PENDETEKSIAN KEMIRIPAN TEKS DESKRIPSI DIRI PADA E-RECRUITMENT KARYAWAN DENGAN METODE RABIN KARP DAN JARO WINKLER DISTANCE

Stephanie [1] Dali Santun Naga [2] Viny Christanti Mawardi [3]

[1] Teknik Informatika, FTI, Universitas Tarumanagara
Jl. Letjen S Parman no 1, Jakarta 11440 Indonesia
Stephanie.535170075@stu.untar.ac.id [1], dalinaga@gmail.com [2], vinym@fti.untar.ac.id [3]

ABSTRACT

E-recruitment is a web-based application that is used to recruit employees for several available companies. The e-recruitment system has a plagiarism checker feature. Plagiarism itself has the meaning of actions that are considered fraud by taking someone's idea or writing without mentioning a reference and claiming it as their own. So that the e-recruitment application requires a plagiarism detector which functions to make it easier for companies to assess the characteristics of their prospective employees. The plagiarism checker feature applies a string matching algorithm to a text to search for common words between texts. There are several algorithms used for string matching, two of which are the Rabin-Karp and Jaro-Winkler algorithms. The Rabin-Karp algorithm is one of the algorithms that is suitable for solving multi-string pattern problems, while the Jaro Winkler Distance algorithm has advantages in terms of accuracy. An e-recruitment application with a plagiarism detection feature is developed and tested on various types of text, namely in the form of a self-description text of the prospective employee. From the experimental results, it was found that the two algorithms can be used to detect plagiarism in the applicant's self-description text, but it still has shortcomings in checking it.

Key words

E-recruitment, Jaro Winkler Distance, Plagiarism Checker, Rabin Karp, String matching

1. Pendahuluan

1.1 Latar Belakang

Perkembangan teknologi informasi yang sangat pesat dalam beberapa tahun terakhir dan pengenalan tentang teknologi ke dalam kehidupan sehari-hari telah meningkatkan jumlah informasi yang tersedia di semua lingkungan sosial manusia tidak terkecuali perkembangan dalam dunia karir. [1]

Di sisi lain, dalam dunia karir terutama dalam perekrutan karyawan banyak perusahaan yang menggunakan pengetahuan sistem manajemen online untuk merekrut karyawan yang biasa disebut sebagai sistem rekrutmen elektronik (*e-recruitment*). Sistem perekrutan manajemen online yang dimaksud adalah aplikasi yang dapat mempublikasikan berbagai lowongan pekerjaan dari berbagai perusahaan. Pada perekrutan karyawan secara *online* juga memiliki manfaat yaitu dapat memudahkan kedua belah pihak, dimana pihak perusahaan dapat mempublikasikan lowongan pekerjaan secara luas dalam internet dan dapat mempermudah pihak pelamar karena mempermudah pencarian pekerjaan yang sesuai dengan posisi yang diinginkan atau perusahaan yang diinginkan dengan gaji yang diharapkan. Pada aplikasi perekrutan karyawan secara online juga terdapat deskripsi diri pelamar dan beberapa informasi lainnya mengenai pelamar yang melamar pada lowongan pekerjaan tersebut yang dapat diinfokan pada perusahaan yang bersangkutan. Selain itu, pada aplikasi ini mempunyai fitur pendeteksi kemiripan teks pada deskripsi diri pelamar sehingga pelamar tidak dapat menyalin deskripsi diri pelamar lainnya yang dapat disebabkan oleh beberapa faktor, salah satunya faktor penyebabnya adalah karena kurangnya kepercayaan diri pelamar.

