TEMU KEMBALI INFORMASI BERITA KEGIATAN PROGRAM STUDI MENGGUNAKAN ALGORITMA PEMBOBOTAN TF-IDF DAN COSINE SIMILARITY

INFORMATION RETRIEVAL FOR STUDY PROGRAM ACTIVITIES NEWS USING TF-IDF WEIGHTING AND COSINE SIMILARITY

Tresna Maulana Fahrudin[1)], Muhammad Haris Hartanto[2)], Alya Setya Paramita[3)], Amanda Aulia[4)], Rizqii Amaliyah Maulana[5)], Iqbal Ramadhan Anniswa[6)]
E-mail : [1)]tresna.maulana.ds@upnjatim.ac.id,
{[2]21083010045,[3]21083010046,[4]21083010048,[5]21083010063,[6]21083010111}@student.upnjatim.ac.id

[1,2,3,4,5,6] Program Studi Sains Data, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional "Veteran" Jawa Timur

Abstrak

Sebagian besar informasi disimpan dalam wujud digital pada media elektronik, salah satunya konten berita kegiatan program studi yang dipublikasikan pada halaman website resmi. Konten berita kegiatan program studi ini dapat berupa kegiatan seminar, workshop/lokakarya, kuliah tamu, dan kegiatan Tridarma lainnya yang dilaksanakan oleh dosen dan mahasiswa yang memungkinkan jumlah konten berita yang terus bertambah. Dalam proses pencarian konten atau informasi yang dibutuhkan oleh pengguna dibutuhkan suatu sistem yang mampu menemu kembalikan informasi secara relevan untuk melakukan pencarian terhadap sekumpulan dokumen teks. Salah satunya dengan pencarian kata-kata pada dokumen yang digunakan untuk mengetahui seberapa mirip isi konten dari suatu dokumen dengan dokumen lainnya. Oleh karena itu, pada penelitian ini bertujuan untuk merancang *prototype* sistem temu kembali informasi yang dapat melakukan pencarian konten berita kegiatan program studi menggunakan algoritma pembobotan TF-IDF dan Cosine Similarity. Dokumen teks yang digunakan dalam penelitian ini dikumpulkan dari konten berita kegiatan program studi yang diperoleh dari website resmi Program Studi Sains Data Universitas Pembangunan Nasional "Veteran" Jawa Timur yang berjumlah sebanyak 22 konten berita. Hasil percobaan menunjukkan bahwa 10 *query* menggunakan kata kunci yang berbeda mendapatkan hasil 100% kesesuaian dengan dokumen berita kegiatan Program Studi.
Kata kunci: *temu kembali informasi, konten berita, program studi, TF-IDF, cosine similarity*

Abstract

Most of the information is stored in digital form on electronic media, one of which is news content of study program activities published on the official website page. The news content of activity can be in the form of seminars, workshops, guest lectures, and other Tridarma activities carried out by lecturers and students that allow the number of news content to continue to grow. In the process of searching for content or information needed by users, a system that is able to find relevant information is needed to search for a set of text documents. One of them is by searching for words in documents that are used to find out how similar the contents of a document are to other documents. Therefore, this study aims to design a prototype of an information retrieval system that can search news content for study program activities using the TF-IDF weighting algorithm and Cosine Similarity. The text document used in this study were collected from news content of study program activities obtained from the official website of the Data Science at Universitas Pembangunan Nasional "Veteran" Jawa Timur, which has totally 22 news content. The experimental results show that 10 queries using different keywords get results 100% matched to the news document of the study program activities.
***Keywords**: information retrieval, news content, study program, TF-IDF, cosine similarity*

1. PENDAHULUAN

Informasi begitu cepat beredar dan mudah untuk diakses berkat kemajuan teknologi informasi dan komunikasi seperti saat ini. Kebutuhan terhadap akses informasi yang relevan

semakin meningkat di tengah volume data yang begitu besar dan masif. Salah satu solusi dari kebutuhan akses informasi tersebut adalah dengan menggunakan sistem temu kembali informasi[1]. Media elektronik menjadi tempat menyimpan informasi dalam wujud digital misalnya seperti dokumen teks dalam bentuk buku digital (*e-book*), karya ilmiah, dan lainnya. Sejumlah besar data yang dibutuhkan pengguna seharusnya dapat ditemukembalikan oleh sebuah sistem[2]. Dalam sebuah konten berita misalnya terkait kegiatan program studi, pencarian informasi terhadap dokumen-dokumen tersebut kurang efisien jika dilakukan prosedur pencarian secara manual karena informasi yang tersimpan memiliki ukuran yang besar dan akan terus berkembang.

