e-ISSN 2620-5157

Perbandingan Algoritma N-gram dan Algoritma Knuth Morris Pratt untuk Mengukur Tingkat Akurasi Plagiarisme pada Dokumen Abstrak Skripsi Berbasis *Website*

Dwi Krisbiantoro [a,1,*], Sofyan Fathur Rohim [a,2], Irfan Santiko [a,3]

[a] Fakultas Ilmu Komputer, Universitas Amikom Purwokerto, Jl.Letjend. Pol.Soemarto, Purwokerto, Jawa Tengah, 53127, Indonesia
[1] dwikris@amikompurwokerto.ac.id *; [2] sofyanfathur@gmail.com; [3] irfan.santiko@amikompurwokerto.ac.id
* Koresponsendi penulis

ARTICLE INFO

Article history
Menerima 8 September 2020
Revisi 25 Juni 2021
Diterima 28 Juli 2021

Kata Kunci
Plagiarism
N-gram Algorithm
KMP Algorithm
Website

ABSTRACT

Plagiarism is a crime that often occurs in the academic world, plagiarism occurs because of theft of other people's work that is illegally recognized as if the work is his own. N-gram is an algorithm by cutting as many characters as N-characters in a sentence or word. While the Knuth Morris Pratt (KMP) algorithm is a string search algorithm, this algorithm is used to maintain information that is used to carry out the number of shifts whenever there is no matched patency in the text. The purpose of this study is to create a system to measure the comparison of the accuracy of the N-gram algorithm with a website-based KMP on a thesis abstract document. This research uses the waterfall system development method which has stages, namely analysis, design, coding, and testing. The KMP test results are better than N-gram where kmp has an average percentage of 3.8% while the N-gram 3.5% results are obtained from an average of 10 trials and 5 documents tested.

1. Pendahuluan

Plagiarisme merupakan suatu tindak kejahatan yang sering terjadi di dunia akademik. Plagiasi merupakan tindakan yang dilakukan dengan cara sengaja mengambil hasil karya orang lain tanpa izin, dengan tujuan untuk mendapatkan nilai yang baik dalam pembuatan karya ilmiah atau skripsi. Plagiasi termasuk perilaku yang dinilai kurang memiliki etika dalam sebuah karya. Baik itu karya tulisan, maupun karya seni monumental [1].

Dalam perkembanganya tindakan plagiat dipengaruhi oleh perkembangan teknologi dan internet yang semakin pesat dan cepat. Menurut asosiasi penyedia jasa internet Indonesia (APJII) tahun 2018 penggunaan internet di Indonesia tumbuh sebesar 10.12% artinya pengguna internet di indonesia mencapai 171,17 juta jiwa yang telah menggunakan internet [2]. Plagiarisme merupakan perilaku umum yang sering dilakukan para pengguna internet dalam membuat sebuah konten atau karya tulis. Tidak banyak juga perilaku saling meng-klaim karya yang justru menimbulkan konflik. Upaya untuk meng-klaim dirasa sering tidak akurat sehingga dapat menimbulkan masalah baru [3].

Pada penelitian ini akan diuji berupa 2 buah algoritma untuk menguji tingkat keakurasian plagiasi agar dapat mengurangi tindakan disclaimer dengan melihat hasil akurasi plagiasi tersebut. Beberapa dekade belakangan ini banyak yang membuat inovasi dalam upaya – upaya mengurangi perilaku plagiarisme dengan menggunakan teknologi [4].

Peneliti terakhir mengangkat sebuah tema yang sama yaitu perbandingan algoritma juga untuk mengukur tingkat akurasi plagiasi pada sebuah text [5]. Pada penelitian sebelumnya juga dilakukan sebuah metode pengukuran untuk mendeteksi tingkat plagiarisme terhadap suatu dokumen berdasarakan tingkat presentase plagiat terhadap suatu dokumen [6].

Penelitian lainnya menyebutkan hasil bahwa N-gram sangat mempengaruhi tingkat similarity atau kemiripan teks dokumen dimana pengimplementasian dari metode Ngram dan *Jaccrad similarity* terhadap algoritma *Winnowing* cukup baik digunakan untuk membandingkan antara dua dokumen, dan cukup baik digunakan untuk meminimalisir tindakan plagiarisme suatu dokumen[7].

