

METODE COSINE SIMILARITY UNTUK REKOMENDASI JUDUL KOMIK

THE COSINE SIMILARITY METHOD FOR COMIC TITLE RECOMMENDATIONS

Rina Candra Noor Santi¹, Sri Eniyati², Sugiyamta³, Arba'i Kurniaji⁴
^{1,2,3,4}Fakultas Teknik Informatika dan Industri, Universitas Stikubank Semarang
r_candra_ns@edu.unisbank.ac.id

ABSTRACT

Comics are the art of still images and their content consists of pictures and writing that are arranged into a story. There are many types of comics that can be found, such as serial physical book comics, online comics in digital format, to comic strips, aka short comics. Comics currently have many types of interesting plots and stories so that readers are easily interested in reading. So, with so many plots, readers are sometimes a little confused about following the story line. So this problem can be solved by creating a recommendation for selecting comics to read using Cosine Similarity.

Keyword: *Comics, Digital Format, Cosine Similarity*

ABSTRAK

Komik adalah seni gambar tidak bergerak dan isinya terdiri dari gambar-gambar serta tulisan yang disusun menjadi sebuah cerita. Ada banyak jenis komik yang bisa ditemukan, seperti komik buku fisik berseri, komik online dalam format digital, hingga komik strip alias komik pendek. Komik pada saat ini memiliki banyak jenis alur dan cerita yang menarik agar pembaca mudah tertarik untuk membaca. Sehingga dari alur yang begitu banyaknya, membuat pembaca kadang sedikit bingung untuk mengikuti jalur ceritanya. Sehingga dari permasalahan ini dapat diselesaikan dengan membuat rekomendasi untuk memilih komik yang akan dibaca dengan menggunakan *Cosine Similarity*.

Kata Kunci: Komik, Format Digital, Cosine Similarity

PENDAHULUAN

Komik adalah cerita bergambar yang umumnya mudah dicerna pembaca karena lucu (KBBI, 2023). Tetapi mengikuti perkembangan jaman komik sendiri sekarang memiliki cerita dan alur yang bermacam-macam serta memiliki genre yang bermacam-macam. Terdapat juga pengertian Komik adalah sebuah media atau karya seni yang menggabungkan gambar dan teks untuk menceritakan cerita atau menyampaikan pesan kepada pembaca.

Secara umum, komik terdiri dari panel-panel gambar yang disusun secara berurutan, yang bisa berisi dialog, narasi, atau kombinasi keduanya. (Soedarso, 2015). Komik juga bisa dikatakan sebagai sastra gambar serta menjadi alat komunikasi visual yang dapat digunakan untuk menyampaikan sebuah informasi agar dapat dengan mudah dimengerti. Komik biasanya dicetak pada kertas yang

biasa kita lihat pada koran dalam bentuk strip atau dicetak dalam bentuk buku (Maharsi, 2011). Banyaknya komik yang beredar serta memiliki alur cerita dan genre yang beragam menjadi masalah bagi orang yang ingin membaca komik. Hal ini membuat orang yang ingin membaca komik mengalami kesusahan untuk memilih komik yang akan dibaca, sehingga perlu adanya rekomendasi atau saran untuk baacn komik. Fungsi rekomendasi ini untuk mempermudah para pembaca dalam mencari genre komik yang disukai. Dan juga fungsi rekomendasi komik adalah untuk membantu pembaca menemukan dan menikmati komik-komik yang sesuai dengan minat dan preferensi mereka. Berikut ini beberapa fungsi utama dari rekomendasi komik:

1. **Memudahkan Pencarian:** Dengan jumlah komik yang sangat banyak, baik dalam bentuk cetak maupun digital, rekomendasi membantu pembaca

