

KOREKSI UJIAN ESSAY OTOMATIS DENGAN *TEXT MINING* PADA *E-LEARNING* STIKOM BALI

Ida Bagus Ketut Surya Arnawa

Program Studi Sistem Informasi

STIKOM Bali

arnawa@sikom-bali.ac.id

ABSTRACT

One of the most important things in the world of education is evaluation in the learning process. The purpose of conducting an evaluation process is to find out the extent to which students understanding a material that has been delivered. There are several ways that can be done in conducting an evaluation, one of them is by giving exam questions to measure students' abilities. STIKOM Bali has implemented online evaluation through e-learning system. The evaluation model developed in the form of essay exam questions where students are asked to input answers to each question. But in its implementation the lecturer is still experiencing difficulties and requires more time in correcting the answers to the exam questions. This can affect the decline in objectivity in the assessment. The efforts that need to be made to overcome these problems need to be developed techniques that can automatically correct the results of essay exams using the Rabin Karp algorithm. The results obtained from the development of automatic correction techniques using the Rabin Karp algorithm are that they have succeeded in making automatic corrections to the results of the essay exam by matching the answers of students with the answer key using parsing-gram and hashing. The final score of the essay exam is a number from 1 to 100.

Keywords: *Rabin-Karp, automatic correction, essay.*

ABSTRAK

Salah satu hal yang terpenting dalam dunia pendidikan adalah evaluasi dalam proses pembelajaran. Adapun tujuan melakukan proses evaluasi adalah untuk mengetahui sejauh mana pemahaman mahasiswa dalam menyerap/memahami suatu materi yang telah disampaikan. Ada beberapa cara yang dapat dilakukan dalam melakukan evaluasi salah satunya dengan memberikan soal ujian guna untuk mengukur kemampuan mahasiswa. STIKOM Bali sudah menerapkan evaluasi online melalui sistem e-learning. Model evaluasi yang dikembangkan berupa soal ujian essay dimana mahasiswa diminta menginputkan jawaban pada masing-masing soal. Namun dalam implementasinya dosen masih mengalami kesulitan dan membutuhkan waktu yang lebih dalam melakukan koreksi jawaban dari soal ujian. Hal ini dapat mempengaruhi penurunan objektivitas dalam penilaian. Adapun upaya yang perlu dilakukan untuk menanggulangi permasalahan tersebut perlu dikembangkan teknik yang dapat mengoreksi otomatis hasil ujian essay dengan menggunakan algoritma Rabin Karp. Hasil yang didapat dari pengembangan teknik koreksi otomatis menggunakan algoritma Rabin Karp yaitu sudah berhasil melakukan koreksi secara otomatis hasil ujian essay dengan cara mencocokkan jawaban mahasiswa dengan kunci jawaban menggunakan parsing -gram dan hashing. Nilai akhir dari penilaian ujian essay berupa angka dari 1 sampai 100.

Kata Kunci : Rabin-Karp, koreksi otomatis, essay.

PENDAHULUAN

Sukses atau tidaknya suatu proses pembelajaran dalam dunia pendidikan dapat diukur dengan melakukan evaluasi. Evaluasi dalam dunia pendidikan sangat penting dilakukan selain dapat mengetahui sukses atau tidaknya suatu proses pembelajaran juga dapat mengetahui tingkat kemampuan peserta didik dalam hal ini adalah mahasiswa dalam menyerap suatu materi yang disampaikan. Ada beberapa cara yang dapat ditempuh untuk melakukan evaluasi salah satunya dengan memberikan ujian tentang materi yang sudah disampaikan sebelumnya. Adapun manfaat dari dilaksanakannya ujian adalah mengukur kemampuan peserta didik dalam menyerap materi yang telah disampaikan (Clara, 2016). Ada banyak model soal ujian yang dapat diberikan untuk mengevaluasi kemampuan peserta didik diantaranya soal ujian berupa pilihan ganda dan soal ujian berupa isian / essay. Soal ujian berupa isian merupakan salah satu bentuk soal yang jawabannya harus diisi sendiri oleh peserta didik (Putri Ratna, et al, 2007).