Selanjutnya penelitian lain juga menghasilkan sebuah metode pengukuran untuk mendeteksi tingkat plagiarisme terhadap suatu dokumen berdasarakan tingkat presentase plagiat terhadap suatu dokumen. Kemudian diperoleh hasil bahwa Ngram sangat mempengaruhi tingkat similarity atau kemiripan teks dokumen dimana pengimplementasian dari metode Ngram dan *Jaccrad similarity* terhadap algoritma *Winnowing* cukup baik digunakan untuk membandingkan antara dua dokumen, dan cukup baik digunakan untuk meminimalisir tindakan plagiarisme suatu dokumen X [8]. N-gram merupakan suatu algoritma dengan melakukan potogan karakter sebanyak N-karakter didalam suatu kalimat atau kata, Sedangkan algorima Knuth Morris Pratt (KMP) merupakan suatu algoritma pencarian string algoritma ini digunakan untuk memelihara informasi yang digunakan untuk melakukan jumlah pergeseran pada setiap kali tidak ditemukan kecocokan *pattren* pada teks [9]. Algoritma ini menggunakan informasi tersebut untuk membuat pergeseran yang lebih jauh, tidak hanya satu karakter. Dengan menggunakan algoritma *Knuth Morris Pratt*, waktu pencarian dapat dikurangi secara signifikan [10]. Hasil penelitian lain membuktikan bahwa hasil yang diperoleh menunjukkan bahwa sistem yang diusulkan yaitu *curpus AraPlagDet* telah meningkatkan kinerja deteksi plagiarisme dalam dokumen Arab dengan pengindeksan semantik dan sistem mutli-agents [11].

Adapun tujuan dari penelitian in adalah untuk mengetahui pola ukur dan perbandingan tingkat akurasi algoritma N-gram dan KMP, serta mengimplementasikan algoritma N-gram dan KMP untuk melakukan pengecekan tingkat plagiarisme dokumen pada abstrak skripsi.

2. Metode Penelitian

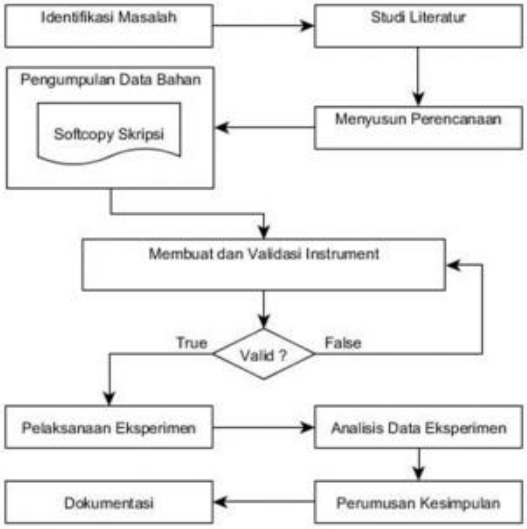


Fig. 1. Alur Penelitian

Tahap pertama adalah identifikasi masalah, peneliti melakukan kajian masalah yang diambil untuk menentukan spesifikasi dan batasan dari objek masalah yang akan diteliti. Hal ini dilakukan agar penelitian fokus pada permasalahan utama yang diteliti dan tujuan penelitian menjadi lebih jelas.

Tahap kedua adalah studi literatur, peneliti mengumpulkan data yang berhubungan dengan topik penelitian yang dilakukan dari berbagai media untuk menambah pengetahuan peneliti tentang riset-riset yang pernah dilakukan oleh peneliti sebelumnya.

Tahap ketiga adalah menyusun perencanaan, peneliti menentukan langkah-langkah yang harus diambil untuk mencapai tujuan penelitian. Identifikasi variabel luar, menentukan cara

kontrol, memilih metode penelitian yang tepat, mengidentifikasi prosedur pengumpulan data, dan rancangan prosedur dalam eksperimen.

Tahap keempat adalah pengumpulan data bahan yang berupa softcopy skripsi. Softcopy skripsi diambil dari petugas perpustakaan universitas Amikom Purwokerto dengan sebelumya melakukan permohonan secara tertulis kepada kepala perpustakaan. Dari softcopy skripsi yang diambil hanya bagian abstrak saja yang digunakan sebagai bahan eksperimen.

