

Similar text sebagai Pengkodean Aplikasi Plagiarisme

Muhammad Wali¹, Safrizal²

¹AMIK Indonesia

²KITA Institute

article info

Article history:

Received 26 Oktober 2017

Received in revised form

23 Desember 2017

Accepted 18 Januari 2018

Available online 24 Januari
2018

Keywords:

Similar text, PHP,
application, Plagiarism

Kata Kunci:

Similar text, PHP, Aplikasi,
Plagiarisme

abstract

This research is intended to make a coding technique with a function similar to a plagiarism detector `similar_text` as text. By using the text function is similar, this research resulted in a detection of the document against the 10 (ten) of selected journal with, the document is the document with extension doc, docx, pdf, and txt. The document will be converted into html form and would henceforth be done making the string with the marker dots (.) And the comma (,) will be fetched a new string. By specifying a percentage string similarity of 90% then produced a text, in this case plagiarism is detected journal journal 1 with string similarity percentage of 48%. The use of similar texts can be classified as coding techniques for detection of plagiarism on its anti plagiarism detection applications.

abstract

Penelitian ini dimaksudkan untuk membuat sebuah teknik pengkodean dengan fungsi mirip `similar_text` sebagai pendeteksi plagiat sebuah teks. Dengan menggunakan fungsi teks yang serupa, penelitian ini menghasilkan sebuah deteksi dokumen terhadap 10 (sepuluh) jurnal yang dipilih dengan bahasa indonesia, dokumen yang merupakan dokumen dengan ekstensi doc, docx, pdf, dan txt. Dokumen tersebut akan dikonversi ke dalam bentuk html dan selanjutnya akan dilakukan pembuatan string dengan penanda titik (.) Dan koma (,) akan terambil sebuah string baru. Dengan menentukan persentase kemiripan string sebesar 90% maka dihasilkan sebuah teks, pada kasus ini jurnal yang terdeteksi plagiat adalah jurnal 1 dengan persentase kemiripan string sebesar 48%. Penggunaan teks serupa dapat digolongkan sebagai teknik pengkodean untuk deteksi plagiarisme pada penggunaan aplikasi deteksi anti plagiat.

*Corresponding author. Email: muhammadwali@amikindonesia.ac.id¹, safrizal@lembagakita.org²

© E-ISSN: 2580-1643.

Copyright © 2018. Published by Lembaga Informasi dan Riset (KITA INFO dan RISET), Lembaga KITA (<http://creativecommons.org/licenses/by/4.0/>).

1. Latar Belakang

Peningkatan dalam hal fungsi Tridharma Perguruan Tinggi adalah melaksanakan penelitian dan mempublikasikan hasil pemikiran dalam bentuk karya ilmiah. Perkembangan karya ilmiah di Indonesia relatif makin baik, terutama sejak diberlakukannya regulasi pemerintah, yang mewajibkan dosen untuk menulis artikel di jurnal ilmiah sebagai salah satu kenaikan fungsional dan pemberian tunjangan. Merujuk Peraturan Menteri Pendayagunaan Aparatur Negara dan Reformasi Birokrasi (PAN RB), Nomor 17 Tahun 2013, dan Peraturan Menteri Pendidikan dan Kebudayaan Nomor 92 Tahun 2014, kenaikan jenjang jabatan akademik dosen mewajibkan untuk publikasi pada jurnal ilmiah nasional terakreditasi dan jurnal internasional bereputasi di bidangnya.

Menariknya Menristekdikti juga menerbitkan Peraturan Menteri Riset, Teknologi, dan Pendidikan Tinggi Nomor 20 Tahun 2017 tentang Pemberian Tunjangan Profesi Dosen dan Tunjangan Kehormatan Profesor yang mengamanatkan bahwa publikasi ilmiah merupakan salah satu indikator untuk melakukan evaluasi terhadap pemberian tunjangan profesi dosen dan tunjangan kehormatan guru besar. Kebijakan ini memberikan suatu arah agar publikasi ilmiah jabatan fungsional Dosen di Indonesia terus ditingkatkan. Adapun tujuan dari publikasi ini adalah mensosialisasikan hasil temuan dari kajian atau penelitian berdasarkan evidence (bukti/kebenaran/fakta/data) di lapangan baik di tingkat lokal, nasional, regional dan internasional. (Nasir, 2017).

Dengan berbagai instrument dan peraturan demi tercapainya peningkatan publikasi ilmiah di Indonesia tidak luput dari kualitas artikel tersebut. Beberapa tahun terakhir, dunia pendidikan dikejutkan oleh munculnya pemberitaan tentang kasus yang terkait dengan plagiarisme terhadap sejumlah karya ilmiah. Plagiarisme telah menciptakan iklim yang buruk, khususnya terhadap dunia pendidikan. Tindakan ini dapat membunuh ide dan gagasan serta menurunkan tingkat kreativitas. Tindakan ini juga mengakibatkan terjadinya degradasi moral dan mental kaum intelektual. Terbitnya Permendiknas No. 7 Tahun

2010 diharapkan mampu mencegah dan menanggulangi tindakan tersebut. Namun, tanpa metode yang tepat tindakan ini sulit dicegah (Afdhal, 2014).