Perguruan tinggi di bidang ICT di Bali yang terkenal salah satunya adalah STIKOM Bali. STIKOM Bali berdiri pada tahun 2002 tepatnya pada tanggal 10 Agustus 2002 dengan izin dari mendiknas RI.No.157/D/O/2002. STIKOM Bali selalu berusaha menjaga eksistensinya sebagai perguruan tinggi ICT yang terbaik. Salah satu upaya yang dilakukan STIKOM Bali dalam menjaga eksistensinya adalah dengan melakukan inovasi dalam meningkatkan kualitas pendidikan tentunya dengan memanfaatkan perkembangan teknologi. Sistem E-learning merupakan salah satu inovasi yang telah dikembangkan oleh STIKOM Bali dalam upaya meningkatkan kualitas pendidikan. Sistem E-learning memiliki beberapa fitur diantaranya upload dan download materi perkuliahan, jadwal

perkuliahan, pengumpulan tugas, forum diskusi dan ujian berupa isian/essay.

Fitur ujian berupa isian/essay merupakan fitur yang paling sering digunakan oleh dosen dalam melakukan proses evaluasi pembelajaran. Bentuknya yang berupa essay mewajibkan mahasiswa harus menulis jawabannya dengan sendiri, hal ini memungkinkan dosen bisa mengevaluasi lebih baik tingkat pemahaman peserta didiknya. Dalam pelaksanaan evaluasi dengan ujian essay dosen masih menggunakan cara manual dalam mengoreksi hasil ujian essay sehingga mengalami kendala pada saat mengoreksi hasil ujian essay yang memerlukan waktu yang cukup lama, hal ini diperparah dengan banyaknya jumlah mahasiswa yang harus dikoreksi sehingga mengakibatkan penurunan objektifitas penilaian.

Dalam upaya mempertahankan objektifitas penilaian dari seorang dosen dalam mengoreksi ujian essay, maka perlu dikembangkan suatu teknik yang dapat mengoreksi hasil ujian essay secara otomatis dan akurat yang nantinya akan diintegrasikan dengan sistem E-learning. Salah satu teknik yang dapat digunakan yaitu teknik pencocokan string dengan menggunakan Text Mining. Ada beberapa algoritma Text Mining yang dapat digunakan dalam pencocokan string salah satunya adalah algoritma Rabin Karp. Algoritma Rabin Karp melakukan pencocokan string dengan cara mencari sebuah pola berupa substring dalam sebuah string menggunakan hashing. Diharapkan dengan dikembangkannya teknik pencocokan string dapat mengatasi permasalahan yang dihadapi dosen dalam mengoreksi jawaban ujian essay.

TINJAUAN PUSTAKA

Algoritma Rabin Karp

Algoritma Rabin Karp merupakan salah satu algoritma yang dapat digunakan untuk mendeteksi kemiripan suatu dokumen teks. Adapun karakteristik dari algoritma Rabin Karp yaitu :

- Memanfaatkan fungsi hashing
- Menggunakan kompleksitas waktu $O(m)$ pada tahap preprocessing
- Pencarian kompleksitas menggunakan $O(mn)$.
- $O(n+m)$ merupakan waktu yang diperlukan.

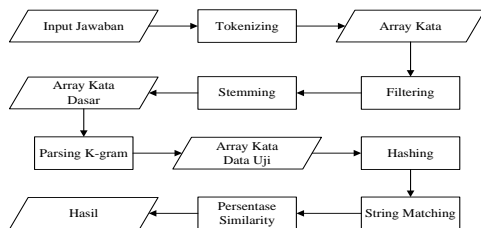
Algoritma Rabin Karp memiliki beberapa tahapan yaitu :

- Preprocessing.
- Rolling Hash.
- Pencocokan.
- Pengukuran Nilai Kemiripan

METODOLOGI PENELITIAN

Arsitektur Sistem

Arsitektur sistem digunakan untuk menggambarkan bagaimana cara mencocokkan jawaban ujian essay dengan kunci jawaban yang sudah ada. Proses diawali dengan menerima inputan berupa jawaban ujian essay, kemudian dicari informasi mengenai jumlah paragraf, jumlah kalimat dan jumlah kata dari inputan yang didapat. Proses selanjutnya dilakukan tokenizing, filtering dan stemming setelah itu baru dilakukan string matching untuk mengetahui kemiripan antara jawaban dan kunci jawaban (Gipp, B. and Meuschke, N. 2011). Gambar 1 merupakan gambaran arsitektur sistem koreksi ujian essay otomatis.



