

SISTEM REKOMENDASI PEMBELIAN LISENSI FILM MENGUNAKAN PENDEKATAN HYBRID FILTERING (STUDI KASUS: FILM ANIMASI JEPANG)

Kuncoro Triandono Mukti, S.T¹⁾ dan Dr. Iffatul Mardhiyah, S.Si., M.Si²⁾

^{1, 2)}Magister Manajemen Sistem Informasi Universitas Gunadarma

Jl Margonda Raya, Depok, Jawa Barat

e-mail: kuncorotriandonomukti@gmail.com¹⁾, iffatul@staff.gunadarma.ac.id²⁾

* Korespondensi: e-mail: kuncorotriandonomukti@gmail.com

ABSTRAK

Perkembangan zaman yang sangat pesat saat ini memunculkan banyak platform hiburan diantaranya streaming video seperti Netflix. Perusahaan pemilik platform streaming video sangat berhati-hati dalam pemilihan dan pembelian suatu lisensi film oleh karena itu pengembangan system rekomendasi untuk pembelian lisensi film animasi jepang menggunakan metode hybrid filtering dengan menggabungkan metode content-based filtering dan collaborative filtering. Data yang digunakan merupakan database dari forum anime terbesar di dunia yaitu myanimelist.com yang tersedia di Kaggle.com. Cosine similarity digunakan untuk menghitung kemiripan kata yang terkandung pada sinopsis film antara 1 film dengan film lainnya. 30 film dengan nilai kemiripan tertinggi dipilih kemudian diurutkan berdasarkan weighted rating IMDB yang kemudian hasilnya merupakan rekomendasi berdasarkan content-based filtering. Collaborative filtering menggunakan preferensi pengguna untuk memprediksi rating yang akan diberikan berdasarkan kemiripan antara pengguna 1 dengan pengguna lainnya. Analisis data dilakukan menggunakan google colab notebook sedangkan platform website dibangun menggunakan bahasa pemrograman python dan framework flask. Pengujian system dilakukan menggunakan confusion matrix dengan nilai presisi yaitu 71%, MAE dengan nilai 0.9 dan RMSE 1.1985. Hasil penelitian menunjukkan bahwa tujuan dari penelitian dapat dipenuhi yaitu membangun system rekomendasi menggunakan metode Hybrid Filtering agar dapat memberikan hasil berupa daftar rekomendasi film dan mengetahui rancang bangun dan implementasi system rekomendasi film pada platform website.

Kata Kunci: content-based filtering, collaborative filtering, hybrid filtering

ABSTRACT

The rapid development of today's era has given rise to many entertainment platforms, including video streaming such as Netflix. The company that owns the video streaming platform is very careful in selecting and purchasing a film license, therefore the development of a recommendation system for purchasing Japanese animation licenses uses a hybrid filtering method by combining content-based filtering and collaborative filtering methods. The data used is a database from the world's largest anime forum, myanimelist.com, which is available on Kaggle.com. Cosine similarity is used to calculate the similarity of words contained in the film synopsis between one film and another. The 30 films with the highest similarity values were selected and then sorted based on the IMDB weighted rating which then the results were recommendations based on content-based filtering. Collaborative filtering uses user preferences to predict the rating that will be given based on the similarity between user 1 and other users. Data analysis was carried out using google colab while the website platform was built using the python programming language and flask framework. System testing was carried out using a confusion matrix with a precision value of 71%, MAE with a value of 0.9, and RMSE 1.1985. The results show that the objectives of the research can be fulfilled, namely building a recommendation system using the hybrid filtering method in order to provide results in the form of a list of film recommendations and to know the design and implementation of a film recommendation system on the website platform.

Keywords: collaborative filtering, content-based filtering, cosine similarity, hybrid filtering.

I. PENDAHULUAN

Pengguna internet Indonesia mencapai 212,35 juta jiwa pada Maret 2021. Dengan jumlah tersebut, Indonesia berada di urutan ketiga dengan pengguna internet terbanyak di Asia, (internetworldstats.com). Data digital 2021 Indonesia pada bulan Januari menunjukkan 98.5% pengguna internet berusia 16 hingga 64 tahun menggunakan internet untuk aktivitas menonton video daring, (datareportal.com). Pada masa pandemic terdapat banyak platform entertainment yang berkembang dengan pesat seperti Youtube, Netflix, WeTV, Hotstar, VIU, dan sebagainya. Data reportal menyatakan pada halaman Top Social Media and Streaming Video Apps jika total waktu yang digunakan pengguna internet pada tahun 2020 untuk mengakses YouTube mencapai 25.9 jam/bulan, Netflix 9.3 jam/bulan, lalu disusul VIU 6.7 jam/bulan. Platform video streaming seperti Netflix menayangkan berbagai Acara TV dan Film dari berbagai negara. Tidak seperti Netflix terdapat platform yang hanya menayangkan acara TV atau Film dari negara tertentu atau kategori tertentu seperti Disney+ HOTSTAR cenderung menayangkan Acara TV dan Film yang dilisensikan oleh Disney dengan mayoritas kategori yaitu animasi dan fantasi. We TV memiliki fokus menyiarkan Acara TV dan Film dari negara Asia seperti Cina, Korea, Indonesia, dan negara asia lainnya. Penyiaran suatu acara TV atau film pada platform streaming harus memiliki sensi hak penyiaran. Untuk mendapatkan suatu lisensi acara TV atau film membutuhkan biaya dengan nominal yang tidak kecil. Pembelian lisensi acara TV atau film dapat mencapai ratusan ribu hingga jutaan dolar, (anievo.id). Oleh karena itu setiap perusahaan pemegang lisensi akan sangat berhati-hati dalam memperoleh lisensi suatu acara TV atau film yang akan ditayangkan. Sistem rekomendasi merupakan salah satu solusi yang dapat digunakan untuk mengatasi masalah tersebut, [1]. Netflix salah satu distributor streaming Acara TV dan Film terbesar di dunia menggunakan metric preferen si pengguna sebagai sumber dalam merekomendasikan suatu acara TV dan film untuk pengguna [2]. Metrik tersebut juga digunakan untuk menginformasikan acara TV dan film yang mungkin akan Netflix lisensikan atau peroleh di masa mendatang. Keberadaan sistem yang dapat membantu perusahaan agar dapat memberikan rekomendasi acara TV dan Film yang akan dilisensikan menjadi suatu hal penting bagi perusahaan distributor penyiaran yang baru akan didirikan, dengan begitu perusahaan dapat mengurangi kerugian dalam pemerolehan lisensi dikarenakan acara TV atau Film yang diperoleh ternyata tidak disukai pengguna.

