2 = (b_x - a_x)^2 + (a_y - b_y)^2 \tag{2.3}$$

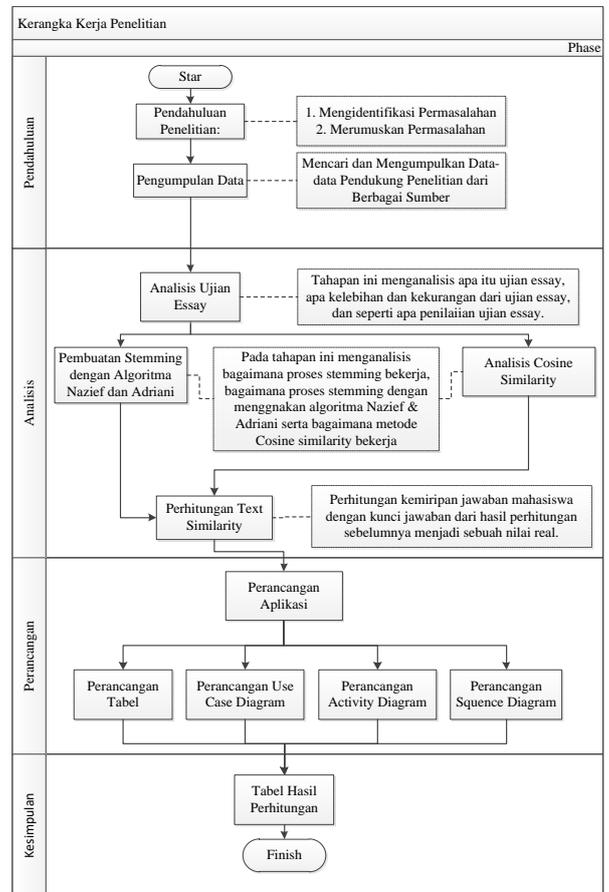
gantikan nilai-nilai tersebut untuk $a, b,$ dan $c,$ sehingga didapatkan:

$$\cos C = \frac{a_x b_x + a_y b_y}{\sqrt{a_x^2 + a_y^2} * \sqrt{b_x^2 + b_y^2}} \tag{4}$$

Ketika dua dokumen identik, sudutnya adalah nol derajat (0°) dan kesamaannya adalah satu (1); dan ketika dua dokumen tidak identik sama sekali, sudutnya adalah 90 derajat (90°) dan kesamaannya adalah nol (0).

1.3 Metodologi Penelitian

Kerangka kerja penelitian yang dilakukan untuk penelitian ini, bisa di lihat pada diagram 3.1 di bawah ini.



Gambar Error! No text of specified style in document..2 Kerangka Kerja Penelitian

2. PEMBAHASAN

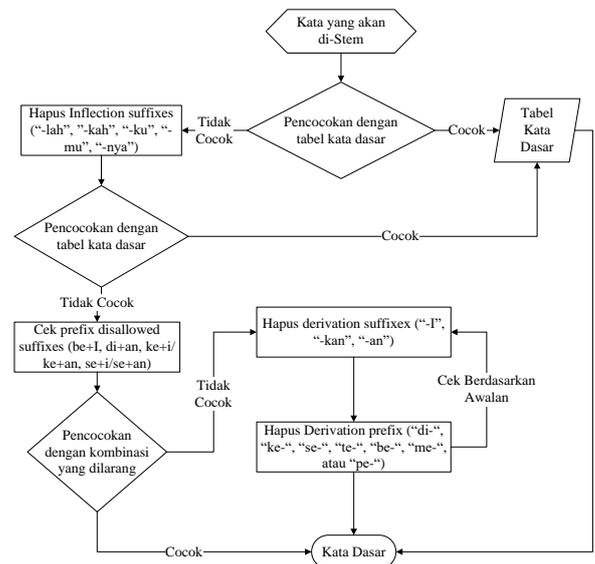
Soal *essay* merupakan salah satu bentuk evaluasi dimana pilihan jawaban dari soal tidak disediakan, dan siswa harus menjawab dengan kalimat, sehingga

jawaban dapat sangat bervariasi sesuai dengan pemikiran masing-masing peserta ujian (Fitri & Asyikin, 2015). Begitupun dengan mahasiswa yang lebih dituntut pemahamannya dalam menjawab pertanyaan dari soal essay. Sehingga akan muncul berbagai bentuk jawaban yang dihasilkan oleh setiap mahasiswa dan membuat penilaian seorang dosen akan memakan waktu lebih lama di bandingkan dalam penilaian dari jawaban pilihan ganda.

