Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI: 10.25126/jtiik...
Vol. x, No. x, April 2014, hlm. x-x p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 36/E/KPT/2019 e-ISSN: 2528-6579

SISTEM REKOMENDASI TOPIK PENELITIAN MENGGUNAKAN METODE CONTENT-BASED FILTERING (STUDI KASUS PROGRAM STUDI TEKNIK INFORMATIKA)

Dafa Rozzi Pratama[*1], Gibran Satya Nugraha[2], Ramaditia Dwiyansaputra [3]

[1,2,3]Universitas Mataram, Mataram
Email: [1]dafarozzipratama@gmail.com, [2]gibransn@unram.ac.id, [3]rama@unram.ac.id
[*]Penulis Korespondensi

(Naskah masuk: dd mmm yyyy, diterima untuk diterbitkan: dd mmm yyyy)

Abstrak

Dalam dunia perkuliahan, topik penelitian sering digunakan dalam pembuatan tugas akhir mahasiswa. Tugas akhir merupakan syarat wajib yang dilakukan mahasiswa untuk mendapatkan gelar sarjana. Namun, permasalahan dalam memilih topik penelitian seringkali menjadi hal yang membingungkan bagi mahasiswa. Permasalahan ini dapat diselesaikan dengan sebuah sistem yang mampu memberikan rekomendasi. Metode *content-based filtering* digunakan pada penelitian ini. Terdapat beberapa tahapan penelitian yang dilakukan, dimulai dari studi literatur, pengumpulan data, *preprocessing text*, menghitung bobot kata, perancangan sistem dan terakhir pengujian sistem. Sistem Rekomendasi Topik Penelitian ini diimplementasikan dalam bentuk *Graphical User Interface* (GUI) pada sebuah *website* dengan menggunakan *framework* Flask dan *database* MySQL. Penelitian ini bertujuan untuk mengimplementasikan sistem rekomendasi topik penelitian menggunakan metode *content-based filtering* dengan menggunakan TF-IDF dan *cosine similarity* berbasis *website*, serta mengetahui keakuratan content-based filtering dalam memberikan rekomendasi topik penelitian. Sistem rekomendasi topik penelitian dengan metode *content-based filtering* pada penelitian ini memiliki nilai *cosine similarity* 0,1658 pada proses *stemming* dan 0,1732 pada proses *non-stemming*. Faktor yang dapat mempengaruhi nilai skor *cosine similarity* tersebut yaitu seperti kata kunci, *dataset*, dan proses *preprocessing data* yang dilakukan. Semakin cocok kata kunci dengan *database*, maka semakin tinggi *cosine similarity*-nya. Jika kata kunci tak ada di *database*, nilai *cosine similarity* tersebut mengalami penurunan.

Kata kunci: *Sistem Rekomendasi, Content-Based Filtering, TF-IDF, Cosine Similarity, Topik Penelitian, Flask, MySQL.*

RESEARCH TOPIC RECOMMENDATION SYSTEM USING CONTENT-BASED FILTERING METHOD (CASE STUDY OF INFORMATICS ENGINEERING STUDY PROGRAM)

Abstract

In the world of lectures, research topics are often used in students' final assignments. The final assignment is a mandatory requirement for students to obtain a bachelor's degree. However, the problem of choosing a research topic is often confusing for students. This problem can be solved with a system that is able to provide recommendations. The content-based filtering method was used in this research. There are several stages of research carried out, starting from literature study, data collection, text preprocessing, calculating word weights, system design and finally system testing. This Research Topic Recommendation System is implemented in the form of a Graphical User Interface (GUI) on a website using the Flask framework and MySQL databases. This research aims to implement a research topic recommendation system using the content-based filtering method using TF-IDF and website-based cosine similarity, as well as knowing the accuracy of content-based filtering in providing research topic recommendations. Research topic recommendation system using the content-based filtering method in research This has a cosine similarity value of 0.1658 in the stemming process and 0.1732 in the non-stemming process. Factors that can influence the value of the cosine similarity score include keywords, dataset, and the data preprocessing process carried out. The more the keyword matches the database, the higher the cosine similarity. If the keyword does not exist in the database, the cosine similarity value decreases.

Keywords: *Recommendation Systems, Content-Based Filtering, TF-IDF, Cosine Similarity, Research Topics, Flask, MySQL.*

1. PENDAHULUAN

Menurut Kamus Besar Bahasa Indonesia (KBBI), topik merupakan pokok pembicaraan dalam diskusi, ceramah, karangan dan sebagainya. Sedangkan penelitian merupakan kegiatan pengumpulan, pengolahan, analisis dan penyajian data yang dilakukan sistematis dan objektif untuk memecahkan suatu persoalan atau menguji suatu hipotesis untuk mengembangkan prinsip-prinsip. Sehingga jika digabungkan topik penelitian merupakan suatu pokok pembicaraan yang dibuat atau dibahas dalam kegiatan pengumpulan, pengolahan, analisis dan penyajian data yang dilakukan sistematis dan objektif untuk memecahkan suatu persoalan atau menguji suatu hipotesis.