II. LANDASAN TEORI

Pada penelitian ini penulis menggunakan metode *hybrid filtering* dengan menggabungkan metode *content-based filtering* dengan *collaborative filtering*.

A. Sistem Rekomendasi

Sistem rekomendasi adalah alat dan teknik perangkat lunak yang bias memberikan saran-saran untuk item yang sekiranya bermanfaat bagi pengguna [4]. Saran-saran tersebut berhubungan dengan proses pengambilan keputusan, seperti item mana yang akan dibeli atau lagu mana yang ingin didengarkan. Oleh karena itu, sistem rekomendasi memerlukan model rekomendasi yang tepat agar yang direkomendasikan sesuai dengan keinginan pengguna, serta mempermudah pengguna mengambil keputusan yang tepat dalam menentukan produk yang akan digunakannya. Sistem rekomendasi memiliki banyak metode diantaranya *demographic filtering*, *content-based filtering*, *collaborative filtering* dan *hybrid filtering*.

B. Content-based filtering

Metode content-based membuat rekomendasi dengan menganalisa deskripsi setiap item untuk mengidentifikasi item mana yang mempunyai hubungan ketertarikan khusus dari seorang user. Deskripsi ketertarikan user diperoleh dari profil user yang didasarkan atas penilaian menari kata tidaknya suatu item yang ada pada recommender sistem ini [6].

C. Collaborative filtering

Pendekatan *collaborative filtering* merekomendasikan item kepada user berdasarkan korelasi antara rating user yang berbeda terhadap item atau transaksi user di dalam sistem. Sementara user merating item, system akan menggunakan data rating user tertentu untuk melakukan perhitungan prediksi dan memberikan rekomendasi item yang belum pernah dirating oleh user tersebut [7].

D. Hybrid Filtering

Secara umum pendekatan hybrid recommendations adalah dengan menggabungkan lebih dari satu metode yang ada pada recommender system, kombinasi yang ada pada teknik ini misalnya dengan menggabungkan metode *content based* dengan *Collaborative Filtering*. (Aizawa, 2002).

E. Cosine Similarity

Cosine similarity berfungsi untuk mengukur kemiripan antara dua dokumen atau teks. Pada *cosine similarity* dokumen atau teks dianggap sebagai *vector*. Untuk pencocokan teks, nilai dari *vector* A dan B adalah *vector term-frequency* dari dokumen. Nilai *cosine similarity* berada pada range 0-1. Pada penelitian ini *cosine similarity* digunakan untuk menghitung jumlah kata istilah yang muncul pada *synopsis* yang diacu pada daftar indeks. Semakin banyak jumlah kata istilah yang muncul pada *synopsis* semakin tinggi nilai *cosine similarity* yang diperoleh. Cosine similarity memiliki persamaan sebagai berikut:

$$\text{similarity}(a,b) = \cos \theta = \frac{a \cdot b}{||a|| ||b||} \quad (1)$$

F. Term Frequency-Inverse Document Frequency (TF-IDF)

TF-IDF (*Term Frequency – Inverse Document Frequency*) TF-IDF digunakan untuk menentukan nilai frekuensi sebuah kata di dalam sebuah dokumen atau artikel dan juga frekuensi di dalam banyak dokumen. Perhitungan ini menentukan seberapa relevan sebuah kata di dalam sebuah dokumen.

Penelitian [8] berpendapat bahwa kata yang muncul dalam dokumen harus diberi nilai sebanding dengan frekuensi kata dan terbalik sebanding dengan frekuensi dokumen. Pembobotan skema yang mengikuti pendekatan ini disebut $TF \times IDF$ (*term frequency × inversedocumentfrequency*). Prosedur dalam implementasi TF-IDF terdapat perbedaan kecil di dalam semua aplikasinya, tetapi pendekatannya kurang lebih sama. TF (*Term Frequency*) merupakan frekuensi sebuah kata terdapat di dalam sebuah dokumen. Nilai dari TF didapat menggunakan persamaan:

$$TF(t) = f_{t,d} / \sum t, d \quad (1)$$

Dimana $f_{t,d}$ merupakan frekuensi sebuah kata (t) muncul di dalam dokumen d, sedangkan $\sum t, d$ merupakan total keseluruhan kata yang terdapat di dalam dokumen d [10].

Kemudian untuk menghitung nilai IDF (*Inverse Document Frequency*) dari sebuah kata di dalam kumpulan dokumen menggunakan persamaan:

$$IDF(t) = \log(|D|/f_{t,d}) \quad (2)$$

Di mana |D| merupakan jumlah dokumen yang ada di dalam koleksi, sedangkan $f_{t,d}$ merupakan jumlah dokumen dimana t muncul di dalam D [11]. Dalam koleksi dokumen D, sebuah kata t dan dokumen individu $d \in D$, dapat dihitung nilai TF-IDF menggunakan rumus:

$$TF - IDF(t) = TF(t) * IDF(t) \quad (3)$$

Diasumsikan bahwa $|D| \sim f_{t,d}$, ukuran dari kumpulan dokumen hampir sama dengan frekuensi t di dalam D. Jika $1 < \log(|D|/f_{t,d}) < c$ untuk sebuah konstanta c dengan nilai yang kecil, maka wd akan lebih kecil daripada $f_{t,d}$ tetapi tetap bernilai positif. Hal berarti bahwa w mempunyai relasi yang biasa dengan seluruh dokumen tetapi tetap menyimpan beberapa informasi yang penting [12]. Jika $f_{t,d}$ bernilai besar dan $f_{t,d}$ bernilai kecil, maka nilai dari $\log(|D|/f_{t,d})$ cenderung tinggi dan juga TF-IDF(t) akan bernilai tinggi. Kata dengan nilai TFIDF(t) yang tinggi berarti bahwa kata tersebut merupakan sebuah kata yang sangat penting di dalam dokumen d tetapi tidak di dalam kumpulan dokumen D.