Tahap kelima adalah membuat instrument eksperimen. Implementasi algoritma sesuai literatur yang didapat dari tahap studi literatur sebelumnya. Semua subproses tersebut diimplementasikan pada beberapa fungsi dalam sebuah aplikasi. Implementasi menggunakan bahasa pemrograman PHP.

Tahap terakhir adalah perumusan kesimpulan, dari proses analisis data yang menyajikan tabulasi perbandingan hasil eksperimen ditariklah sebuah kesimpulan. Rumusan kesimpulan dikaitkan dengan rumusan masalah agar dapat menjawab pertanyaan yang ada dalam rumusan masalah.

3. Hasil dan Pembahasan

3.1 Desain

3.1.1 Perancangan sistem

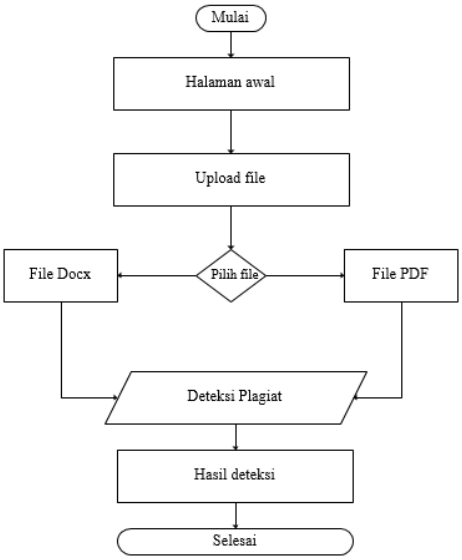


Fig. 2. Perancangan sistem

3.1.2 *Use case* diagram

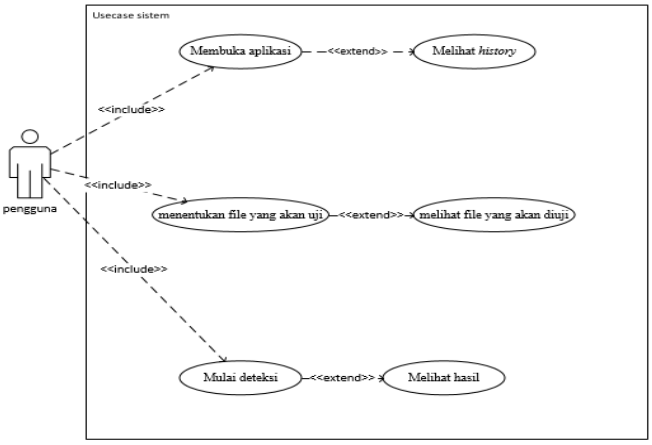


Fig.3. Use case diagram

3.1.3 Desain Antarmuka

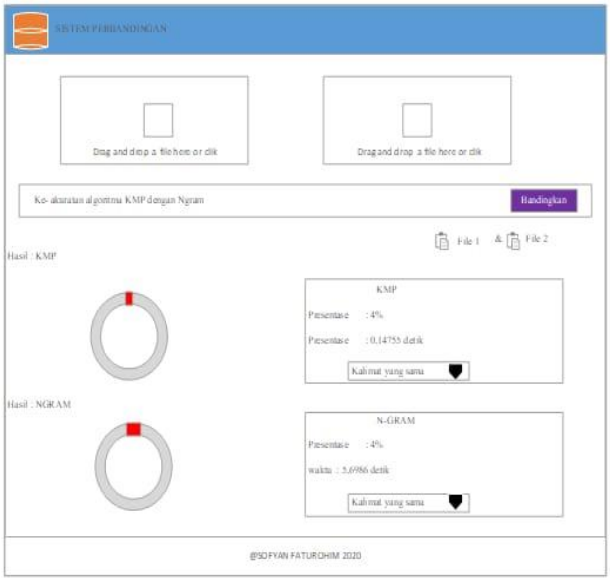


Fig.4. Antarmuka halaman hasil perbandingan

3.2 Pengkodean

```
public function kmp_compute_prefix($P) {
    $m = strlen($P);
    $pi = array();
    $pi[1] = 0;
    $k = 0;
    for ($q = 1; $q < $m; $q++) {
        while ($k > 0 && $P[$k] != $P[$q]) {
            $k = $pi[$k];
        }
        if ($P[$k] == $P[$q]) {
            $k++;
        }
        $pi[$q+1] = $k;
    }
    return $pi;
}

public function kmp_search_prefix($T, $prefix)
{
    $matches = array();
    $P = $prefix->str;
    $m = $prefix->len;
    $pi = $prefix->P;
    $n = strlen($T);
    $q = 0;
    $l = 0;
    for ($i = 0; $i < $n; $i++) {
        while ($q > 0 && $P[$q] != $T[$i]) {
            $q = $pi[$q];
        }
        if ($P[$q] == $T[$i]) {
            $q = $q + 1;
        }
        if ($q == $m) {
            $matches[] = $i - $m + 1;
            $l = $i;
            $q = $pi[$q];
        }
    }
    return $matches;
}
```

Fig.5. Source code algoritma KMP

```
function ngramSearch($word,$item){
    $excerpts = [
        $word,
        $item
    ];
    // dd($item);
    $joinedExcerpts = implode(".\n", $excerpts);
    $sentences = preg_split('/[^\s]\pL]/umi', $joinedExcerpts, -1, PREG_SPLIT_NO_EMPTY);
    $wordsSequencesCount = array();
    $id = 0;
    if($sentences){
    foreach($sentences as $index=>$sentence) {
        $words = array_map('mb_strtolower',
                    preg_split('/[^\pL+]/umi', $sentence, -1, PREG_SPLIT_NO_EMPTY));
        foreach($words as $index => $word) {
            $wordsSequence = '';