- menemukan komik-komik baru yang mungkin belum mereka ketahui sebelumnya.
2. **Personalisasi:** Rekomendasi komik sering kali disesuaikan dengan preferensi pembaca berdasarkan genre favorit, tema cerita, gaya gambar, atau penulis/penyunting yang disukai pembaca.
 3. **Menambah Diversitas Bacaan:** Pembaca dapat menemukan komik-komik dari genre yang berbeda atau dari budaya yang beragam, yang mungkin tidak mereka perhatikan sebelumnya.
 4. **Meningkatkan Pengalaman Membaca:** Dengan rekomendasi yang baik, pembaca dapat menemukan cerita-cerita yang mendalam, menarik, dan memuaskan untuk dinikmati.
 5. **Mendorong Interaksi dan Diskusi:** Rekomendasi komik sering kali dipertukarkan antar pembaca melalui forum, platform daring, atau media sosial, yang dapat meningkatkan pengalaman membaca dengan berbagi pendapat dan kesan.
 6. **Promosi untuk Pembuat Komik:** Bagi pembuat komik, rekomendasi dapat meningkatkan eksposur karya mereka kepada audiens yang lebih luas, membantu mereka mendapatkan pengakuan dan dukungan yang lebih besar.

Dengan demikian, rekomendasi komik tidak hanya membantu pembaca menemukan hiburan yang sesuai dengan minat mereka, tetapi juga memainkan peran penting dalam memajukan industri komik dengan memperluas jangkauan dan apresiasi terhadap karya-karya yang ada

METODE

Ada beberapa metode yang umum digunakan untuk melakukan rekomendasi komik kepada pembaca. Salah satunya **Content-Based Filtering (Filtering Berbasis Konten)**. **Content-Based Filtering** menggunakan informasi terkait konten komik seperti genre, tema,

karakter, dan elemen-elemen cerita lainnya untuk membuat rekomendasi. Sistem ini mencari kesamaan antara komik yang telah disukai oleh pengguna dengan komik lain berdasarkan analisis terhadap atribut-atribut tersebut.

Cosine similarity adalah salah satu teknik yang sering digunakan dalam content-based filtering untuk menentukan seberapa mirip antara item (seperti komik) berdasarkan atribut-atribut mereka. Berikut ini adalah kaitannya:

1. **Pengukuran Kemiripan Berbasis Konten:** Dalam content-based filtering, setiap item (misalnya komik) direpresentasikan sebagai vektor fitur berdasarkan atribut-atribut tertentu seperti genre, tema, karakter, dan elemen cerita lainnya. *Cosine similarity* digunakan untuk mengukur seberapa mirip vektor fitur dari dua item. Semakin tinggi nilai *cosine similarity* antara dua komik, semakin mirip mereka dalam konteks atribut-atribut ini.
2. **Perhitungan Similarity:** Untuk setiap pasangan item (A dan B), *cosine similarity* dihitung dengan menggunakan vektor fitur dari masing-masing item.
3. **Penggunaan dalam Rekomendasi:** *Cosine similarity* digunakan dalam content-based filtering untuk menyajikan rekomendasi kepada pengguna berdasarkan kesamaan antara komik yang disukai oleh pengguna dengan komik lainnya. Misalnya, jika seorang pengguna menikmati komik dengan genre fantasy dan tema petualangan, sistem rekomendasi dapat menggunakan *cosine similarity* untuk menemukan komik lain yang memiliki atribut serupa.
4. **Keuntungan Cosine Similarity:** Salah satu keuntungan *cosine similarity* adalah kemampuannya untuk menangkap kemiripan berdasarkan orientasi atau arah vektor fitur, tanpa memperhatikan magnitudo (besar) dari vektor tersebut. Hal ini cocok dalam

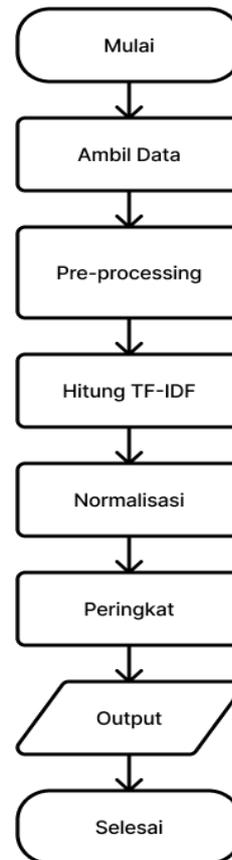
konteks content-based filtering di mana penting untuk menentukan sejauh mana dua item mirip berdasarkan karakteristik mereka.