Dalam dunia perkuliahan, topik penelitian sering digunakan dalam pembuatan tugas akhir mahasiswa. Tugas akhir merupakan syarat wajib yang dilakukan mahasiswa untuk mendapatkan gelar sarjana. Namun, permasalahan dalam memilih topik penelitian seringkali menjadi hal yang membingungkan bagi mahasiswa, seperti kurangnya referensi yang didapatkan, kurangnya pengetahuan mengenai topik penelitian yang akan diambil, banyaknya pilihan dalam memilih topik penelitian dan lain-lain. Pada hakikatnya, banyak mahasiswa yang masih kesulitan dalam menentukan topik penelitian tugas akhir. Berdasarkan survei yang dilakukan pada tanggal 3 November 2022 sampai 28 Februari 2023 oleh penulis terhadap 48 responden mahasiswa PSTI angkatan 2015-2020 diperoleh sekitar 10,4% mahasiswa PSTI tidak mengalami kesulitan memilih judul tugas akhir, 22,9% mahasiswa PSTI sedikit mengalami kesulitan memilih judul tugas akhir dan 66,6% mahasiswa PSTI yang mengalami kesulitan memilih judul tugas akhir. Kesulitan dalam menentukan topik tugas akhir tersebut dikarenakan kurangnya referensi yang didapatkan, kurangnya konsep dan ruang lingkup cakupan terkait judul terkait studi kasus. Alasan lainnya, mahasiswa mengalami kebingungan dalam memilih masalah yang ingin diangkat, sehingga kesulitan untuk menemukan ide penelitian yang menarik dan orisinal.

Salah satu upaya untuk mencari dan menemukan ide atau topik penelitian adalah dengan cara mencari referensi jurnal atau artikel di internet seperti di Google Scholar, Scopus, IEEE, dan lain-lain. Dengan banyaknya referensi jurnal atau artikel memudahkan mahasiswa dalam mencari topik atau ide penelitian. Namun, banyaknya referensi jurnal atau artikel yang tersebar luas membuat mahasiswa (khususnya mahasiswa PSTI) mengalami kesulitan mencari informasi keterbaharuan judul yang sudah diambil atau dipilih. Permasalahan ini dapat diselesaikan dengan sebuah sistem yang mampu memberikan rekomendasi.

Sistem rekomendasi adalah perangkat lunak dan teknik yang menyediakan saran mengenai item tertentu untuk digunakan oleh *user* (Badriyah et al., 2017). Sistem ini digunakan dalam berbagai contoh seperti rekomendasi musik, rekomendasi film, rekomendasi buku dan lain-lain. Sistem rekomendasi menggunakan teknik analisis data dan kecerdasan buatan untuk memproses data *user* dan item yang tersedia dan kemudian menghasilkan rekomendasi yang paling sesuai dengan keinginan *user*, dengan begitu sistem dapat menentukan item yang mungkin akan disukai oleh *user*. Ada beberapa sistem rekomendasi yang digunakan untuk memberikan rekomendasi kepada *user*, yaitu *content-based filtering, collaborative filtering, knowledge-based filtering*.

Content-based filtering mempertimbangkan preferensi dan minat individu dari *user* untuk memberikan rekomendasi yang relevan. Metode ini menggunakan fitur dan atribut *item* untuk mencocokkan dengan keinginan *user*. Dengan demikian, rekomendasi yang diberikan akan lebih disesuaikan dengan kebutuhan dan minat *user*. *Content-based filtering* memiliki perbedaan dengan *collaborative filtering*, yang dimana *collaborative filtering* didasarkan pada kesamaan kemiripan antar *user* (Gantini, 2016). Sedangkan metode *knowledge-based filtering* bergantung pada dua jenis data, yaitu data kumpulan aturan (batasan) atau metrik yang sama dan kumpulan *item* (Novandra & Heryanto, 2021).

Dari perbedaan beberapa jenis sistem rekomendasi di paragraf sebelumnya, metode *content-based filtering* dapat digunakan untuk membuat sistem rekomendasi topik penelitian. Metode *content-based filtering* lebih tepat digunakan untuk merekomendasikan topik penelitian sesuai dengan keinginan dan aktivitas *user*. Berbeda dengan *collaborative filtering* dan *knowledge-based filtering*, *collaborative filtering* membutuhkan komunitas dengan cara mencari kemiripan antar *user*, sedangkan *knowledge-based filtering* sangat bergantung terhadap data. *knowledge-based filtering* memiliki keterbatasan dalam penanganan data yang belum diketahui, bila data yang masuk tidak sesuai dengan aturan, maka *knowledge-based filtering* mungkin tidak efektif dalam memberikan rekomendasi. Contoh dalam penelitian (Putra & Santika, 2020) dengan menggunakan data lagu sebanyak 11.737 serta 133.501 total riwayat lagu semua *user*, didapatkan bahwa sistem ini dapat berjalan baik pada *dataset* yang belum begitu banyak, namun akan bermasalah jika menggunakan *dataset* yang lebih besar, terutama pada *performance* dari sistem ini. Pada penelitian (Rizqi & Zayyad, 2021) menggunakan *dataset* sebanyak 5143 buku, didapatkan hasil dari pengujian sistem rekomendasi berbasis konten dengan menggunakan algoritma TF-

IDF dan *cosine similarity* menghasilkan rata-rata nilai *precision* sebesar 85%. Pada penelitian (Salim et al., n.d.) menggunakan lebih dari 5000 data film yang mengandung judul, tahun terbit, sinopsis, nama aktor, nama direktur, dan *genre*. Pada penelitian tersebut didapatkan bahwa dengan menggunakan metode *content-based filtering*, sangat mudah terhadap pengguna dalam mencari judul film yang mirip, walaupun film yang direkomendasikan sudah lampau atau baru dipublikasikan, serta penggunaan *dataset* yang diperhitungkan dapat diperbanyak variasi. Seperti penambahan tahun terbit, bahasa, dan *reviews* supaya nilai kemiripan antar film menjadi maksimal.

