

Implementasi Metode *Hybrid Filtering (Collaborative dan Content-based)* untuk Sistem Rekomendasi Pariwisata

Implementation of Hybrid Filtering (Collaborative and Content-based) Methods for the Tourism Recommendation System

Yohana Imelda Lubis¹, Dohar Josua Napitupulu², Arie Satia Dharma³

^{1,2,3}Prodi S1 Informatika, Fakultas Informatika dan Teknik Elektro, Institut Teknologi Del
Jln. Sisingamangaraja Sitoluama, Laguboti, Tobasa, Sumatera Utara, Indonesia, 22381

1ifs16005@students.del.ac.id, 2ifs16034@students.del.ac.id,

[3ariesatia@del.ac.id](mailto:ariesatia@del.ac.id)

Abstrak— Kemajuan teknologi informasi menyebabkan pertumbuhan informasi pariwisata berkembang dengan pesat yang mengakibatkan para wisatawan kesulitan menemukan informasi yang relevan sesuai dengan kebutuhan wisatawan. Sistem rekomendasi merupakan salah satu solusi untuk merekomendasikan tempat wisata sesuai dengan preferensi pengguna. Teknik yang sering digunakan adalah *Collaborative filtering* (CF) dan *content-based* (CB). Namun, teknik CB dan CF memiliki kelebihan dan kekurangan. Salah satu kekurangan yang dimiliki yaitu *sparsity data* dimana hanya terdapat beberapa rating dari item yang menyebabkan kekurangan informasi untuk proses rekomendasi dan menyebabkan efektivitas rekomendasi menurun. Pada penelitian ini, metode *hybrid filtering* digunakan membangun sistem rekomendasi yaitu penggabungan metode *content-based* dan *Collaborative filtering* yang diharapkan mampu mengatasi *sparsity data*. Eksperimen dilakukan berdasarkan jenis *item similarity* menggunakan *pearson correlation* dan pemilihan sebanyak k konten pada algoritma k-NN. Setiap eksperimen akan dievaluasi menggunakan persamaan MAE (*Mean absolute error*) dimana hasil eksperimen pada algoritma yang menghasilkan nilai MAE paling rendah akan dipilih untuk memperoleh rekomendasi objek wisata. Hasil akhir dari penelitian ini menunjukkan bahwa metode *hybrid filtering* dapat digunakan untuk membangun sistem rekomendasi objek wisata dengan hasil eksperimen dengan MAE terendah adalah 0,3741 dengan k 25%. Metode *hybrid filtering* terbukti lebih baik dari metode *content-based filtering* yang menghasilkan nilai MAE yaitu 1,174201 dan metode *Collaborative filtering* yang menghasilkan nilai MAE 0,3768 dengan k 25%. Berdasarkan hasil tersebut, pengisian data sparse terbukti efektif berdasarkan nilai MAE yang diperoleh pada metode *hybrid filtering*.

Kata Kunci— sistem rekomendasi; *Collaborative filtering*; *content-based filtering*; k-NN; *hybrid filtering*; *pearson correlation*

Abstract— Advances in information technology have led to the rapid growth of tourism information which has made it difficult for tourists to find information relevant to the needs of tourists. The recommendation system is one solution to recommend tourist attractions according to user preferences. The techniques often used are *Collaborative filtering* (CF) and *content-based* (CB). However, CB and CF techniques have advantages and disadvantages. One of the shortcomings is *sparsity data* where there are only a few rating of items that

causes a lack of information for the recommendation process and causes the effectiveness of recommendations to decrease. In this study, the hybrid filtering method is used to build a recommendation system that is the merging of content-based and Collaborative filtering methods that are expected to overcome sparsity data. Experiments were carried out based on the type of similarity items using the Pearson correlation and the selection of k content in the k-NN algorithm. Each experiment will be evaluated using the MAE (Mean absolute error) equation where the experimental results on the algorithm that produces the lowest MAE value will be selected to obtain a tourist attraction recommendation. The final results of this study indicate that the hybrid filtering method can be used to build a tourist recommendation system with the lowest experimental yield of MAE being 0.3741 with k 25%. The hybrid filtering method is proven to be better than the content-based filtering method that produces a MAE value of 1.174201 and a Collaborative filtering method that produces a MAE value of 0.3768 with a 25% k. Based on these results, sparse data filling proved to be effective based on the MAE values obtained on the hybrid filtering method.

