Implementation of Rabin Karp Algorithm for Essay Writing Test System on Organization xyz

M Misbah Musthofa
Departement of Informatics
Universitas AMIKOM Yogyakarta
Jl Ringroad Utara, Condongcatur, Depok, Sleman, Yogyakarta
Indonesia 55283
misbah.musthofa@students.amikom.ac.id

Ainul Yaqin
Informatics, Faculty of Computer Science
Universitas AMIKOM Yogyakarta
Jl Ringroad Utara, Condongcatur, Depok, Sleman, Yogyakarta
Indonesia 55283
ainulyaqin@amikom.ac.id

Abstract*— Every year the xyz organization conducts written tests for participants who will be administrators of the xyz organization, written tests are chosen because participants can convey information verbally. But it is constrained for correction, because to correct essay answers manually will require a lot of time, while the answers from participants 1 and other participants are different. By kaera the written test system owned by the xyz organization needs to be developed to facilitate correction of participants' answers to the system, so that the assessment process can be done easily and quickly. To correct answers quickly and accurately, a system is needed that can correct the answers to essays by applying the Rabin Karp algorithm. Rabin Karp is a very efficient multiple pattern search algorithm that searches for many pattern sets using the hash method in searching for words. The results of this system are expected to improve the answers to essay questions more easily and accurately. This theory is rarely used to find one word, but it is very important and very effective when used to look for several patterns. For this reason, in this study Rabin Karp's ability to detect similarities in the similarity of the participants' answers to the system will be proven by giving a similarity to the sentence or word. The process of equating sentences or words through preprocessing stages, so that the data becomes "clean", so it is feasible to do the hashing process. After hashing, the Rabin Karp algorithm is applied for each word and its similarity measurement uses the dice coefficient, which results in the same sentence value from the participant's answer and the system. The better the stemming process in the preprocessing process, the higher the value of the similarity.

Keywords*— Rabin karp, writing test, hashing, preprocessing, Similarity

I. INTRODUCTION

Tests or tests are one of the comprehensive, systematic and objective evaluations whose results can be used as the basis for decision making in the teaching process carried out by the teacher or examiner [1]. Tests are things that need to be done for someone who is learning to know the level of understanding of the material being studied. At present the exam starts working on a computer directly and can produce test results quickly. Examination by system was introduced in the world of education with the emergence of a computerized learning system which is often referred to as e-learning. In the e-learning process the type of exam that is widely used is a test with multiple choice questions and short entries. The use of multiple choice questions and short entries is chosen because it is considered easier in the assessment. But with these types of questions, it doesn't measure the ability of e-learning users. The other type of question is about essays. With the essay questions, e-learning users can practice expressing their own answers verbally and can be used to measure the level of understanding more deeply.

Essay questions are rarely used in e-learning because they are constrained in the difficulty of correction. Assessment must use a manual method by correcting one by one the answers. One effective way to create a correction system is to develop and implement an algorithm. Correction of essay questions is to match the string of answers with the answer key. The method that can be used to match strings is the string matching method. String matching is used to find a string called a pattern in a string called text [2]. The algorithms that are widely used in string matching are Wusnch Algorithm Need, Smith Watermen Algorithm, Boyer-Moore Algorithm, Brute Force Algorithm and Rabin-Karp Algorithm. Each algorithm or method certainly has its own strengths and weaknesses in matching strings.

In previous studies (Enola D'Souza1, B ShaliniPai and Ms.Suchetha Vijayakumar, 2016) compared the performance of three string matching algorithms. The algorithms compared are Brute-Force, Rabin-Karp, and Boyer-Moore algorithms. From the results of these studies it is stated that the Rabin-Karp algorithm is an effective algorithm as string matching. Rabin-Karp is also considered to be faster for matching documents with the mix-letter type (a mixture of small and large letters) compared to the brute-force algorithm. In addition, a Nazief and Andriani stemming algorithm was also inserted to obtain a good level of accuracy on similarity values.