Demi peningkatan publikasi yang berkualitas muncul beberapa permen, kebijakan hingga undang-undang terkait plagiat dan hak cipta. Sebagian, beberapa Kopertis Wilayah menerapkan aturan kecil dalam pengecekan publikasi sebagai syarat kenaikan jabatan fungsional. Selain itu, munculnya hibah bagi penerbit jurnal di Indonesia menjadi semangat tersendiri bagi penerbit untuk menerbitkan karya tulisan. Sisi positif dari hal ini adalah terjadinya musim penerbitan tulisan maupun penerbit di Indonesia. Ini terbukti bahwa peneliti di Indonesia telah menerbitkan 43.296 Jurnal, 3.178 Book Chapters, 18.372 Conference Papers, 4.280 Buku (Sumber : sinta.ristekdikti.go.id). Tak hanya itu, statistik ISSN menunjukkan bahwa tingkatan registrasi ISSN naik signifikan sejak pada Tahun 2015 yaitu sebesar 14.2 %, tetapi jika diukur dengan media publikasi yang mendaftar terbitan, P-ISSN masih mendominasi dengan 70.2 %, jauh dengan terbitan E-ISSN sebesar 29.6 % dari jumlah 34.511 (Sumber : ISSN LIPI). Kecenderungan dari jumlah penerbit jurnal yang masih dalam bentuk cetak dikarenakan karena “ketakutan” akan hal plagiarisme publikasi jika dimuat dalam online.

Di Indonesia, telah mendorong pergerakan dan perubahan perilaku kehidupan masyarakat dalam organisasi maupun perekonomian (Wali dan Ahmad, 2017). Seiring dengan perkembangan teknologi informasi dan komunikasi, muncul sejumlah aplikasi anti-plagiat sebagai alternatif untuk mencegah tindakan plagiarisme. Aplikasi seperti turnitin, plagiarism checker, seo tools menjadi alternative terbaik digunakan oleh penerbit. Banyak aplikasi yang bisa direkomendasikan untuk digunakan oleh penerbit jurnal di Indonesia, tetapi yang terbaik adalah yang berbayar dan tidak gratis.

Maraknya kasus penjiplakan yang dilakukan oleh golongan intelektual menjadi suatu tragedi dalam dunia pendidikan Indonesia. Akibat banyaknya informasi tersedia secara online yang memprihatinkan tersebut dapat dengan mudah dilakukan dengan kebiasaan copy-paste tanpa menyebutkankan referensi (Purwitasari, 2017). Menurut

Sastroasmoro (2007), plagiarisme adalah penggunaan ide, pikiran, data, kalimat orang lain seolah-olah sebagai miliknya tanpa menyebutkan sumbernya. Plagiarisme merupakan salah satu pelanggaran serius yang bersifat universal terhadap etika akademis (*scientific misconduct*). Tindakan plagiarisme ini merupakan sebuah tindakan tercela, karena merupakan ketidakjujuran alias kebohongan. Tulisan atau potongan tulisan yang diambil dari tulisan orang lain, sengaja atau tidak sengaja, bila tidak dirujuk dengan baik dan benar, dapat dikategorikan sebagai plagiarisme (Isa dan Abidin, 2013). Yadav, Rawal dan Baxi (2016) menjelaskan praktek mengambil gagasan orang lain atau menyalin teks sepenuhnya atau dengan mengulang sedikit merupakan plagiarisme. Plagiarisme adalah bentuk serius dari kesalahan ilmiah. Literatur ilmiah penuh dengan bukti di mana manuskrip dibatalkan karena plagiarisme.

Terkait dengan hal tersebut, muncul beberapa metode pengecekan plagiarisme serta beragam algoritma untuk menghasilkan hasil plagiarisme dari publikasi ilmiah. Hermansyah, Pramono dan Subardin (2016) melakukan penelitian menggunakan metode deteksi kemiripan kode menggunakan analisis leksikal dapat digunakan untuk merubah kode menjadi token unik. Token ini digunakan untuk mengidentifikasi struktur program yang diperiksa. Sebelum proses pencocokan token menggunakan array, terlebih dahulu token yang terbentuk dari kedua source code yang diperiksa di sorting atau diurutkan berdasarkan abjad. Setelah itu dilakukan proses pencocokan string menggunakan array per posisi array. Semakin besar kemiripan alur dan kode program maka semakin besar kemungkinan kode tersebut merupakan plagiat. Nilai presentase kemiripan kode program yang dihasilkan dari proses pencocokan token ini yang nantinya digunakan sebagai acuan penentuan plagiarisme dengan ketentuan jika nilai presentase lebih besar atau sama dengan 70 % maka source dianggap plagiat, namun jika lebih kecil dari 70% tidak dianggap sebagai plagiat.