Keywords—recommendation system; *Collaborative filtering*; *content-based filtering*; k-NN; *hybrid filtering*; *pearson correlation*

I. PENDAHULUAN

Pariwisata merupakan sektor yang menjanjikan untuk peningkatan nilai tambah ekonomi terhadap sebuah produk khususnya aset kepariwisataan nasional baik alam, budaya maupun khusus/buatan. Dampak langsung pengembangan pariwisata adalah peningkatan kunjungan wisatawan yang dapat dilihat pada peningkatan penerimaan devisa negara. Berdasarkan data Kementerian Pariwisata menjelaskan devisa dari pariwisata meningkat secara konsisten [1].

Objek wisata dapat berupa wisata alam seperti gunung, sungai, laut, danau, pantai atau berupa objek bangunan seperti museum, benteng, situs peninggalan sejarah, dan lain-lain. Seiring dengan perkembangan pariwisata, Informasi mengenai pariwisata juga berkembang dengan pesat.. Ada banyak informasi yang menyediakan tempat wisata untuk wisatawan. Namun, kendala yang dihadapi yaitu sulitnya menemukan informasi tempat wisata yang tepat bagi pengguna. Oleh karena itu dibutuhkan informasi yang relevan untuk wisatawan dalam memperoleh tempat

wisata yang dibutuhkannya. Dengan masalah ini, untuk meningkatkan kepuasan wisatawan dan sekaligus meningkatkan pariwisata, maka dibutuhkan sebuah sistem rekomendasi yang berfungsi untuk menawarkan serta merekomendasikan tempat wisata sesuai dengan preferensi pengguna.

Sistem rekomendasi mampu menangani permasalahan kelebihan informasi bagi pengguna dengan memberikan layanan rekomendasi berupa konten atau *item* sesuai personalisasi pengguna [2]. Dalam pembuatan sistem rekomendasi terdapat berbagai macam metode yang dapat digunakan. Metode yang digunakan dalam penelitian ini adalah *Collaborative filtering* (CF) dan *content-based* (CB). Metode *Collaborative filtering* adalah yang paling sering digunakan dalam *recommender system*. Rekomendasi yang dihasilkan dengan metode ini didasarkan pada *rating* oleh *user* yang memiliki preferensi yang mirip [3].

Metode *Collaborative filtering* tidak memberikan rekomendasi berdasarkan analisa dari konten-konten yang ada, yang biasanya berupa informasi *item* ataupun profil pengguna. Namun, terbatasnya jumlah pendapat dari beberapa *user* tentang beberapa produk menyebabkan *sparsity problem*, yang mengakibatkan rekomendasi yang buruk. Metode *Collaborative filtering* juga mengalami masalah *cold start problem* dimana ketika terdapat *item* atau pengguna baru, *item* tersebut tidak dapat langsung dijadikan rekomendasi karena harus menunggu pengguna untuk memberi *rating* terlebih dahulu [4].

Metode *content-based filtering* merupakan metode rekomendasi yang didasarkan pada data yang terkait dengan fitur *item*. *Content* yang dimaksud disini adalah deskripsi. Di dalam *content-based filtering*, *rating* dan *behavior* dari *user* dikombinasikan dengan informasi konten yang tersedia pada *item*. Atribut dari *item* yang pernah berinteraksi dengan *user* akan dipakai untuk menemukan *item* sejenis yang memiliki atribut yang mirip ataupun serupa untuk dijadikan rekomendasi [5]. Metode ini memiliki kelemahan, seperti merekomendasikan sesuatu yang tidak terduga atau yang disebut *serendipity problem* dan tidak dapat memberikan informasi kepada pengguna baru yang belum pernah melakukan aktivitas apapun dan tidak memiliki profil *user* yang cukup (*Cold Start Problem*).

Untuk menutupi kelemahan pada metode-metode tersebut dapat dilakukan dengan menggunakan metode *hybrid filtering* yaitu menggabungkan metode *Collaborative filtering* (CF) dan *content-based* (CB) untuk menghasilkan *item* rekomendasi yang sesuai dengan keinginan pengguna yang menangani masalah *sparsity* dan meningkatkan akurasi nilai prediksi. Untuk mendukung cara kerja metode tersebut maka diperlukan algoritma yang akan mendukung kemampuan sistem rekomendasi dalam memberikan informasi yang sesuai yaitu algoritma k-NN [4].