Dengan menggunakan cosine similarity dalam content-based filtering, sistem rekomendasi dapat memberikan rekomendasi yang lebih personal dan relevan kepada pengguna berdasarkan preferensi mereka terhadap atribut-atribut tertentu dari komik yang mereka nikmati.

Metode Cosine Similarity

Metode cosine similarity adalah salah satu metode yang digunakan untuk mengambil keputusan seberapa mirip antara item (seperti komik) berdasarkan atribut-atributnya. Dalam pembuatan sistem pendukung keputusan dengan metode cosine similarity ada metode yang bisa digunakan dalam mengidentifikasi item atau data berupa teks yaitu dengan menggunakan metode TF-IDF (*Term Frequency Inverse Document Frequency*).

Beberapa tahapan yang dilakukan pada sistem rekomendasi ini adalah *pre-processing*, perhitungan tf-idf dan normalisasi.



Gambar 1. Alur Pembuatan Rekomendasi
1. Pre-Processing

Pre-processing (pre-pemrosesan) adalah tahap yang penting dalam analisis data atau pengolahan informasi di mana data mentah atau input awal dipersiapkan, dibersihkan, dan diubah menjadi format yang lebih sesuai untuk analisis atau aplikasi tertentu.

Tujuan dari pre-processing adalah untuk meningkatkan kualitas data, menghilangkan noise atau gangguan, serta mengubah data ke dalam format yang lebih mudah diolah dan diinterpretasikan oleh sistem atau algoritma yang akan digunakan..

Tahapan yang dilakukan pada proses ini pertama clean data dimana data teks akan diubah menjadi huruf kecil, menghilangkan tanda baca dan menghilangkan angka. Tahapan *pre-processing* adalah tahapan untuk mempersiapkan item atau data yang berupa teks untuk dapat dilakukan analisa dan perhitungan. Dalam tahapan *pre-processing* ini meliputi proses *clean data*, tokenisasi, dan *filtering*. *Clean data* adalah

tahapan dimana data akan dibersihkan dari *noise* seperti mengubah data menjadi huruf kecil, menghapus angka dan menghapus tanda baca. Tokenisasi adalah proses dimana data berupa teks akan dipisah menjadi kata per kata. Hal ini bertujuan untuk mempermudah proses perhitungan term nantinya. Dan *Filtering* adalah proses memfilter atau menghapus kata-kata yang tidak dibutuhkan pada proses perhitungan atau analisis data. Proses ini juga bisa disebut dengan *stop word removal*.

2. Perhitungan TF-IDF

Setelah data dibersihkan kemudian data dapat dihitung dengan menggunakan metode TF-IDF. TF-IDF adalah nilai statistik yang mencerminkan atau menunjukkan kemiripan suatu term dengan beberapa data dalam dokumen yang membuat term tersebut bisa menjadi kata kunci dari dokumen tersebut. Nilai TF-IDF diambil dari perhitungan *Term Frequency* (TF) yang menggambarkan frekuensi kemunculan istilah atau kata yang muncul dari dokumen yang ada, dan *Inverse Document Frequency* (IDF) yang menggambarkan frekuensi dokumen yang memiliki istilah dalam koleksi. Berikut adalah rumus perhitungan dari TF :

$$tf(d, t) = 1 + \log(f_{dt})$$

Keterangan :

tf (d,t) = Nilai TF untuk kata ke-t dalam dokumen ke-d

f dt = Jumlah kata ke-t dalam dokumen ke-d

Berikut adalah rumus perhitungan dari IDF:

$$IDF(t) = \log \frac{D}{df}$$

Keterangan :