Ulfa dan Mustikasari (2017) menggunakan algoritma winnowing yang merupakan salah satu algoritma pada metode document fingerprinting. Metode ini akurat dalam mengidentifikasi penyalinan teks termasuk bagian kecil yang mirip dalam sekumpulan dokumen melalui fingerprint

yang dihasilkan. Melalui pencocokan fingerprint akan diperoleh nilai similarity antar dokumen. Semakin kecil tingkat persentase kesamaan dokumen teks yang diuji, maka dokumen tersebut tidak termasuk plagiat, tetapi jika hasil dari pengujian pada dua dokumen semakin besar, makadisimpulkan bahwa dokumen tersebut menyerupai tindakan plagiat.

Sistem deteksi plagiarisme umumnya menerapkan algoritma pencocokan string dalam dokumen teks untuk mencari kata-kata umum di antara dokumen. Ada beberapa algoritma yang digunakan untuk pencocokan string, dua di antaranya adalah algoritma Rabin-Karp dan Jaro-Winkler Distance. Algoritma Rabin-Karp adalah salah satu algoritma yang kompatibel untuk memecahkan masalah beberapa pola string, sedangkan algoritma Jaro-Winkler Distance memiliki kelebihan dalam hal waktu. Aplikasi pendeteksian plagiat dikembangkan dan diuji pada berbagai jenis dokumen, yaitu doc, docx, pdf dan txt. hasil eksperimen yang diperoleh bahwa kedua algoritma ini dapat digunakan untuk melakukan deteksi plagiarisme terhadap dokumen tersebut, namun dari segi keefektifannya, algoritma Rabin-Karp jauh lebih efektif dan lebih cepat dalam proses mendeteksi dokumen dengan ukuran. lebih dari 1000 KB (Leonardo dan Hansun, 2017).

Yudhana dan Djayali (2017) menggunakan metode dalam penelitiannya untuk mewakili dua kumpulan objek dibandingkan dalam bentuk probabilitas. Algoritma Rabin-Karp yang dipilih karena algoritma tersebut merupakan algoritma pencocokan string yang menggunakan fungsi hash sebagai perbandingan antara string pencarian (m) dan Explodeing pada teks (n). Jika kedua nilai hash sama maka perbandingan akan dilakukan sekali lagi untuk karakter. Sistem yang dihasilkan adalah aplikasi berbasis web yang menunjukkan nilai kesamaan dua set objek. Sekarwati *et al* (2017) mengembangkan sistem untuk mengukur kemiripan yang ada terhadap suatu dokumen dengan dokumen lain yang telah disimpan dalam database internal. Hasil perhitungan kesamaan berupa sebuah dokumen yang membandingkan persentase kesamaan. Penelitian ini dilakukan menunjukkan bahwa algoritma (*similar text*) tersebut dapat digunakan untuk menguji kesamaan dokumen.

Penelitian dilakukan Sochenkov *et al* (2016) yang

menyajikan ikhtisar tentang *Exactus Like { a plagiarism detection system}*. Penguraian yang dalam untuk keselarasan teks membantu system untuk melihat bentuk plagiarisme tersembunyi. Hasil dari penelitian ini menghasilkan sebuah *prototype* yang diberinama Exactus Like yang tersedia secara online di <http://like.exactus.ru/index.php/en/>. Penelitian ini difokuskan pada deteksi plagiarisme yang sangat disamakan.

Al-Shamery dan Ghani (2016) melakukan penelitian untuk mengetahui cara yang tepat untuk mendeteksi plagiarisme semantic pada makna dan memanfaatkan sinonim dan menggantinya dari kata-kata asli. Penelitian ini juga bertujuan untuk menerapkan pre-processing untuk kata-kata penelitian dengan menggunakan *tokenization* dan *stop word remove processes*, kemudian diuji apakah penelitian masuk di bawah spesialisasi ilmu komputer atau tidak, dimana hanya penelitian yang akan terdeteksi plagiarisme semantic dengan menggunakan WordNet. Penelitian ini memberikan cara yang efektif untuk mendeteksi plagiarisme semantic penelitian tertulis, terutama oleh siswa yang memiliki plagiarisme besar dalam penelitian. Vani dan Gupta (2016) juga melakukan penelitian dengan menfokuskan dari teknik deteksi plagiat dan *tools* yang digunakan. Tetapi hanya menyajikan ulasan singkat tentang alat dan teknik plagiarisme ekstrinsik teks.