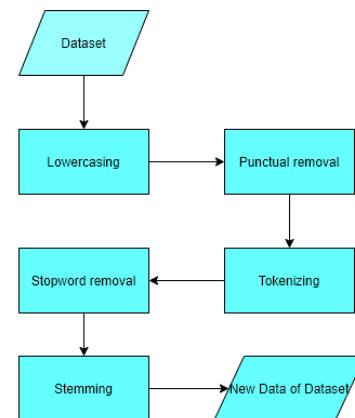
Dalam metode k-NN, *rating* yang diberikan oleh pengguna langsung disimpan oleh sistem dan kemudian digunakan untuk memprediksi *rating* untuk *item* baru [6]. Menurut penelitiannya, Kleef menyatakan bahwa algoritma k-Nearest Neighbors (k-NN) mudah untuk diimplementasikan daripada kebanyakan algoritma

rekомendasi lainnya, cara kerja matematika yang mendasarinya mudah untuk dipahami dan berskala baik terhadap konten yang telah dinilai berulang kali oleh beberapa pengguna. Salah satu langkah penting pada algoritma k-NN yaitu mengukur kemiripan antar konten (*item similarity*) dan kemudian memilih beberapa konten yang paling mirip dengan konten yang akan dilakukan prediksi *rating* [7]. Suatu konten yang memiliki nilai kemiripan yang lebih tinggi dengan konten lain mampu berdampak pada kualitas prediksi *rating* konten lebih tinggi [8]. Atas dasar pada pengerjaan makalah ilmiah ini, diharapkan mampu memperoleh rekomendasi tempat wisata bagi pengguna berdasarkan hasil dari evaluasi model algoritma yang memiliki nilai *error* yang paling kecil.

II. METODE PENELITIAN

A. Preprocessing Data

Dataset pada penelitian ini diperoleh dari website resmi Tripadvisor.com dan lonelyplanet.com. Tripadvisor dan Lonelyplanet adalah salah satu sumber informasi wisata yang terbesar dan terpercaya di dunia, dimana pengunjung dapat menemukan ulasan tentang hotel, restoran, dan tempat tujuan wisata lainnya. Pengguna yang ada pada data berjumlah 362, yang berarti akan ada rekomendasi kepada 0 sampai 361 dengan *rating* berjumlah 11474. Jumlah objek wisata pada data adalah 358 objek wisata. Preprocessing bertujuan untuk mengurangi volume dari kosakata serta menyeragamkan bentuk dari kata sehingga data lebih terstruktur. Untuk memberikan rekomendasi tempat wisata kepada pengguna, tempat wisata tersebut akan dilihat keterkaitan atau kemiripannya dengan tempat wisata lain berdasarkan deskripsi dari tempat wisata tersebut. Data deskripsi tempat wisata yang diperoleh belum sepenuhnya siap digunakan untuk proses pembuatan rekomendasi, untuk itu dibutuhkan *data preprocessing*. Proses *data processing* pada deskripsi tempat wisata akan menghasilkan karakteristik tempat wisata yang akan menjadi inputan pada teknik *content-based filtering* untuk menghasilkan rekomendasi.



Gambar 1. Tahapan Preprocessing Data

- 1) *Lowercasing*: *Lowercasing* digunakan untuk mengubah seluruh teks dalam deskripsi tempat wisata menjadi huruf kecil secara keseluruhan
- 2) *Punctual removal*: *punctuational removal* memproses noise berupa karakter yang tidak berpengaruh dan tidak mempunyai arti seperti tanda baca, simbol dan karakter yang tidak dikenali seperti atribut yang tidak penting karakter simbol (~!@#\$%^&*()_+?:{}[]])
- 3) *Tokenizing*: *Tokenizing* akan memisahkan kalimat ke dalam satuan kata berdasarkan karakter penghubung kata yaitu enter, spasi dan tanda baca.
- 4) *Stopword removal*: *Stopword removal* menghapus kata penghubung yang tidak memiliki arti seperti kata penghubung untuk mengurangi kata pembanding antar kata dalam deskripsi pariwisata.
- 5) *Stemming*: *Stemming* merupakan proses mengekstraksi kata yang dihasilkan dari tahap stopword menjadi kata dasarnya. Hasil dari tahapan ini adalah kata sifat, kata kerja dasar dan kata benda tunggal yang tidak memiliki kata dasar yang sama

B. Content-based Filtering

Content-based Recommender system, *attribute* dari *item* digunakan untuk membuat rekomendasi. *content* yang dimaksud disini adalah deskripsi. Di dalam *content-based*, *rating* dan *behavior* dari *user* dikombinasikan dengan informasi konten yang tersedia pada *item*. Atribut dari *item* yang pernah berinteraksi dengan *user* akan dipakai untuk menemukan *item* sejenis yang memiliki atribut yang mirip ataupun serupa untuk dijadikan rekomendasi [5]. Untuk menghitung bobot dari masing-masing atribut dapat dengan menggunakan rumus berikut [9].