IDF(t) = Nilai IDF untuk kata ke-t pada item

Df = Jumlah dokumen yang mengandung kata ke-t

D = Total jumlah dokumen pada item

Berikut adalah rumus yang digunakan dalam proses TF-IDF :

$$Wd, t = TF(d, t) \times IDF(t)$$

Keterangan :

d = dokumen ke-d

t = kata ke-t dari kata kunci

W = bobot dokumen d terhadap kata ke-t

3. Normalisasi

Tahapan normalisasi pada cosine similarity adalah proses untuk mengubah vektor fitur dari item-item yang akan dibandingkan ke dalam bentuk yang setara atau terstandarisasi, sehingga hasil perhitungan cosine similarity tidak terpengaruh oleh magnitudo (besarnya) dari vektor tersebut. Normalisasi dilakukan untuk memastikan bahwa cosine similarity hanya mempertimbangkan orientasi atau arah dari vektor fitur, bukan besarnya.. *Cosine Similarity* sering digunakan untuk melakukan analisis data dan untuk pembuatan sistem rekomendasi. Berikut adalah rumus dari cosine similarity.

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

Keterangan :

A = vektor

B = vektor

A_i = bobot term i dalam blok A_i

B_i = bobot term i dalam blok B_i

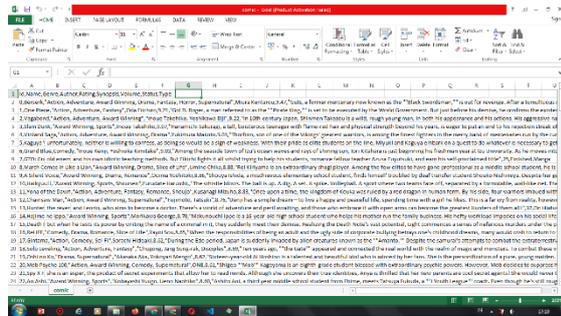
I = jumlah term dalam kalimat

N = jumlah vektor

HASIL DAN PEMBAHASAN

Dalam sistem rekomendasi ini proses pengumpulan data menggunakan metode dataset statistik yaitu dengan mengumpulkan atau mendapatkan data melalui *website* penyedia dataset. Data yang dikumpulkan pada penelitian ini akan dipilih lagi dan dipisah menjadi 9 kolom yaitu id, nama, genre, komikus, rating, sinopsis, volume, status dan tipe. Sedangkan kolom yang akan dijadikan kriteria adalah kolom genre, komikus dan sinopsis. Jumlah data yang akan digunakan pada pembuatan sistem rekomendasi ini adalah 200 data komik

Berikut data yang dikumpulkan dalam format csv:



Dalam sistem rekomendasi ini tahapan *pre-processing* meliputi *clean data*, tokenisasi dan filtering.

1. Clean Data

Berikut adalah tabel proses yang terjadi pada tahapan clean data beserta kode program untuk mengimplementasikannya.

```
df_comic['clean_plot'] = df_comic['Synopsis'].str.lower()
df_comic['clean_plot'] = df_comic['clean_plot'].apply(lambdax: re.sub('[^a-zA-Z]', '',str(x)))
df_comic['clean_plot'] = df_comic['clean_plot'].apply(lambdax: re.sub('/s+', '',str(x)))
```

2. Tokenisasi

Tahapan tokenisasi merupakan tahapan yang bertujuan untuk mengubah data teks menjadi kata-kata yang terpisah atau memisahkan kata-kata. Berikut adalah tabel dan kode untuk mengimplementasi pada proses tokenisasi.

```
df_comic['clean_plot'] = df_comic['clean_plot'].apply(lambdax: nltk.word_tokenize(x))
```

3. Filtering

Tahapan filtering merupakan tahapan yang berguna untuk memfilter atau menghilangkan kata-kata yang tidak dibutuhkan dalam proses analisis dokumen. Berikut adalah table dan kode untuk mengimplementasikan proses filtering.

```
stop_words = nltk.corpus.stopwords.words('english')
plot = []
for sentence in df_comic['clean_plot']:
    temp = []
```

```
for word in sentence:
    if word notin stop_words
orlen(word) >= 3:
        temp.append(word)
plot.append(temp)
```