G. *IMDB Weighted Rating*

Rating yang pengguna input akan memengaruhi nilai sebuah film. Nilai yang diinputkan seorang pengguna tidak langsung merubah nilai *rating* sebuah film namun jika dipopulasikan dengan *registereduser* lainnya yang jumlahnya mencapai jutaan, tentu akan ada perubahan secara signifikan. menggunakan sistem *weightedaverage* (pembobotan). Banyak parameter yang diikuti sertakan dalam formula IMDB untuk menghasilkan nilai akhir *userrating* sebuah film. Alasan di balik penggunaan sistem tersebut adalah untuk menghilangkan unsur *ballotstuffing* atau *votestuffing*, yakni pemberian vote berkali-kali oleh satu *user* dengan mengubah-ubah *vote* yang sudah diberikan (karena pada dasarnya ketika satu *user* telah memberi *vote*, maka *vote* tersebut akan tercatat di account IMDB pengguna tersebut walaupun itu masih bisa diubah-ubah di kemudian hari). IMDB weighted rating memiliki persamaan sebagai berikut:

$$\text{Weighted Rating (WR)} = \left(\frac{v}{v+m} \cdot R \right) + \left(\frac{m}{v+m} \cdot C \right) \quad (6)$$

Di mana:

- ⌚ v: jumlah *votes* untuk film tersebut
- ⌚ m: jumlah minimum *votes* yang dibutuhkan supaya dapat masuk dalam *chart* (60% dari total member)
- ⌚ R: rata-rata *rating* dari film tersebut
- ⌚ C: rata-rata jumlah *votes* dari seluruh semesta film (saat ini 6.92)

H. *Confusion Matrix*

Precision digunakan untuk mengukur kinerja sistem. *Precision* adalah kecocokan antara bagian data yang diambil dengan informasi yang dibutuhkan. *Recall* merupakan tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. *Accuracy* adalah tingkat kedekatan antara nilai yang didapat terhadap nilai sebenarnya. *Precision*, *recall* dan *accuracy* dapat dihitung menggunakan *confusionmatrix*. Namun pada penelitian yang dilakukan pengujian sistem yang dilakukan hanya presisi saja. Presisi dapat diketahui menggunakan persamaan berikut:

$$\text{precision} = \frac{TP}{TP + FP} \quad (4)$$

Di mana:

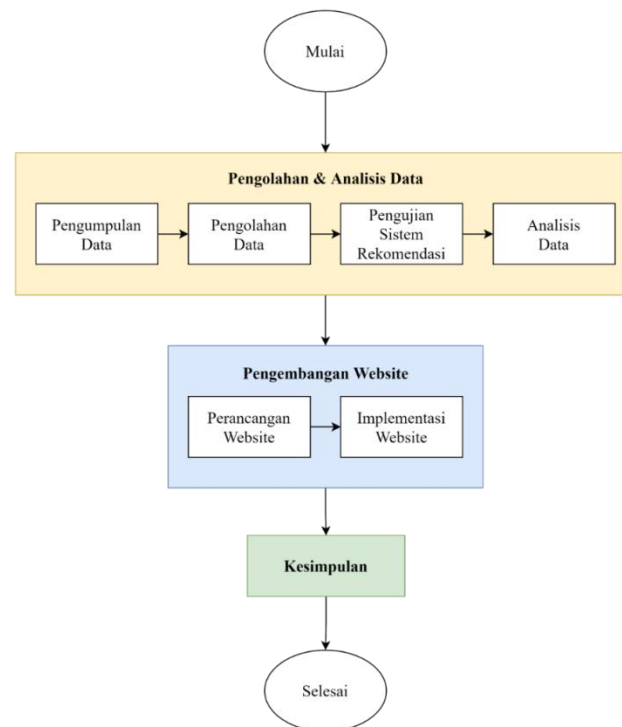
- ⌚ TP = Banyak data dengan nilai sebenarnya positif dan nilai prediksi positif
- ⌚ FP = Banyak data dengan nilai sebenarnya negatif dan nilai prediksi positif

I. *Flask Framework*

Terdapat banyak *framework* yang dapat digunakan untuk membuat website berbasis pythondiantaranya Flask, Django, CherryPiy, dan lain-lain. Pada penelitian ini website akan dibangun menggunakan *framework* Flask. Flask adalah kerangka kerja aplikasi web (*web framework*) yang bersifat micro, istilah ini dipakai karena flask menyediakan pustaka atau alat bantu inti yang digunakan untuk proses membangun sebuah aplikasi web, jadi istilah micro ini bukan berarti bahwa mengembangkan aplikasi web dengan flask hanya dilakukan dalam skala kecil. Apabila ada kebutuhan lain yang dibutuhkan, flask mengizinkan pengguna untuk menambah fitur lain yang spesifik sebagai ekstensi.

III. METODOLOGI PENELITIAN

Secara khusus penelitian akan dilakukan melalui 3 tahap yaitu analisis & pengembangan data, pengembangan *website* dan kesimpulan. Setiap kegiatan memiliki langkah-langkah yaitu pengumpulan data, pengolahan data, pengujian sistem rekomendasi, analisis data, perancangan *website*, implementasi *website*, dan kesimpulan.



Gambar. 3.1. Alur metodologi penelitian

A. Pengumpulan Data

Data yang akan digunakan merupakan data yang telah dikumpulkan dari komunitas serta database anime dan manga daring paling aktif di dunia yaitu *myanimelist.net*. Kumpulan data dapat disebut sebagai dataset. Dataset berisi informasi tentang 17.562 anime dan preferensi dari 325.772 pengguna yang berbeda. Secara khusus, dataset berisi daftar anime, daftar pengguna dan rating anime. Sebagian besar pola pengguna memberi rating tersebut dari 1 sampai dengan 10 sehingga dapat menggambarkan preferensi pengguna terhadap item.

B. Pengolahan Data

Sistem rekomendasi akan dirancang menggunakan diagram blok. Rancangan yang telah dibuat kemudian akan dikembangkan menggunakan bahasa pemrograman python dan menggunakan beberapa *library* atau *toolbox machine learning* diantaranya Scikit, Pandas, Matplotlib dan Surprise.

- 1) Pandas, digunakan untuk membaca data.
- 2) Scikit, digunakan untuk mengolah teks menjadi *vector* atau biasa disebut *vectorizer*, *Vectorizer* yang digunakan yaitu TF-IDF.
- 3) Matplotlib, digunakan untuk menampilkan data dalam bentuk grafik.
- 4) Surprise, digunakan untuk pengolahan data menggunakan SVD.

Sistem rekomendasi yang telah dikembangkan kemudian akan di uji secara manual oleh penulis dengan metode *confusion matrix* untuk mencari nilai *Precision*.

Penelitian diawali dengan memuat data pada aplikasi pengolah data yang akan digunakan yaitu Google Colaboratory. Data yang sudah dimuat selanjutnya akan diolah terlebih dahulu. Sebelum melakukan pengolahan akan dilakukan pemeriksaan data terlebih dahulu untuk menemukan anomali pada data. Anomali pada data dapat berupa informasi yang hilang, duplikat informasi, dan informasi yang inkonsisten.