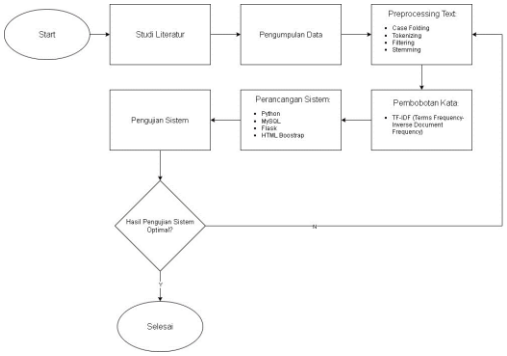
Sistem rekomendasi topik penelitian dengan metode *content-based filtering* dapat memanfaatkan metode *cosine similarity* dan metode TF-IDF (Term Frequency-Inverse Document Frequency). Metode *cosine similarity* adalah sebuah metode yang dapat mengukur kemiripan pada setiap topik penelitian dengan menggunakan perhitungan antar vektor (Rizqi & Zayyad, 2021). Metode *cosine similarity* ini sangat cocok digunakan pada dataset topik penelitian karena metode *cosine similarity* tersebut tidak terpengaruh oleh panjang dokumen. Metode *cosine similarity* hanya memperhatikan sudut antara vektor. Dengan begitu, metode ini sangat berguna ketika membandingkan dokumen yang memiliki jumlah kata yang berbeda.

Sedangkan metode TF-IDF adalah cara pemberian bobot hubungan suatu kata (term) terhadap dokumen (Amrizal, 2018). Metode TF-IDF tersebut sangat cocok digunakan dalam *text processing* karena bobot suatu kata dalam dokumen dihitung berdasarkan seberapa sering kata tersebut muncul dalam dokumen tersebut, serta seberapa umum kata tersebut di seluruh dokumen. Dengan begitu, kata-kata yang sering muncul dalam dokumen tetapi jarang muncul di dokumen lain akan memiliki bobot yang tinggi dan dianggap penting. Dalam metode *content-based filtering*, metode TF-IDF akan membobotkan kata pada item yang direkomendasikan. Bobot kata tersebut digunakan untuk menghitung kesamaan antara *item* yang direkomendasikan dengan *item* yang sudah pernah di akses oleh *user* sebelumnya dengan menggunakan metode *cosine similarity*. Dengan begitu, sistem dapat memperhitungkan bobot kata yang lebih relevan dan signifikan dalam atribut atau konten item yang akan di rekomendasikan. Dengan berbasis *web*, sistem rekomendasi topik penelitian ini dapat dengan mudah di akses oleh mahasiswa melalui *smartphone*, komputer, laptop, tablet dan lain lain.

Berdasarkan ulasan tersebut, penulis mengusulkan sebuah sistem rekomendasi topik penelitian menggunakan metode *content-based filtering* berbasis *web* untuk mempermudah khususnya mahasiswa PSTI dalam mendapatkan rekomendasi topik penelitian berdasarkan aktivitas yang mereka lakukan.

2. METODE PENELITIAN

Metode *content-based* filtering digunakan pada penelitian ini. Pada bab ini, terdapat beberapa tahapan penelitian yang dilakukan, dimulai dari studi literatur, pengumpulan data, *preprocessing text*, menghitung bobot kata, perancangan sistem dan terakhir pengujian sistem. Adapun rancangan penelitian yang akan dilakukan digambarkan dengan diagram alir pada Gambar 1.



Gambar 1. Alur Penelitian

2.1. Studi Literatur

Studi literatur dilakukan dengan mempelajari buku-buku, jurnal-jurnal penelitian sebelumnya serta sumber lain yang berkaitan dengan permasalahan yang diangkat pada penelitian ini. Adapun materi yang dipelajari dalam studi literatur berkaitan dengan content-based filtering serta materi lain yang berkaitan dengan penelitian yang dilakukan.

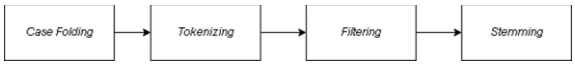
2.2. Pengumpulan Data

Data yang digunakan dalam sistem rekomendasi topik penelitian tersebut berjumlah 358 judul tugas akhir yang berasal dari transkrip alumni PSTI. Selanjutnya, dilakukan pencarian abstrak dari semua judul tugas akhir tersebut di open-source journal. Setelah abstrak berhasil didapatkan, dilakukan klasifikasi judul tugas akhir berdasarkan laboratorium penelitian yang tersedia di PSTI.

2.3. Preprocessing Text

Preprocessing adalah teknik yang dilakukan sebelum dilakukan analisis atau pembangunan model pada data tersebut. Tujuan dari preprocessing tersebut yaitu mempersiapkan data agar dapat diolah lebih efektif dan efisien. Preprocessing pada penelitian ini menggunakan Natural Language Processing (NLP). NLP adalah disiplin ilmu komputer yang bertujuan untuk memahami konsep dan maksud dari Bahasa manusia [31]. Selain itu, peneliti juga menggunakan Natural Language Toolkit (NLTK) yang merupakan rangkaian library dan program pengolahan bahasa simbolik dan statistic alami (NLP) yang ditulis dalam Bahasa pemrograman Python (Rifano et al., 2020). NLTK ini sangat mendukung proses pengolahan Bahasa natural seperti *case folding*, *tokenizing*,

filtering, *stemming* dan lain-lain. Alur pada *preprocessing text* ini dapat dilihat pada Gambar 2.