Hal berbeda dilakukan oleh Abdul-Rahman *et al* (2017), untuk memudahkan interaksi manusia dalam proses analisis visual untuk deteksi kesamaan teks, diawali dengan memetakan masalah perbandingan urutan berpasangan dengan pemrosesan gambar, yang memungkinkan pola kemiripan divisualisasikan sebagai pixelmap 2D, kemudian merancang antarmuka visual untuk memungkinkan pengguna membuat dan bereksperimen dengan detektor yang berbeda menggunakan *primitive metrics*, dengan cara yang serupa dengan pembuatan *pipeline* pengolahan citra. Selanjutnya, Esteki dan Esfahani (2016) melakukan penelitian yang menggunakan fitur statistik dan semantik untuk menentukan fungsionalitas *Support Vector Machines* (SVMs) dalam mendeteksi tindakan plagiarisme dalam bahasa Persia. Untuk meningkatkan akurasi, stemmer dirancang untuk membendung kata-kata Persia. Fitur statistik dan

semantik digunakan untuk melatih dan menerapkan SVM. Gambaran statistik yang digunakan adalah koefisien Jaccard, koefisien Dice, jarak Levenshtein, dan persamaan umum terpanjang. Untuk mendeteksi kesamaan semantik, sebuah metode baru yang disebut "*Index Words Replacement*" diusulkan. Kerangka yang diusulkan diuji pada data PAN. Hasilnya menunjukkan ketepatan 0,93337, mengingat 0,70124 dan Plagdet dari 0,80083.

Mengingat banyaknya metode dan algoritma yang diterapkan untuk mendeteksi plagiarisme, maka peneliti akan mengambil beberapa algoritma sebagai pengujian deteksi dokumen, tetapi penelitian ini hanya menfokuskan fungsi *similar text* pada bahasa pemrograman PHP.

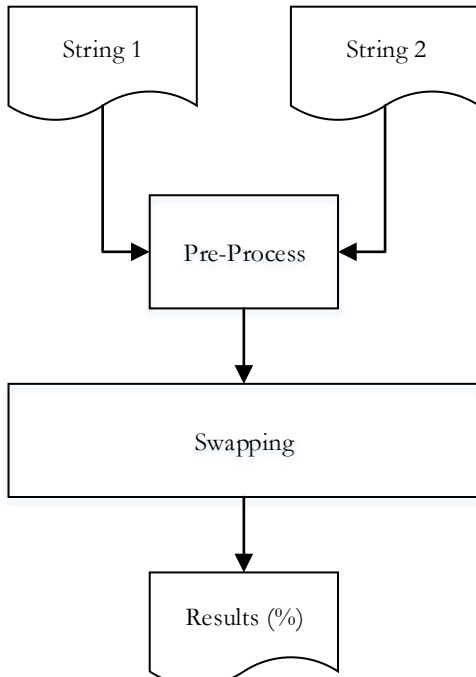
Similar text merupakan salah satu fungsi PHP yang merupakan algoritma sederhana untuk menghitung kesamaan antara dua string. String dimaksud dapat berupa angka, karakter, huruf atau kata yang dikelompokkan secara Bersama maupun terpisah sebagai string dengan variable panjang. Dalam prakteknya, *similar text* banyak digunakan di berbagai situs web untuk mencari keunikan teks, menghitung jumlah duplikasi hingga mengidentifikasi konten duplikat.

2. Metode Penelitian

Ada beberapa metode serta algoritma dalam mendeteksi sebuah dokumen, penelitian ini mengusulkan metode baru berdasarkan dokumen yang diunggah dengan menggunakan beberapa algoritma. Algoritma yang disarankan dalam bahasa PHP. Skema usulan pendekatan disajikan pada gambar 1. Fase yang berbeda dari usulan Proses pendeteksian plagiarisme akan dibahas di bawah ini.

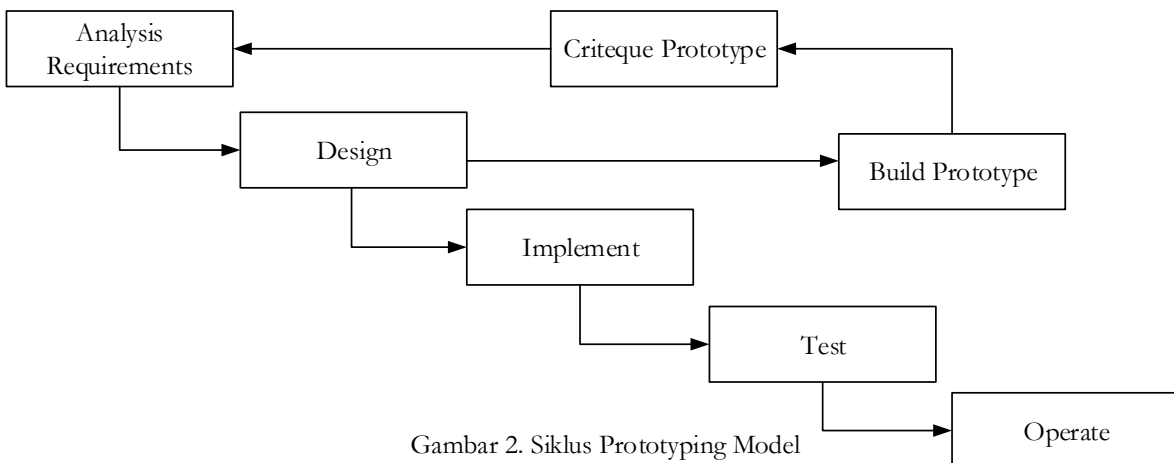
Parameter *similar text* diawali dengan string 1, string 2, dan seterusnya string *n*, yang selanjutnya *similar text* akan menghitung kemiripan string dalam bentuk persen. Dengan model perhitungan yaitu; membagi hasil *similar text* dengan rata-rata panjang string sehingga akan menghasilkan hasil persentase yang tertinggi 100. Hasil persentase hasil *similar text* juga dapat diatur sesuai dengan berapa persentase yang diambil dari string tersebut. Sebagai contoh: string 1 berisi "similarity PHP" dan string 2 berisi "PHP similarity" maka jika ingin mengambil hasil