            $count = 0;
            foreach(array_slice($words, $index) as $nextWord) {
                $wordsSequence .= $wordsSequence ? (' ' . $nextWord) : $nextWord;

                // if !item, continue
                if($wordsSequence != strtolower($item)){
                    continue;
                }
                if( !isset($wordsSequencesCount[$wordsSequence]) ) {
                    $wordsSequencesCount[$id] = [
                        'name'=>$wordsSequence,
                        'count'=>0
                    ];
                }
                $wordsSequencesCount[$id++] = [
                    'name'=>$wordsSequence,
                    'count'=>++$count
                ];
                continue;
            }
        }
    }
    }
    $ngramsCount = array_filter($wordsSequencesCount, function($count) { return $count > 1; });
    $result =collect($ngramsCount)->count();
    return $result-1;
}
```

Fig. 6. Source code algoritma N-gram

3.3 Hasil pengujian

3.3.1 Pengujian N-gram manual

Dalam melakukan pengujian N-gram secara manual dilakukan dengan cara memecahkan kalimat menjadi kata berdasarkan panjang karakter n-gram, dan kemudian dicocokan antar kata. Pada kasus ini panjang n-gram yang digunakan, yaitu 4. Berikut ini merupakan hasil dari pemecahan kalimat menjadi kata dan hasil dari pencocokan dokumen uji dengan dokumen sumber berikut ini:

Teks dokumen sumber : pelangiarime merupakan tindakan melanggar hak cipta

pela elan lang angi ngia giar iari aris rism isme smem meme emer meru erup rupa upak paka akan kant anti ntin tind inda ndak daka akan kanm anme nmel mela elan lang angg ngga ggar garh arha rhak hakc akci kcip cipt ipta

Teks dokumen uji : plagiarisme melanggar hak cipta

pela elan lang angi ngia giar iari aris rism isme smem meme emel mela elan lang angg ngga ggar garh arha rhak hakc akci kcip cipt ipta

selanjutnya hitung jumlah kata yang terdapat pada dokumen dan hitung jumalah kata sementara dokumen uji dan dokumen sumber.

Jumlah N-Gram pada dokumen sumber = 44 kata
Jumlah N-Gram dokumen uji = 27kata
Jumlah kata sama = 25 kata.

Setelah didapat hasil kata yang sama selanjutnya dilakukan perhitungan persentase *sismilarity* menggunakan persamaan

$$S = \frac{2\ X\ 25}{44+27} = \frac{50}{71} = 0{,}704225352$$

Maka kemiripan dari dokumen asli dan dokumen uji adalah sebesar 0,704 atau sebesar 70,4%.