Penilaian adalah salah satu proses untuk mengambil keputusan dengan menggunakan informasi yang didapat melalui pengukuran hasil belajar baik yang menggunakan instrumen tes maupun yang tidak. Penilaian dengan metode essay menjadi pilihan dalam mengevaluasi tingkat kemampuan dari siswa walaupun faktanya tidak mudah dalam memberikan penilaian yang objektif pada setiap siswa pada saat mengevaluasi jawaban dari soal essay. Evaluasi pembelajaran dengan menggunakan bentuk esai ini oleh banyak peneliti dianggap alat yang sangat sesuai untuk mengukur hasil pembelajaran, begitu juga untuk mengamati kemahiran dan pemahaman berpikir tingkat tinggi seperti sintesis dan analisis (Fitri & Asyikin, 2015). Dari pemahaman di atas penilaian jawaban dari soal bertipe essay tidaklah mudah memberikan nilai secara objektif kepada setiap peserta ujian dalam kasus ini adalah para mahasiswa. Mengingat bahwa dosen sendiri bukanlah mesin yang mampu bertindak secara sama persis dalam setiap kegiatannya. Untuk itu akan selalu ada proses penilaian yang tidak akan sama walau jawaban mahasiswanya sama.

2.1 Analisis Stemming Menggunakan Algoritma Nazief & Adriani

Stemming merukapan proses mengubah kata menjadi kata dasar. Proses stemming yang digunakan pada sistem ini menggunakan algoritma Nazief-Adriani dikarenakan bahasa yang digunakan pada sampel ujian adalah bahasa Indonesia. Berikut ini merupakan alur dari algoritma stemming Nazief-Adriani yang dibuat (Sulistyo, Saptono, & Asshidiq, 2015).



Gambar Error! No text of specified style in document.3Sistem Stemming Berdasarkan Algoritma Nazief&Adriani

Sumber : (Sulistyo, Saptono, & Asshidiq, 2015) Diagram diatas menunjukkan bagaimana kerja sistem stemming berdasarkan Algoritma Nazief dan Adriani. Dari mulai awal pemilihan kata yang akan di stem sampai kata tersebut benar-benar sudah menjadi kata dasar, jika awalnya terdapat imbuhan. Berikut contoh-contoh aturan yang terdapat pada awalan sebagai pembentuk kata dasar.

- a. **Awalan SE-**
Se + semua konsonan dan vokal tetap tidak berubah
Contoh :
1) Se + bungkus = sebungkus
- b. **Awalan ME-**
Me + vokal (a,i,u,e,o) menjadi sengau “meng”
Contoh :
1) Me + inap = menginap
Me + konsonan b menjadi “mem”
Contoh :
1) Me + beri = memberi
Me + konsonan c menjadi “men”
Contoh :
1) Me + cinta = mencinta
Me + konsonan d menjadi “men”
Contoh :
1) Me + didik = mendidik
Me + konsonan g dan h menjadi “meng”
Contoh :
1) Me + gosok = menggosok
Me + konsonan j menjadi “men”
Contoh :
1) Me + jepit = menjepit
Me + konsonan k menjadi “meng” (luluh)
Contoh :
1) Me + kukus = mengukus
Me + konsonan p menjadi “mem” (luluh)
Contoh :
1) Me + pesona = mempesona

Me + konsonan s menjadi “meny” (luluh)
 Contoh :
 1) Me + sapu = menyapu
 Me + konsonan t menjadi “men” (luluh)
 Contoh :
 1) Me + tanam = menanam
 Me + konsonan (l,m,n,r,w) menjadi tetap “me”
 Contoh :
 1) Me + lempar = melempar

2.2 Analisis Metode Cosine Similarity

Tahapan ini mmberikan contoh manual perhitungan kemiripan teks jawban dengan menggunakan metode Cosine Similarity. Diketahui terdapat 5 dokumen (D1 s.d D5) sebagai beriku:

- D1. Timeline atau garis waktu adalah komponen untuk mengatur atau mengontrol jalannya animasi.
- D2. Timeline atau garis waktu merupakan komponen yang digunakan untuk mengatur atau mengontrol jalannya animasi.
- D3. Jalannya animasi dapat di kontrol atau di atur dengan menggunakan komponen garis waktu atau timeline.
- D4. Timeline adalah komponen yang terdapat dalam flash.
- D5. Timeline disebut juga layar atau panggung.

Dan terdapat kunci jawaban dalam Query (Q) : “timeline adalah komponen untuk mengatur atau mengontrol jalannya animasi”. Dengan demikian daftar dokumen yang paling relevan dengan Query dapat di tentukan dengan langkah-langkah sebagai berikut:

Langkah 1. *Preprocessing* terhadap semua (n= 6) dokumen yang terlibat, yaitu Q, D1, D2, D3, D4 dan D5.