II. EXAM CORRECTION PROBLEM

Essay questions are questions that are used to measure (goals) the achievement of learning outcomes in complex aspects. And it is recommended that the test designer measure the ability of the test participant in the form of analysis, organize and express ideas about something.[3]

problems related to the background above, namely as follows:

1) How to implement the rabin karp algorithm for correction on the essay write test system in the XYZ organization.
2) What is the percentage level of accuracy of the rabin karp algorithm on the essay writing test system in the xyz organization.

III. BASIC THEORY

A. text preprocessing

In this process, semantic analysis (truth of meaning) and syntax (arrangement of truths) on the text are carried out. The purpose of text processing is to prepare text into data that will undergo further processing. The process of text processing includes case folding, tokenizing, filtering, and stemming. [4]

B. K-gram

K-Gram is a series of terms with length K. Most of which are used as terms are words. K-Gram is a method that is applied to the generation of words or characters. The K-Gram method is used to extract letters from a number of characters from a word that is continuously read from the source text to the end of the document.[4]

C. Hashing

Hashing is a way to transform a string into a fixed-length unique value that serves as the string marker. Hash function or hash function is a way of creating fingerprints from various input data. The hash function will replace or transpose the data to create a fingerprint, which is commonly called a hash value. [5]. One of the frequently used hashes is rolling hashes. Following are the hash rolling algorithm equations:

$$H_{(a1\ldots ak)} = C_1 * b^{(k-1)} + b^{(k-2)} + \cdots + C_{(k-1)} * b^k + C_k$$

Information:

c: character ASCII value

b: prime number base

k: lots of characters

Example of searching for hashes from the word "I" with base 2.

a has an ASCII value of 97

k has an ASCII value of 107

u has an ASCII value of 117

$$H_{(aku)} = (97 * 2^2) + (107 * 2^1) + (117 * 2^0)$$
$$H_{(aku)} = 388 + 214 + 117$$
$$H_{(aku)} = 719$$

D. Rabin Karp Algorithm

Basically, the Rabin-Karp algorithm will compare the hash values of the input strings and substrings in the text. If it is the same, then a comparison will be made of the characters. If not the same, then the substring will shift to the right. The main key to the performance of this algorithm is the efficient calculation of the substring hash value at the time of the shift. [5]

The formula of Rabin Karp's algorithm is:

$$H_{(s)} \leftarrow (s[i] * b^{(n-1)} + s[i+1] * b^{(n-2)} + \cdots + s[i+n-1])$$

Where:

s: string length sought
i: array location
b: base
n: number of string lengths searched

E. Measurement of Similarity Value

To calculate the similarity value of the obtained fingerprint document, Dice's Similarity Coeficients are used by calculating the value of the number of K-Gram used in the two documents tested, while the fingerprint document is obtained from the same number of K-Gram values. The similarity value can be calculated using the formula [5]:

$$S = \frac{2C}{A+B}$$

Where:

S: Similarity value

A: The number of grams of text 1

B: Number of k-grams of text 2

C: The number of k-grams from text 1 and 2

F. Validation Value formula

Measurement is the process of comparison between an unknown quantity with a standard quantity obtained, which includes the relationship of a measuring instrument in the system with consideration and observation of the results of the response to the instrument. Measurements obtained are measurements of quantities called true values, but it is very difficult to define the actual price.

Levels where a measurement is in accordance with the expected price shown in the error requirements of the measurement. Other errors may be indicated by absolute errors or percentage errors. Absolute errors can be defined as the difference between the expected value variable and the measurement value variable. If we expect to show an error as a percentage, then we can formulate it as a result of an absolute error with the expected price multiplied by 100%. The percentage of errors can be seen in the presentation error. This percentage is needed to express the measurement for the level of accuracy. [6]

Error Percentage Formula:

$$\%Error = \frac{Yn - Xn}{Yn}$$

Information :
Y n: Expected value
X n: Measurement value

Accuracy Level Formula:

$$@ = 100\% - \%Error$$

G. Percentage Similarity

In the Annis Prastyanti study, the range of presentation similarities used are as follows. [6]