3.3.2 Pengujian N-gram dengan sistem

Table 1. Hasil pengujian N-gram dengan sistem

No	Dokumen asli	Dokumen uji	Jmlh *word* dok.1	Jmlh *word* dok. 2	Jmlh kata yang mirip	tingkat plagiat	Waktu (detik)
1	Dokumen 1	Dokumen 2	265	248	9	4	3.4
2	Dokumen 1	Dokumen 3	265	330	5	1	3.2
3	Dokumen 1	Dokumen 4	265	247	2	2	3.4
4	Dokumen 1	Dokumen 5	265	411	5	1	3.8
5	Dokumen 2	Dokumen 3	248	330	23	10	3.5
6	Dokumen 2	Dokumen 4	248	247	11	5	3.1
7	Dokumen 2	Dokumen 5	248	411	5	1	3.4
8	Dokumen 3	Dokumen 4	330	247	9	4	3.2
9	Dokumen 3	Dokumen 5	330	411	4	0	3.1
10	Dokumen 4	Dokumen 5	247	411	5	1	3.5

3.3.3 Pengujian KMP secara manual

Pada tahap pengujian dokumen dengan menggunakan algoritma Knuth Morris Pratt (KMP) dilakukan dengan menggunakan beberapa langkah berikut merupakan cara menguji algoritma KMP

String : karya ilmiah (S)
Pattern : ilmiah (P)

K	a	r	y		A		i	L	m	i	a	h

I	l	m	i	a	H

1) Langkah pertama bandingkan antara (P- 1) dengan (S-1)

K	a	r	y	a		i	l	M	i	a	h

i	l	m	i	a	H

Pada langkah pertama (P-1) tidak cocok dengan (S-1) maka akan bergeser. Tapi ada (P-5) dengan (S-5) maka KMP akan menyimpan informasi.

2) Langkah kedua bandingkan antara (P-1) dengan (S-2)

K	a	r	y	a		i	l	M	i	a	h

I	l	m	I	a	h

Pada langkah kedua (P-1) dengan (S-2) tidak memiliki kecocokan maka bergeser kekanan sebanyak 1 kali

3) Langkah ketiga bandingkan antara (P-1) dengan (S-3)

K	a	r	y	a		i	l	M	i	a	h

i	l	m	I	a	H

Pada langkah kedua (P-1) dengan (S-3) tidak memiliki kecocokan maka bergeser kekanan sebanyak 1 kali.

4) Langkah ketiga bandingkan antara (P-1) dengan (S-4)

K	a	r	y	a		i	l	M	i	a	h

i	l	m	I	a	h

Pada langkah kedua (P-1) dengan (S-4) tidak memiliki kecocokan maka bergeser kekanan sebanyak 1 kali

5) Langkah ketiga bandingkan antara (P-1) dengan (S-5)

				i	l	m	i	a	h

K	a	r	y	a		i	l	m	i	a	h

Pada langkah kedua (P-1) dengan (S-5) tidak memiliki kecocokan maka bergeser kekanan sebanyak 1 kali

6) Langkah ketiga bandingkan antara (P-1) dengan (S-6)

K	a	r	y	a		i	l	m	i	a	h

i	l	m	i	a	h

Pada langkah kedua (P-1) dengan (S-6) tidak memiliki kecocokan karena bertemu dengan string kosong, maka otomatis bergeser kekanan sebanyak 1 kali

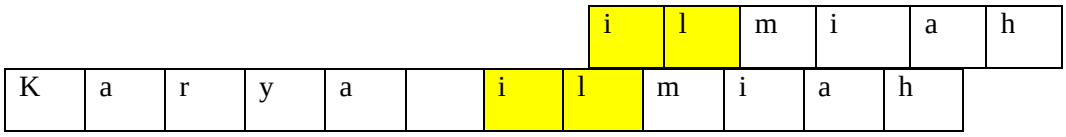
7) Langkah ketiga bandingkan antara (P-1) dengan (S-7)

							i	l	m	i	a	h

K	a	r	y	a		i	l	m	i	a	h

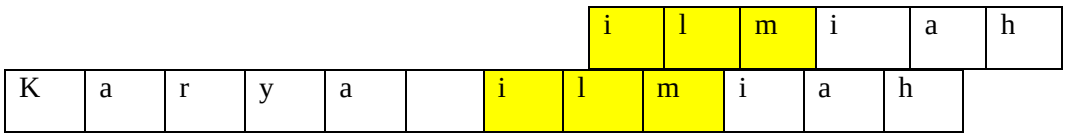
Pada langkah kedua (P-1) dengan (S-7) memiliki kecocokan maka KMP menyimpannya sebagai informasi, selanjutnya akan dilanjutkan antara (P-2) dengan (S-8).