persentase similar 0 maka hasil menampilkan >70, jika hasil persentase similar 100% maka hasil akan menampilkan deteksi string 0%.



Gambar 1. Proses Similar PHP

Sedangkan untuk proses pengembangan sistem menggunakan pendekatan *prototipe (prototyping)*. Metode ini sangat baik digunakan untuk menyelesaikan masalah kesalahpahaman antara *user* dan analis yang timbul akibat *user* tidak mampu. Proses pembuatan *prototipe* merupakan proses yang interaktif dan berulang-ulang yang menggabungkan langkah-langkah siklus pengembangan tradisional. *Prototipe* dievaluasi beberapa kali sebelum pemakai akhir menyatakan *prototipe* tersebut diterima. Gambar 2 di bawah ini mengilustrasikan proses pembuatan *prototipe* :

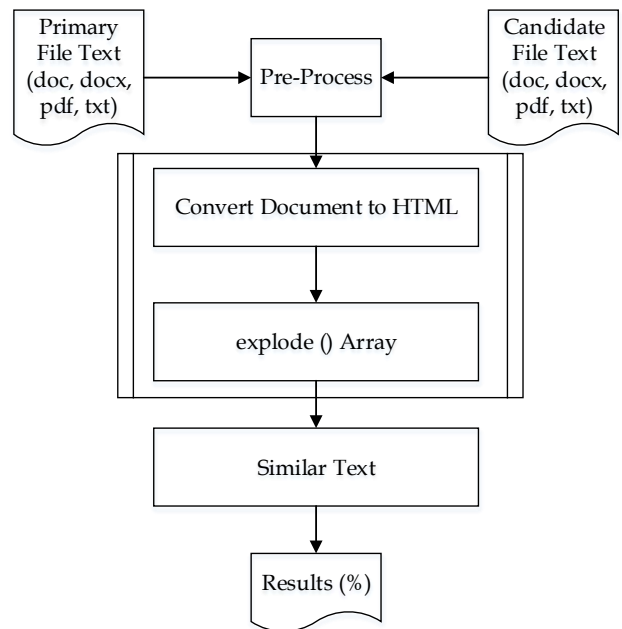


Gambar 2. Siklus Prototyping Model

3. Pembahasan

3.1. Analisa Sistem

Pada proses pembuatan aplikasi plagiarisme, dibutuhkan fase pada proses konversi *file* menjadi sebuah text yang telah ditentukan berdasarkan penentuan string yang akan diambil sebagai sebuah kalimat yang akan digunakan untuk proses *similar text*, yaitu setiap string akan ditetapkan dengan tanda titik (.) dan koma (,) dianggap sebuah kalimat string. Hal ini digunakan sebagai *array* dari sebuah dokumen dan akan dihitung dengan fungsi *similar text*. Proses tersebut akan terlihat pada gamabr 3 berikut.



Gambar 3. Similar text PHP dengan Fungsi Konversi dan Explode

Gambar 3 menunjukkan sebuah proses diawali dengan dokumen sebagai string baik berupa file word (doc, docx), Pdf, Txt akan dikonversi sebelumnya menjadi sebuah file dalam bentuk HTML sehingga dapat memudahkan dalam proses pengulangan string. Seperti dijelaskan sebelumnya, setiap string pertama akan dibuat ketika sebuah kata-kata pada sebuah dokumen diakhiri dengan tanda titik (.) atau koma (,) sehingga hasil dari pemisahan tersebut akan terbentuk sebuah string 1,2, n pada setiap dokumen baik dokumen awal (*primary*) atau target (*candidate*).