Gambar 2. Alur *Preprocessing Text*

Pada tahap *case folding* ini dilakukan untuk mengubah kata-kata atau kalimat judul penelitian dan abstrak menjadi huruf kecil. Selain mengubah kata-kata atau kalimat menjadi huruf kecil, tahap ini akan melakukan pembersihan kalimat dari angka, simbol dan tanda baca. Tujuan dilakukannya proses tersebut yaitu untuk memberikan kesamaan tiap kata pada dokumen data topik penelitian serta menghilangkan *noise* pada saat pengambilan informasi.

Selanjutnya pada tahap *tokenizing* ini dilakukan untuk memisahkan kalimat menjadi potongan-potongan berupa token, bisa berupa huruf, kata atau kalimat. Pada proses ini, data judul dan abstrak dari topik penelitian tersebut akan dilakukan pemotongan kata dengan tujuan untuk mengubah ukuran dokumen menjadi lebih kecil sehingga proses pencarian informasi dilakukan dengan lebih cepat. Proses *tokenizing* ini menggunakan *library* Python NLTK.

Kemudian tahap *Filtering* atau *stopword removal* adalah proses yang dilakukan untuk mengambil kata-kata penting dari hasil *tokenizing* tadi. Biasanya kata yang muncul dan tidak memiliki makna disebut *stopword*. Contohnya, penggunaan kata dan, yang, serta, di dan lain-lain. Tujuan dilakukannya *filtering* yaitu untuk mengurangi volume kata pada dokumen, sehingga proses pencarian informasi dapat dilakukan dengan lebih tepat.

Terakhir, tahap *Stemming* adalah tahapan yang diperlukan untuk memperkecil jumlah indeks yang berbeda sehingga sebuah kata yang memiliki *suffix* dan *prefix* akan kembali ke bentuk dasarnya. Proses *stemming* ini menggunakan *library* Python dari Sastrawi yang dapat mengubah kata menjadi bentuk dasar.

2.4. Pembobotan Kata

Pembobotan kata ini dilakukan setelah melakukan preprocessing data. Pembobotan kata tersebut menggunakan algoritma TF-IDF. Algoritma ini akan mengevaluasi seberapa penting sebuah kata di dalam sebuah dokumen atau dalam sekelompok kata. Pada pembobotan menggunakan algoritma TF-IDF, bobot akan semakin besar frekuensinya jika kemunculan kata semakin tinggi, sebaliknya bobot akan semakin berkurang bila kata tersebut semakin sering muncul pada topik penelitian yang lain. Persamaan Pembobotan TF-IDF dapat dilihat pada persamaan (1), (2) dan (3).

$$tf_{i,j} = f_{i,j} \quad (1)$$

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

$$Wdt = tf_{i,j} \times idf_i \quad (3)$$

Keterangan:
Wdi = bobot dokumen ke-*d* terhadap kata ke-*i*
$tfdi$ = banyaknya kata yang dicari pada sebuah dokumen
$Idfi$ = *Inversed Document Frequency* ($\log(N/df)$)
N = total dokumen
df = banyak dokumen yang mengandung kata yang dicari

2.5. Perhitungan Similarity

Pada perhitungan similarity ini akan menggunakan algoritma *cosine similarity*. Perhitungan menggunakan algoritma *cosine similarity* dilakukan untuk mengukur kemiripan antara dua vektor atau dua dokumen pada ruang vektor. Dengan menggunakan algortima ini, sistem dapat menentukan kemiripan antara satu dokumen dengan dokumen lainnya. Dalam melakukan perbandingan dokumen, data yang sudah dilakukan *preprocessing* tidak hanya besaran dari setiap pembobotan kata (TF-IDF) yang akan dibandingkan, tetapi juga sudut antar dokumen-dokumen. Dalam melakukan perhitungan *cosine similarity* dapat dilakukan dengan menggunakan persamaan (4).

$$cosSim(d_j, q_k) = \frac{\sum_{i=1}^{n}(td_{ij} \times tq_{ik})}{\sqrt{\sum_{i=1}^{n} td_{ij^2} \times \sum_{i=1}^{n} tq_{ik^2}}} \quad (4)$$

Keterangan:
$cosSim(d_j, q_k)$ = tingkat kesamaan dokumen dengan *query* tertentu
td_{ij} = term ke-*i* dalam vektor untuk dokumen ke-*j*
tq_{ik} = term ke-*i* dalam vektor untuk *query* ke-*k*
n = jumlah *term* yang unik dalam *dataset*

2.6. Perancangan Sistem

Perancangan sistem ini didasarkan pada alur penelitian sebelumnya. Perhitungan yang sudah dilakukan pada subbab sebelumnya akan diimplementasi menggunakan bahasa pemrograman Python dan *framework web* Flask. Salah satu perhitungan yang dilakukan yaitu *preprocessing data* (case folding, tokenizing, stemming, filtering) dan perhitungan pembobotan.

2.7. Pengujian Sistem

Pada proses skenario pengujian sistem ini akan dilakukan dengan cara menguji kata kunci yang dimasukan dan pengujian menggunakan dataset yang sudah dilakukan proses *stemming* dan *non-stemming*.

Dalam skenario pengujian menggunakan kata kunci ini akan menguji seberapa banyak dokumen relevan yang keluar dalam sistem rekomendasi.