4. Penggabungan Data

Pada tahapan ini adalah menggabungkan data sinopsis yang sudah dilakukan *pre-processing* dengan data genre dan komikus

Berikut adalah kodenya.

```
df_comic['Author'] = df_comic['Author'].apply(lambdax: str(x).split(','))
df_comic['Genre'] = df_comic['Genre'].apply(lambdax: str(x).split(','))
defclean(sentences):
    temp = []
    for word in sentences:
        temp.append(word.lower().replace(' ', ''))
    return temp
df_comic['Author'] = [clean(x) for x in df_comic['Author']]
df_comic['Genre'] = [clean(x) for x in df_comic['Genre']]
```

5. Hitung TF-IDF

TF-IDF (*Term Frequency - Inverse Document Frequency*) merupakan sebuah metode yang digunakan untuk menghitung bobot kata pada sebuah dokumen atau data. Dalam penggunaan metode ini mempertimbangkan frekuensi kemunculan kata pada sebuah dokumen dan frekuensi kemunculan kata pada semua dokumen. Pada pembuatan sistem rekomendasi ini dibantu dengan menggunakan *library python* yaitu *scikit-learn* dengan memanggil `TfidfVectorizer()`. Kemudian menggunakan `method fit_transform()` pada objek `TfidfVectorizer()` tersebut untuk menghasilkan vektor fitur yang merupakan representasi TF-IDF dari dataset yang telah dibuat. Hasil vektor

kemudian disimpan pada *features*. Berikut adalah kode untuk mengimplementasikannya.

```
tfidf = TfidfVectorizer()
features =
tfidf.fit_transform(df_comic['clean'
])
```

Hasil dan Pembahasan Cosine Similarity

1. Normalisasi

Pada tahapan normalisasi dilakukan untuk menormalisasikan nilai pada TF-IDF agar memiliki rentang nilai yang sama. Normalisasi yang dilakukan pada sistem rekomendasi ini menggunakan *Cosine Similarity*.

Proses *Cosine Similarity* ini akan dilakukan setelah pengguna menginputkan judul komik yang dijadikan preferensi. *Cosine similarity* akan menghitung kemiripan antara dua dokumen dalam bentuk vektor berdasarkan sudut antara vektor-vektor tersebut. Semakin kecil sudut antara vektor-vektor, maka semakin mirip pula dokumen-dokumen tersebut.

Berikut adalah kode dari implementasi tahapan normalisasi.

```
cosine_sim_matrix =
cosine_similarity(features,
features)
def get_recommendations(comic_title,
cosine_sim_matrix=cosine_sim_matrix,
comic_df=df_comic):
indices =
pd.Series(comic_df.index,
index=comic_df['Name'])
idx = indices[comic_title]
cosine_scores =
list(enumerate(cosine_sim_matrix[idx]))
cosine_scores =
sorted(cosine_scores,
key=lambda x: x[1], reverse=True)
cosine_scores =
cosine_scores[1:11]
comic_indices = [i[0]
for i in cosine_scores]
```

```
recommendation_scores = [i[1]
for i in cosine_scores]
return comic_df['Name'].iloc[comic_indices],
recommendation_scores
```

Berikut merupakan tabel hasil cosine similarity dengan preferensi judul komik contoh Ao Ashi

Tabel 1. Perhitungan Cosine Similarity

Judul Komik	Cosine Similarity
Darwin's Game	0.08803
Kuroko's Basketball	0.10375
Eyeshield 21	0.09680
Be Blues! Ao ni Nare	0.09921
.....
Days	0.09085

2. Pemeringkatan

Tahapan pemeringkatan adalah tahapan dimana sistem akan mengurutkan dokumen-dokumen berdasarkan hasil dari *Cosine Similarity* dan kemudian akan diambil peringkat 10 besar dokumen teratas secara menurun. Kode pada tahap ini ada pada bagian normalisasi berikut adalah kodenya.

```
cosine_scores =
list(enumerate(cosine_sim_matrix[idx]))
cosine_scores =
sorted(cosine_scores,
key=lambda x: x[1], reverse=True)
cosine_scores =
cosine_scores[1:11]
comic_indices = [i[0]
for i in cosine_scores]
```