Konten berita kegiatan suatu program studi secara berkala dipublikasikan melalui website program studi, misalnya kegiatan seminar, workshop/lokakarya, kuliah tamu, dan kegiatan Tridarma lainnya yang dilaksanakan oleh dosen, mahasiswa, dan civitas akademik. Dalam proses pencarian dokumen atau informasi yang dibutuhkan oleh pengguna dibutuhkan suatu sistem yang mampu menemukembalikan informasi secara relevan. Hal ini diperlukan suatu algoritma yang dirancang untuk dapat menemukan informasi dari dalam dokumen yang relevan dengan kata kunci (*keyword*) yang dimasukkan oleh pengguna ke dalam sistem.

Penelitian tentang temu kembali informasi telah dilakukan oleh beberapa peneliti. Arif Amrulloh, dkk, meneliti dengan membangun sebuah sistem pencarian similaritas judul tugas akhir menggunakan metode TF-IDF untuk memberikan solusi bagi dosen dan mahasiswa dalam menyusun judul tugas akhir. Cara kerja dari sistem yang dibangun yakni user menginputkan judul tugas akhir, lalu akan dipraproses mengikuti tahapan *case folding, tokenizing, stopword* atau *filtering*, pemeriksaan kata bantu (kata konkrit dan kata abstrak), pencocokan kata kunci dengan judul tugas akhir, dan pemeringkatan kata kunci. Hasil uji coba dari penelitian ini ditemukan kecocokan antara kata kunci sebanyak 99 judul tugas akhir dari 384 judul tugas akhir yang ada. Kekurangan pada sistem yang telah dikembangkan ini yaitu sistem belum bisa mendeteksi kesalahan ejaan kata kunci dan belum mampu mendeteksi dua kata yang sebenarnya masih satu kesatuan [3].

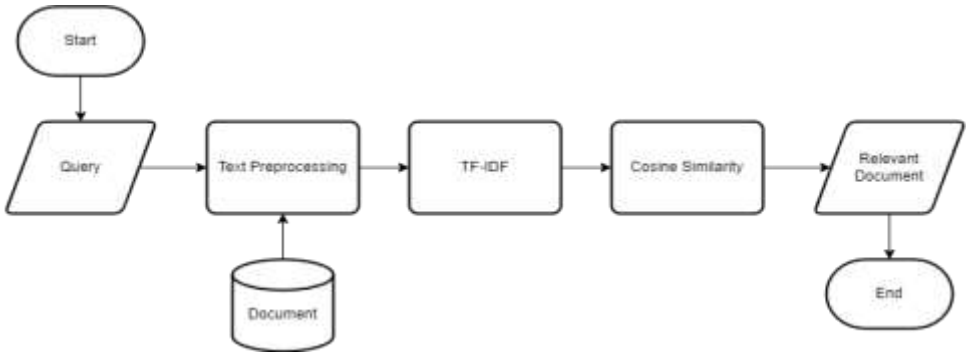
Nuzul hikmah juga meneliti tentang pemanfaatan text mining dalam pencarian ayat Al-Quran menggunakan metode TF-IDF dan Cosine Similarity yang nantinya akan memudahkan umat muslim dalam mencari informasi yang terkandung di dalam dokumen teks terjemahan Al-Quran. Cara kerja dari sistem yakni dengan memasukkan teks berupa topik yang dicari ke dalam sistem, lalu diproses menggunakan *text normalizer,* tokenisasi, *stopword removal, stemming, term documen matrix,* pencocokan vektor *query* dan dokumen teks terjemahan ayat Al-Quran menggunakan Cosine Similarity hingga menghasilkan ranking ayat Al-Quran yang relevan dan sesuai dengan yang diminta oleh user. Hasil dari penelitiannya diharapkan dapat bermanfaat bagi umat muslim yang ingin mencari topik tertentu dan menghasilkan temu kembali informasi teks terjemahan Al-Quran yang dicari [4]

