
DETEKSI KEMIRIPAN ARTIKEL MELALUI *KEYWORDS* DENGAN METODE *FUZZY STRING MATCHING* DALAM *NATURAL LANGUAGE PROCESSING*

Humuntal Rumapea

Program Studi Sistem Informasi, Universitas Methodist Indonesia, Medan, Indonesia

Email: hrumapea16@gmail.com

DOI: <https://doi.org/10.46880/jmika.Vol5No1.pp60-66>

ABSTRACT

Natural Language Processing (NLP) is part of artificial intelligence which focuses on natural language processing. That is the language commonly used by humans in communicating with each other. In writing, articles can also be applied. To find out the similarity of the contents of a scientific article to another will make it easier for readers to make selections and make it easier to collect similar documents. Likewise with differences in article content even though they use the same keywords. Measurements are made using the keywords contained in the abstract which consists of several words and the number of keywords. The method used is to use fuzzy string matching to get the number of keywords used in an article. Then the calculation will be done for each keyword used and will be sorted by priority according to the position of the keyword. The search will be carried out starting from the title, abstract, keywords, content, and references. The number of keywords found in each article is shared across all keywords found to generate a similarity percentage level. If the results found are the same in number from one article to another, it can even be categorized that the contents of the article have similar content. The testing process is carried out by counting the number of keywords in the source then comparing all the source keys to the destination articles in the database by searching and comparing each word.

Keyword: Article Similarity, Keywords, Fuzzy String Matching, Natural Language Processing, Artificial Intelligence, Search.

ABSTRAK

Natural Language Processing (NLP) merupakan bagian dari kecerdasan buatan yang berfokus pada pemrosesan bahasa natural. Itulah bahasa yang biasa digunakan manusia dalam berkomunikasi satu sama lain. Dalam menulis, artikel juga bisa diterapkan. Untuk mengetahui kemiripan isi suatu artikel ilmiah dengan yang lain akan memudahkan pembaca dalam melakukan seleksi dan memudahkan untuk mengumpulkan dokumen sejenis. Begitu juga dengan perbedaan konten artikel meski menggunakan kata kunci yang sama. Pengukuran dilakukan dengan menggunakan kata kunci yang terdapat pada abstrak yang terdiri dari beberapa kata dan jumlah kata kunci. Metode yang digunakan adalah dengan menggunakan pencocokan string fuzzy untuk mendapatkan jumlah kata kunci yang digunakan dalam sebuah artikel. Kemudian akan dilakukan perhitungan untuk setiap kata kunci yang digunakan dan akan diurutkan berdasarkan prioritas sesuai dengan posisi kata kunci tersebut. Pencarian akan dilakukan mulai dari judul, abstrak, kata kunci, konten, dan referensi. Jumlah kata kunci yang ditemukan di setiap artikel dibagikan di semua kata kunci yang ditemukan untuk menghasilkan tingkat persentase kesamaan. Jika hasil yang ditemukan sama jumlahnya dari satu artikel ke artikel lainnya, bahkan dapat dikategorikan bahwa konten artikel tersebut memiliki konten yang serupa. Proses pengujian dilakukan dengan menghitung jumlah kata kunci pada sumber kemudian membandingkan semua kunci sumber dengan artikel tujuan di database dengan mencari dan membandingkan setiap kata.

Kata Kunci: Kemiripan Artikel, Kata Kunci, Pencocokan String Fuzzy, Natural Language Processing, Kecerdasan Buatan, Pencarian.