8) Langkah ketiga bandingkan antara (P-2) dengan (S-8)



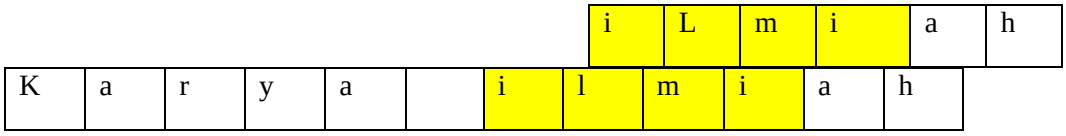
Pada langkah kedua (P-2) dengan (S-8) memiliki kecocokan maka KMP menyimpannya sebagai informasi, selanjutnya akan dilanjutkan antara (P-3) dengan (S-9).

9) Langkah ketiga bandingkan antara (P-3) dengan (S-9)



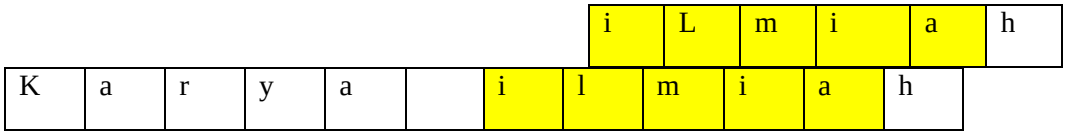
Pada langkah kedua (P-3) dengan (S-9) memiliki kecocokan maka KMP menyimpannya sebagai informasi, selanjutnya akan dilanjutkan antara (P-4) dengan (S-10).

10) Langkah ketiga bandingkan antara (P-4) dengan (S-10)



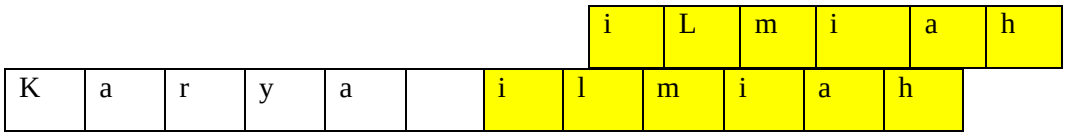
Pada langkah kedua (P-4) dengan (S-10) memiliki kecocokan maka KMP menyimpannya sebagai informasi, selanjutnya akan dilanjutkan antara (P-5) dengan (S-11).

11) Langkah ketiga bandingkan antara (P-5) dengan (S-11)



Pada langkah kedua (P-5) dengan (S-11) memiliki kecocokan maka KMP menyimpannya sebagai informasi, selanjutnya akan dilanjutkan antara (P-6) dengan (S-12).

12) Langkah ketiga bandingkan antara (P-6) dengan (S-12)



Pada langkah kedua (P-6) dengan (S-12) memiliki kecocokan maka KMP menyimpannya sebagai informasi, maka pencarian akan dihentikan karena semua pattern (P) memiliki kecocokan sebesar 100% dengan string (S -7) samapai (S-12).

2.4.4 pengujian KMP dengan sistem

Table 2. Hasil pengujian KMP dengan sistem

No	Dokumen asli	Dokumen uji	Jmlh *word* dok.1	Jmlh *word* dok. 2	Jmlh kata yang mirip	tingkat plagiat	Waktu (detik)
1	Dokumen 1	Dokumen 2	265	248	9	4	0.074
2	Dokumen 1	Dokumen 3	265	330	6	2	0.090
3	Dokumen 1	Dokumen 4	265	247	5	2	0.075
4	Dokumen 1	Dokumen 5	265	411	7	1	0.085
5	Dokumen 2	Dokumen 3	248	330	24	12	0.078
6	Dokumen 2	Dokumen 4	248	247	12	5	0.087
7	Dokumen 2	Dokumen 5	248	411	5	1	0.090
8	Dokumen 3	Dokumen 4	330	247	9	4	0.085
9	Dokumen 3	Dokumen 5	330	411	4	0	0.085
10	Dokumen 4	Dokumen 5	247	411	5	1	0.078