Pada penelitian yang lain, mekanisme pencarian judul skripsi dengan metode Term-Frequency-Inverse Document Frequency juga diteliti oleh Meiga Ayu Ariyanti Nur Fitroh, dkk. Sistem yang dikembangkan bertujuan untuk membantu mahasiswa dalam menentukan judul skripsi dan tugas akhir serta meminimalisir pengajuan judul yang memiliki kemiripan secara morfologi kata, tetapi hanya berbeda lokasi saja. Sistem yang dikembangkan bernama SISINTA yakni kependekan dari Sistem Informasi Skripsi dan Tugas Akhir. Pengujian dari sistem yang dikembangkan menggunakan metode *white-box* dan *black-box.* Pengujian *white-box* menghasilkan rata-rata persentase sebesar 92% dengan metrik akurasi, presisi, dan sensitifitas, sedangkan pengujian *black-box* (divalidasi oleh ahli sistem) menghasilkan rata–rata sebesar 100% sehingga metode yang diusulkan sangat valid untuk mekanisme pencarian judul [5].

Seperti yang telah diuraikan di atas, beberapa peneliti telah mengimplementasikan temu kembali informasi untuk melakukan pencarian similaritas judul tugas akhir dan terjemahan ayat Al-Quran. Oleh karena itu, pada penelitian ini mengusulkan untuk merancang *protoype* sistem temu kembali informasi yang dapat melakukan pencarian konten berita kegiatan program studi. Dalam penelitian ini dilakukan beberapa tahapan dimulai dari pengumpulan data, praproses teks (*tokenizing*, *stopword removal*, *case folding*, dan *stemming*), pembobotan TF-IDF dan Cosine Similarity. Hasil yang diharapkan dari penelitian ini adalah sistem yang dibangun dapat menemukembalikan informasi yang sesuai antara kata kunci dengan konten berita yang tersedia.

2. METODOLOGI

Desain sistem pada penelitian ini ditunjukkan pada Gambar 1, di mana terdapat diagram alir dimulai dari input *query* (pencarian) dan data berita kegiatan program studi yang berupa data tabular yang disimpan pada database. Selanjutnya dilakukan praproses data terhadap *query* dan data konten berita kegiatan program studi. Setelah didapatkan data yang bersih dari hasil praproses data, selanjutnya dilakukan perhitungan pembobotan TF-IDF dan Cosine Similarity. Terakhir, hasil dari perhitungan Cosine Similarity setiap dokumen diurutkan berdasarkan nilai yang terbesar untuk didapatkan dokumen yang relevan dengan *query*.



Gambar 1. Desain Sistem pada Penelitian yang Diusulkan

2.1 Pengumpulan Data

Tahap pertama yang dilakukan yaitu pengumpulan data konten berita kegiatan program studi yang didapatkan melalui website Program Studi Sains Data Universitas Pembangunan Nasional "Veteran" Jawa Timur yang dapat diakses di http://sada.upnjatim.ac.id/category/berita/. Terdapat sebanyak 22 publikasi berita kegiatan program studi meliput webinar, workshop/lokakarya, seminar nasional, kuliah tamu, sosialisasi, dan kegiatan pengabdian kepada masyarakat. Pada *prototype* sistem yang dikembangkan, judul dan isi berita diinputkan ke dalam database sistem sebagai input data dan agar dapat dilanjutkan ke proses berikutnya.