Proses normalisasi data akan dilakukan apabila terdapat anomaly pada data. Pada kasus informasi yang hilang akan dilakukan perbaikan dengan mengisi kolom kosong dengan 0 atau *null* agar pada saat pengolahan data sistem dapat mengetahui apabila pada *cell* tersebut bernilai 0 untuk tipe data angka dan *null* pada tipe data teks. Informasi yang ganda akan dihilangkan dan informasi yang inkonsisten akan diperbaiki sesuai dengan informasi seharusnya. Proses normalisasi data akan dipastikan jika isi dari informasi aktual tidak akan berubah sehingga hasil penelitian dapat dinyatakan *valid*.

Data yang telah dinormalisasi akan diolah menggunakan metode *hybridfiltering*. Hybridfiltering yang digunakan menggunakan 2 metode sistem rekomendasi yaitu *Content-Based Filtering* dan *Collaborative Filtering*. Masukan yang dibutuhkan yaitu judul film animasi jepang. Keluaran dari *hybridfiltering* berupa rekomendasi daftar film animasi jepang.

C. Pengujian Sistem Rekomendasi

Hasil dari rekomendasi yang telah dihasilkan oleh system akan diuji validitas menggunakan *confusion matrix* untuk mendapatkan nilai *precision*. Nilai *precision* dapat dicari menggunakan persamaan (5).

D. Analisis Data

Analisis data akan dilakukan guna mendapatkan parameter-parameter terbaik yang akan digunakan untuk menyusun siste mrekomendasi seperti pengaruh banyaknya jumlah data pada data latih dan data uji, berapa banyak perbandingan data latih dan data uji agar me ndapatkan MAE dan RMSE terbaik. Perbedaan hasil rekomendasi pada tahap *content-based filtering* dan *collaborative filtering*.

E. Perancangan Website

Sistem rekomendasi yang telah dikembangkan akan diolah menggunakan *framework* Flask. Website yang dibangun sangat sederhana dengan tujuan hanya untuk menampilkan hasil rekomendasi rancangan website. Website yang dibangun hanya memuat halaman awal saja karena tidak terdapat fitur-fitur khusus seperti *sign in*, *signup*, *database* dan sebagainya yang membutuhkan perancangan menggunakan *Unified Modeling Language* (UML).

Masukkan Judul Anime...
Rekomendasi
Item 1
Item 2
Item 3
Item 4
Item 5
Item 6
Item 7
Item 8
Item 9
Item 10

Gambar 3.2. Rancangan halaman awal *website*

F. Implementasi Website

Pengolahan data dilakukan pada google collaborator sedangkan web dibangun menggunakan *framework* Flask. Sistem rekomendasi yang telah dibuat pada google collaborator disatukan menjadi suatu fungsi dengan cara menyatukan seluruh tahap system rekomendasi menjadi satu fungsi system rekomendasi. Pembuatan web dapat dibagi menjadi beberapa tahap yaitu membuat fungsi system rekomendasi, instalasi *environment*, dan membangun web.

IV. HASIL DAN PEMBAHASAN

A. Sistem Rekomendasi

TABEL I
PERBANDINGAN HASIL REKOMENDASI
CONTENT-BASED FILTERING DAN HYBRID FILTERING

Rank	<i>Content-BasedFiltering</i>	<i>HybridFiltering</i>
1	Naruto: Shippuuden	Naruto: Shippuuden
2	Naruto: Shippuuden Movie 1	The Last: Naruto the Movie
3	The Last: Naruto the Movie	Naruto: Shippuuden Movie 6 - Roadto Ninja
4	Boruto: Naruto NextGenerations	Boruto: Naruto the Movie
5	Mayoiga	Naruto: Shippuuden Movie 4 - The Lost Tower
6	Naruto: Shippuuden Movie 6 - Roadto Ninja	Kamisamano Inai Nichiyoubi
7	Boruto: Naruto the Movie	Naruto: Shippuuden Movie 1
8	Naruto Movie 1: Dai Katsugeki!! Yuki HimeShinobuHoujouDattebayo!	Naruto: Shippuuden Movie 2 - Kizuna
9	Naruto: Shippuuden Movie 4 - The Lost Tower	Naruto Movie 1: Dai Katsugeki!! Yuki HimeShinobuHoujouDattebayo!
10	Kamisamano Inai Nichiyoubi	Naruto Movie 2: Dai Gekitotsu! MaboroshinoChiteiisekiDattebayo!
11	Naruto: Shippuuden Movie 2 - Kizuna	Mayoiga
12	Naruto Movie 2: Dai Gekitotsu! MaboroshinoChiteiisekiDattebayo!	Boruto: Naruto NextGenerations

Pengujian dilakukan sebanyak 10 kali dengan judul anime berbeda-beda. Pengujian dilakukan untuk mengukur rekomendasi yang dihasilkan oleh sistem dengan cara menilai hasil rekomendasi yang diberikan sistem lalu dihitung menggunakan persamaan *confusionmatrix*.

Pengujian 1. Keyword “Naruto”

Gambar 4.29 menampilkan hasil rekomendasi *keyword* Naruto dengan hasil rekomendasi yang tepat sebanyak 10 anime yaitu Naruto: Shippuuden, The Last: Naruto the Movie, Naruto: Shippuuden Movie 6 - Roadto Ninja, Boruto: Naruto the Movie, Naruto: Shippuuden Movie 4 - The Lost Tower, Naruto: Shippuuden Movie 1, Naruto: Shippuuden Movie 2 – Kizuna, Naruto Movie 1: Dai Katsugeki!! Yuki HimeShinobuHoujouDattebayo!, Naruto Movie 2: Dai Gekitotsu! MaboroshinoChiteiisekiDattebayo!, Boruto: Naruto NextGenerations. 2 dari 12 rekomendasi yang dihasilkan tidak sesuai yaitu Kamisamano Inai Nichiyoubi, Mayoiga. 10 rekomendasi yang sesuai dengan keyword merupakan *sequel* atau *prequel* dari *keyword* judul anime yang dimasukkan.