4. Kesimpulan

Dari hasil perancangan aplikasi perbandingan algoritma N-gram dan Knuth Morris Pratt (KMP) untuk mengukur tingkat plagiarisme pada dokumen abstrak skripsi, dengan melakukan pengujian terhadap 5 dokumen yang diujikan sebanyak 10 kali pengujian, dihasilkan beberapa butir kesimpulan diantaranya sistem yang dibangun dapat berjalan sesuai dengan apa yang diharapakan. Algoritma KMP lebih baik tingkat akurasinya jika dibandingkan dengan algoritma N-gram. Dimana hasil akurasi rata-rata KMP yaitu sebesar 3,8 % sedangkan algoritma N-gram yang hanya mencapai 3,5 %. Saran untuk penelitian selanjutnya antara lain. Pembuatan database untuk menyimpan data sumber sehingga setiap digunakan pengguna hanya memasukan satu dokumen yang akan diuji dan tidak perlu lagi memasukan dokumen ke dua untuk membandingkannya. Tampilkan sumber-sumber atau alamat berdasarkan teks yang terdeteksi plagiarisme.

Ucapan Terima Kasih

Terima kasih penulis ucapkan kepada Lembaga Penelitian dan Pengabdian Kepada Masyarakat Universitas Amikom Purwokerto atas dukungan yang diberikan selama penelitian dilakukan. Para contributor dalam penelitian yang telah dilakukan.

Daftar Pustaka

[1] G. Solís, A. C. Garcinu, L. Alsina, M. De Lara, and C. Rey, "Plagiarism and ethics in scientific publications ☪ Plagio y ética en las publicaciones científicas," vol. 90, no. 1, pp. 2018–2019, 2019, doi: 10.1016/j.anpede.2018.10.005.

[2] S. Zakir, S. Defit, and V. Vitriani, "Indeks Kesiapan Perguruan Tinggi dalam Mengimplementasikan Smart Campus," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 3, p. 267, 2019, doi: 10.25126/jtiik.201963986.

[3] C. Demartini and L. Benussi, "Do Web 4.0 and Industry 4.0 Imply Education X.0?," *IT Prof.*, vol. 19, no. 3, pp. 4–7, 2017, doi: 10.1109/MITP.2017.47.

[4] M. F. Abad-garcía, "Plagiarism and predatory journals : A threat to scientific integrity ☙," *An. Pediatría (English Ed.*, vol. 90, no. 1, pp. 57.e1-57.e8, 2019, doi: 10.1016/j.anpede.2018.11.006.

[5] S. Zouaoui and K. Rezeg, "Multi-Agents Indexing System (MAIS) for Plagiarism Detection," *J. King Saud Univ. - Comput. Inf. Sci.*, no. xxxx, 2020, doi: 10.1016/j.jksuci.2020.06.009.

[6] D. Sakamoto and K. Tsuda, "ScienceDirect ScienceDirect A Detection Method for Plagiarism Reports of Students A Detection Method for Plagiarism Reports of Students," *Procedia Comput. Sci.*, vol. 159, pp. 1329–1338, 2019, doi: 10.1016/j.procs.2019.09.303.

[7] H. Carter, J. Hussey, and W. Forehand, "Plagiarism in nursing education and the ethical implications in practice," no. November 2018, 2019, doi: 10.1016/j.heliyon.2019.e01350.

[8] W. Merkel, "Collage of confusion : An analysis of one university ' s multiple plagiarism policies," *System*, vol. 96, p. 102399, 2021, doi: 10.1016/j.system.2020.102399.

[9] J. Goodwin and J. Mccarthy, "Explaining Plagiarism for Nursing Students : An Educational Tool," *Teach. Learn. Nurs.*, vol. 15, no. 3, pp. 198–203, 2020, doi: 10.1016/j.teln.2020.03.004.

[10] O. Karnalim, "IR-based technique for linearizing abstract method invocation in plagiarism-suspected source code pair," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 31, no. 3, pp. 327–334, 2019, doi: 10.1016/j.jksuci.2018.01.012.

[11] S. Alzahrani and H. Aljuaid, "Identifying cross-lingual plagiarism using rich semantic features and deep neural networks : A study on Arabic-English plagiarism cases," *J. King Saud Univ. - Comput. Inf. Sci.*, no. xxxx, 2020, doi: 10.1016/j.jksuci.2020.04.009.