2.2 Praproses Teks

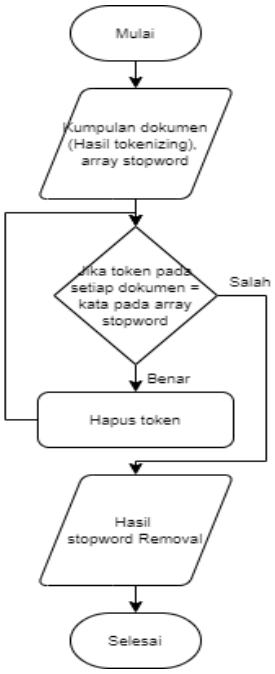
Praproses teks merupakan serangkaian proses yang diimplementasikan terhadap data tidak terstruktur dalam bentuk teks dengan tujuan untuk mendapat teks yang lebih bersih dan dapat dibaca oleh komputer. Pada tahap ini dilakukan restrukturisasi teks dengan menghilangkan karakter selain huruf dan juga mengubah huruf pada teks menjadi huruf kecil yang sebelumnya dalam bentuk huruf kapital (*case folding*), memisahkan teks menjadi setiap kata (tokenisasi), menghilangkan imbuhan pada setiap kata (*stemming*), dan melakukan proses penghilangan kata yang tidak relevan pada dokumen (*stopward*) [4]. Praproses data pada *query* dan isi dari dokumen dengan tahapan *tokenizing*, *stopword removal*, *case folding*, dan *stemming* dapat dijelaskan sebagai berikut:

a. Praproses Teks

Tokenizing atau tokenisasi merupakan proses untuk pemotongan *string* berdasarkan setiap kata yang tersusun menjadi beberapa kalimat [6]. Pada tahap ini menggunakan bantuan pustaka NLTK (*Natural Language Toolkit*) dengan *module* Regex Tokenizer dengan dengan parameter (\w+). Dalam Regex (*Regular expression*), \w+ akan mencocokkan pola kata pada suatu string yang setiap karakternya ekuivalen dengan a-z, A-Z, 0-9, dan simbol *underscore* (_) [7]. Artinya, setiap kata yang ditokenisasi hanya akan berupa *alphanumeric* [A-Za-z0-9_] dan *underscore*, dan akan langsung menghilangkan tanda baca atau karakter lainnya. Pada tahap ini juga dilakukan pengubahan huruf kapital menjadi huruf kecil dengan menggunakan method lower() agar dapat dilanjutkan pada tahap stopword removal. Tahapan dapat dilihat pada gambar 2.

Gambar 2. Tahapan *Tokenizing*

b. Stopword Removal



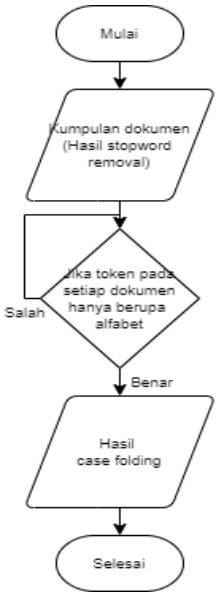
Gambar 3. Tahapan *Stopword Removal*

Stopword removal merupakan proses penghilangan kata-kata yang tidak memberikan informasi penting yang termuat dalam daftar *stopword* [4]. Pada tahap ini menggunakan pustaka PySastrawi dengan *module* StopwordRemoverFactory. Pada tahap ini dilakukan ekstraksi kata yang termasuk ke dalam daftar kata *stopword* dan dilakukan pencocokan antara kata pada isi dokumen dengan daftar kata *stopword*. Jika pada isi dokumen terdapat kata yang sama dengan daftar kata *stopword*, kata tersebut akan difilter dari daftar kata pada dokumen.

c. Case Folding

Case Folding merupakan cara memproses karakter string pada data teks menjadi seragam. Pada tahap ini dilakukan proses pengubahan seluruh huruf alfabet kapital (A-Z) menjadi huruf kecil dan karakter selain huruf alfabet (tanda baca dan angka) akan dihilangkan dari data teks [6]. Pada tahap ini digunakan *method* lower() untuk mengubah semua huruf besar (kapital) menjadi

huruf kecil dan *method* isalpha() untuk menghapus karakter selain huruf ‘a-z’ pada string [8]. Namun, karena pada tahap *stopword removal* dokumen sudah harus berupa huruf kecil, maka *method* lower() diaplikasikan pada tahap *tokenizing*, sedangkan isalpha() merupakan *method* yang mengembalikan nilai *True* jika pada string yang dikenakan *method* tersebut hanya terdiri dari karakter ‘A-Z’ atau ‘a-z’.