Pengujian 2. Keyword “Dragon Ball”

Gambar 4.20 menampilkan 10 judul anime yang sesuai dengan *keyword* yang dimasukkan, sedangkan 2 lainnya tidak memiliki keterkaitan dengan *keyword* yang dimasukkan. Judul yang sesuai diantaranya Dragon Ball Z, Dragon Ball Kai, Dragon Ball Super Movie: Broly, Dragon Ball Z Movie 15: Fukkatsuno "F", Dragon Ball Z Movie 14: Kami to Kami, Dragon Ball GT, Dragon Ball Z Movie 01: Ora noGohanwoKaese!!, Dragon Ball Movie 1: Shen Long noDensetsu, Dragon Ball Super, Dragon Ball Z Movie 12: FukkatsunoFusion!! Gokuuto Vegeta sedangkan rekomendasi yang tidak sesuai yaitu SeikokunoDragonar, Big Order (TV). Judul yang sesuai merupakan sequel atau prequel dari *keyword* terkait namun 2 lainnya tidak memiliki latar yang sama dengan *keyword* yang dimasukkan.

Pengujian 3. Keyword “Doraemon”

Gambar 4.31 menampilkan hasil rekomendasi dari pengujian ke-3 dengan *keyword* Doraemon. Seluruh rekomendasi yang dihasilkan berkaitan dengan *keyword* yang dimasukkan yaitu Stand By Me Doraemon, Doraemon Movie 11: NobitatoAnimal Planet, Doraemon Movie 31: ShinNobitatoTetsujinHeidan - HabatakeTenshi-tachi, Doraemon (1979), Doraemon Movie 15: NobitatoMugenSankenshi, Doraemon Movie 28: Nobitato Midori noKyojin Den, Doraemon Movie 01: NobitanoKyouryuu, Doraemon Movie 13: NobitatoKumonoOukoku, Doraemon Movie 12: NobitanoDorabianNights, Doraemon Movie 19: NobitanoNankaiDaibouken, Doraemon (2005), Doraemon Movie 26: NobitanoKyouryuu 2006. Daftar yang dihasilkan merupakan prequel dan sequel dari judul anime yang dimasukkan.

Pengujian 4. Keyword “CrayonShin-chan”

Gambar 4.32 menampilkan hasil rekomendasi dari pengujian ke-4 dengan *keyword* CrayonShin-chan namun berbeda dengan pengujian sebelumnya pada pengujian ke-4 hasil rekomendasi yang diberikan oleh sistem tidak seakurat pengujian lainnya di mana hanya 5 dari 12 yang terkait dengan *keyword* yang dimasukkan. Adapun rekomendasi yang dihasilkan oleh sistem yaitu Area 88, ShinChuukaIchiban! 2nd Season, HokutonoKen Zero: Kenshirou Den, Jashin-chanDropkick', CrayonShin-chan Movie 06: Dengeki! Buta noHizumeDaisakusen, CrayonShin-chan Movie 12: ArashiwoYobu! YuuhinoKasukabe Boys, Kenjano Mago, Mai-OtomeZweiSpecial, ShinKoihime†Musou: OtomeTairan - Gakuensaidayo! ZeninShuugouno Koto, AshitaSekaigaOwarutoshitemo, KodomonoJikan OVA, Jashin-chanDropkick. 5 judul anime yang terkait dengan crayonshin-chan yaitu Jashin-chanDropkick', CrayonShin-chan Movie 06: Dengeki! Buta noHizumeDaisakusen, CrayonShin-chan Movie 12: ArashiwoYobu! YuuhinoKasukabe Boys, KodomonoJikan OVA, Jashin-chanDropkick di mana terdapat sequel dan prequelcrayonshin-chan dan lainnya memiliki genres atau latar cerita yang sama dengan *keyword* yang dimasukkan.

Rekomendasi yang kurang sesuai pada pengujian ini dikarenakan anime dengan judul CrayonShin-chan tidak memiliki banyak sequel dan prequel seperti judul anime pada pengujian sebelumnya sehingga sistem memberikan rekomendasi hasil yang lain dengan kemiripan terbesar berdasarkan synopsis dan preferensi pengguna lain. Namun dapat dilihat apabila hasil rekomendasi yang diberikan memiliki nilai prediksi yang sangat tinggi yaitu 9.5 dan yang paling rendah adalah 7. Rekomendasi yang diberikan oleh sistem layak untuk dicoba terlebih dahulu.

Pengujian 5. Keyword “DetectiveConan”

Gambar 4.33 menampilkan hasil rekomendasi dari pengujian ke-5 dengan 11 rekomendasi yang sesuai yaitu DetectiveConan Movie 01: The TimedSkyscraper, DetectiveConan Movie 05: Countdownto Heaven, DetectiveConan Movie 13: The Raven Chaser, DetectiveConan Movie 04: Captured in HerEyes, DetectiveConan Movie 10: RequiemoftheDetectives, DetectiveConan Movie 09: StrategyAbovetheDepths, DetectiveConan Movie 02: The Fourteenth Target, DetectiveConan Movie 15: QuarterofSilence, DetectiveConan Movie 11: Jolly Roger in theDeepAzure, DetectiveConan Movie 14: The LostShip in the Sky, DetectiveConan Movie 08: MagicianoftheSilver Sky. Dari rekomendasi yang diberikan terdapat 1 anime yang tidak sesuai yaitu NodameCantabile. NodameCantabile dinominasikan menjadi rekomendasi di nomor 1

dengan pertimbangan preferensi pengguna berdasarkan preferensi pengguna jika judul anime tersebut akan lebih disukai dibandingkan dengan sequel atau prequel dari *keyword* yang dimasukkan.

Pengujian 6. Keyword “One Piece”

Gambar 4.34 menampilkan hasil rekomendasi dari pengujian ke-6 dimana hasil rekomendasi yang diberikan hampir sempurna dengan 11 dari 12 rekomendasi yang diberikan sesuai dengan keyword. 11 judul yang sesuai yaitu Gintama Movie 2: Kanketsu-hen - YorozuyayoEienNare, MouretsuPirates, One Piece Film: Strong World, One Piece Film: Z, One Piece Movie 6: OmatsuriDanshakutoHimitsuno Shima, One Piece 3D2Y: Ace noshiwoKoete! LuffyNakama Tono Chikai, One Piece Movie 1, One Piece Movie 2: Nejimaki-jimanoDaibouken, One Piece Movie 3: Chinjuu-jimanoChopperOukoku, One Piece Movie 14: Stampede, One Piece Film: Strong World Episode 0. Namun terdapat 1 yang tidak sesuai yaitu The Big O.