PENDAHULUAN

Dengan meningkatnya jumlah artikel dalam bentuk dokumen digital terutama pada bidang akademik seperti skripsi, tesis, jurnal, prosiding dan sejenisnya. Sehingga kemungkinan kesamaan atau pun kemiripan dokumen dapat terjadi seperti pada judul

tulisan, abstrak, permasalahan, metode yang digunakan, objek penelitian, pembahasan, bahkan isinya maupun hasilnya. Sehingga untuk menghindari pengulangan penulisan yang sama perlu dilakukan antisipasi agar kesamaan dokumen dapat dihindari (Ariantini, Lumenta, & Jacobus, 2016). Mendeteksi

adanya kemiripan suatu dokumen/artikel ke beberapa dokumen dengan cara dibandingkannya konten dokumen sehingga dapat menghasilkan bobot atau penilaian kemiripan dari dokumen tersebut. Salah satu kegunaan perbandingan isi dokumen adalah untuk membantu pengguna dalam pengelompokan dokumen dan juga memungkinkan pengguna mengetahui apakah isi dokumen yang satu merupakan dokumen yang pada dasarnya sama dengan dokumen yang lain (D. R. Manalu, 2016). Hal ini berfungsi untuk mengetahui apakah sebuah dokumen mirip dengan dokumen (Ariantini et al., 2016).

Pengujian kemiripan ini dilakukan dengan cara beberapa teknik, seperti teknik pencarian informasi, penghitungan statistik, atau menggunakan informasi sintaktik dari kalimat perkalamatnya (IlmuSkripsi.com, 2016). Pendekatan ini tidaklah sempurna sekali, karena masih terdapat beberapa kekurangan dan kelemahan, seperti penghitungan statistik yang membandingkan frekuensi kata dari dokumen satu dengan dokumen lainnya, serta mengabaikan atau tidak memperhatikan struktur dari kalimat. Sedangkan pada teknik sintaktik kalimat lain, pengurutan kata pada kalimat diperiksa unsur semantiknya dengan cara mengolah letak kata sesuai tata bahasanya maupun dengan penggantian sebuah kata sinonim dari kata tersebut. Cara ini mempunyai kelemahan, setiap kata dikelompokkan pada label masing-masing untuk diketahui struktur kalimatnya.

Tulisan ini bertujuan mengembangkan pengukuran kemiripan satu artikel dengan artikel lainnya yang terdapat dalam satu database. Proses yang akan dilakukan adalah dengan membandingkan jumlah penggunaan kata kunci (*keywords*) yang ada pada sebuah artikel menggunakan metode *fuzzy string matching*. Pengujian artikel dapat dibandingkan dengan beberapa artikel lain yang telah dikelompokkan dalam sebuah server atau pusat database. Dimana informasi terhadap sebuah dokumen tersebut sudah dimasukkan terlebih dahulu dalam bentuk digital seperti kata kunci, abstrak, isi (pedahuluan, teori pendukung, metode yang digunakan, pengujian, hasil, kesimpulan serta referensi yang digunakan. Untuk itu diharapkan artikel yang diuji kemiripannya dapat menghasilkan persentase kemiripan antara artikel dipilih oleh pengguna dibandingkan dengan sekumpulan artikel lain dalam database. Maka pengguna dokumen dapat terbantu dalam mengetahui isi dari sebuah dokumen/karya ilmiah tanpa harus membaca isi secara keseluruhan dokumen tersebut.

Metode Fuzzy string matching digunakan dalam pencarian kata yang terdapat dalam sebuah artikel

untuk mendapatkan jumlah penggunaan kata dan valid dengan memanfaatkan logika fuzzy untuk mencari tingkat kebenaran dimana melakukan pencarian string yang benar sama dan juga string yang mendekati dengan string lain terkumpul meskipun tidak sama persis susunan karakternya. Metode ini sangat membantu dalam proses pencarian kata yang sama serta yang mirip dengan kata kunci yang dipergunakan. Cara ini melakukan pencarian pada string yang serupa dan juga string yang mendekati pada string lainnya yang terkumpul dalam sebuah kamus.