3.2. Mengekstrak Fitur Pengkodean

Ada beberapa Teknik yang digunakan sehingga menghasilkan sebuah proses *similar text*:

a. Konversi File Dokumen

Sebelumnya file akan dikonversi ke HTML, hal ini diuntungkan karena file tidak perlu diunggah ke dalam database nantinya. Pada proses ini dibutuhkan beberapa reader dokumen.

b. Explode ()

Fungsi explode adalah membagi sebuah string dengan sebuah string, pada fungsi PHP dapat digunakan kode :

```
array explode ( string $delimiter ,
string $string [, int $limit =
PHP_INT_MAX ] )
```

c. Persentase *Similar text*

Sebelum menampilkan sebuah hasil *similar text*, akan ada proses hasil duplikasi diambil dari 2 (dua) buah string atau lebih string dan ditentukan dengan persentase persamaan string tersebut.

```
int similar_text ( string $first , s
tring $second [, float &$percent ] )
```

d. Hasil *Similar text*

Dengan ketiga proses diatas akan dibuat sebuah kesimpulan persamaan string dengan menampilkan persentase plagiat dari string tersebut.

3.3. Analisa Kode *Similar text*

Dari Analisa system dan memproses fitur pengkodean *similar text* maka dapat diterapkan kode PHP berikut:

```
<?php
include('db.class.php');
include('docx_reader.php');

$id = $_GET['similar-text'];

$detail = mysqli_fetch_array(mysqli_query($db,
"select * from table1 where id = $id"));
$list_compaire = mysqli_query($db, "select * from
table1 where id <> $id");

$doc = new Docx_reader();
$doc->
setFile('uploads/dokumen/'.$detail['content']);

if(!$doc->get_errors()) {
    $html = $doc->to_html();
    $plain_text = $doc->to_plain_text();

    $total_plagiat = 0;
    $final_read = '';
    while ($data =
mysqli_fetch_array($list_compaire)) {
        $doc->
setFile('uploads/dokumen/'.$data['content']);
        $read_doc = $doc->to_plain_text();

        $final_read = explode('.',
$read_doc);

        foreach (explode('.', $plain_text) as
$key => $text) {
            foreach ($final_read as
$compare) {
                similar_text($text,
$compare, $percent);

                if ($percent > 90) {
                    echo $compare;
                    echo '<br/>';
                    echo
round($percent, 2). ' persen';
                    echo '<br/>';
                    echo 'dari
'.$data['field_id'];
                    echo '<br/>';

                    $total_plagiat
+= $percent;
                }
            }
        }

        mysqli_query($db, "update table1 set
persentase_plagiat = ".round(($total_plagiat *
mysqli_num_rows($list_compaire)) /100, 2)." where id
= $id");

        echo round(($total_plagiat
mysqli_num_rows($list_compaire)) /100, 2);
    } else {
        echo implode(' ', $doc->get_errors());
    }
}
```

Penjelasan dari kode diatas melibatkan 1 (satu) buah tabel dengan nama *table1* dan *field* (id, content, persentase_plagiat). Ada beberapa file *include* terdiri dari 2 (dua) yaitu; *db.class.php* merupakan sebuah file untuk koneksi ke database MySQLi dan *docx_reader.php* merupakan sebuah file untuk mengeksekusi file dokumen menjadi sebuah string HTML dan ditampilkan pada sebuah *browser*.

Hasil konversi dokumen akan dilakukan pemotongan string sebagai sebuah string dengan penggunaan fungsi `explode()`, karakter sebuah string akan dipisahkan dengan tanda titik dan koma. Pada kode diatas hasil plagiat akan dimunculkan dengan kesamaan string sebesar 90 persen. Dengan menggunakan matematika sederhana maka hasil dari string array () akan didapati dan dibagi dengan jumlah dokumen yang dianalisa.

3.4. Analisa Output

Data keluaran (*Output*) yang dihasilkan dari pengkodean ini adalah didapatkan sejumlah kesamaan string dengan persentase berbeda. Pada pengujian dilakukan melibatkan 10 (sepuluh) jurnal dengan rata-rata memiliki kata 4.500, maka didapati string tertinggi pada jurnal 1 dengan jumlah 562 dan terendah pada jurnal 9 dengan jumlah 487, rincian string dapat dilihat pada tabel 1 berikut:

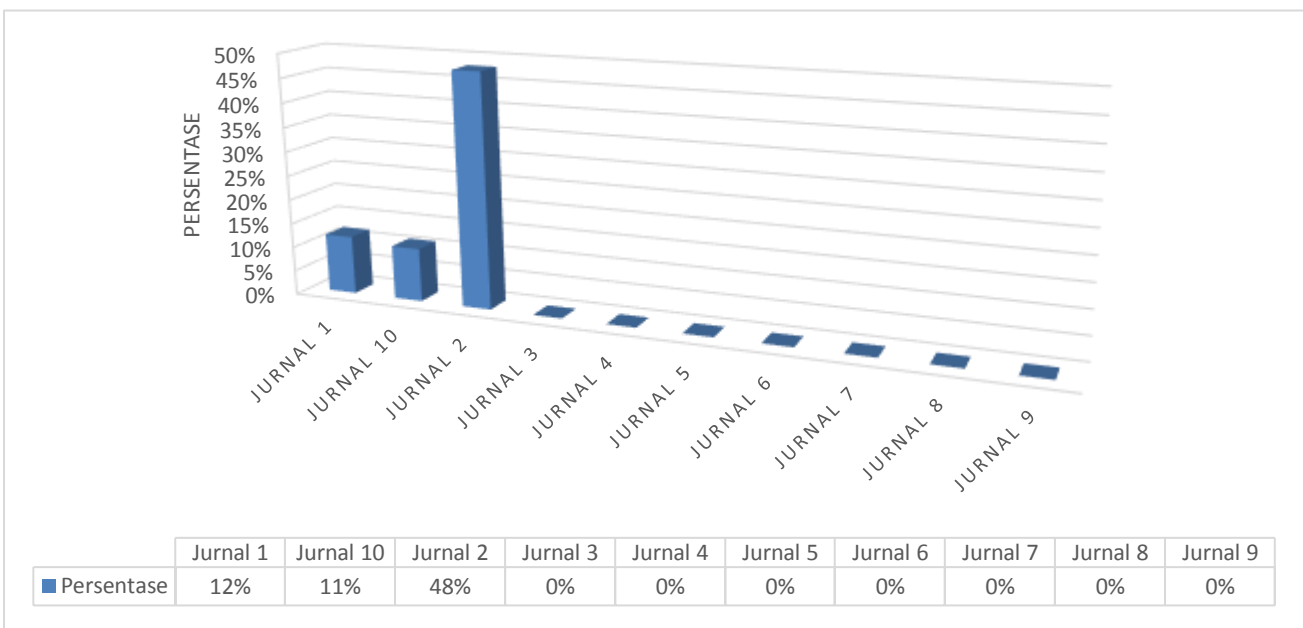
Tabel 1. Dokumen yang di Uji (Jurnal)

Dokumen	Jumlah String
Jurnal 1	562
Jurnal 2	480
Jurnal 3	570
Jurnal 4	640
Jurnal 5	526
Jurnal 6	209
Jurnal 7	368
Jurnal 8	344
Jurnal 9	487
Jurnal 10	521

Dari masing-masing jurnal tersebut, maka didapati suatu string. Dimana dari suatu string tersebut akan diuji tingkat kesamaan kedalam fungsi *similar text*. Maka dari pengujian jurnal dengan kode diatas maka dihasilkan persentase kesamaan secara rinci pada tabel 2.

Tabel 2. Hasil Pengujian Dokumen

Dokumen	Persentase Plagiat
Jurnal 1	12 %
Jurnal 2	48 %
Jurnal 3	0 %
Jurnal 4	0 %
Jurnal 5	0 %
Jurnal 6	0 %
Jurnal 7	0 %
Jurnal 8	0 %
Jurnal 9	0 %
Jurnal 10	11 %



Gambar 4. Hasil Evaluasi

3.5. Evaluasi

Data yang digunakan dalam pengujian ini menggunakan dokumen dengan ekstensi (doc, docx, pdf dan txt). Dokumen yang terlibat diambil dari berbagai jurnal berbahasa Indonesia sebanyak 10 (sepuluh). Setiap string yang terbentuk dengan penanda titik dan koma dijadikan sebuah string. Sebagai hasil kesamaan string, penelitian ini menggunakan persamaan string dengan tingkat kesamaan 90% kemiripan string. Hasil evaluasi dapat dilihat pada gambar 4.

4. Kesimpulan dan Saran

Dengan menggunakan fungsi *similar text*, penelitian ini menghasilkan sebuah deteksi dokumen terhadap 10 (sepuluh) jurnal dipilih dengan bahasa Indonesia, dokumen yang diuji merupakan dokumen dengan ekstensi doc, docx, pdf, dan txt. Dokumen tersebut akan dikonversi ke dalam bentuk html dan selanjutnya akan dilakukan pembuatan string dengan penanda titik (.) dan koma (,) akan dianggap sebuah string baru. Dengan menentukan persentase kemiripan string sebesar 90% maka dihasilkan sejumlah persamaan teks, pada kasus ini jurnal yang terdeteksi plagiat terbanyak adalah jurnal 1 dengan persentase kemiripan string sebesar 48%.

Penggunaan *similar text* dapat diusulkan sebagai Teknik pengkodean untuk deteksi plagiarisme pada penggunaan aplikasi deteksi anti plagiat. Dengan penggunaan *similar text* dapat menentukan nilai kesamaan persentase plagiat. Tetapi, jika string yang lebih panjang dicek, fungsi PHP `similar_text()` bisa sangat lambat, yang dapat mempengaruhi kinerja sebuah situs web. Karena itu, fitur ini hanya bisa digunakan untuk string pendek. Selain itu, setiap string dengan perubahan bentuk, tempat akan dinilai dengan persentase kemiripan berbeda pula.