$$W(u, j_k) = \frac{1}{|I_j|} \sum_{i \in I_u} x(i, j) r(u, i) \quad (1)$$

$w(u, j_k)$: Bobot yang dimiliki oleh pengguna u terhadap fitur j_k

I_u : Satu set *item* yang telah *dirating* oleh pengguna u

$x(i, j)$: Jumlah kata dalam dokumen j

$r(u, i)$: *rating* yang diberikan pengguna u terhadap *item* i

Untuk menghitung prediksi *rating* yang akan diberikan seorang pengguna terhadap sebuah *item* dapat menggunakan formula berikut ini [9].

$$R'(u, i) = \frac{1}{|D_i|} \sum_{j \in D_i} w(u, j) \quad (2)$$

$R'(u, i)$: Prediksi *rating* pengguna u terhadap *item* i

D_i : Fitur yang muncul di dalam *item* i

$w(u, j)$: Bobot yang dimiliki oleh pengguna u terhadap fitur j_k

Salah satu masalah pada teknik rekomendasi *Collaborative filtering* (CF) yang terjadi yaitu *data sparsity* yang diakibatkan jika kurangnya informasi ketika hanya beberapa dari jumlah total *item* yang tersedia dan telah diakses oleh pengguna. Tanpa *rating* yang cukup,

efektivitas prediksi kepada pengguna akan sangat berkurang. *Data sparsity* juga menciptakan tantangan untuk perhitungan kesamaan *item* yang kuat ketika jumlah *item* yang dinilai bersama antara dua pengguna tersedia sedikit [5]. Pada Gambar 2 hanya terdapat beberapa *item* yang telah *dirating* oleh *user* (NaN menunjukkan *item* yang belum *dirating* oleh *user*).

| destination_id | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | ... | 349 | 350 | 351 | 352 |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| user_id | 0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | ... | NaN | NaN | NaN | NaN |
| 1 | 1.0 | 1.0 | NaN | NaN | NaN | NaN | NaN | 1.0 | NaN | NaN | ... | NaN | NaN | NaN | NaN |
| 2 | NaN | NaN | NaN | NaN | 2.0 | 2.0 | NaN | NaN | NaN | ... | NaN | 1.0 | 1.0 | NaN | NaN |
| 3 | NaN | NaN | NaN | 1.0 | 1.0 | 1.0 | NaN | NaN | NaN | ... | NaN | NaN | NaN | NaN | NaN |
| 4 | NaN | ... | NaN | NaN | NaN | NaN | NaN |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 357 | 1.0 | NaN | NaN | NaN | NaN | NaN | NaN | 1.0 | 1.0 | NaN | ... | NaN | NaN | NaN | NaN |
| 358 | NaN | ... | NaN | NaN | NaN | NaN | NaN |
| 359 | NaN | ... | NaN | NaN | NaN | NaN | NaN |
| 360 | NaN | ... | NaN | NaN | NaN | NaN | NaN |
| 361 | NaN | NaN | NaN | 1.0 | NaN | NaN | NaN | NaN | NaN | ... | NaN | NaN | NaN | NaN | NaN |

Gambar 2. Sparse Rating

Untuk mengatasi masalah ini, hasil prediksi *rating* menggunakan metode *content-based filtering* digunakan untuk mengisi *rating sparse* yang ada pada dataset sehingga data *rating* yang telah diisi menggunakan prediksi *rating* pada metode *content-based filtering* akan digunakan dalam metode *collaborative filtering* untuk memprediksi *item* yang akan direkomendasikan kepada *user*. Berikut dataset *sparse* yang telah diisi menggunakan prediksi *rating* pada metode *content-based filtering*.