TABEL 1 Precentage Similarity

Percentage	Information
0%	The 0% test result means that the two sentences are completely different, in

	terms of the overall word
<15%	Test results 15% means that the two sentences have only a few similarities
15% - 50%	Test results 15% - 50% means that the sentence has a moderate level of similarity.
>50%	Test results of more than 50% means that it can be done that the sentence is close to similarity
100%	Test results 100% indicate that the sentence has the same resemblance because from the beginning to the end are the same

H. Confusion Matrix

Confusion matrix is a measurement tool that can be used to calculate the performance or level of truth of the classification process. With confusion matrix, it can be analyzed how well the classifier can recognize records from different classes.[7]

IV. IMPLEMENTATION OF THE RABIN KARP ALGORIHM FOR WRITEN TEST CORECCTION

A. System Description

The basis of the application for correcting essay questions on written examinations in the xyz organization uses the rabin karp algorithm. The way the application works is by receiving an answer inputted by the member and will match the key from the administrator that has been inputted by the administrator.

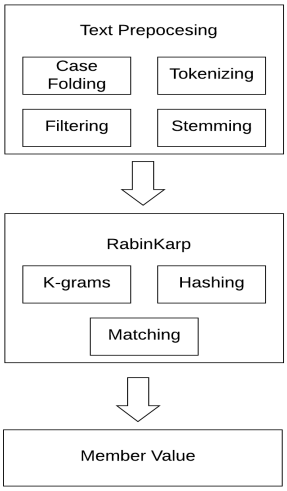

Figure 1 Step Rabin Karp Algorithm

Explanation of Figure 1

1) *Case Folding*
   At this stage there is a process of removing punctuation, numbers, and converting member answers into lowercase letters.
2) *Tokenzing*
   At this stage a member answer is broken down into words.
3) *Filtering*
   At this stage the removal of conjunctions is carried out.
4) *Stemming*
   At this stage a root search (basic word) is carried out from the word fltering.
5) *K-Grams*
   Is the stage where all stemming words are reconnected without using spaces and the combined words are divided into a number of K-grams values, in this application K-Gram value = 3
6) *Hashing*
   Is the stage of converting each word that has been divided based on the K-Grams value into a unique hash value.
7) *Matching*
   Is the process of comparing the hash value of a member's answer with the hash value of the answer key that was previously inputted by the active management of the xyz organization.
8) *Member value*
   Is a similarity calculation, calculating the same number of hashes between member answers with the answer key that was previously inputted by the active management of the xyz organization.

V. EXPERIMENTAL RESULTS

A. System Validation Test Results

TABEL 2 *Validasi Test Results*

No	Answer	Answer key	simil arity	Hash Value		accur acy
				M	S	
1	perintah untuk login super user	perintah untuk login sebagai super user	100.0 0%	11	11	100%
2	man	man	100.0 0%	1	1	100%
3	mencari diskripsi perintah sesuai dengan perintah yang dicari	mencari diskripsi perintah pada linux yang sesuai dengan kata kunci yang dicari	81.58 %	31	31	100%
4	mencari diskripsi perintah pada linux berdasar kan perintah yang	mencari diskripsi perintah pada linux yang mengandu ng kata kunci yang dicari	75.29 %	32	32	100%