Kunci dari teknik pencarian ini ialah cara untuk memutuskan bahwa sebuah string yang dapat dicari memiliki kesamaan dengan string dalam kamus, walaupun tidak sama persis pada susunan karakternya. Untuk memutuskan kesamaan dipergunakan sebuah fungsi yang diibaratkan sebagai *similarity function*. Dalam hal ini kecerdasan buatan sangat berperan penting dalam mempercepat proses pengujian untuk kemiripan sebuah artikel salah satunya adalah dengan penerapan Pemrograman Bahasa Alami. Adapun tujuan dari pembuatan program yang memiliki kemampuan untuk memahami bahasa manusia. Tujuan dari pemrograman bahasa alami ialah melakukan proses pembuatan model komputasi dari bahasa sehingga bisa terjadi suatu interaksi antara manusia dengan komputer dengan perantara bahasa alami.

Konsep tentang logika fuzzy diperkenalkan oleh Prof. Lotfi Astor Zadeh pada tahun 1962. Logika fuzzy merupakan sebuah metodologi sistem kontrol dalam pemecahan masalah, sesuai untuk diimplementasikan pada sebuah sistem, mulai dari sistem sederhana, sistem kecil hingga *embedded system*, jaringan komputer, multi-channel atau workstation berbasis akuisisi data, serta sistem control.

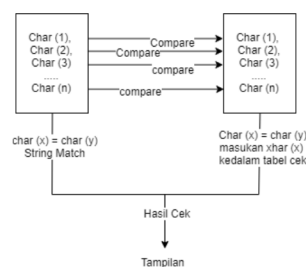
TINJAUAN PUSTAKA

Fuzzy String Matcing

Algoritma pencarian string atau dapat disebut juga algoritma pencocokan string yaitu algoritma untuk dapat melakukan pencarian semua kemunculan teks/string pendek dan dan panjang, untuk string pendek yang disebut pola/pattern dan string yang lebih panjang yang disebut teks. string pendek adalah pattern [0..n-1] string panjang adalah teks [0..m-1]. Algoritma yang digunakan dalam pencarian string ini dapat diklasifikasikan ke 3 bagian menurut arah pencariannya . Arah yang paling alami yaitu dimulai dari kiri ke kanan, kemudian yang merupakan arah untuk membaca, algoritma ini adalah (Fauzy & Kusriani, 2019): Algoritma Brute Force dan Algoritma dari Morris & Pratt, yang selanjutnya dikembangkan Knuth,

Morris, dan Pratt. Fuzzy String Matching merupakan salah satu metode pencarian string dengan menggunakan proses pendekatan pada pola dari string yang akan dicari. Dalam melakukan proses pencarian pada string yang sama dan juga string mendekati string lainnya akan terkumpul dalam sebuah kamus. Kunci konsep pencarian seperti ini ialah bagaimana memutuskan bahwa sebuah string yang akan dicari dapat memiliki kesamaan dengan string di kamus, meskipun tidak sama persis dalam susunan karakternya. Untuk memutuskan kesesuaian dipergunakan sebuah fungsi yang disebut sebagai *similarity function* (Fauzy & Kusri, 2019). Fungsi ini bertugas dalam memutuskan string hasil pencarian jika ada ditemukan string hasil pendekatan (aproksimasi). *Approximate String Matching* adalah pencocokan string dengan dasar kemiripan dari segi penulisannya, tingkat kemiripan ditentukan dengan jauh tidaknya beda penulisan dengan dua buah string dibandingkan itu.

Operasi perubahan string tersebut dapat berupa melakukan perubahan dari satu huruf ke huruf lainnya, menghapus satu huruf dari string, atau memasukkan satu huruf ke dalam string (Wangsanegara & Subaeki, 2015). Operasi ini digunakan dalam menghitung jumlah perbedaan dapat diperlukan dalam pertimbangan kecocokan suatu string dengan string sumber, jumlah perbedaan tersebut diperoleh dari penjumlahan semua perubahan yang terjadi pada masing-masing operasi. Operasi untuk mengubah string ada 3 operasi, yaitu: 1) Operasi Penghapusan yang dilakukan dengan menghapus karakter pada suatu kata supaya menyesuaikan string sumber (S) dengan string target (T); 2) Operasi Penyisipan dilakukan dengan menyisipkan karakter pada suatu indeks tertentu untuk menyamakan string sumber (S) dengan string target (T); 3) Operasi Pertukaran dilakukan dengan mengganti suatu karakter untuk menyamakan string sumber (S) dengan string target (T) (IlmuSkripsi.com, 2016).