Dalam pengujian ini, *user* akan memasukan 5 sampai 7 kata kunci dalam pengujian. Setelah dokumen relevan tersebut keluar, selanjutnya akan dilakukan pengukuran performansi menggunakan nilai *cosine similarity*.

Dalam skenario pengujian menggunakan *stemming* dan *non-stemming* ini akan menguji seberapa relevan dokumen yang di rekomendasikan bila menggunakan *stemming* atau *non-stemming*. Dalam pengujian ini menggunakan pengukuran performansi nilai *cosine similarity*.

3. HASIL DAN PEMBAHASAN

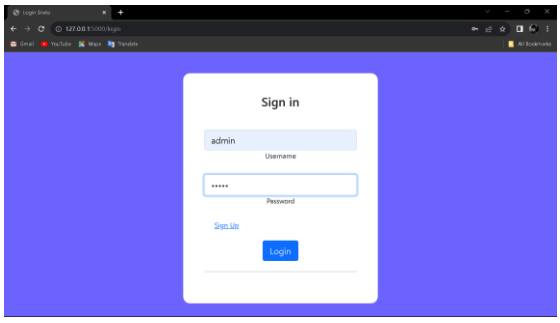
3.1. Pengumpulan Data

Data yang digunakan pada penelitian ini di dapatkan dari *open-source journal* seperti JTika, JCosine dan beberapa *open-source journal* lainnya. Data yang diambil pada penelitian ini terdapat 356 data topik penelitian dan variabel penting yang dibutuhkan untuk membuat sistem rekomendasi pada penelitian ini adalah judul, abstrak, penulis, lab dan mata kuliah terkait.

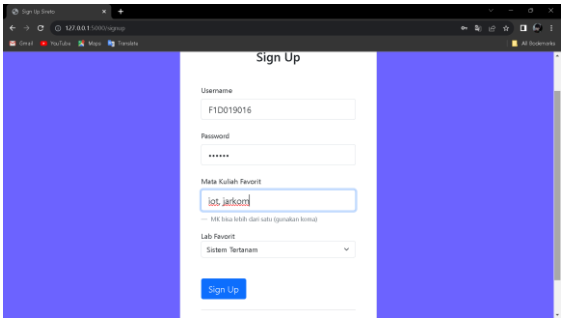
3.2. Implementasi GUI

Sistem Rekomendasi Topik Penelitian ini diimplementasikan dalam bentuk *Graphical User Interface* (GUI) pada sebuah *website* dengan menggunakan *framework* Flask dan *database* MySQL. Sedangkan untuk pembuatan HTML sistem rekomendasi tersebut dibuat dengan menggunakan Bootstrap 5.0, CSS, dan Javascript sehingga tampilan situs web tersebut terlihat lebih bagus dan *responsive*.

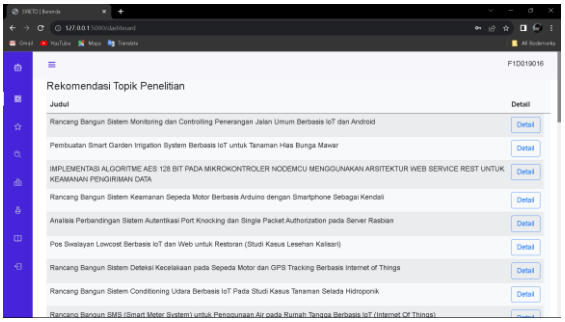
Gambar 3. Halaman Utama



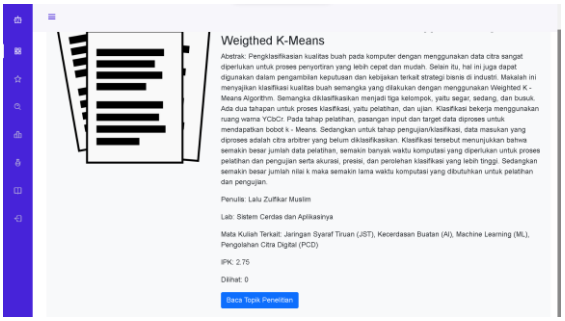
Gambar 4. Halaman *Login*



Gambar 5. Halaman *Sign Up*



Gambar 6. Halaman Beranda



Gambar 7. Halaman *Detail*

3.3. Hasil Penelitian

Pada penelitian ini bertujuan untuk mengimplementasikan sistem rekomendasi topik penelitian menggunakan metode *content-based filtering* dengan menggunakan TF-IDF dan *cosine similarity* berbasis *website*, serta mengetahui keakuratan *content-based filtering* dalam memberikan rekomendasi topik penelitian. Pembuatan sistem rekomendasi topik penelitian ini menggunakan data yang berasal dari transkrip alumni sebelumnya serta beberapa sumber dari *open-source journal*. Pengujian ini dilakukan menggunakan *cosine similarity* dengan menggunakan 356 sampel topik penelitian untuk mengukur seberapa mirip kata kunci yang dimasukkan dengan topik penelitian yang tersedia. Dimana dalam pengujian ini terdapat 10 user yang akan mencoba menguji sistem tersebut.

3.4. Analisis Pengujian Stemming Menggunakan Kata Kunci

Dalam skenario pengujian menggunakan kata kunci ini akan menguji seberapa banyak dokumen relevan yang keluar dalam sistem rekomendasi.

Pengujian ini dilakukan menggunakan 5 atau 7 kata kunci yang dimana pengujian ini dilakukan oleh 10 *user* yang dimana *user* tersebut merupakan mahasiswa PSTI. Tabel 1. merupakan hasil dari pengujian yang dilakukan terhadap 10 *user* tersebut.