Gambar 4. Tahapan *Case Folding*

d. Stemming

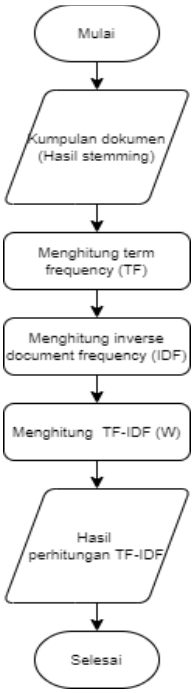
Gambar 5. Tahapan *Stemming*

Stemming merupakan proses untuk memperoleh suatu kata dasar dari suatu kata yang terdapat dalam kalimat dengan cara memisahkan masing-masing kata dari kata dasar dan imbuhannya baik awalan (*prefix*) maupun akhiran (*suffix*) [9]. Pada tahap ini digunakan pustaka PySastrawi dengan *module* StemmerFactory. Setiap kata yang melalui proses ini akan berubah menjadi kata dasar tanpa memerhatikan kaidah kebahasaan yang benar, berbeda dengan *lemmatization* yang akan mengubah suatu kata menjadi kata dasar sesuai dengan kaidah kebahasaan yang benar.

2.3 Pembobotan TF-IDF

TF-IDF merupakan suatu metode pemberian bobot hubungan suatu kata terhadap suatu dokumen. Pada tahap ini dilakukan metode penggabungan dua konsep untuk menghitung bobot yaitu frekuensi kemunculan kata dalam sebuah dokumen tertentu (*Term Frequency*) dan *inverse* frekuensi dokumen yang mengandung kata-kata tersebut (*Inverse Document Frequency*) [10].

Metode ini bertujuan untuk mengubah dokumen teks menjadi frekuensi dalam bentuk numerik dengan menentukan hubungan masing-masing kata pada suatu dokumen berdasarkan bobot [4].



Gambar 6. Tahapan Pembobotan TF-IDF

Setelah tahap praproses data dan didapatkan data teks yang bersih, tahap selanjutnya yaitu dilakukan pembobotan kata dengan menggunakan fungsi yang telah dibuat sesuai dengan rumus pembobotan kata (TF-IDF) dengan bantuan modul log10 dari pustaka Math yang digunakan pada perhitungan *Inverse Document Frequency* (IDF). Pada tahap ini, dihitung kata yang muncul pada *query* dan setiap dokumen, sesuai dengan rumus:

$$tf_{ij} = f_{ij} \quad (1)$$

Keterangan:
tf_{ij} : frekuensi *term i* pada dokumen *j*
f_{ij} : kemunculan *term i* pada dokumen *j*

Nilai IDF didapatkan melalui perhitungan dengan menggunakan rumus:

$$idf_i = log\left(\frac{N}{df_i}\right) \quad (2)$$

Keterangan:
idf_i : *inverse document frequency* pada *term i*
N : jumlah keseluruhan dokumen
df_i : jumlah kemunculan *term i* pada keseluruhan dokumen

Nilai bobot kata didapatkan melalui perhitungan dengan rumus:

$$W_{ij} = tf_{ij} \times idf_i \quad (3)$$

Keterangan:
W_{ij} : bobot *term i* dalam dokumen *j*
tf_{ij} : frekuensi *term* i dalam dokumen *j*
idf_i : *inverse document frequency* pada *term i*

2.4 Cosine Similarity

Gambar 7. Tahapan Cosine Similarity

Cosine Similarity merupakan suatu metode yang digunakan untuk mengukur tingkat kemiripan antar teks berdasarkan dua vektor. Metode pengukuran kemiripan teks ini termasuk yang paling populer diantara metode lainnya. Pengukuran ini dapat memberikan peringkat suatu dokumen sesuai dengan skor kemiripan yang dihasilkan [4]. Setelah dilakukan perhitungan bobot kata, dilakukan perhitungan tingkat kemiripan antara *query* yang diinputkan oleh pengguna dengan dokumen yang ada pada database menggunakan fungsi yang telah dibuat sesuai dengan rumus Cosine Similarity dengan bantuan modul sqrt dari pustaka Math.

$$Simillarity = cos(\theta) = \frac{A \cdot B}{||A||\,||B||} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}} \quad (4)$$

Keterangan:

$A \cdot B$: *dot product* antara vektor A dan B
$||A||\,||B||$: *cross product* antara $||A||$ dan $||B||$
$||A||$: panjang vektor A
$||B||$: panjang vektor B
A_i : bobot *query* pada *term* i (WQ_i)
B_i : bobot dokumen pada *term* i dan dokumen j (WD_{ij})
i : jumlah *term* dalam kalimat
n : jumlah vektor

3. HASIL DAN PEMBAHASAN

Dokumen teks konten berita kegiatan program studi yang digunakan dalam penelitian ini diperoleh melalui website resmi Program Studi Sains Data Universitas Pembangunan Nasional "Veteran" Jawa Timur sebanyak 22 konten berita. Pada tahap pengujian dilakukan pencarian terhadap konten berita kegiatan program studi dengan menggunakan kata kunci "industri". Hasil dari pencarian menggunakan kata kunci tersebut setelah melalui tahap praproses data diperoleh

perhitungan bobot kata dan Cosine Similarity sebagai berikut:

Tabel 1. Hasil Perhitungan Bobot Kata menggunakan TF-IDF

Terms	TF						DF	IDF	W					
	Q	**D1**	**D2**	**…**	**D21**	**D22**			**Q**	**D1**	**D2**	**…**	**D21**	**D22**
abdi	0	0	0	…	0	0	1	1.342	0	0	0	…	0	0
absensi	0	0	0	…	1	0	1	1.342	0	0	0	…	1.342	0
acara	0	0	0	…	0	0	9	0.388	0	0	0	…	0	0
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
industri	**1**	**0**	**1**	**…**	**0**	**0**	**8**	**0.439**	**0.439**	**0**	**0.439**	**…**	**0**	**0**
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
youtube	0	1	1	…	0	0	13	0.228	0	0.228	0.228	…	0	0
yuhana	0	0	1	…	0	0	2	1.041	0	0	1.041	…	0	0
zoom	0	1	1	…	1	1	16	0.138	0	0.138	0.138	…	0.138	0.318

Pada Tabel 1 merupakan hasil perhitungan pembobotan kata terhadap *query* "industri" dan 22 konten berita kegiatan program studi, terlihat hasil frekuensi dan bobot tiap *terms* baik pada *query* maupun dokumen.

Tabel 2. Hasil Perhitungan Cosine Similarity

Terms	W^2						$WQ \times WD$				
	Q	**D1**	**D2**	**…**	**D21**	**D22**	**D1**	**D2**	**…**	**D21**	**D22**
abdi	0	0	0	…	0	0	0	0	…	0	0
absensi	0	0	0	…	1.802	0	0	0	…	0	0
acara	0	0	0	…	0	0	0	0	…	0	0
…	…	…	…	…	…	…	…	…	…	…	…
industri	**0.193**	**0**	**0.193**	**…**	**0**	**0**	**0**	**0.193**	**…**	**0**	**0**
…	…	…	…	…	…	…	…	…	…	…	…
youtube	0	0.052	0.052	…	0	0	0	0	…	0	0
yuhana	0	0	1.084	…	0	0	0	0	…	0	0
zoom	0	0.019	0.019	…	0.019	0.019	0	0	…	0	0
Total	0.193	66.652	75.944	…	191.20	181.25	0	0.193	…	0	0
$\sqrt{Total}$	0.439	8.164	8.715	…	13.823	13.463					

Tabel 2 merupakan hasil perhitungan beberapa variabel yang terdapat pada rumus Cosine Similarity, seperti $WQ_i^2 = A_i$, $WD_{ij}^2 = B_i$, $WQ_i \times WD_i = A_i \times B_i$, dan $\sqrt{Total}$. Setelah didapatkan nilai dari masing-masing variabel, selanjutnya dilakukan perhitungan Cosine Similarity dengan menggunakan nilai-nilai variabel tersebut sesuai dengan rumus sebagai berikut:

$$D1 = \frac{\sum_{i=1}^{n} Q_i \times D1_i}{\sqrt{\sum_{i=1}^{n} Q_i^2} \times \sqrt{\sum_{i=1}^{n} D1_i^2}} = \frac{0}{0.439 \times 8.164} = 0 \quad (5)$$

$$D2 = \frac{\sum_{i=1}^{n} Q_i \times D2_i}{\sqrt{\sum_{i=1}^{n} Q_i^2} \times \sqrt{\sum_{i=1}^{n} D2_i^2}} = \frac{0.193}{0.439 \times 8.715} = 0.050 \quad (6)$$