Gambar 1. Proses Alur Pencocokan String

Natural Language Processing (NLP)

NLP adalah bagian dari ilmu komputer dan ilmu linguistik yang mengkaji interaksi antara komputer dengan bahasa (alami) manusia. NLP sering dianggap sebagai cabang dari kecerdasan buatan dan bidang kajiannya bersinggungan dengan linguistik internasional. Kajian NLP mencakup segmentasi tuturan (*speech segmentation*), penandaan kelas kata (*part-of-speech tagging*), segmentasi teks (*text segmentation*), penentuan makna (*word sense disambiguation*). Meskipun demikian kajian ini dapat mencakup teks, pemrosesan kata (*speech processing*) telah berkembang menjadi suatu bidang kajian terpisah. Secara umum *Natural Language Processing* terbagi menjadi dua, yaitu *text-based application* dan *dialog-based application*. *Text-based application* adalah segala macam aplikasi yang melakukan proses terhadap teks tertulis seperti dokumen, e-mail, buku, dan lain sebagainya (Rohman, Utami, & Raharjo, 2019). *Dialog-based application* adalah segala macam aplikasi yang melakukan proses terhadap bahasa lisan atau pengenalan suara, akan tetapi bisa juga memasukan interksi dialog dengan mengetikkan teks pertanyaan melalui keyboard.

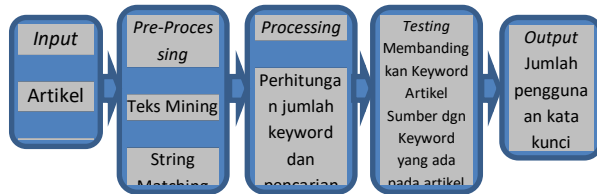
Natural Language Processing (NLP) merupakan salah satu cabang ilmu AI yang berfokus pada pengolahan bahasa natural (Gurning, Zarnelly, & Adawiyah, 2016; Rohman et al., 2019). Bahasa natural secara umum dapat digunakan oleh manusia pada saat berkomunikasi satu dengan yang lain. Bahasa tersebut akan diterima oleh komputer membutuhkan waktu diproses serta dipahami terlebih dahulu agar maksud dari pengguna dapat dipahami dengan baik komputer. Ada beberapa terapan aplikasi yang digunakan dari NLP. Seperti Chatbot (digunakan untuk pengguna dapat seolah-olah melakukan hubungan/komunikasi dengan komputer), Lemmatization dan Stemming (pemotongan kata pada bahasa tertentu diubah pada bentuk dasar pengenalan fungsi setiap kata suatu kalimat), Summarization (adalah ringkasan dari bacaan), Translation Tools (menerjemahkan bahasa) serta aplikasi lain yang dimungkinkan peralatan komputer dapat mampu memahami perintah/instruksi bahasa yang dimasukkan oleh user.

METODOLOGI

Skema Proses dan Aliran Data

Cara kerja pengukuran kemiripan artikel ini adalah dimulai dengan pengumpulan data sesuai dengan kebutuhan penelitian, kemudian akan dilakukan *prepossessing (text mining)* untuk menghindari data yang tidak valid dalam pengujian. Selanjutnya akan

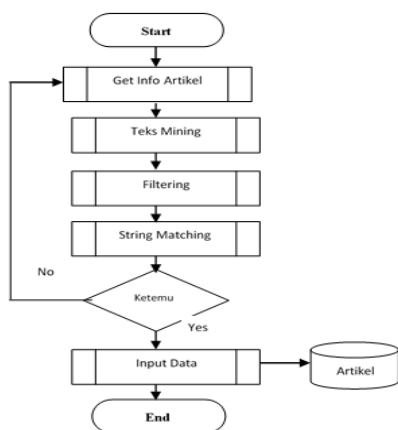
dilakukan proses input data kedalam database sesuai dengan format yang telah ditentukan. Proses selanjutnya adalah pemilihan artikel untuk diuji dan melakukan pencocokan kata (*string matching*) untuk menghindari kata kunci (*keywords*) ganda pada setiap dokumen. Kemudian akan masuk ke prosesing pencarian kata kunci. Untuk mendapatkan hasil kemiripan dokumen dilakukan perhitungan. Kerangka alur kerja data dilihat pada Gambar 1 berikut ini.