1.2 Rumusan Masalah

Berdasarkan uraian di atas, maka dapat ditulis perumusan masalah yaitu:

1. Bagaimana caranya merancang fitur plagiarisme pada aplikasi *e-recruitment* ?

2. Bagaimana caranya memeriksa akurasi plagiarisme yang dilakukan ?

1.3 Tujuan Rancangan

Berdasarkan perumusan masalah di atas, maka tujuan yang hendak dicapai, yaitu :

1. Memudahkan pihak administrator dalam mengelola data serta memproses *recruitment* calon karyawan.
2. Memudahkan administrator dalam melakukan pengecekan dan pembuatan laporan mengenai jumlah calon karyawan yang melamar lowongan kerja yang dapat diberikan kepada perusahaan yang bersangkutan dalam bentuk file PDF.
3. Menghindari adanya plagiarisme pada jawaban esai dari pelamar mengenai deskripsi diri pelamar.

2. Landasan Teori

2.1 *E-recruitment*

E-recruitment adalah suatu cara dalam memanfaatkan penggunaan internet untuk merekrut karyawan yang memiliki kemampuan di bidang tertentu untuk masuk ke dalam suatu perusahaan, termasuk di dalamnya adalah penggunaan dari situs perusahaan itu sendiri, yakni organisasi dan penggunaan papan pengumuman lowongan pekerjaan komersial secara *online*. Menurut Galanaki (2002), *e-recruitment* didefinisikan sebagai proses rekrutmen secara *online* mengacu pada posting lowongan di situs web perusahaan atau website vendor rekrutmen online, dan memungkinkan pelamar untuk mengirimkan resume atau curriculum vitae (CV) pelamar dalam format elektronik melalui e-mail atau dalam beberapa format elektronik lainnya.[2]

2.2 *Plagiarism*

Secara etimologis plagiat berasal dari bahasa Inggris yaitu plagiarism, apabila dirunut sebenarnya berasal dari bahasa Yunani yaitu plagiarius berarti penculik, pencuri karya tulis, atau pencuri ide atau gagasan.[3] Plagiat juga merupakan tindakan yang tidak baik karena mengakui karya orang lain atau tulisan orang lain sebagai milik sendiri.[4] Dalam penentuan plagiat yang digunakan fitur pendeteksi plagiat pada aplikasi ini berdasarkan jenis polanya yaitu total dan parsial, dimana plagiat total dapat diartikan sebagai penjiplakan tulisan pihak lain secara menyeluruh sedangkan plagiat parsial diartikan sebagai penjiplakan sebagian tulisan dari pihak lain.

2.3 *Pre-processing*

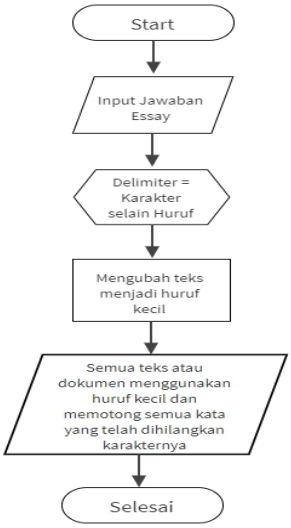
Pada umumnya, teks yang dilakukan preprocessing adalah teks yang memiliki beberapa karakteristik di antaranya adalah memiliki dimensi yang tinggi, terdapat noise pada data, dan terdapat struktur teks yang tidak baik. Maka diperlukan tahap pada preprocessing yang dilakukan secara umum dalam teks pada dokumen , yaitu sebagai berikut[5]:

1. Case Folding
   Case folding merupakan proses untuk melakukan perubahan terhadap semua huruf dalam dokumen menjadi huruf kecil, sehingga hanya huruf ‘a’ sampai dengan huruf ‘z’ yang diterima. Case folding dilakukan untuk mempermudah pencarian, karena tidak semua penulisan teks konsisten dalam penggunaan huruf kapital. Oleh karena itu, peran case folding, dibutuhkan dalam mengonversi keseluruhan teks dalam dokumen menjadi suatu bentuk standar (huruf kecil). Di dalam case folding terdapat beberapa proses lain seperti menghapus tanda baca, menghapus whitespace (karakter kosong). Contoh proses *case folding* dapat dilihat pada **Tabel 1**.