Berikut merupakan tabel hasil cosine similarity dan pemeringkatan dengan preferensi judul komik Ao Ashi :

Tabel 2. Pemeringkatan Berdasarkan Cosine Similarity

Judul Komik	Cosine Similarity
Giant Killing	0.1390
Blue Lock	0.1212
Ariadne in the Blue Sky	0.1119
Ace of the Diamond	0.1086
Kuroko's Basketball	0.1037

	5
Be Blues! Ao ni Nare	0.0992 1
Tokyo Ghoul:re	0.0985 9
Eyeshield 21	0.0968 0
Days	0.0908 5
That Time I Got Reincarnated as a Slime	0.0904 6

3. Pengujian Dan Analisa

Pada fase ini, kami menguji sistem dengan melakukan eksperimen untuk mengambil rekomendasi buku komik dan menganalisis skor kemiripan.

Pada penelitian ini pengujiannya menggunakan komik Aoaashi sebagai komik pilihan. Sistem mencari skor kemiripan yang paling dekat dengan komik Aoashi. Setelah dilakukan perhitungan, nilai cosine-similarity pada manga "Aoashi" adalah sebesar 0,13901. Tabel 3 menunjukkan hasil rekomendasi, nilai cosine-similarity, dan tingkat kemiripan pada pengujian dengan Komik Aoashi sebagai komik prioritas.

Tabel 3. Tabel Perhitungan Cosine Similarity dan Persentase Kemiripan

Judul Komik	Cosine Similarity	Persentase Kemiripan
Giant Killing	0.13901	100%
Blue Lock	0.12123	87.21%
Ariadne in the Blue Sky	0.11194	80.53%
Ace of the Diamond	0.10865	78.16%
Kuroko's Basketball	0.10375	74.64%
Be Blues! Ao ni Nare	0.09921	71.37%
Tokyo Ghoul:re	0.09859	70.92%
Eyeshield 21	0.09680	69.64%
Days	0.09085	65.36%
That Time I Got Reincarnated as a Slime	0.09046	65.08%

Dari Tabel 3 terlihat bahwa nilai peringkat pertama yaitu komik Giant Killing mempunyai nilai cosine kemiripan yang sama dengan komik pilihan yaitu 0,13901 dan tingkat kemiripannya sebesar 100%. Komik

peringkat 10 "Saat Itu Aku Bereinkarnasi sebagai Slime" memiliki skor kesamaan kosinus 0,09046 dan tingkat kesamaan 65,08%. Seluruh hasil rekomendasi yang diperoleh diberikan skor kemiripan cosinus yang mendekati atau mirip dengan komik Aoashi, dan seluruh tingkat kemiripan yang diperoleh berada di atas 50%. Rata-rata hasil rekomendasi menghasilkan nilai 0,10604 yang merupakan tingkat kemiripan sebesar 76,38%.

Hal ini menandakan bahwa hasil rekomendasi yang diberikan kompatibel dan mirip dengan komik atau Aoashi yang disukai. Tingkat kemiripan sebesar 76,38% juga menunjukkan bahwa sistem berfungsi dan memberikan hasil rekomendasi yang baik

SIMPULAN

Hal ini didasarkan pada penelitian yang dilakukan dengan menggunakan teknik content-based filtering menggunakan TF-IDF dan cosine kemiripan pada sistem rekomendasi komik untuk mendapatkan hasil rekomendasi yang sesuai berdasarkan konten komik. Metode TF-IDF digunakan untuk menghitung kemiripan kata dalam komik berdasarkan frekuensi kemunculannya, dan cosine-similarity digunakan untuk menghitung kemiripan antar komik berdasarkan representasi vektor TF-IDF. Kriteria yang digunakan dalam sistem rekomendasi ini adalah konten, genre, dan artis manga.