Gambar 2. Skema Proses Global dan Aliran Data

Preprocessing

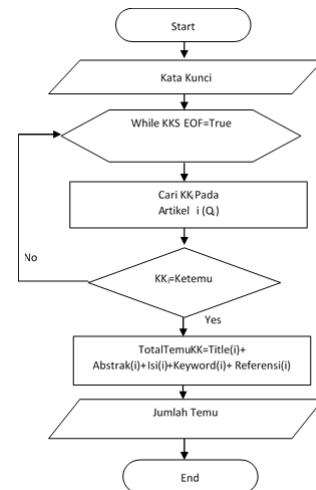
Proses yang dilakukan pada tahapan ini adalah teks mining dan *string matching*. Teks mining digunakan untuk membentuk kata yang terstruktur dengan baik sehingga memiliki arti dan merupakan. Kemudian akan dilakukan proses string matching pada saat pemasukan kata kunci untuk satu dokumen, sehingga terhindar dari redundan kata kunci dalam satu dokumen. Adapun kata kunci yang digunakan adalah kata kunci yang terdapat pada abstrak. Proses dapat digambarkan pada Gambar 2 *flowchart preprocessing* berikut ini:



Gambar 3. Flowchart Preprocessing

Metode Pengujian

Adapun proses pengujian kemiripan dokumen adalah seperti gambar 3 berikut ini:



Gambar 4. Proses Pengujian Kemiripan

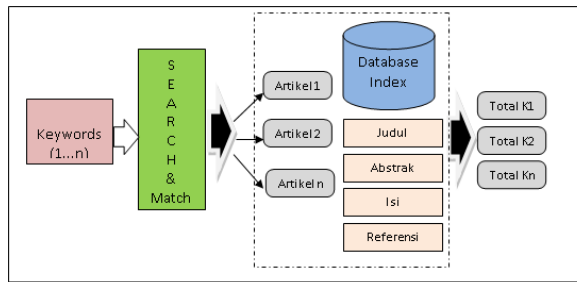
Dari gambar diatas menjelaskan proses pengujian yaitu dari kata kunci yang ditemukan akan dibandingkan mulai dari Kata Kunci pertama (KK_1) ke setiap record yang ada pada database mulai dari record awal sampai dengan record terakhir. String Matching antar KK akan dilakukan pada atribut judul jurnal, abstrak jurnal, keyword abstrak, isi, dan referensi jurnal. Jika kata tersebut ditemukan maka jumlah kata yang ditemukan akan bertambah 1. Dapat dituliskan dalam algoritma berikut ini:

```

while Art EOF
    Count(KK(i))=TemuKK(i)Art(i)[tit(i)],[abs(i)],[key
    (i)], [Isi(i)], [ref(i)]+1
    Movenext Art(i)
Loop
TotalKK(i)[ Art(i)]=Count(KK(i))
    
```

Perhitungan Kemiripan

Tahapan yang dilakukan adalah dengan membuat query dalam menyeleksi artikel yang akan dibandingkan dengan database jurnal. Kemudian akan menghasilkan kata kunci (Darwis Robinson Manalu, Rajagukguk, Siringoringo, Siahaan, & Sihombing, 2019). Akan dilanjutkan dengan perbandingan dengan dokumen yang ada pada database dengan menggunakan query pada judul tulisan, kata kunci, abstrak, isi dan referensi sebuah artikel yang telah deindex sebelumnya. Pada database telah dilakukan pengelompokan berdasarkan kriteria seperti kategori jurnal/paper, bidang ilmu, tahun dan lainnya.