Tabel 1 Proses *case folding*

Teks *Input*	Teks *Output*
Teknologi informasi adalah alat yang membantu melakukan tugas informasi.	teknologi informasi adalah alat yang membantu melakukan tugas informasi

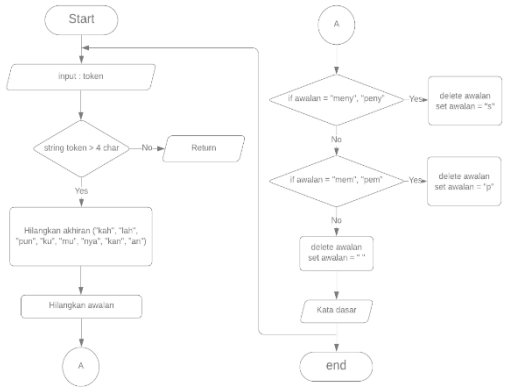
2. Tokenizing
   Tahap tokenizing adalah tahap yang dilakukan untuk memotong string yang diinput berdasarkan tiap kata yang menyusunnya. Karakter selain huruf dihilangkan dan dianggap delimiter. Delimiter adalah urutan satu karakter atau lebih yang dipakai untuk membatasi atau memisahkan data yang disajikan dalam kalimat. Salah satu contoh dari delimiter adalah tanda koma, titik koma, atau titik dua. Untuk alur proses *case folding* dan *filtering* dapat dilihat pada **Gambar 1.**



Gambar 1 Proses Case Folding dan Tokenizing

3. *Filtering*
   Filtering adalah proses untuk mengumpulkan kata-kata penting dari hasil tokenizing. Filtering dapat menggunakan algoritma stoplist (membuang kata yang kurang penting) atau wordlist (menyimpan kata penting). Pembuata aplikasi ini menggunakan stoplist. Stoplist atau yang biasa juga disebut stopword adalah kata-kata yang tidak deskriptif yang dapat dibuang dalam pendekatan *bag-of-words* (struktur kalimat tidak diperhatikan).

4. Stemming
   Tahap stemming adalah proses pencarian kata dasar dari tiap kata hasil filtering. Pada tahap ini dilakukan proses pengembalian berbagai bentukan kata ke dalam suatu representasi yang sama. Terdapat alur proses stemming dapat dilihat pada **Gambar 2**.



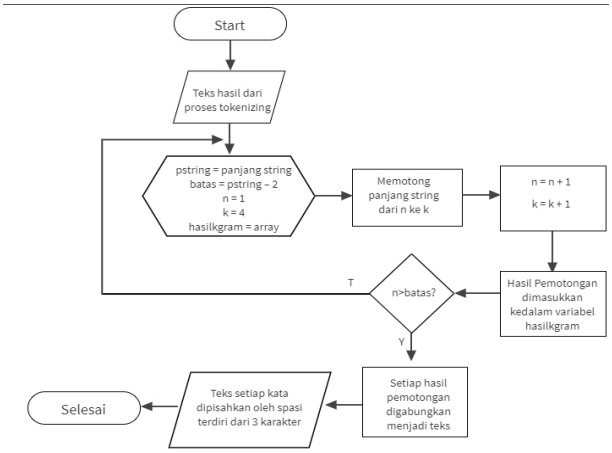
Gambar 2 Flowchart Stemming

2.4 Algoritma Rabin Karp

Algoritma Rabin-Karp adalah salah satu algoritma pencarian string yang dikembangkan oleh Michael O. Rabin dan Richard M. Karp yang menggunakan fungsi hashing untuk menemukan *pattern* di dalam string teks. [6]

Algoritma Rabin-Karp memiliki beberapa karakteristik yaitu menggunakan K-Gram dan hashing. Penerapan algoritma Rabin Karp dilakukan setelah melewati tahapan preprocessing. Berikut merupakan tahapan algoritma Rabin Karp , yaitu :