| destination_id | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------------|----------|----------|----------|----------|----------|----------|----------|----------|
| user_id | 0 | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 1.000000 |
| 1 | 1.000000 | 1.000000 | 0.047548 | 0.081621 | 0.077209 | 0.045328 | 0.045838 | 1.000000 |
| 2 | 0.129425 | 0.122222 | 0.100845 | 0.104833 | 2.000000 | 2.000000 | 0.095590 | 0.101018 |
| 3 | 0.099179 | 0.098422 | 0.079379 | 0.093842 | 1.000000 | 0.079175 | 1.000000 | 0.093781 |
| 4 | 0.087002 | 0.094255 | 0.071316 | 0.092448 | 0.082138 | 0.068243 | 0.078277 | 0.082087 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 357 | 1.000000 | 0.089318 | 0.070178 | 0.079104 | 0.071439 | 0.082329 | 0.068584 | 1.000000 |
| 358 | 0.171125 | 0.166685 | 0.160322 | 0.142385 | 0.113442 | 0.116086 | 0.140102 | 0.143326 |
| 359 | 0.122020 | 0.115582 | 0.095820 | 0.094623 | 0.112278 | 0.115243 | 0.081456 | 0.103675 |
| 360 | 0.114880 | 0.136773 | 0.110437 | 0.105016 | 0.109178 | 0.101941 | 0.103507 | 0.140982 |
| 361 | 0.112366 | 0.114575 | 0.092559 | 0.105486 | 1.000000 | 0.097580 | 0.092472 | 0.139096 |

Gambar 3. Hasil Pengisian Sparse Rating

C. TF-IDF

TF-IDF banyak digunakan dalam *content-based filtering*. Dalam penelitian kali ini TF – IDF digunakan untuk membangun *profil* untuk *item* dalam *content-based filtering* [10]. TF (*Term Frequency*) digunakan untuk mengetahui frekuensi dari kata dalam seberkas dokumen, jika frekuensi kata tinggi maka dapat disimpulkan bahwa kata tersebut penting dan dapat digunakan untuk membangun *profil* dari *item* [11].

$$TF_{ij} = \frac{f_{ij}}{\max_k f_{kj}} \quad (3)$$

TF_{ij} : *Term Frequency* kata i dalam dokumen j
 f_{ij} : Frekuensi kemunculan i dalam dokumen j
 $\max_k f_{kj}$: Jumlah kata dalam dokumen j

Namun frekuensi juga kadang belum menggambarkan bahwa sebuah kata itu merupakan kata penting misalnya kata “The” yang jumlahnya bisa sangat banyak didalam suatu dokumen tetapi tidak memiliki arti apa – apa, IDF

adalah keseluruhan frekuensi dokumen dari antar seluruh *corpus* dari dokumen.

$$IDF_i = \log \frac{N}{n_i} \quad (4)$$

- IDF_i : Frekuensi kata i dalam *corpus*
- N : Jumlah total keseluruhan dokumen dalam *corpus*
- n_i : Jumlah dokumen yang mengandung fitur (kata) i

Sehingga Pembobotan TF – IDF dapat disimpulkan untuk meniadakan efek dari kata dengan jumlah frekuensi tinggi dalam menentukan pentingnya suatu fitur. Profil dari suatu dokumen adalah kumpulan fitur hasil komputasi dari TF-IDF *score* yang tertinggi untuk keseluruhan fitur dalam dokumen.

$$\text{TF-IDF score : } w_{ij} = TF_{ij} \times IDF_i \quad (5)$$

D. Collaborative Filtering

Collaborative Filtering Recommender system menggunakan kemampuan kolaboratif dari *rating* yang diberikan oleh banyak *user* untuk membuat rekomendasi. Pada proses *Collaborative filtering*, terdapat 3 tahapan untuk memperoleh nilai *similarity* dan *item* hasil rekomendasi. Tahap pertama yaitu tahap awal yang berisi persiapan data yang diperlukan yaitu parameter *userID*, *itemID* dan *rating*. Selanjutnya akan dipilih pengguna yang akan diberi rekomendasi dan konten yang belum pernah diakses oleh *user*. Tahapan kedua yaitu perhitungan prediksi konten yang berisi perhitungan *item similarity* atau kesamaan antara semua konten dengan konten yang ingin dilakukan prediksi *rating*, kemudian hasil perhitungan tersebut akan diambil beberapa konten terpilih yang memiliki nilai *item similarity* terbesar sebanyak ‘ k ’ dan konten yang terpilih akan menjadi bagian untuk memprediksi *rating item*. Dengan nilai *item similarity* tertinggi dari *item* yang terpilih akan digunakan pada tahap prediksi *rating*. Tahapan ketiga yaitu pemberian rekomendasi konten sesuai hasil prediksi *rating item* dengan nilai tertinggi.