Gambar 5. Perhitungan Persentase Kemiripan

Setelah dilakukan pencarian pada database dan jumlah kata kunci yang terdapat pada setiap dokumen akan dibagikan dengan total semua kata kunci yang ditemukan pada database. Proses perhitungan seperti berikut ini;

For $i = 1$ to Jumlah_Art_Temu
 Persentase Mirip Art(i) = $\frac{\text{TotalTemuKK_Art}(i)}{\text{TotalTemuKK_All}}$
 Next i

Tahapan perhitungan tersebut adalah:

1. Pertama sekali dilakukan adalah teks/string dicocokkan oleh kamus data kemudian apabila memiliki kesesuaian maka hasil perhitungan akan ditampilkan.
2. Jika tidak atau belum ditemukan string/teks tersebut dilanjutkan pada tahapan yang berikutnya yaitu string dicek kesesuaian huruf awal serta huruf akhir, apakah memiliki panjang (*len*) yang sama, jika ditemukan kecocokan akan dilakukan perhitungan dan menampilkannya.
3. Jika belum ditemukan string tersebut akan dilanjutkan pada tahap berikutnya yaitu hanya memeriksa huruf awal dan akhir dengan mengabaikan ukuran panjang/len, jika ditemukan kecocokan akan dilakukan perhitungan.
4. Selanjutnya adalah bila belum ditemukan kesesuaian sama sekali dengan kamus data tersebut, maka teks/string akan dihitung sesuai dengan yang dimasukkan oleh pengguna.

PEMBAHASAN

Dalam penjelasan sebelumnya tujuan dari tulisan ini adalah mengetahui kemiripan sebuah artikel dengan artikel lainnya, agar informasi yang didapat dari beberapa artikel yang memiliki kategori yang sama atau bidang ilmu/topik artikel; yang sama. Sehingga berdasarkan pengujian kemiripan dapat memberikan informasi lebih awal tentang isi artikel tersebut. Maka yang menjadi pembanding yang digunakan adalah kata kunci yang terdapat pada abstrak setiap tulisan yang

ada. Aspek kedua yang akan dibahas adalah peringkat kemiripan artikel dari database yang tersedia.

Dengan banyaknya artikel saat ini terutama dalam bentuk digital semakin memudahkan penulis mencari referensi. Namun dalam kenyataannya ada dokumen yang sangat mirip dalam kasus yang serupa, termasuk referensi yang digunakan. Bagi pembaca atau user yang akan merujuk atau untuk mengetahui isi dari sebuah tulisan tentunya akan membutuhkan waktu yang banyak menyelesaikan itu. Sehingga perlu dilakukan pengujian kemiripan dokumen dengan menggunakan kata kunci yang ada pada setiap artikel. Faktanya ada artikel yang memiliki kata kunci yang mirip dan bahkan sama tahap.

Adapun tahapan yang dilakukan dalam proses pencarian dan pencocokan string sebagai berikut:

1. Memilih atau menentukan artikel yang akan dibandingkan dengan artikel lainnya setelah memilih topik, kategori atau bidang ilmu. Tujuannya adalah untuk menghindari pengujian yang tidak sesuai dengan topik.
2. Menentukan Kata kunci yang akan digunakan dengan mengurutkan berdasarkan posisi penulisan, jika lebih dari satu kata maka akan diurutkan sesuai posisi kata kemudian dilanjutkan pada kata kunci berikutnya.

Contoh:

Keywords: *Fuzzy String Matching, Natural Language Processing, Kemiripan Artikel, Kecerdasan Buatan, Pencarian*

Dari kata kunci ini akan dilakukan indeks dengan memisahkan setiap kata menjadi.