1. K-gram
   K-gram merupakan rangkaian terms dengan panjang K. Kebanyakan pembagian yang digunakan sebagai terms adalah huruf, namun pada pembuatan aplikasi ini pembagian yang digunakan sebagai terms adalah kata. Pada pembuatan aplikasi ini menggunakan K = 3. Proses K-gram dapat dilihat pada **Gambar 3**. Pembagian K-gram pada pembuatan aplikasi ini bukan berdasarkan huruf, melainkan berdasarkan katanya, untuk lebih jelas dapat dilihat pada **Tabel 2**.



Gambar 3 Flowchart K-gram

Tabel 2 Contoh K-gram berdasarkan kata

Kalimat	Komputer adalah perangkat keras
Pre-Processing	komputeradalahperangkatkeras
K-Gram k = 3	{komputeradalahperangkat} {adalahperangkatkeras}

2. Hashing
   Hashing merupakan salah satu cara untuk mengubah karakter string menjadi integer yang disebut nilai hash. Algoritma Rabin Karp memiliki kekurangan yaitu sulitnya keakuratan antarkata yang mirip, karena pada algoritma Rabin Karp masih digunakan fungsi Hash untuk mengubah kata menjadi sebuah bilangan desimal, di mana fungsi hash suatu string

S1 mungkin sama dengan string S2. Penerapan hashing pada program ini menggunakan rumus rolling hash pada **Persamaan 1**.

$$H(c_1 \dots c_k) = \left(c_1 * b^{(k-1)} + c_2 * b^{(k-2)} + \cdots + c_{(k-1)} * b^k + c_k\right) mod\ q \quad \textbf{(1)}$$

Dimana H adalah substring, c adalah nilai ASCII perkarakter, b adalah konstan bilangan prima, k adalah banyaknya karakter, dan q adalah modulo bilangan prima.

2.5 Algoritma Jaro Winkler Distance

Jaro Winkler Distance adalah algoritma yang berfungsi untuk menghitung nilai jarak kedekatan antara dua teks. Pada perhitungan algoritma Jaro Winkler Distance setiap token dicek pengejaannya dengan daftar kata yang terdapat dalam basis data. Pada basis data terdapat data deskripsi diri pelamar sebelumnya yang digunakan sebagai pembanding. Nilai normal pada Jaro Winkler Distance adalah 0 yang menunjukkan tidak ada kesamaan dan 1 yang menunjukkan adanya kesamaan yang tepat. Untuk melakukan perhitungan Jaro Winkler Distance diperlukan perhitungan Jaro Distance dengan rumus pada **Persamaan 2** dengan ketentuan pada **Persamaan 3**, kemudian dilanjutkan perhitungan Jaro Winkler Distance pada **Persamaan 4**.

$$dj\ (s1, s2) = \frac{1}{3}\left(\frac{m}{|s1|} + \frac{m}{|s2|} + \frac{m-t}{m}\right) \quad \textbf{(2)}$$

Dimana m adalah jumlah karakter yang sama, |s1| adalah panjang string 1, |s2| adalah panjang string 2, t adalah jumlah transposisi.

$$\frac{\max|s1|, |s2|}{s} < -1 \quad \textbf{(3)}$$

$$dw = dj + \left(l \times p \times (1 - dj)\right) \times 100\% \quad \textbf{(4)}$$

Dimana dw adalah Jaro Winkler Distance, dj adalah Jaro Distance, l adalah panjang prefix umum di awal string (maksimal 4 kata atau indeks), p adalah konstanta scaling factor. Nilai standar untuk konstanta (p) menurut Winkler adalah 0,1.