Pengujian 7. Keyword “FairyTail”

Gambar 4.35 menampilkan hasil rekomendasi dari pengujian ke-7. Hasil rekomendasi yang ditampilkan dapat dikatakan cukup baik karena dari 12 daftar terdapat 6 yang sesuai atau 50% dari total rekomendasi sesuai. Rekomendasi yang sesuai yaitu anime dengan judul FairyTail: Final Series, FairyTail (2014), FairyTail x Rave, FairyTail Movie 1: Hououno Miko, FairyTail OVA, FairyTail Movie 2: Dragon Cry yang merupakan sequel atau prequel dari *keyword*. Sedangkan rekomendasi yang tidak sesuai yaitu KokorogaSakebitagatterunda., 7 Seeds, Kuma Miko, Ano NatsudeMatteru: BokutachiwaKoukouSaigononatsuwoSugoshinagara, Ano NatsudeMatteiru, Kurokono Basket: OshaberiShiyokka, WizardBarristers: BenmashiCecil. Hasil rekomendasi pada pengujian ke-7 ini sama dengan hasil pada penelitian ke-5 dimana *keyword* yang dimasukkan tidak memiliki banyak prequel dan sequel oleh karena itu sistem memberikan rekomendasi judul anime lain yang memiliki nilai kemiripan lebih tinggi dibandingkan yang lainnya berdasarkan synopsis.

Pengujian 8. Keyword “Bleach”

Gambar 4.36 menampilkan hasil rekomendasi dari pengujian ke-8. Terdapat 7 rekomendasi judul yang sesuai dengan keyword yaitu SoulEater, Bleach: Memories in the Rain, Bleach Movie 3: Fadeto Black - Kimino Na woYobuBleach: The SealedSwordFrenzy, Bleach Movie 2: The DiamondDustRebellion - MouHitotsunoHyourinmaru, Bleach Movie 1: MemoriesofNobody, Bleach Movie 4: Jigoku-hen. Sedangkan 5 diantaranya tidak sesuai yaitu IchigoMashimaro, Tokyo MewMew, Yume-iroPâtisserie, JigokuShoujoFutakomori, Colorful (Movie). Kemiripan yang terdapat pada *keyword* dengan hasil rekomendasi yang tidak sesuai yaitu nama pemeran utama pada judul anime yang dijadikan *keyword* yaitu ichigo atau dalam Bahasa Indonesia berarti stroberi. Dapat dilihat daftar rekomendasi yang tidak sesuai berjudul ichigo di mana kemiripan pada sistem rekomendasi akan menilai terdapat kesamaan pada konten kedua anime tersebut.

Pengujian 9. Keyword “Hunter x Hunter”

Gambar 4.37 menampilkan hasil rekomendasi dari pengujian ke-9 yaitu Hunter x Hunter (2011), Angel Heart, Hunter x Hunter: Greed Island Final, Hunter x Hunter: Original Video Animation, Hakuouki Movie 1: Kyoto Ranbu, City Hunter, Iria: Zeiram The Animation, Hunter x Hunter Movie 2: The LastMission, Hunter x Hunter Movie 1: PhantomRouge, Monster Hunter Stories: Ride On, Hunter x Hunter: Greed Island, Vampire Hunter D (2000). Terdapat 7 rekomendasi yang sesuai dengan *keyword* yaitu *sequel* atau *prequel* anime tersebut. Sedangkan yang tidak sesuai memiliki konten kemiripan yang tinggi pada kata *hunter* oleh karena itu sistem memberikan rekomendasi tersebut.

Pengujian 10. Keyword “CaptainTsubasa”

Gambar 4.38 merupakan hasil rekomendasi dari pengujian ke-10 dengan keyword *captaintsubasa* dimana sistem berhasil memberikan rekomendasi yang sesuai sebanyak 7 judul anime yaitu InazumaEleven, InazumaEleven Go, Area noKishi, One Piece: YumenoSoccerOu!, InazumaEleven Go: ChronoStone, CaptainTsubasa (2018), CaptainTsubasa: Roadto 2002. Sedangkan yang tidak sesuai adalah SenkiZesshouSymphogear, SenkiZesshouSymphogear G, B-Project: Kodou*Ambitious, GatchamanCrowdsInsight, Tsubasa: Shunraiki. Adapun rekomendasi yang tidak sesuai didapatkan dari konten dengan kata *tsubasa* yang merupakan pemeran utama sedangkan konten yang sesuai memiliki kemiripan yaitu konten mengenai sepak bola.

VI. Hasil Pengujian

Pengujian yang telah dilakukan dapat diringkas dengan menuliskan hasil rekomendasi yang sesuai dan yang tidak sesuai. Rekomendasi yang sesuai akan disebut *Sebatatruepositives* sedangkan yang tidak sesuai akan disebut *falsepositive*. Pada tabel 2 terdapat kolom presisi yang merupakan hasil persamaan *confusionmatrix* untuk mencari nilai presisi. Nilai TP akan dibagi dengan nilai TP ditambah TF. Hasil dari persamaan tersebut ditampilkan pada kolom presisi yang apabila dijumlahkan dapat dicari rata-ratanya dengan nilai 0,71 atau dapat dikatakan sistem rekomendasi yang dibangun memiliki nilai presisi sebesar 71%.

TABEL II
PERBANDINGAN HASIL REKOMENDASI
CONTENT-BASED FILTERING DAN HYBRID FILTERING

No	Judul Anime	Pengujian		
		TP	FP	Presisi
1	Naruto	10	2	0,833333333
2	Dragon Ball	10	2	0,833333333
3	Doraemon	12	0	1
4	Shin-chan	5	7	0,416666667
5	DetectiveConan	11	1	0,916666667
6	One Piece	11	1	0,916666667
7	FairyTail	6	6	0,5
8	Bleach	7	5	0,583333333
9	Hunter x Hunter	7	5	0,583333333
10	CaptainTsubasa	7	5	0,583333333
Rata-Rata				0,716666667