Tabel 1. Keyword pada Artikel Sumber

Keyword 1	<i>Fuzzy String Matching</i>	<i>(Fuzzy, String, Matching)</i>
Keyword 2	<i>Natural Language Processing</i>	<i>(Natural, Language, Processing)</i>
Keyword 3	<i>Kemiripan Artikel</i>	<i>(Kemiripan, Artikel)</i>
Keyword 4	<i>Kecerdasan Buatan</i>	<i>(Kecerdasan, Buatan)</i>
Keyword 5	<i>Pencarian</i>	<i>(Pencarian)</i>

3. Membuat index setiap kata kunci menjadi:

Tabel 2. Index Keyword pada Artikel Sumber

No	Kata	Index
1	Fuzzy	0
2	String	1
3	Matching	2
4	Natural	3
5	Language	4
6	Processing	5
7	Kemiripan	6
8	Artikel	7
9	Kecerdasan	8
10	Buatan	9
11	Pencarian	10

Dari tabel di atas apabila ditemukan tanda baca dan spasi seperti “’*&[]: dan lain-lain akan dihilangkan sehingga panjang kata (*len*) sesuai dengan yang karakter alphabet.

$$K_{total} = \{K_1 K_2 K_3 \dots K_n\}$$

Jika pembentukan indeks kata kunci telah selesai maka dapat dilanjutkan pada proses berikutnya.

4. Setelah kata kunci artikel sumber sudah ditentukan maka artikel lainnya juga akan diindeks mulai dari judul, abstrak, kata kunci, isi tulisan dan referensi. Maka kata kunci yang pertama (Kata Kunci Sumber = KKs) akan dibandingkan dengan kata yang ada pada artikel sumber. Sehingga diperoleh jumlah kata yang ditemukan pada artikel sumber. Selanjutnya kata kunci yang pertama (Kata Kunci Sumber = KKs) akan dibandingkan dengan kata yang ada pada artikel tujuan lainnya (Kata Kunci Tujuan = KKt).

Untuk kata yang benar-benar match (sama persis setiap karekater pada kata)

```

For i = 1 to count(KKs)
    Jlh=0
    For j = 1 to count(KKt)
        If KKs(i) = KKt(j) Then
            Jlh=Jlh+1
        End if
    Next j
    TotalKKs(i)=jlh
Next i
    
```

Untuk kata yang sama panjang dan huruf awal dan akhir sama

```

For i = 1 to count(KKs)
    Jlh=0
    
```

```

For j = 1 to count(KKt)
    k=len(KKt(j))
    If KKs(i).chr(1) = KKt(j).Chr(1) and
    KKs(i).chr(k) = KKt(j).Chr(k) and Len(KKs(i)) =
    Len(KKt(j)) Then
        Jlh_mirip=Jlh_mirip +1
    End if
Next j
TotalKKs(i)=jlh_mirip
Next i
    
```

Contoh : String → Strong, Pencarian → Pencairan

Maka akan ditampilkan hasil seperti berikut ini untuk kata yang sama dan sama panjang kata, huruf awal dan huruf akhir:

Tabel 3. Jumlah Kata Kunci Yang Ditemukan

No	Kata	Sumber					Tujuan				
		Art1	Art 2	Art 3	Art 4	Art N	Art 1	Art 2	Art 3	Art 4	Art N
1	Fuzzy	5	16	12	25	...					
2	String	15	16	20	18	...					
3	Matching	5	16	18	18	...					
4	Natural	8	8	10	13	...					
5	Language	11	12	12	13	...					
6	Processing	10	5	10	13	...					
7	Kemiripan	20	15	14	8	...					
8	Artikel	11	21	14	8	...					
9	Kecerdasan	5	9	8	6	...					
10	Buatan	4	3	5	4	...					
11	Pencarian	11	26	34	16	...					
	Total	105	147	157	142	N					

5. Menghitung dan membandingkan jumlah kata yang sudah dicari dan akan diurutkan berdasarkan jumlah kata yang sama paling besar.