Langkah 1a: Lakukan tokenisasi, *stop words removal* dan *stemming*. Hasilnya diperlihatkan pada tabel 3.1 berikut :

Tabel Error! No text of specified style in document..3 Term hasil Setemming

Dokumen	Term yang mewakili dokumen
Q	timeline komponen atur kontrol jalan animasi
D1	Timeline garis waktu komponen atur kontrol jalan animasi
D2	Timeline garis waktu rupa komponen guna atur kontrol jalan animasi
D3	Jalan animasi dapat kontrol atur guna komponen garis waktu timeline
D4	Timeline komponen dapat dalam flash
D5	Timeline sebut layar panggung

Langkah 2c: Terapkan rumus Cosine Similarity. Hitung kemiripan Q dengan D1, D2 dan seterusnya

sampai dengan D5. Berikut adalah cara menghitungnya :

$$\begin{aligned} \text{Cos (Q, D1)} &= 0.130145/(0.36*0.56) = 0.130145/0.2016 = 0.645 \\ \text{Cos (Q, D2)} &= 0.130145/(0.36*1.07) = 0.130145/0.3852 = 0.340 \\ \text{Cos (Q, D3)} &= 0.130145/(0.36*0.87) = 0.130145/0.3132 = 0.415 \\ \text{Cos (Q, D4)} &= 0.006241/(0.36*1.20) = 0.006241/0.432 = 0.014 \\ \text{Cos (Q, D5)} &= 0/(0.36*1.35) = 0 /0.486 = 0 \end{aligned}$$

Dari hasil perhitungan diatas dapat dimasukkan kedalam tabel 3.6 Hasil Perhitungan dibawah ini:

Tabel Error! No text of specified style in document..4 Hasil Perhitungan

D1	D2	D3	D4	D5
0.645	0.340	0.415	0.014	0

Langkah 3: Urutkan hasil perhitungan kemiripan, diperoleh data sebagai mana terlihat pada tabel 3.7 dibawah ini:

Tabel Error! No text of specified style in document..5 Urutan Hasil Perhitungan

1	2	3	4	5
D1	D3	D2	D4	D5

Dokumen yang relevan dengan Query “timeline adalah komponen untuk mengatur atau mengontrol jalannya animasi” berturut-turut dari nilai terbesar yaitu D1, D3 dan D2.

Jika lima dokumen tersebut diserahkan kepada pengguna,maka:

$$\text{Recall} = 5/5 \times 100 \% = 100\%$$

Jika diberikan kepada pengguna keenam dokumen tersebut, padahal hanya 3 dokumen yang relevan, maka:

$$\text{Presisi} = 3/5 \times 100 \% = 60\%$$

Recall dan Presisi ini digunakan sebagai parameter untuk mengetahui bagus dan tidaknya hasil pemrosesan terhadap query yang dilakukan oleh sistem temu balik informas.

2.3 Perhitungan Text Similarity

Nilai kemiripan yang dihasilkan sebelumnya dikonversi menjadi nilai jawaban ujian essay mahasiswa berdasarkan rentang nilai versi penilaian manusia (*human rates*), rentang nilai ini mengacu pada penelitian yang dilakuakn oleh Fuat tahun 2010. Adapun rentang nilai tersebut adalah seperti yang ditunjukkan oleh tabel 3.8 berikut ini:

Tabel Error! No text of specified style in document..6 Rentang Nilai Jawaban Mahasiswa

Perbandingan Rentang Nilai	
Nilai Kemiripan	Nilai Human Rates
0.01-0.10	10
0.11-0.20	20

Perbandingan Rentang Nilai	
Nilai Kemiripan	Nilai Human Rates
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
0.91-1.00	100

Sumber : (Fitri & Asyikin, 2015)

Maka konversi nilai dari dokumen jawaban mahasiswa yang sudah dihitung sebelumnya dapat dilihat pada tabel 3.9 berikut:

Tabel Error! No text of specified style in document..7 Hasil Konversi Nilai

Dokumen	Nilai Kemiripan	Nilai Human Rates
D1	0.645	70
D2	0.340	40
D3	0.415	50
D4	0.014	10
D5	0	0

3. KESIMPULAN

Penelitian yang penulis lakukan dapat disimpulkan menjadi beberapa point sebagai berikut:

- a. Dalam penilaian hasil dari ujian essay mahasiswa waktu yang digunakan seorang dosen menjadi lebih efisien;
- b. Tersedianya panduan bagi dosen yang digunakan dalam penilaian terhadap hasil dari jawaban mahasiswanya menjadi lebih konsisten.
- c. Dalam penilaian hasil dari ujian Essay yaitu dengan menggunakan Metode *Cosine Similarity* dalam perhitungan nilainya.
- d. Telah dibuatnya aplikasi untuk penilaian jawaban dari soal essay yang dapat digunakan dalam pelaksanaan ujian essay Mata Kuliah Praktikum di Fakultas Teknik Program Studi Informatika Universitas Majalengka.

PUSTAKA

- Dewey, J. (1997). *Democracy and Education*. Free Press.
- Firdaus, A., Ernawati, & Vatesia, A. (2014, April). Aplikasi Pendeteksi Kemiripan Pada Dokumen Teks Menggunakan Algoritma Nazief & Adriani Dan Metode Cosine Similarity. *Jurnal Teknologi Informasi*, 10 Nomor 1, 1-14.
- Fitri, R., & Asyikin, A. N. (2015). Aplikasi Penilaian Ujian Essay Otomatis Menggunakan

Metode Cosine Similarity. *POROS TEKNIK*, 88-94.

Sulistyo, M. E., Saptono, R., & Asshidiq, A. (2015, Juli). Penilaian Ujian Essay Menggunakan Metode Text Similarity. *TELEMATIKA*, 146-158.