Tabel 1. Pengujian *Stemming* Menggunakan 5-7 Kata Kunci Dengan 10 Hasil Dokumen

Kata Kunci	Banyak Kata Kunci	Nilai Cosine Similarity Tertinggi	Nilai Cosine Similarity Terendah	Rata-Rata
Sistem Penjadwalan Matakuliah Otomatis Berbasis Web Menggunakan	5 kata kunci	0.1729	0.0763	0.1173
	6 kata kunci	0.1596	0.1096	0.1333
	7 kata kunci	0.1596	0.1096	0.1333
penggunaan blockchain pada bidang militer di Indonesia	5 kata kunci	0.1059	0.0435	0.0585
	6 kata kunci	0.1059	0.0435	0.0585
	7 kata kunci	0.0966	0.0548	0.0692
pengembangan aplikasi e-commerce dengan fitur produk filtering	5 kata kunci	0.1613	0.0605	0.0929
	6 kata kunci	0.3642	0.0655	0.1208
	7 kata kunci	0.2883	0.0525	0.0974
perancangan sistem informasi manajemen pelanggan berbasis crm	5 kata kunci	0.1951	0.1462	0.1626
	6 kata kunci	0.1951	0.1462	0.1626
	7 kata kunci	0.1951	0.1462	0.1626
kinerja protokol routing DSDV DSR AOMDV AODV	5 kata kunci	0.5310	0.2078	0.3645
	6 kata kunci	0.5196	0.2990	0.3885
	7 kata kunci	0.4873	0.2828	0.3693
website sistem informasi akademik menggunakan clean arsitektur	5 kata kunci	0.5103	0.1138	0.2048
	6 kata kunci	0.3537	0.0788	0.1419
	7 kata kunci	0.3199	0.0975	0.1748
sistem keamanan pintu otomatis dengan sidik jari	5 kata kunci	0.3395	0.0604	0.1260
	6 kata kunci	0.3696	0.0731	0.1788
	7 kata kunci	0.6018	0.0622	0.2375
performa algoritma routing dalam jaringan ad-hoc terstruktur	5 kata kunci	0.2626	0.1533	0.1867
	6 kata kunci	0.3060	0.1909	0.2400
	7 kata kunci	0.3060	0.1909	0.2400
perbandingan citra rgb dan grayscale untuk klasifikasi	5 kata kunci	0.1736	0.0732	0.1090
	6 kata kunci	0.1736	0.0732	0.1090
	7 kata kunci	0.1813	0.1031	0.1370
kinerja machine learning dalam deteksi anomali jaringan	5 kata kunci	0.1842	0.1046	0.1331
	6 kata kunci	0.1842	0.1046	0.1331
	7 kata kunci	0.1842	0.1046	0.1331

Berdasarkan hasil pada Tabel 1 tersebut, nilai rata-rata *cosine similarity* pada percobaan menggunakan kata kunci tersebut cenderung meningkat ketika lebih banyak kata kunci dimasukkan. Hal ini disebabkan oleh pengaruh pembobotan TF-IDF pada setiap kata kunci terhadap perhitungan *cosine similarity*. Hal ini melibatkan perhitungan sejauh mana kesesuaian item yang ada dengan pembobotan TF-IDF yang telah diimplementasikan.

3.5. Analisis Pengujian Non-Stemming Menggunakan Kata Kunci

Dalam pengujian menggunakan *stemming* dan *non-stemming* ini akan menguji seberapa relevan dokumen yang di rekomendasikan bila menggunakan *stemming* atau *non-stemming*. Pengujian ini dilakukan menggunakan 5 atau 7 kata kunci yang dimana pengujian ini dilakukan oleh 10 *user* yang dimana *user* tersebut merupakan mahasiswa PSTI. Tabel 2. merupakan hasil dari pengujian yang dilakukan terhadap 10 *user* tersebut.

Tabel 2. Pengujian *Non-Stemming* Menggunakan 5-7 Kata Kunci Dengan 10 Hasil Dokumen

Kata Kunci	Banyak Kata Kunci	Nilai Cosine Similarity Tertinggi	Nilai Cosine Similarity Terendah	Rata-Rata
Sistem Penjadwalan Matakuliah Otomatis Berbasis Web Menggunakan	5 kata kunci	0.3235	0.0612	0.1287
	6 kata kunci	0.3283	0.0909	0.1448
		0.3283	0.0909	0.1448