E. Pearson Correlation

Metode *Pearson Correlation Coefficient* merupakan metode yang paling sering digunakan untuk menghitung kemiripan antara objek-objek berdasarkan perhitungan korelasi linear antara dua himpunan yang berbeda. Hasil pengukuran *pearson correlation* berupa klasifikasi kriteria *similarity* dengan rentang nilai -1 (sangat tidak mirip) sampai 1 (sangat mirip) dengan 0 sebagai nilai tengah (netral) dimana *rating* yang akan dihitung kemiripannya akan dinormalisasi terlebih dahulu menggunakan *mean centering normalization*.

$$sim(i,j) = corr_{ij} = \frac{\sum_{j=1}^k (R_{ui} - \bar{R}_i)(R_{uj} - \bar{R}_j)}{\sqrt{\sum_{j=1}^k (R_{ui} - \bar{R}_i)^2 \sum_{j=1}^k (R_{uj} - \bar{R}_j)^2}} \quad (6)$$

- $sim(i,j)$: Kemiripan *item i* dan *item j*
- R_{ui} : Rating yang pernah diberikan pengguna *u* terhadap *item i*
- \bar{R}_i : Rata-rata rating dari *item i*

- R_{uj} : Rating yang diberikan pengguna *u* terhadap *item j*
- \bar{R}_j : Rata-rata rating dari *item j*

F. Algoritma k-Nearest Neighbors

Perhitungan prediksi digunakan untuk memprediksi nilai *rating* yang diberikan oleh *user* untuk *item* tertentu. Salah satu metode dalam teknik *Collaborative filtering* yang mampu memberikan rekomendasi yang akurat dan dipersonalisasi adalah metode neighborhood-based atau sering disebut K-Nearest Neighbor. Dalam metode neighborhood-based, *rating* yang diberikan oleh pengguna langsung disimpan oleh sistem dan kemudian digunakan untuk memprediksi *rating* untuk *item* baru. Untuk prediksi *rating* konten, diawali dengan perhitungan kesamaan antara dua konten (*item similarity*) yang disimbolkan dengan $sim(i,j)$ dan memilih sejumlah *item* (disimbolkan dengan k) tetangga terdekat, kemudian menghitung prediksi nilai *rating*. Pada penelitian ini, untuk mengukur *item similarity* akan menggunakan persamaan pearson correlation. Adapun bentuk persamaan algoritma k-NN adalah sebagai berikut:

$$p_{u,i} = \frac{\sum_{j=1}^k (sim(i,j) \times R_{uj})}{\sum_{j=1}^k (sim(i,j))} \quad (7)$$

- $p_{u,i}$: Prediksi *rating* untuk *user u*
- $sim(i,j)$: Kemiripan *item i* dan *item j*
- R_{uj} : Rating yang diberikan pengguna *u* terhadap *item j*

Evaluasi yang dilakukan di algoritma k-NN berdasarkan eksperimen yang dilakukan antara lain nilai *item* pemilihan k konten sebanyak 100%, pemilihan k konten sebanyak 75%, pemilihan k konten sebanyak 50% dan pemilihan k konten sebanyak 25%. Semua faktor pertimbangan akan dieksperimenkan terhadap beberapa kondisi pembagian data *training* dan data *test* antara lain 60:40, 75:25, 80:20, 85:15 dan 90:10. Keempat eksperimen akan membentuk model-model algoritma yang menghasilkan nilai *error* berdasarkan kondisi faktor pertimbangan dan pembagian data.

G. Hybrid Filtering

Menurut Burke, *Hybrid recommender system* adalah metode yang mengkombinasikan dua atau lebih teknik rekomendasi untuk meningkatkan performansi rekomendasi, yang biasanya digunakan untuk memecahkan masalah yang ada di masing-masing metode yang digunakan [4]. Untuk membuat sistem *hybrid* terdapat 7 teknik yang bisa dipakai, yaitu *weighted*, *switching*, *mixed*, *feature combination*, *feature augmentation*, *cascade* dan *meta-level* [4]. Dalam bukunya, Aggarwal mengelompokan teknik *hybrid* menjadi 3, yaitu *monolithic*, yang meliputi *combination* dan *meta-level*, kemudian *ensemble design*, yang meliputi *feature augmentation*, *cascade*, *weighted*, dan *switching*, dan yang terakhir *mixed system* yang berdiri sendiri dan tidak dikategorikan sebagai *monolithic* ataupun *ensemble* [5].