	dicari					
5	open	cat	0%	0	0	100%
6	berfungsi menampilkan isi direktori	menampilkan isi dari direktori yang aktif	74.42 %	16	16	100%
7	menampilkan isi direktori	menampilkan isi dari direktori yang aktif	86.49 %	16	16	100%
8	as	alias	0%	0	0	100%
9	mencari letak sebuah file	untuk menemukan dimana letak sebuah file pada Linux	51.61 %	8	8	100%
10	mengetahui di direktori mana sekarang berada	mengetahui di direktori mana sekarang berada	100 %	7	7	100%
11	file sistem pada linux	file sistem pada linux	100 %	8	8	100%
12	short data	sort data	50%	2	2	100%
13	uniqe data	uniq data	50%	2	2	100%
14	menampilkan data yang ada string Adhe didalam file data	menampilkan baris yang ada string Adhe didalam file data	81.63 %	20	20	100%
15	menambah data dengan nama Linus Torvalds didalam file data	menambah data dengan nama Linus Torvalds didalam file data	100 %	25	25	100%
16	Membuat direktori	Membuat direktori	100 %	11	11	100%
17	membuat direktori fossil amikom yang didalamnya ada direktroi member pengurus yang didalamnya ada	membuat direktori fossil amikom yang didalamnya terdapat direktori member pengurus yang didalamnya terdapat	71.43 %	50	50	100%
	direktori rapat	direktori laporan rapat				
18	memindahkan file data ke dalam direktori pengurus	memindahkan file data ke dalam direktori pengurus	100 %	28	28	100%
19	direktori pengurus	direktori pengurus	100 %	15	15	100%
20	menghapus direktori Amikom dan isinya	menghapus direktori Amikom dan isinya	100 %	21	21	100%
21	untuk menginstall paket aplikasi psada linux turunan debian	untuk menginstall paket aplikasi pada linux turunan debian	100 %	34	34	100%
22	menginstal aplikasi dengan format paket .deb	untuk menginstal aplikasi dengan format paket .deb	100 %	20	20	100%
23	tempat untuk menyimpan paket aplikasi pada linux turunan debian	tempat untuk menyimpan paket aplikasi pada linux turunan debian	100 %	39	39	100%
24	mengupdate informasi paket	mengupdate informasi paket debian	85.71 %	18	18	100%
25	mengupgrade semua paket debian yang versinya rendah	mengupgrade semua paket debian yang versinya lebih rendah dari versi repositori	78.26 %	27	27	100%
26	membuka partisi dis	partisi dis	80.00 %	8	8	100%
27	#mount /dis	#umount /dis	25.00 %	1	1	100%
28		mnt	0%	0	0	100%

29	digunakan untuk melihat lis semua sda pada direktori dev	perintah yang digunakan untuk melihat lis semua sda pada direktori dev	83.33 %	20	20	100%
30	digunakan untuk memformat harddisk sda2 yang berada didalam folder dev	perintah yang digunakan untuk memformat harddisk sda2 yang berada didalam folder dev	84.62 %	22	22	100%
31	membatasi kemampuan user untuk mengelola data pada sebuah sistem	membatasi kemampuan user untuk mengelola data pada sebuah sistem	100 %	28	28	100%
32	membuat user tanpa ada home direktori, membuat user tanpa memberi keterangan lengkap mengenai user yang dibuat dan password	membuat user tanpa mengreset password, membuat user tanpa ada home direktori, membuat user tanpa memberi keterangan lengkap mengenai user yang dibuat	92.73 %	51	51	100%
33	membuat user dengan setting password terlebih dahulu, beserta home direktorinya,dan memberikan keterangan lengkap user yang dibuat	membuat user dengan setting password terlebih dahulu, beserta home direktorinya,dan memberikan keterangan lengkap user yang dibuat	100 %	48	48	100%
34	bilangan dengan username yang menjadi acuan sistem	Bilangan numerik dengan username yang menjadi acuan sistem	77.55 %	19	19	100%
35	karena sebagai pengaman awal pada saat login	karena, setiap user wajib memiliki password sebagai pengaman awal	38.71 %	6	6	100%
36	user tidak dapat login ke sistem	user tidak dapat login ke sistem	100 %	18	18	100%
37	groupadd fossil	groupadd fossil	100 %	12	12	100%
38	membuat user baru dengan nama fossil dan didalmnya terdapat group yang bernama amikom	membuat user baru dengan nama fossil dan didalmnya terdapat group yang bernama amikom	100 %	36	36	100%
39	didalam group fossil jadi ada user amikom, sedangkan di user amikom tetap masih ada user amikom	didalam group fossil menjadi ada user amikom, sedangkan di user amikom tetap masih ada user amikom	100 %	9	9	100%
40	hanya menampilkan group yang sedang aktif	yang terjadi adalah hanya akan menampilkan group yang sedang	100 %	9	9	100%