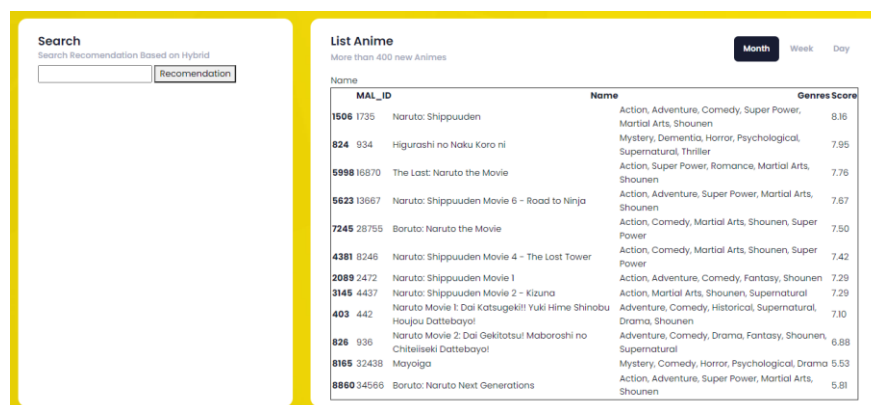
B. Implementasi Website

Website dibangun menggunakan *framework* flask di mana bahasa pemrograman yang digunakan menggunakan python, html, css dan javascript. Pertama dilakukan impor seluruh library yang digunakan, selanjutnya merupakan kerangka flask berupa statement dan routes terdapat fungsi *get* dan *post* guna melakukan *bypass* dari input halaman awal agar dapat diproses oleh sistem. Input yang di *bypass* ke sistem rekomendasi yang telah disusun berada pada file yang berbeda yaitu *hybrid_recommendation_system_with_anime_df.py*.

```
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: on
* Restarting with stat
* Debugger is active!
* Debugger PIN: 646-914-498
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Gambar 4.1 Pemberitahuan *website* berhasil dijalankan pada *localhost*

Flask dapat dijalankan menggunakan *commandline* (CLI) atau *code editor* yang memiliki terminal. Cara untuk menjalankan *framework* flask yaitu dengan mengetik sintak *py app.py*. Sistem akan memberikan keluaran pada terminal seperti yang ditampilkan pada gambar 3. Untuk membuka halaman tersebut dapat dengan mengetikkan alamat ip diikuti port 127.0.0.1:5000. Halaman *website* dibagi menjadi 2 bagian, bagian kiri merupakan tempat pencarian dan bagian kanan merupakan tabel hasil pencarian. Untuk mencari rekomendasi dari suatu judul anime dapat menggunakan kolom disebelah kiri dan menekan enter atau menekan tombol *recommendation*. Sedangkan hasil rekomendasi akan ditampilkan pada tabel yang berada pada bagian kanan seperti yang ditampilkan pada gambar 4.



MAL_ID	Name	Genres	Score
1506 1735	Naruto: Shippuuden	Action, Adventure, Comedy, Super Power, Martial Arts, Shounen	8.16
824 934	Higurashi no Naku Koro ni	Mystery, Dementia, Horror, Psychological, Supernatural, Thriller	7.95
5998 16870	The Last: Naruto the Movie	Action, Super Power, Romance, Martial Arts, Shounen	7.76
5623 13667	Naruto: Shippuuden Movie 6 - Road to Ninja	Action, Adventure, Super Power, Martial Arts, Shounen	7.67
7245 28755	Boruto: Naruto the Movie	Action, Comedy, Martial Arts, Shounen, Super Power	7.50
4381 8248	Naruto: Shippuuden Movie 4 - The Last Tower	Action, Comedy, Martial Arts, Shounen, Super Power	7.42
2089 2472	Naruto: Shippuuden Movie 1	Action, Adventure, Comedy, Fantasy, Shounen	7.29
3145 4437	Naruto Movie 2 - Kizuna	Action, Martial Arts, Shounen, Supernatural	7.29
403 442	Naruto Movie 1: Dai Katsugeki!! Yuki Hime Shinobu Houjou Dattebayo!	Adventure, Comedy, Historical, Supernatural, Drama, Shounen	7.10
826 936	Naruto Movie 2: Dai Gekitotsu! Maboroshi no Chitoseki Dattebayo!	Adventure, Comedy, Drama, Fantasy, Shounen, Supernatural	6.88
8165 32438	Mayoiga	Mystery, Comedy, Horror, Psychological, Drama	5.53
8860 34566	Boruto: Naruto Next Generations	Action, Adventure, Super Power, Martial Arts, Shounen	5.81

Gambar 4.2. Tampilan *website*

V. KESIMPULAN

Penelitian yang dilakukan dapat memenuhi tujuannya itu membangun system rekomen dari menggunakan metode Hybrid Filtering agar dapat memberikan hasil berupa daftar rekomendasi film dan mengetahui rancang bangun dan implementasi system rekomen dari film pada platform *website*. Sistem rekomendasi dipengaruhi oleh banyak faktor, semakin banyak atribut yang digunakan semakin besar perancangan system rekomendasi yang harus disusun, pada penelitian ini hanya digunakan atribut synopsis sebagai konten. Terdapat banyak atribut yang dapat digunakan sebagai konten diantaranya pemain, produser, pengisi suara, studio, genre dan sebagainya.

Metode yang digunakan juga sangat mempengaruhi hasil rekomendasi. Pada tahap content-based filtering dilakukan sortir anime sebanyak 30 judul maka hasil yang akan direkomendasikan tidak akan lebih dari 30 judul. Sedangkan pada tahap Collaborative filtering terdapat tahap weighted rating dimana terdapat batasambang (threshold) yang membatasi anime dengan jumlah penilai dibawah batasambang tidak dapat dijadikan rekomendasi. Pada sistem yang dibangun hasil rekomendasi yang dihasilkan hanya berjumlah 12 judul. Untuk memperbanyak judul rekomendasi dapat dilakukannya saja dari 12 judul yang telah ditampilkan nilai kemiripan terendah yaitu 0,08 atau 8%, di mana 8% sudah dianggap sangat kecil untuk dapat dinyatakan sebagai mirip.

Sistem rekomendasi yang dibangun dapat diimplementasikan dalam bentuk *website* dengan menggunakan *framework* flask. Pengembangan *website* dapat dilakukan dengan berbagai bahasa pemrograman lainnya. *Framework* lainnya yang dapat digunakan yaitu *django* yang merupakan *framework* bahasa pemrograman python juga. Hanya saja pengembangan *website* hanya sampai pada tahap *localhost*.

DAFTAR PUSTAKA

- [1] A S Girsang, B Al Faruq, H R Herlianto, and S Simbolon. (2020). *Collaborative recommendation system in users of anime films*. *Journal of Physics: Conference Series*, 1566:012057.
- [2] Netflix. (2022, Januari 02). Bagaimanacara Netflix memberilisensi acara TV dan film. Tersedia: <https://help.netflix.com/id/node/4976>
- [3] Francesco Ricci, LiorRokach, Bracha Shapira. (2010).*Recommender Systems Handbook.Recommender Systems Handbook*. pp.1-35
- [4] McGinty, L. and Smyth, B. (2006). *Adaptive selection: analysis of critiquing and preference based feedback in conversational recommender systems*. *International Journal of Electronic Commerce*, 11(2), pp 35–57.
- [5] Pazzani, Michael J. & Billsus, Daniel. (2007). *Content-Based Recommendation Systems*. Springer-Verlag Berlin Heidelberg.
- [6] Leimstoll, U. & Stormer, H. (2007). *Collaborative Recommender Systems for Online Shops*. Journal: AMCIS 2007, Keystone, CO.