…

$$D21 = \frac{\sum_{i=1}^{n} Q_i \times D21_i}{\sqrt{\sum_{i=1}^{n} Q_i^2} \times \sqrt{\sum_{i=1}^{n} D21_i^2}} = \frac{0}{0.439 \times 13.823} = 0 \quad (7)$$

$$D22 = \frac{\sum_{i=1}^{n} Q_i \times D22_i}{\sqrt{\sum_{i=1}^{n} Q_i^2} \times \sqrt{\sum_{i=1}^{n} D22_i^2}} = \frac{0}{0.439 \times 13.463} = 0 \quad (8)$$

Berdasarkan hasil perhitungan Cosine Similarity di atas, terdapat 8 dokumen (berita) yang memiliki skor Cosine Similarity > 0 atau merupakan dokumen (berita) yang relevan dengan *query* atau pencarian. Setelah mendapatkan skor hasil perhitungan Cosine Similarity dan dokumen (berita) yang relevan dengan *query* atau pencarian, dokumen-dokumen tersebut diurutkan berdasarkan skor Cosine Similarity yang terbesar ke yang terkecil.

a. Peringkat Hasil Pencarian Konten Berita

Pada Tabel 3 ditunjukkan peringkat hasil pencarian konten berita berdasarkan skor Cosine Similarity yang telah diurutkan mulai dari yang terkecil hingga yang terbesar.

Tabel 3. Peringkat Hasil Pencarian Konten Berita Kegiatan Program Studi Berdasarkan Skor Cosine Similarity

No.	Judul Berita	Skor Cosine Similarity
1.	iNTERVAL #3: Perspektif Sains Data di Industri	0.107
2.	Kuliah Tamu : "Pengenalan Sains Data Spasial dan Peran Sains Data Dalam Dunia Kerja"	0.096
3.	5 Dosen Sains Data Menjadi DPL KKN Tematik 2021	0.059
4.	iNTERVAL #2 : Strategi Penyusunan Kurikulum dan RPS	0.050
5.	Kuliah Tamu: "Teknologi Internet of Things (IoT)"	0.046
6.	Pelatihan SPPS dan Fundamental R	0.036
7.	Pembelajaran IoT dengan Modul ESP32 di SMKN 1 Dlanggu Mojokerto	0.024
8.	Survey Lokasi KKN Tematik UPN "Veteran" Jawa Timur di Daerah Gresik	0.018

b. Evaluasi Hasil Pencarian Konten Berita

Pada Tabel 4 ditunjukkan evaluasi hasil percobaan yang dilakukan dengan 10 kali pencarian menggunakan kata kunci yang berbeda dan kesesuaian hasil pencarian dengan kata kunci yang digunakan. Berdasarkan evaluasi menunjukkan bahwa 10 percobaan *query* menggunakan kata kunci yang berbeda mendapatkan hasil 100% kesesuaian dengan dokumen berita kegiatan Program Studi yang dicari.

Tabel 4. Evaluasi Hasil Temu Kembali Informasi Pencarian Konten Berita Kegiatan Program Studi

No.	Hasil Pencarian Berita (Peringkat 1)	Kata kunci	Skor	Keterangan
1.	iNTERVAL #3: Perspektif Sains Data di Industri	Industri	0.107	Sesuai
2.	Kuliah Tamu: "Teknologi Internet of Things (IoT)"	Kuliah tamu IoT	0.640	Sesuai
3.	5 Dosen Sains Data Menjadi DPL KKN Tematik 2021	Dosen sains data	0.067	Sesuai
4.	Pelatihan Penyusunan Kurikulum Merdeka Belajar Kampus Merdekan (MBKM) dengan Dunia Industri	Kurikulum	0.327	Sesuai
5.	Seminar Sains Data - Seri Bela Negara	Bela negara	0.492	Sesuai
6.	Kuliah Tamu: "Paten Bidang Sains Data dan Teknologi Informasi"	Teknologi informasi	0.114	Sesuai
7.	Pelatihan Penulisan Publikasi Pengelolahan Jurnal Terakreditrasi	Pelatihan penulisan	0.222	Sesuai
8.	iNTERVAL Seri #4 : Kontribusi Sains Data dalam Pencegahan dan Penyebaran COVID-19 di Tempat Kerja	Covid-19	0.215	Sesuai
9.	Kunjungan Kerja : "Sharing Ilmu Dalam Rangka Pengembangan Aplikasi SITUK dan SiOBEL"	Pengembangan aplikasi	0.309	Sesuai
10.	Survey Lokasi KKN Tematik UPN "Veteran" Jawa Timur di Daerah Gresik	KKN	0.636	Sesuai
	Persentase Kecocokan			100% Sesuai