Fitur pendeteksi kemiripan teks pada aplikasi ini dengan metode Jaro Winkler Distance yang digunakan untuk menghitung persentase kemiripan pada teks deskripsi diri pelamar. Persentase dikategorikan menjadi 4 bagian yaitu 0% yang menyatakan non-plagiat, 1% sampai dengan kurang dari 30% dapat dikatakan sebagai plagiat ringan, 30% - 70% dapat dikatakan sebagai plagiat sedang, dan diatas 70% sampai batas maksimal yaitu 100% dapat dikatakan sebagai plagiat berat atau tinggi.

3. Hasil Pengujian

3.1 Pengujian Perhitungan Akurasi

Pengujian akurasi merupakan pengujian terhadap fitur yang terdapat pada aplikasi *e-recruitment* ini yaitu pengujian dengan melakukan perhitungan akurasi dengan menggunakan hasil *similarity* yang diperoleh pada sistem pendeteksi plagiarisme deskripsi diri pada aplikasi e-*recruitment* dan hasil similarity yang diperoleh dari *software* Plagiarism Checker X. Pada proses perhitungan akurasi ini menggunakan perhitungan dari *table* confusion matrix dengan perhitungan rumus akurasi pada **Persamaan 5**.

$$Akurasi = (TP + TN)/(TP + TN + FP + FN) \quad \textbf{(5)}$$

Persamaan 5 digunakan untuk perhitungan akurasi dengan menggunakan 10 data deskripsi diri pelamar yang telah dikumpulkan melalui *google form.* Klasifikasi data uji terhadap data yang telah dikumpulkan, disajikan pada **Tabel 3**.

Tabel 3 Klasifikasi Data Uji

Data		Aplikasi E-recruitment	Plagiarisme Checker X	Klasifikasi
1	2	Non	Non	TP
1	3	Non	Non	TP
1	4	Non	Non	TP
1	5	Non	Non	TP
1	6	Non	Non	TP
1	7	Non	Non	TP
1	8	Non	Non	TP

Tabel 3 (Lanjutan)

Data		Aplikasi E-recruitment	Plagiarisme Checker X	Klasifikasi
1	9	Non	Non	TP
1	10	Non	Non	TP
2	1	Non	Non	TP
2	3	Ringan	Ringan	TN
2	4	Non	Non	TP
2	5	Ringan	Ringan	TN
2	6	Ringan	Ringan	TN
2	7	Non	Non	TP
2	8	Ringan	Ringan	TN
2	9	Non	Non	TP
2	10	Non	Non	TP
3	1	Non	Non	TP
3	2	Ringan	Ringan	TN

3	4	Non	Non	TP
3	5	Ringan	Ringan	TN
3	6	Sedang	Ringan	FN
3	7	Non	Non	TP
3	8	Non	Non	TP
3	9	Non	Non	TP
3	10	Non	Non	TP
4	2	Non	Non	TP
4	3	Non	Non	TP
4	5	Non	Non	TP
4	6	Non	Non	TP
4	7	Non	Non	TP
4	8	Non	Non	TP
4	9	Non	Non	TP
4	10	Ringan	Ringan	TN
5	1	Non	Non	TP
5	2	Ringan	Ringan	TN
5	3	Ringan	Ringan	TN
5	4	Non	Non	TP
5	6	Non	Non	TP
5	7	Non	Non	TP
5	8	Non	Non	TP
5	9	Non	Non	TP
5	10	Non	Non	TP

Dari tabel di atas dilakukan perhitungan akurasi dengan perolehan data true positive(TP), true negative (TN), false negative(FP), false negative(FN).
TP= 35
TN= 9
FP= 0
FN= 1
Akurasi = (35+9)/(35+0+1+9)
Akurasi = 44/45
Akurasi = 0,98 × 100 = 98%

Dari perhitungan di atas diperoleh nilai yang cukup baik yaitu 98%.

3.2 Skenario Uji Coba

Skenario uji coba ini terdiri dari 5 kasus yang dilakukan uji coba. Uji coba ini dilakukan dengan menggunakan salah satu data deskripsi diri pelamar. Berikut merupakan kasus uji coba yang dilakukan.