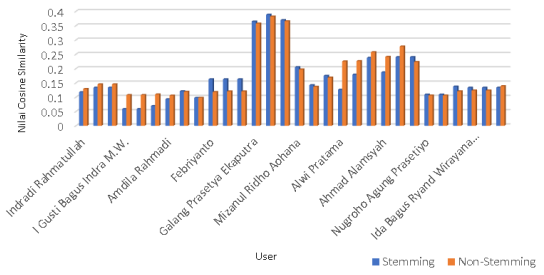
penggunaan blockchain pada bidang militer di Indonesia	7 kata kunci			
	5 kata kunci	0.1836	0.0809	0.1073
	6 kata kunci	0.1836	0.0809	0.1073
	7 kata kunci	0.1535	0.0837	0.1096
pengembangan aplikasi e-commerce dengan fitur produk filtering	5 kata kunci	0.1423	0.0786	0.1055
	6 kata kunci	0.3303	0.0722	0.1188
	7 kata kunci	0.2693	0.0630	0.0977
perancangan sistem informasi manajemen pelanggan berbasis crm	5 kata kunci	0.1337	0.1040	0.1179
	6 kata kunci	0.1398	0.1041	0.1202
	7 kata kunci	0.1398	0.1041	0.1202
kinerja protokol routing DSDV DSR AOMDV AODV	5 kata kunci	0.5079	0.2165	0.3573
	6 kata kunci	0.5177	0.2995	0.3819
	7 kata kunci	0.4734	0.2870	0.3654
website sistem informasi akademik menggunakan clean arsitektur	5 kata kunci	0.4854	0.1134	0.1969
	6 kata kunci	0.3365	0.0786	0.1365
	7 kata kunci	0.3044	0.0961	0.1685
sistem keamanan pintu otomatis dengan sidik jari	5 kata kunci	0.3970	0.0820	0.2252
	6 kata kunci	0.3302	0.1525	0.2260
	7 kata kunci	0.5517	0.1413	0.2572
performa algoritma routing dalam jaringan ad-hoc terstruktur	5 kata kunci	0.3069	0.2126	0.2409
	6 kata kunci	0.3486	0.2242	0.2773
	7 kata kunci	0.2805	0.1804	0.2231
perbandingan citra rgb dan grayscale untuk klasifikasi	5 kata kunci	0.1799	0.0736	0.1059
	6 kata kunci	0.1799	0.0736	0.1059
	7 kata kunci	0.1721	0.0946	0.1204
kinerja machine learning dalam deteksi anomali jaringan	5 kata kunci	0.1756	0.0947	0.1232
	6 kata kunci	0.1756	0.0947	0.1232
	7 kata kunci	0.1771	0.1183	0.1384

3.6. Analisis Pengujian Stemming dan Non-Stemming Menggunakan Kata Kunci

Tabel 2. Pengujian *Non-Stemming* Menggunakan 5-7 Kata Kunci Dengan 10 Hasil Dokumen

Kata Kunci	Banyak Kata Kunci	Nilai Cosine Similarity Tertinggi	Nilai Cosine Similarity Terendah	Rata-Rata
Sistem Penjadwalan Matakuliah Otomatis Berbasis Web Menggunakan	5 kata kunci	0.3235	0.0612	0.1287
	6 kata kunci	0.3283	0.0909	0.1448
	7 kata kunci	0.3283	0.0909	0.1448
penggunaan blockchain pada bidang militer di Indonesia	5 kata kunci	0.1836	0.0809	0.1073
	6 kata kunci	0.1836	0.0809	0.1073
	7 kata kunci	0.1535	0.0837	0.1096
pengembangan aplikasi e-commerce dengan fitur produk filtering	5 kata kunci	0.1423	0.0786	0.1055
	6 kata kunci	0.3303	0.0722	0.1188
	7 kata kunci	0.2693	0.0630	0.0977
perancangan sistem informasi manajemen pelanggan berbasis crm	5 kata kunci	0.1337	0.1040	0.1179
	6 kata kunci	0.1398	0.1041	0.1202
	7 kata kunci	0.1398	0.1041	0.1202
kinerja protokol routing DSDV DSR AOMDV AODV	5 kata kunci	0.5079	0.2165	0.3573
	6 kata kunci	0.5177	0.2995	0.3819
	7 kata kunci	0.4734	0.2870	0.3654
website sistem informasi akademik menggunakan clean arsitektur	5 kata kunci	0.4854	0.1134	0.1969
	6 kata kunci	0.3365	0.0786	0.1365
	7 kata kunci	0.3044	0.0961	0.1685

sistem keamanan pintu otomatis dengan sidik jari	5 kata kunci	0.3970	0.0820	0.2252
	6 kata kunci	0.3302	0.1525	0.2260
	7 kata kunci	0.5517	0.1413	0.2572
performa algoritma routing dalam jaringan ad-hoc terstruktur	5 kata kunci	0.3069	0.2126	0.2409
	6 kata kunci	0.3486	0.2242	0.2773
	7 kata kunci	0.2805	0.1804	0.2231
perbandingan citra rgb dan grayscale untuk klasifikasi	5 kata kunci	0.1799	0.0736	0.1059
	6 kata kunci	0.1799	0.0736	0.1059
	7 kata kunci	0.1721	0.0946	0.1204
kinerja machine learning dalam deteksi anomali jaringan	5 kata kunci	0.1756	0.0947	0.1232
	6 kata kunci	0.1756	0.0947	0.1232
	7 kata kunci	0.1771	0.1183	0.1384
Rata-Rata			0.1658	0.1732