Tabel 4. Hasil Pengujian

Artikel	Jumlah Kata Kunci	Kemiripan Pemakaian Kata Kunci dengan Artikel 1
1	105	-
2	147	71,43%
3	157	66,88%
4	142	73,94%
5	180	58,33%
N

Dari hasil pengujian di atas terdapat jumlah kata kunci pada artikel 1 sebanyak 105 jika dibandingkan dengan artikel yang ada dalam database, kemudian telah memilih kategori, bidang ilmu dan topik yang sejenis setelah dibandingkan maka pada artikel ke 4 memiliki jumlah penggunaan kata kunci yang lebih tinggi sebesar 73,94%. Dapat dikategorikan artikel ini memiliki kemiripan yang lebih besar.

KESIMPULAN

Setelah membahas dan menguji kemiripan artikel melalui penggunaan kata kunci dalam sebuah tulisan maka dapat diberikan beberapa kesimpulan antara lain:

1. Pengukuran kemiripan artikel dapat dilakukan dengan baik sesuai dengan metode yang digunakan dalam waktu yang cepat dibandingkan pada pengujian dengan membaca isi dokumen
2. Dari pengujian data yang dilakukan diperoleh hasil kemiripan yang bervariasi karena kata yang benar sama ditambahkan dengan kata yang mirip dengan kata kunci pada huruf awal dan akhir serta panjang kata tersebut.
3. Dari persentase yang dihasilkan dapat menggambarkan jika artikel tersebut memiliki isi yang mirip dan juga memiliki perbedaan dalam konteks kemiripan artikel dengan penggunaan kata kunci.

DAFTAR PUSTAKA

Ariantini, D. A. R., Lumenta, A. S. M., & Jacobus, A. (2016). Pengukuran Kemiripan Dokumen Teks Bahasa Indonesia Menggunakan Metode Cosine Similarity. *Jurnal Teknik Informatika*, 9(1).
<https://doi.org/10.35793/jti.9.1.2016.13752>

Fauzy, M. N., & Kusriani, K. (2019). Chatbot Menggunakan Metode Fuzzy String Matching Sebagai Virtual Assistant Pada Pusat Layanan Informasi Akademik. *Jurnal Informa*, 5(1), 61–67.

Gurning, A. I. A., Zarnelly, Z., & Adawiyah, A. (2016). Penerapan Fuzzy String Matching Pada Aplikasi Pencarian Tugas Akhir Mahasiswa Jurusan Sistem Informasi Berbasis Web (Studi Kasus: Fakultas Sains Dan Teknologi UIN Suska Riau). *Jurnal Ilmiah Rekayasa Dan Manajemen Sistem Informasi*, 2(1), 54–59.
<https://doi.org/10.24014/rmsi.v2i1.1810>

IlmuSkripsi.com. (2016). Algoritma Pencarian String.

Manalu, D. R. (2016). Pengujian Tingkat Kemiripan Skripsi Mahasiswa Dengan Algoritma Genetika Menggunakan POSI Formulation. *Jurnal METHODIKA*, 2(2), 75–82.

Manalu, Darwis Robinson, Rajagukguk, E., Siringoringo, R., Siahaan, D. K., & Sihombing, P. (2019). The Development of Document Similarity Detector by Jaccard Formulation. *2019 International Conference of Computer Science and Information Technology (ICoSNIKOM)*, 1–4.
<https://doi.org/10.1109/ICoSNIKOM48755.2019.9111494>

Rohman, A. N., Utami, E., & Raharjo, S. (2019). Deteksi Kondisi Emosi pada Media Sosial Menggunakan Pendekatan Leksikon dan Natural Language Processing. *Eksplora Informatika*, 9(1), 70–76.
<https://doi.org/10.30864/eksplora.v9i1.277>

Wangsanegara, N. K., & Subaeki, B. (2015). Implementasi Natural Language Processing Dalam Pengukuran Ketepatan Ejaan Yang Disempurnakan (EYD) Pada Abstrak Skripsi Menggunakan Algoritma Fuzzy Logic. *Jurnal Teknik Informatika*, 8(2).
<https://doi.org/10.15408/jti.v8i2.3185>