1. Kasus I
   Uji coba pada kasus pertama dilakukan dengan menggunakan data deskripsi diri 1 yang terdiri dari 8 kalimat dan 4 kalimat awal yang merupakan potongan dari data deskripsi diri 1. Pada uji coba ini dapat disimpulkan bahwa hasil persentase yang diperoleh merupakan tingkat plagiarisme tinggi.
2. Kasus II
   Uji coba pada kasus kedua dilakukan dengan menggunakan data deskripsi diri 1 dan menggunakan data deskripsi 1 dengan mengubah keterangan pada data deskripsi 1. Perubahan keterangan pada data deskripsi 1 yaitu keterangan waktu, keterangan nama tempat, dan keterangan lainnya. Pada uji coba ini dapat disimpulkan bahwa hasil persentase yang diperoleh merupakan tingkat plagiarisme sedang.
3. Kasus III
   Uji coba pada kasus ketiga dilakukan dengan menggunakan data deskripsi diri 1 dan menggunakan data deskripsi 1 dengan menukar 4 kalimat akhir menjadi 4 kalimat awal dan 4 kalimat awal menjadi 4 kalimat akhir. Pada uji coba ini dapat disimpulkan bahwa hasil persentase yang diperoleh merupakan tingkat plagiarisme sedang.
4. Kasus IV
   Uji coba pada kasus kelima dilakukan dengan menggunakan data deskripsi diri 1 dan menggunakan data deskripsi 1 dengan menambah keterangan yang lebih lengkap daripada data deskripsi 1 sebelumnya. Pada uji coba ini dapat disimpulkan bahwa hasil persentase yang diperoleh merupakan tingkat plagiarisme sedang.
5. Uji coba ini dilakukan dengan menggunakan data utama dan data perbandingannya adalah data deskripsi diri 1. Pada uji coba ini dapat disimpulkan bahwa hasil persentase yang diperoleh merupakan tingkat plagiarisme tinggi.

3.3 Pengujian Kecepatan Waktu

Pengujian Kecepatan waktu merupakan pengujian yang dilakukan terhadap fitur pendeteksi kemiripan teks untuk data deskripsi diri pelamar. Pengujian ini dilakukan dengan cara mengukur waktu proses terhadap lamanya program melakukan deteksi terhadap data deskripsi diri pelamar yang terdapat pada basis data. Pengujian kecepatan waktu dilakukan dengan menggunakan 5 data deskripsi diri pelamar yang disajikan dalam **Tabel 4** sebagai berikut.

Tabel 4 Pengujian Kecepatan Waktu

Banyak Data	Waktu Proses	Keterangan
1	0,15 detik	Pengujian dengan 1 data memperoleh waktu tersingkat yaitu 0,15 detik karena pada data tersebut tidak memiliki pembanding untuk dideteksi plagiarismenya.
2	14,79 detik	Pengujian dengan 2 data yang memiliki kemiripan teks memiliki waktu lebih banyak dibandingkan data yang tidak memiliki kemiripan teks dengan persentase 0%.

3	12,63 detik	Pengujian dengan 3 data dengan persentase 0% memiliki waktu yang lebih cepat dibandingkan dengan pengujian 2 data yang memiliki persentase
4	27,74 detik	Pengujian dengan 4 data dengan 2 data yang memiliki nilai persentase lebih dari 0%.
5	44,41 detik	Pengujian dengan 5 data dengan 3 data yang memiliki nilai persentase lebih dari 0%.