- [7] Salton, G. and Buckley, C. (1988) *Term-Weighting Approaches in Automatic Text Retrieval*. *Information Processing & Management*, 24, 513-523 (Berger, et al, 2000)
- [8] Y. Koren, R. Bell, and C. Volinsky. *Matrix factorization techniques for recommender systems*. *Computer*, 42(8):30–37, 2009.
- [9] J. Yuan, W. Shalaby, M. Korayem, D. Lin, et al. *Solving cold-start problem in large-scale recommendation engines: A deep learning approach*. (2016) IEEE International Conference on Big Data (Big Data), pages 1901–1910, Los Alamitos, CA, USA.
- [10] Jie Lu, Dianshuang Wu, Mingsong Mao, Wei Wang, and Guangquan Zhang. *Recommender system application developments: A survey*. *Decision Support Systems*, 74:12 – 32, 2015.
- [11] Robin Burke. (2007). *Hybrid Web Recommender Systems*, pages 377–408. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [12] Noor Ifada, Triyani Fatchur Rahman, Mochammad Kautsar Sophan. (2020). *Comparing Collaborative Filtering and Hybrid based Approaches for Movie Recommendation*. *Information Technology International Seminar (ITIS)*.
- [13] Rizki Tri Wahyuni, Dhidik Prastiyanto, dan Eko Suprpto. (2017). Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi. *Jurnal Teknik Elektro* Vol. 9 No. 1.
- [14] Rut Samuel, Ripa Natan, Fitria, Umami Syafiqoh. (2018). Penerapan Cosine Similarity dan K-Nearest Neighbor (K-NN) pada Klasifikasi dan Pencarian Buku. *Journal of Big Data Analytic and Artificial Intelligence*, Vol. 1, No. 1, Tahun 2018.
- [15] Novita Hidayati, Muhammad Ihsan Jambak, Danny Matthew Saputra. (2017). Pengaruh Singular Value Decomposition Terhadap metode-metode Clustering. *Seminar Nasional Teknologi Informasi 2017*.
- [16] Hug, N., (2020). *Surprise: A Python library for recommender systems*. *Journal of Open Source Software*, 5(52), 2174. <https://doi.org/10.21105/joss.02174>
- [17] Aggarwal, C. C., & others. (2016). *Recommender systems (Vol. 1)*. Springer. doi:10.1007/978-3-319-29659-3
- [18] Behnel, S., Bradshaw, R., Citro, C., Dalcin, L., Seljebotn, D. S., & Smith, K. (2011). Cython: The best of both worlds. *Computing in Science & Engineering*, 13(2), 31–39. doi:10.1109/MCSE.2010.118
- [19] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., et al. (2013). *API design for machine learning software: Experiences from the scikit-learn project*. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning* (pp. 108–122).
- [20] Gantner, Z., Rendle, S., Freudenthaler, C., & Schmidt-Thieme, L. (2011). *MyMediaLite: A Free Recommender System Library*. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys 2011)*.

- [21] Guo, G., Zhang, J., Sun, Z., & Yorke-Smith, N. (n.d.). *LibRec: A Java Library for Recommender Systems*.
- [22] Harper, F. M., & Konstan, J. A. (2015). *The MovieLens Datasets: History and Context*. ACM Transactions on Interactive Intelligent Systems (TIIS), 5(4), 1–19.
- [23] Koren, Y., Bell, R., & Volinsky, C. (2009). *Matrix factorization techniques for recommender systems*. Computer, 42(8), 30–37.
- [24] Kula, M. (2015). Metadata Embeddings for User and Item Cold-start Recommendations. In T. Bogers & M. Koolen (Eds.), *Proceedings of the 2nd Workshop on New Trends on Content-Based Recommender Systems co-located with 9th ACM Conference on Recommender Systems (RecSys 2015)*, Vienna, Austria, September 16-20, 2015., CEUR workshop proceedings (Vol. 1448, pp. 14–21). CEUR-WS.org. Retrieved from <http://ceur-ws.org/Vol-1448/paper4.pdf>
- [25] Walt, S. vander, Colbert, S. C., & Varoquaux, G. (2011). *The numpy array: A structure for efficient numerical computation*. Computing in Science & Engineering, 13(2), 22–30. doi:10.1109/MCSE.2011.37
- [26] Lee, D. D., & Seung, H. S. (2001). *Algorithms for non-negative matrix factorization*. In *Advances in Neural Information Processing Systems* (pp. 556–562).
- [27] Yang, L., Bagdasaryan, E., Gruenstein, J., Hsieh, C.-K., & Estrin, D. (2018). *OpenRec: A Modular Framework for Extensible and Adaptable Recommendation Algorithms*. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM '18* (pp. 664–672). New York, NY, USA: Association for Computing Machinery. doi:10.1145/3159652.3159681
- [28] McKinney, W. (2010). *Data Structures for Statistical Computing in Python*. In S. vander Walt & J. Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (pp. 51–56). doi:10.25080/Majora-92bf1922-00a
- [29] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., et al. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825–2830.
- [30] F. O. Isinkaye, Y. O. Folajimi, dan B. A. Ojokoh. (2015). *Recommendation systems: Principles, methods and evaluation*. Egypt. Inform. J., vol. 16, no. 3, hlm. 261–273. doi: 10.1016/j.eij.2015.06.005.
- [31] B. Sarwar, G. Karypis, J. Konstan, dan J. Reidl. (2001). *Item-based collaborative filtering recommendation algorithms*. Proceedings of the tenth international conference on World Wide Web - WWW '01, Hong Kong, Hong Kong. hlm. 285–295, doi: 10.1145/371920.37207. Retrieved: <http://portal.acm.org/citation.cfm?doid=371920.372071>.
- [32] G. Linden, B. Smith, dan J. York. (2003). *Amazon.com recommendations: item-to-item collaborative filterin*. IEEE Internet Comput., vol. 7, no. 1, hlm. 76–80. doi: 10.1109/MIC.2003.1167344.
- [33] Christian Wibison, Lucky Surya Haryadi, Juan Elisha Widyaya, Swat Lie Liliawati. (2021). Sistem Rekomendasi Suku Cadang Berdasarkan *Item Based Filtering*. Jurnal Teknik Informatika dan Sistem Informasi Vol.7 No.1. <http://dx.doi.org/10.28932/jutisi.v7i1.3036>