4. KESIMPULAN DAN SARAN

Pencarian konten berita kegiatan program studi dengan menggunakan algoritma pembobotan TF-IDF dan Cosine Similarity memberikan hasil pencarian yang akurat. Sistem yang dikembangkan mampu memberikan hasil berupa pencarian konten berita yang relevan dengan kata kunci yang dimasukkan dengan melalui serangkaian tahapan *text preprocessing* berupa *tokenizing*, *stopword removal*, *case folding* dan *stemming* hingga perhitungan bobot TF-IDF dan Cosine Similarity. Konten berita kegiatan program studi yang digunakan dalam percobaan ini diperoleh dari website resimi Program Studi Sains Data Universitas Pembangunan Nasional "Veteran" Jawa Timur sebanyak 22 konten berita. Hasil percobaan menunjukkan bahwa 10 percobaan *query* menggunakan kata kunci yang berbeda mendapatkan hasil 100% kesesuaian dengan dokumen berita kegiatan Program Studi yang dicari.

5. DAFTAR RUJUKAN

[1] A. A. Maarif, "Penerapan Algoritma TF-IDF Untuk Pencarian Karya Ilmiah," *eprints UDiNus Repository,* pp. 1-4, 2015.

[2] M. N. Saadah, R. W. Atmagi, D. S. Rahayu dan A. Z. Arifin, "Sistem Temu Kembali Dokumen Teks dengan Pembobotan TF-IDF dan LCS," *Jurnal Ilmiah Teknologi Informasi (JUTI),* vol. 11, no. 1, pp. 17-20, 2013.

[3] A. Amrulloh dan I. F. Adam, "Sistem Pencarian Similaritas Judul Tugas Akhir Menggunakan Metode TF-IDF," *CoreIT: Jurnal Hasil Penelitian Ilmu Komputer dan Teknologi Informasi,* vol. 7, no. 2, pp. 74-82, 2021.

[4] N. Hikmah, "Pemanfaatan Text Mining dalam Pencarian Ayat AlQuran menggunakan TF-IDF dan Cosine Similarity," *Jurnal Antartika,* vol. 8, no. 1, pp. 13-22, 2018.

[5] M. A. Ariyanti, A. P. Wibawa and U. Pujianto, "Metode Term Frequency - Invers Document Frequency pada Mekanisme Pencarian Judul Skripsi," *Jurnal Teknologi, Elektro, dan Kejuruan,* vol. 28, no. 2, pp. 177-190, 2018.

[6] F. S. Jumeilah, "Penerapan Support Vector Machine (SVM) untuk Pengkategorian Penelitian," *Jurnal RESTI,* vol. 1, no. 1, pp. 19-25, 2017.

[7] G. Skinner, "RegExr," Gskinner, 1 June 2020. [Online]. Available: https://regexr.com/. [Diakses 15 July 2022].

[8] Python Software Foundation, "Python 3.10.5 documentation," 30 July 2022. [Online]. Available: https://docs.python.org/. [Accessed 15 July 2022].

[9] D. Wahyudi, T. Susyanto dan D. Nugroho, "Implementasi dan Analisis Algoritma Stemming Nazief & Adriani dan Porter pada Dokumen Berbahasa Indonesia," *Jurnal Ilmiah SINUS,* vol. 15, no. 2, pp. 49-56, 2017.

[10] K. D. Putung, A. Lumenta dan A. Jacobus, "Penerapan Sistem Temu Kembali Informasi pada Kumpulan Dokumen Skripsi," *Jurnal Teknik Informatika,* vol. 8, no. 1, pp. 18-23, 2016.