5. Daftar Pustaka

Abdul-Rahman, A., Roe, G., Olsen, M., Gladstone, C., Whaling, R., Cronk, N., Morrissey, R. and Chen, M., 2017, January. Constructive visual analytics for text similarity detection. In Computer Graphics Forum (Vol. 36, No. 1, pp. 237-248).

Ajie, H. and Bangsa, A.S., 2017. Aplikasi Pendeteksi Dugaan Awal Plagiarisme Pada Tugas Siswa Dan

Mahasiswa Berdasarkan Kemiripan Isi Teks Menggunakan Algoritma Levenshtein Distance. PINTER-Jurnal Pendidikan Teknik Informatika dan Komputer, 1(1), pp.25-32.

Al-Shamery, E.S. and Ghenni, H.Q., 2016. Plagiarism detection using semantic analysis. Indian Journal of Science and Technology, 9(1).

Azmi, Afdhal & Chalis, Taufan & A. Gani, Taufiq. 2014. Analisa Perbandingan Aplikasi Pendeteksi Plagiat Terhadap Karya Ilmiah. Conference: Seminar Nasional dan Expo Teknik Elektro 2014, At Hotel Santika Premiere Dyandra, Medan - Sumatera Utara, Volume: ISSN: 2088-9984.

Esteki, F. and Esfahani, F.S., 2016. A Plagiarism Detection Approach Based on SVM for Persian Texts. In FIRE (Working Notes) (pp. 149-153).

Gupta, D., 2016. Study on Extrinsic Text Plagiarism Detection Techniques and Tools. Journal of Engineering Science & Technology Review, 9(5).

Hermansyah, H., Pramono, B. and Subardin, S., 2016. Deteksi Kemiripan Source Code Pada Bahasa Pemrograman Java Menggunakan Metode Analisis Leksikal. *semanTIK*, 2(1).

Isa, T.M. and Abidin, T.F., 2013. Mengukur Tingkat Kesamaan Paragraf Menggunakan Vector Space Model untuk Mendeteksi Plagiarisme. Australian Journal of Politics and History, 53, p.4.

Leonardo, B. and Hansun, S., 2017. Text Documents Plagiarism Detection using Rabin-Karp and Jaro-Winkler Distance Algorithms. Indonesian Journal of Electrical Engineering and Computer Science, 5(2), pp.462-471.

Nasir., M. 2017. Menristekdikti Dorong Dosen Tingkatkan Publikasi Ilmiah. <http://ristekdikti.go.id/menristekdikti-dorong-dosen-tingkatkan-publikasi-ilmiah/>. Diakses tanggal 17 Mei 2017 (23.23).

Peraturan Menteri Riset, Teknologi, dan Pendidikan Tinggi Nomor 20 Tahun 2017 tentang Pemberian Tunjangan Profesi Dosen dan Tunjangan

Kehormatan Profesor. Berita Negara Republik Indonesia Tahun 2017 Nomor 173. Jakarta.

Peraturan Menteri Pendidikan Dan Kebudayaan Republik Indonesia Nomor 92 Tahun 2014 Tentang Petunjuk Teknis Pelaksanaan Penilaian Angka Kredit Jabatan Fungsional Dosen. Menteri Pendidikan Dan Kebudayaan Republik Indonesia. Jakarta.

Purwitasari, D., 2017. Deteksi keberadaan kalimat sama sebagai indikasi penjiplakan dengan algoritma hashing berbasis N-Gram. *Jurnal Ilmiah KURSOR*, 6(1).

Sastroasmoro, S., 2007. Beberapa Catatan Tentang Plagiarisme. *Majalah Kedokteran Indonesia*, 57(8), pp.239-244.

Sekarwati, K.A., Wiryana, I.M., Banowosari, L.Y. and Kerami, D., Semantic Space for Document Similarity Test.

Sochenkov, I., Zubarev, D., Tikhomirov, I., Smirnov, I., Shelmanov, A., Suvorov, R. and Osipov, G., 2016, March. Exactus Like: Plagiarism detection in scientific texts. In *European Conference on Information Retrieval* (pp. 837-840). Springer, Cham.

Ulfa, N.F. and Mustikasari, M., 2017. Pembuatan Aplikasi Pengukuran Tingkat Kemiripandokumen Berbasis Web Menggunakan Algoritma Winnowing. *Jurnal Ilmiah Informatika Komputer*, 21(3).

Wali, M. and Ahmad, L., 2017. Perancangan Aplikasi Source code library Sebagai Solusi Pembelajaran Pengembangan Perangkat Lunak. *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, 1(1), pp.39-47.

Yadav, S., Rawal, G. and Baxi, M., 2016. Plagiarism- A serious scientific misconduct. *Int J Health Sci Res*, 6(2), pp.364-366.

Yudhana, A. and Djayali, A.D., 2017. Implementation of Pattern Matching Algorithm for Portable Document Format. (*IJACSA*) *International Journal of Advanced Computer Science and Applications*, 8(11), pp. 509-512.