		aktif saja				
41	Linus Torvalds	Linus Torvalds	0%	0	0	100 %
42	Penguin	Penguins	90.91 %	5	5	100%
43	PC atau Laptop yang didalamnya terdapat dua sistem operasi yang bisa dijalankan bergantian	PC / Laptop yang didalamnya terdapat dua sistem operasi yang bisa dijalankan bergantian	100 %	43	43	100%
44	dapat digunakan secara bebas dan gratis	dapat digunakan secara bebas dan gratis	100 %	18	18	100%
45	distro	distro	100 %	4	4	100%
46	unix	unix	100 %	2	2	100%
47	slackwer	slackware	0%	0	0	100%
48	ubuntu	suse	0%	0	0	100%
49	debian	debian	100 %	4	4	100%
50	tanda bahwa shell prompt masuk sebagai root	tanda bahwa shell prompt masuk sebagai root	100 %	23	23	100%

B. Confusion Matrix Test Results

The Confusion Matrix test results from the presentation of system similarities with 50 question test data are presented below.

TABEL 3 Confusion Matrix Test Result

	True Positive	**True Negative**	
False Positive			**(*precicion*)**
False Negative	1	5	**0% (*fallout*)**
	97.56% (*sensitivity*)	**55.56% (*specificity*)**	

From the table of test results above, it can be seen the results of the presentation of the system similarity with 50 test data which obtained 90% Accuracy value with 10% Error Rate.

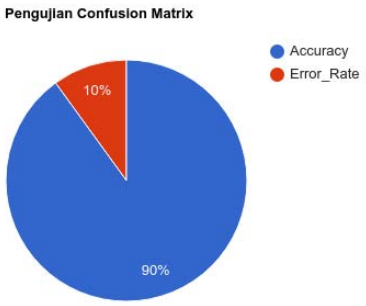

Figure 2 Diagram Pie Confusion Matrix Test Result

VI. CONCLUSION

Rab*in Karp*s algorithm can be implemented to correct essay answer questions on written test systems in xyz organizations. This study has tested the superiority of Rabin Karp algorithm by matching the answers of members and answer keys with a total of 50 answers in Indonesian and obtained 90% Accuracy and Error Rate of 10%.

REFERENCES

[1] Djaali dan Muljono, (2008). Pengukuran dalam Bidang Pendidikan,. (Jakarta: Grasindo)

[2] Charras, C. Dan Lecroq. T. (2003), *Jewels of Stringology Text Algorithm, (*Singapore)

[3] Anwar, Syafri. 2009. Penilian Berbasis Kompetensi. Padang: UNP Press

[4] Yoga, Kadek Versi Yana. (2012). "Pengembangan Aplikasi Pendeteksi Plagiarisme Pada Dokumen Teks Menggunakan Algoritma Rabin –Karp." Bali: Fakultas Teknik dan Kejuruan Universitas Pendidikan Ganesha.

[5] Moch Nurhalimi, (2017). "Implementasi Algoritma Lesk Untuk Synonim Recognition dan Rabbin Karp Pada Pendektesian Plagiarisme" Jurnal Ilmu Komputer dan Informatika (KOMPUTA)

[6] Annis Prastyanti, (2014). "Sistem Deteksi Kemiripan Kata Pada Dua Dokumen Menggunakan Algoritma Rabin-Karp" Universitas Diponegoro

[7] Triowali Rosandy, (2016). "Perbandingan Metode Nave Bayes Classifier Dengan Metode Decision Tree (C4.5) Untuk MEnganalisa Kelancaran Pembiayaan" Jurnal TIM Darmajaya