Gambar 1. Arsitektur Sistem

HASIL DAN PEMBAHASAN

Implementasi Rabin Karp

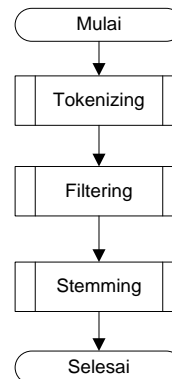
Dalam penelitian ini mahasiswa akan diwajibkan melakukan login dengan cara memasukan NIM (Nomor Induk Mahasiswa) dan password yang sudah diberikan sebelumnya. Setelah itu akan muncul halaman matakuliah yang akan dilaksanakan quiz. Mahasiswa dapat mengerjakan quiz dengan catatan waktu yang diberikan masih tersedia, untuk memulai quiz mahasiswa memilih quiz yang akan dikerjakan. Format quiz dalam bentuk essay, quiz akan menampilkan soal dan kolom untuk mengisi jawaban dari soal tersebut. Salah satu contoh soal yang dipakai dalam uji coba. Soal ini diambil dari matakuliah Pengantar Teknologi Informasi. Berikut adalah contoh soalnya :

Soal : Apakah fungsi dari motherboard ?

Kunci Jawaban : Menghubungkan seluruh komponen penyusun sebuah komputer

Jawaban Mahasiswa : Menghubungkan seluruh komponen penyusun sebuah komputer

Flowchart Preprocessing



Gambar 2. Flowchart Preprocessing

Gambar 2 menggambarkan tahapan – tahapan dalam melakukan Preprocessing, dimana tahap pertama dilakukan tokenizing yaitu tahap pemotongan string input berdasarkan tiap kata yang menyusunnya. Tahap kedua dilakukan filtering yaitu mengambil kata – kata penting dari hasil tokenizing. Tahap selanjutnya dilakukan stemming dimana tahap ini dilakukan untuk menemukan kata dasar dari sebuah kata.

Tokenizing

Tahap Tokenizing adalah tahap pemotongan string input berdasarkan tiap kata yang menyusunnya.

Tabel 1. Hasil Tokenizing

Kunci jawaban	Jawaban Mahasiswa
menghubungkan	menghubungkan
seluruh	seluruh
komponen	komponen
penyusun	penyusun
sebuah	sebuah
komputer	komputer

Filtering

Tahap filtering adalah tahap mengambil kata-kata penting dari hasil token.

Tabel 2. Hasil Filtering

Kunci jawaban	Jawaban Mahasiswa
menghubungkan	menghubungkan
seluruh	seluruh
komponen	komponen
penyusun	penyusun
sebuah	sebuah
komputer	komputer

Stemming

Stemming merupakan suatu tahap untuk menemukan kata dasar dari sebuah kata.

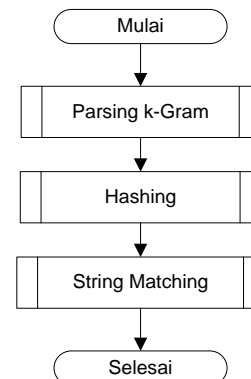
Tabel 3. Hasil Stemming

Kunci jawaban	Jawaban Mahasiswa
hubung	hubung
seluruh	seluruh
komponen	komponen
susun	susun
sebuah	sebuah
komputer	komputer

Parsing k-gram

Merupakan sebuah metode yang diaplikasikan untuk pembangkitan kata atau karakter. Metode ini digunakan untuk

Flowchart Algoritma Rabin Karp



Gambar 3. Algoritma Rabin Karp

Gambar 3 menggambarkan tahapan yang ada pada algoritma Rabin Karp. Tahap pertama dalam algoritma Rabin Karp adalah parsing-gram, dalam tahap ini dilakukan pengambilan potongan karakter huruf sejumlah K dari sebuah kata secara kontinuitas dari teks sumber. Hasing merupakan tahapan selanjutnya dimana pada proses hashing dilakukan dengan bantuan kode strlen, ord dan pow untuk mentransformasikan sebuah string menjadi nilai yang unik yang berfungsi sebagai penanda string.

pengambilan potongan karakter huruf sejumlah K dari sebuah kata yang secara kontinuitas dibaca dari teks sumber.

Tabel 4. Hasil Parsing k-gram

	Kunci Jawaban	Jawaban Mahasiswa
text	Hubungseluruhkomponensusunsebuahkomputer	Hubungseluruhkomponensusunsebuahkomputer
k-gram	{hubu} {ubun} {bung} {ungs} {ngse} {gsel} {selu} {elur} {luru} {uruh} {ruhk} {uhko} {hkom} {komp} {ompo} {mpon} {pone}	{hubu} {ubun} {bung} {ungs} {ngse} {gsel} {selu} {elur} {luru} {uruh} {ruhk} {uhko} {hkom} {komp} {ompo} {mpon} {pone}

	{onen} {nens} {ensu} {nsus} {susu} {usun} {suns} {unse} {nseb} {sebu} {ebua} {buah} {uahk} {ahko} {hkom} {komp} {ompu} {mput} {pute} {uter}	{onen} {nens} {ensu} {nsus} {susu} {usun} {suns} {unse} {nseb} {sebu} {ebua} {buah} {uahk} {ahko} {hkom} {komp} {ompu} {mput} {pute} {uter}
--	---	---

Hashing

Proses hashing merupakan proses lanjutan dari proses parsing sebelumnya. Proses hashing dilakukan dengan menggunakan bantuan kode strlen, ord dan

pow untuk mentransformasikan sebuah string menjadi suatu nilai yang unik dengan panjang tertentu yang berfungsi sebagai penanda string.