Hasil rekomendasi akan menemukan 10 rekomendasi manga berdasarkan preferensi manga Anda

DAFTAR PUSTAKA

- Alkaff, M., Khatimi, H., & Eriadi, A. (2020). Sistem Rekomendasi Buku Menggunakan Weighted Tree Similarity dan Content Based Filtering. *MATRIK J. Manajemen, Tek. Inform. Dan Rekayasa Komput*, 20(1), 193–202.

- Arfisko, H. H., & Wibowo, A. T. (2022). Sistem Rekomendasi Film Menggunakan Metode Hybrid Collaborative Filtering Dan Content-Based Filtering. *EProceedings of Engineering*, 9(3).
- Bonneff, M. (1998). *Komik indonesia*. Kepustakaan Populer Gramedia.
- Christian, Y., & Kelvin, K. (2022). Rancang Bangun Aplikasi Kursus Online Berbasis Web Dengan Sistem Rekomendasi Metode Content-Based Filtering. *Rabit: Jurnal Teknologi Dan Sistem Informasi Univrab*, 7(1), 23–36.
- Fajriansyah, M., Adikara, P. P., & Widodo, A. W. (2021). Sistem Rekomendasi Film Menggunakan Content Based Filtering. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 5(6), 2188–2199.
- Fathoroni, A., Fatonah, R. N. S., Andarsyah, R., & Riza, N. (2020). *Buku tutorial sistem pendukung keputusan penilaian kinerja dosen menggunakan metode 360 degree feedback*. CV. Kreatif Industri Nusantara.
- Limbong, T., Muttaqin, M., Iskandar, A., Windarto, A. P., Simarmata, J., Mesran, M., Sulaiman, O. K., Siregar, D., Nofriansyah, D., & Napitupulu, D. (2020). *Sistem Pendukung Keputusan: Metode & Implementasi*. Yayasan Kita Menulis.
- Lindang, D. N., Muniar, A. Y., Halid, A., Muhajirin, M., & Amiruddin, A. (2023). Sistem Penentuan Kemiripan Antar Skripsi Menggunakan Metode Cosine Similarity Pada Perpustakaan. *Seminar Nasional Teknik Elektro Dan Informatika (SNTEI)*, 8(1), 321–324.
- Maharsi, I. (2011). *Komik*. Dwi-Quantum.
- Nastiti, P. (2019). Penerapan Metode Content Based Filtering dalam Implementasi Sistem Rekomendasi Tanaman Pangan. *Teknika*, 8(1), 1–10.
- Putra, A. I., & Santika, R. R. (2020). Implementasi Machine Learning dalam Penentuan Rekomendasi Musik dengan Metode Content-Based Filtering. *Edumatic: Jurnal Pendidikan Informatika*, 4(1), 121–130.
- Putri, I. H., Nurakhmadyavi, S. M. K. H., & Wahyudi, E. E. (2022). Literature Review: Sistem Rekomendasi untuk Buku dan Film. *Prosiding Seminar Nasional Teknik Elektro, Sistem Informasi, Dan Teknik Informatika (SNESTIK)*, 1(1), 75–81.
- Soedarso, N. (2015). Komik: karya sastra bergambar. *Humaniora*, 6(4), 496–506.
- Sugiyono. (2013). *Metodelogi Penelitian Kuantitatif dan Kualitatif Dan R&D*. ALFABETA.
- Turban, E. (2003). *Decision Support System and Intelligen System (Sistem Pendukung Keputusan dan Sistem Cerdas)*, Jilid 1, Andi Offset. Yogyakarta.
- Ula, N., Setianingsih, C., & Nugrahaeni, R. A. (2021). Sistem Rekomendasi Lagu Dengan Metode Content Based Filtering Berbasis Website. *EProceedings of Engineering*, 8(6).
- Widjaja, A. A., & Palit, H. N. (2022). Hybrid Recommendation System untuk Peminjaman Buku Perpustakaan dengan Collaborative dan Content-Based Filtering. *Jurnal Infra*, 10(2), 1–6.