Gambar 8. Grafik Perbandingan *Stemming* dan *Non-Stemming*

Dalam pengujian dengan menggunakan stemming dan tanpa stemming, perbandingan hasilnya dapat dilihat dalam Tabel 3. Hasil dari pengujian sistem rekomendasi berbasis konten dengan menggunakan algoritma TF-IDF dan *Cosine Similarity* menghasilkan nilai rata-rata *cosine similarity* pada percobaan *stemming* sebesar 0,1658 dan percobaan *non-stemming* sebesar 0,1732. Hasil tersebut menunjukkan kemampuan sistem dalam mencari atau memberikan rekomendasi berdasarkan kemiripan dari data yang dimiliki oleh tiap topik penelitian. Performa sistem rekomendasi topik penelitian ini sudah tergolong bagus, mengingat nilai *cos* memiliki rentang nilai dari -1 hingga 1. Secara khusus, nilai rata-rata proses *stemming* sebesar 0,1658 dan proses *non-stemming* sebesar 0,1732, kedua nilai tersebut berada di atas nilai 0. Pada percobaan perbandingan antara proses *stemming* dan *non-stemming*, Nilai rata-rata *cosine similarity* pada hasil tersebut cenderung lebih tinggi dibandingkan dengan percobaan menggunakan *stemming*. Ini disebabkan bahwa pada percobaan *non-stemming dataset* tidak mengalami proses *stemming*. Proses *stemming* adalah proses untuk menghilangkan kata dasar dari kata yang memiliki imbuhan yang dapat mempengaruhi pembobotan TF-IDF dan perhitungan *cosine similarity* saat mencocokkan kata kunci. Sebagai contoh, kata "Dirancang" memiliki kata dasar "Rancang". Dalam perhitungan TF-IDF, kata-kata ini memiliki perbedaan karena imbuhan yang dimilikinya. Oleh karena itu, dalam proses pembobotan TF-IDF, perbandingan antara kata "Dirancang" dan kata "Rancang" dianggap sebagai dua kata yang memiliki makna kata berbeda, sehingga hasil pembobotan TF-IDF tidak sesuai dengan seharusnya.

Selain itu, faktor lain yang menyebabkan nilai *cosine similarity* pada percobaan *non-stemming* tersebut lebih tinggi adalah karena konteks dataset yang digunakan, terutama data judul dan abstrak pada topik penelitian, lebih tepat untuk mempertahankan kata-kata dalam bentuk dasarnya atau tidak mengubahnya menjadi kata dasarnya, karena hal tersebut akan menjadikan makna kata-kata tersebut lebih spesifik.

Terakhir, faktor yang menyebabkan nilai *cosine similarity* pada percobaan *non-stemming* tersebut lebih tinggi adalah faktor *library* yang digunakan dalam melakukan proses *stemming*. *Library* yang digunakan dalam proses *stemming* yaitu Sastrawi. Library ini masih kurang optimal ketika menggunakan *dataset* yang berkaitan dengan topik penelitian. Sebagai contoh kata “Berbasis” seharusnya akan menjadi kata “Basis” ketika melakukan proses *stemming*. Setelah mencoba *library* tersebut, kata “Berbasis” tersebut berubah menjadi kata “Bas”, sehingga pengolahan proses *stemming* tersebut menjadi kurang optimal dilakukan dengan menggunakan *library* Sastrawi.

4. KESIMPULAN DAN SARAN

Berdasarkan hasil yang didapatkan pada penelitian ini, penggunaan teknik pembobotan TF-IDF dan algoritma *cosine similarity* dalam sistem rekomendasi topik penelitian memungkinkan user untuk menerima rekomendasi yang sesuai berdasarkan kesamaan antara data topik penelitian yang ada dalam *database*.

Sistem rekomendasi topik penelitian dengan metode *content-based filtering* pada penelitian ini memiliki nilai rata-rata *cosine similarity* 0,1658 pada proses *stemming* dan 0,1732 pada proses *non-stemming*. Faktor yang dapat mempengaruhi nilai skor *cosine similarity* tersebut yaitu seperti kata kunci, *dataset*, dan proses *preprocessing data* yang

dilakukan. Semakin cocok kata kunci dengan *database*, maka semakin tinggi *cosine similarity*-nya. Jika kata kunci tidak ada di *database*, nilai *cosine similarity* tersebut mengalami penurunan.

DAFTAR PUSTAKA

Amrizal, V. (2018). Penerapan Metode Term Frequency Inverse Document Frequency (Tf-Idf) Dan Cosine Similarity Pada Sistem Temu Kembali Informasi Untuk Mengetahui Syarah Hadits Berbasis Web (Studi Kasus: Hadits Shahih Bukhari-Muslim). Jurnal Teknik Informatika, 11(2), 149–164. https://doi.org/10.15408/jti.v11i2.8623

Badriyah, T., Restuningtyas, I., & Setyorini, F. (2017). Sistem Rekomendasi Collaborative Filtering Berbasis User Algoritma Adjusted Cosine Similarity.

Gantini, T. (2016). Penerapan Metode Content-Based Filtering Pada Sistem Rekomendasi Kegiatan Ekstrakulikuler (Studi Kasus di Sekolah ABC). In Jurnal Teknik Informatika dan Sistem Informasi (Vol. 2).

Novandra, R. R., & Heryanto, H. (2021). Perancangan Sistem Rekomendasi Influencer Menggunakan Knowledge-Based Filtering. In Media Informatika (Vol. 20, Issue 3).

Putra, A. I., & Santika, R. R. (2020). Edumatic: Jurnal Pendidikan Informatika Implementasi Machine Learning dalam Penentuan Rekomendasi Musik dengan Metode Content-Based Filtering. 4(1). https://doi.org/10.29408/edumatic.v4i1.2162

Rifano, E. J., Fauzan, Abd. C., Makhi, A., Nadya, E., Nasikin, Z., & Putra, F. N. (2020). Text Summarization Menggunakan Library Natural Language Toolkit (NLTK) Berbasis Pemrograman Python. ILKOMNIKA: Journal of Computer Science and Applied Informatics, 2(1), 8–17. https://doi.org/10.28926/ilkomnika.v2i1.32

Rizqi, M., & Zayyad, A. (2021). Sistem Rekomendasi Buku Menggunakan Metode Content Based Filtering.

Salim, E., Pragantha, J., & Lauro, M. D. (n.d.). Perancangan Sistem Rekomendasi Film menggunakan metode Content-based Filtering.