Tabel 5. Hasil Hashing

	Kunci Jawaban	Jawaban Mahasiswa
k-gram	{hubu} {ubun} {bung} {ungs} {ngse} {gsel} {selu} {elur} {luru} {uruh} {ruhk} {uhko} {hkom} {komp} {ompo} {mpon} {pone} {onen} {nens} {ensu} {nsus} {susu} {usun} {suns} {unse} {nseb} {sebu} {ebua} {buah} {uahk} {ahko} {hkom} {komp} {ompu} {mput} {pute} {uter}	{hubu} {ubun} {bung} {ungs} {ngse} {gsel} {selu} {elur} {luru} {uruh} {ruhk} {uhko} {hkom} {komp} {ompo} {mpon} {pone} {onen} {nens} {ensu} {nsus} {susu} {usun} {suns} {unse} {nseb} {sebu} {ebua} {buah} {uahk} {ahko} {hkom} {komp} {ompu} {mput} {pute} {uter}
hash	153776 168982 145908 170285 160239 152227 166591 148900 159276 170912 167142 169599 152701 157159 162273 159962 163814 162272 159956 149123 161727 168604 171039 168547 170403 161534 166481 147673 145766 168715 142979 152701 157159 162279 160034 164606 170988	153776 168982 145908 170285 160239 152227 166591 148900 159276 170912 167142 169599 152701 157159 162273 159962 163814 162272 159956 149123 161727 168604 171039 168547 170403 161534 166481 147673 145766 168715 142979 152701 157159 162279 160034 164606 170988

Pada tahap string matching dilakukan setelah mendapatkan nilai hash. Tahap string matching yaitu tahap melakukan perhitungan nilai yang sama antara kunci jawaban dan jawaban mahasiswa dengan menggunakan Dice's Similarity Coefisient dengan rumus sebagai berikut :

$$S = \frac{2C}{A+B} \times 100 \dots\dots\dots \text{Persamaan (1)}$$

Keterangan:

S = nilai similarity antar kedua teks

A = jumlah dari kumpulan K-grams dalam teks 1

B = jumlah dari kumpulan K-grams dalam teks 2

C = jumlah dari K-grams yang sama dari teks yang dibandingkan.

$$S = \frac{2 * 37}{37 + 37} \times 100\%$$

$$S = 100 \%$$

Perhitungan string matching menggunakan Dice's Similarity Coefisient yang dilakukan ditemukan nilai kemiripan 100% yang artinya jawaban mahasiswa dengan kunci jawaban sama.

SIMPULAN

Algoritma Rabin Karp ini dapat digunakan untuk mengkoreksi otomatis jawaban ujian essay dengan mencocokkan kunci jawaban dengan jawaban mahasiswa. Hasil akhir dari koreksi dengan algoritma Rabin Karp berupa nilai kesamaan antara jawaban mahasiswa dengan kunci jawaban dengan menggunakan parsing k-gram dan hasing.

DAFTAR PUSTAKA

- [1] Gipp, B. and Meuschke, N., (2011). Citation pattern matching algorithms for citation-based plagiarism detection: greedy citation tiling, citation chunking and longest common citation sequence. In Proceedings of the 11th ACM symposium on Document engineering 249-258.
- [2] Putri Ratna, A. A., Budiardjo, B., & Hartanto, D. (2007). SIMPLE : Sistem Penilaian Esei Otomatis Untuk Menilai Ujian Dalam Bahasa Indonesia. Makara, Teknologi, 11(1). 5-11.
- [3] Raharja, I. Putu Dharma Ade, and I. Putu Dharma Ade Raharja. (2015). Perancangan dan Implementasi Sistem Penilaian Jawaban Esai Otomatis Menggunakan Algoritma Rabin-Karp. PhD diss., Universitas Udayana.
- [4] S. Pressman, Roger. (2012). Rekayasa Perangkat Lunak Pendekatan Praktisi Buku1 dan 2. Yogyakarta: Andi.