Perhitungan rata-rata kecepatan waktu pada pengujian 5 data sebagai berikut:

Rata-rata= ((0,15+14,79+12,63+27,74+44,41))/5
Rata-rata = 19,94 detik

4. Kesimpulan

Setelah dilakukan pengujian terhadap fitur pendeteksi kemiripan teks atau dapat disebut dengan plagiarisme checker, diperoleh kesimpulan sebagaimana berikut :

1. Algoritma Rabin Karp dan algoritma Jaro Winkler Distance berhasil diimplementasikan pada fitur pendeteksi kesamaan teks pada deskripsi diri pelamar dalam aplikasi e-recruitment.
2. Uji skenario pada kasus I dan kasus V dapat disimpulkan bahwa aplikasi dapat berjalan dengan baik untuk memeriksa kemiripan teks deskripsi diri pelamar yang memiliki struktur kata yang identik atau sama dengan bobot 100%. Hal ini dikarenakan urutan kata-kata yang dibandingkan sangat sesuai.
3. Uji skenario pada kasus II sampai kasus IV dapat disimpulkan bahwa dalam mendeteksi kemiripan teks deskripsi diri pelamar aplikasi ini kurang mampu mendeteksi kemiripannya. Hal ini dikarenakan dalam teks utama dan pembanding memiliki perbedaan dalam struktur atau urutan katanya.
4. Pengujian kecepatan waktu pada fitur pendeteksi dengan menggunakan 5 data dapat disimpulkan bahwa perhitungan waktu sangat dipengaruhi oleh banyaknya data yang memiliki nilai persentase plagiarisme atau memiliki kemiripan teks antara data satu dengan data lainnya.
5. Hasil perhitungan akurasi dengan confusion matrix diperoleh nilai 98% dari 10 perbandingan data deskripsi diri pelamar.
6. Fitur pendeteksi kemiripan teks pada aplikasi ini dapat menampilkan warna berbeda untuk teks yang sama antara teks utama dan teks pembanding.

Dari hasil kesimpulan tersebut dapat dinyatakan bahwa aplikasi *e-recruitment* dengan fitur pendeteksi kemiripan teks pada deskripsi diri pelamar belum mampu mendeteksi kesamaan teks utama dengan teks pembanding yang diambil dari internet. Namun, fitur ini sudah mampu melakukan marking pada kata yang sama dan juga mampu melakukan pendeteksian plagiarisme atau kemiripan teks dengan *range* persentase dari 0% sampai 100%.

REFERENSI

[1] Neuman, Clifford, "Prospero : A tool of organizing internet resource", Internet Research, Vol. 20, No. 1, 2010.Purnomo, Tommy Septian. "Recruitment online (eRecruitment) sebagai suatu indovasi dalam perekrutan perusahaan". Jurnal JIBEKA, Vol.7 No.3, (Agustus,2013).

[2] Galanaki, Eleanna. The Decision to Recruit Online: A Descriptive Study, https://www.researchgate.net/publication/228299307_The_Decision_t_Recruit_Online_A_Descriptive_Study, 26 September 2020.

[3] Soelistyo, Henry. "Plagiarisme: Pelanggaran Hak Cipta dan Etika". (Yogyakarta: PT Kanisius, 2011).

[4] Risparyanto, Anton. TURNITIN SEBAGAI ALAT DETEKSI PLAGIARISME, Jurnal Perpustakaan, Vol. 11 No.2, 2020.

[5] Rosid, Mochamad Alfan; Fitrani,Arif Senja; Astutik, Ika Ratna Indra; Mulloh, Nasrudin Iqrok; Gozali, Haris Ahmad. "Improving Text Preprocessing For Student Complaint Document Classification Using Sastrawi". IOP, Vol 1, 2020.

[6] Siswanto, Eric dan Giap, Yo Ceng. "IMPLEMENTASI ALGORITMA RABIN-KARP DAN COSINE SIMILARITY UNTUK PENDETEKSI PLAGIARISME PADA DOKUMEN", Jurnal ALGOR, Vol. 1 No. 2, 2020.

Stephanie, mahasiswa program studi Fakultas Teknologi Informasi di Universitas Tarumanagara.