Dalam penelitian ini, metode *hybrid filtering* menggunakan penggabungan metode *content-based filtering* dan *collaborative filtering* yang diharapkan mampu menyelesaikan masalah *sparsity*. Metode *content-based filtering* digunakan untuk melakukan prediksi *rating* yang digunakan untuk pengisian data *rating sparse*. Kemudian data *rating sparse* yang telah diisi digunakan kembali dalam metode *collaborative filtering* untuk perhitungan prediksi *rating* menggunakan algoritma k-NN untuk memberikan rekomendasi item kepada *user*. Metode *hybrid* yang digunakan dengan tipe *feature augmentation*.

H. Evaluasi Algoritma

MAE (*Mean absolute error*) adalah formula yang digunakan untuk menghitung tingkat akurasi atau besar *error* untuk menghitung prediksi *rating* dari sistem terhadap *rating* yang sebenarnya yang *user* berikan terhadap suatu sistem. MAE diperoleh dengan menghitung *error* absolut dari N pasang *rating* asli dan prediksi, kemudian akan dihitung rata-ratanya. Nilai MAE yang semakin mendekati 0 maka hasil prediksi akan semakin baik.

$$MAE_i = \frac{1}{n} \sum_{i \in E} (|pi - ri|) \quad (8)$$

- | | | |
|---------|---|--|
| MAE_i | : | Model evaluasi algoritma |
| ri | : | Nilai <i>rating</i> yang sebenarnya |
| pi | : | Nilai prediksi <i>rating</i> |
| n | : | Banyaknya pasang <i>rating</i> asli dan prediksi |

MAE pada penelitian kali ini diaplikasikan untuk menghitung performansi dari *hybrid recommender system*. Hasil MAE terendah akan dipilih untuk memperoleh rekomendasi objek wisata berdasarkan data pada dataset.

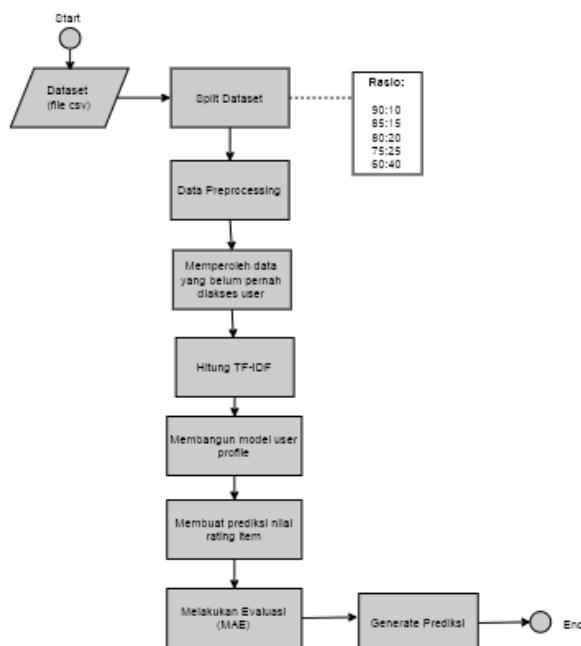
III. HASIL DAN PEMBAHASAN

Pada bab ini dijelaskan mengenai hasil dan pembahasan dari implementasi yang telah dilakukan pada penelitian ini. Implementasi yang dilakukan terhadap algoritma dengan beberapa eksperimen. Hasil dari eksperimen terhadap algoritma akan memperoleh nilai *error* pada model algoritma. Model algoritma dengan nilai *error*

terendah terpilih untuk menghasilkan rekomendasi objek wisata kepada *user*.

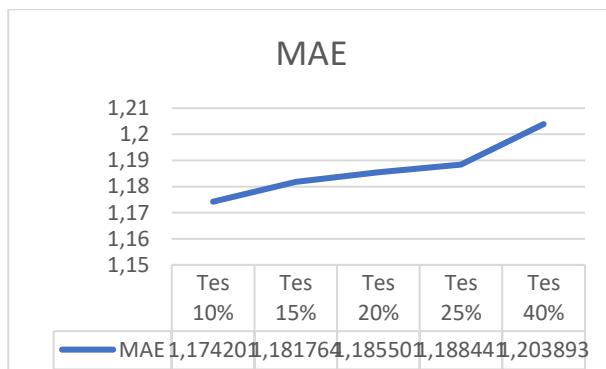
A. Content-based Filtering

Tahapan-tahapan pada metode *content-based filtering* dapat dilihat pada flow berikut:



Gambar 4. Desain Percobaan *Content-based Filtering*

Pada Gambar 4, dilakukan percobaan berdasarkan pembagian (*split*) data yaitu data *training* dan data *test*. *Data training* dibuat sebagai data latih sebagai faktor pertimbangan untuk *data testing* untuk dilakukan prediksi. Data hasil prediksi akan dihitung akurasinya menggunakan persamaan MAE untuk memperoleh perbandingan data aktual dengan data prediksi. Data akan dibagi menjadi rasio 90:10, 80:20, 85:15, 75:25, dan 60:40. Berikut hasil eksperimen metode *content-based filtering*:



Gambar 5. Grafik Eksperimen *Content-based Filtering*

Berdasarkan Gambar 5, diperoleh nilai *error* terendah dari setiap eksperimen. Pada eksperimen yang dilakukan, diperoleh nilai *error* terendah pada pembagian data 90:10. Berikut nilai *error* terendah eksperimen pada *content-based filtering*.

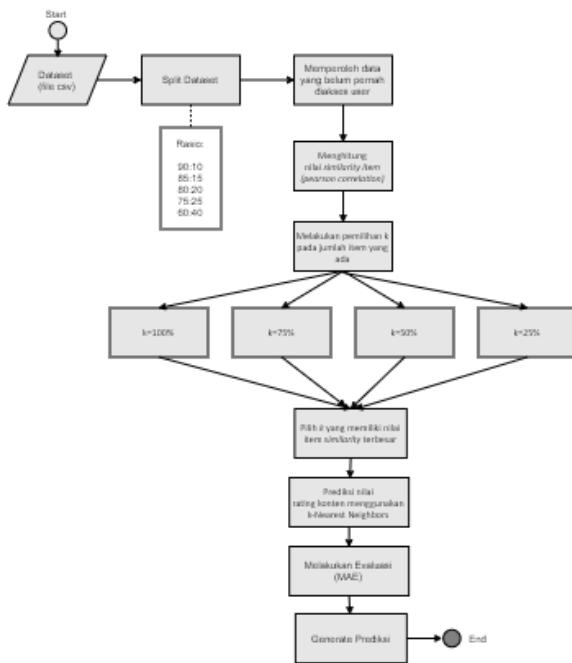
Tabel 1. Hasil Eksperimen *Content-based Filtering*

| Rasio Pembagian Data | Nilai MAE |
|----------------------|-----------|
| 90:10 | 1,174201 |
| 85:15 | 1,181764 |
| 80:20 | 1,185501 |
| 75:25 | 1,188441 |
| 60:40 | 1,203893 |

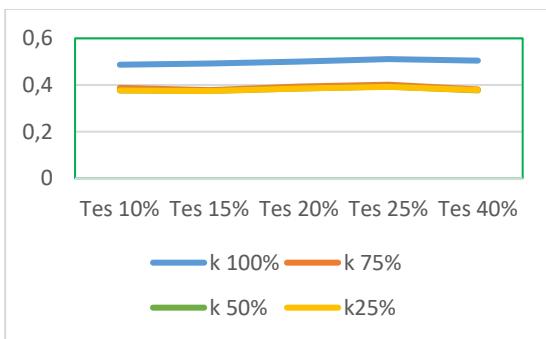
Berdasarkan Tabel 1, nilai *error* terendah di pembagian data 90:10 dengan nilai 1,174201.

B. Collaborative Filtering

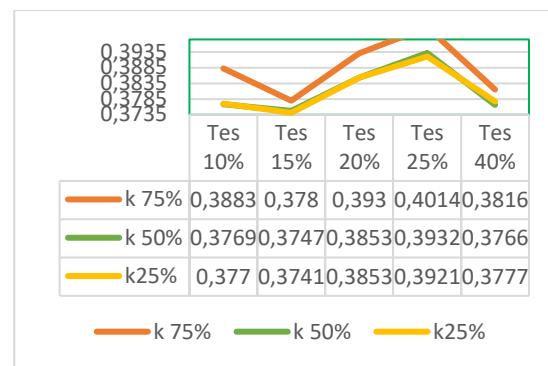
Tahapan-tahapan pada metode *Collaborative filtering* dapat dilihat pada *flow* berikut:

Gambar 6. Desain Percobaan *Collaborative Filtering*

Pada Gambar 6, dilakukan percobaan berdasarkan pembagian (*split*) data yaitu data *training* dan data *test*. Data hasil prediksi akan dihitung akurasinya menggunakan persamaan MAE untuk memperoleh perbandingan data aktual dengan data prediksi. Data akan dibagi menjadi rasio 90:10, 80:20, 85:15, 75:25, dan 60:40. Berikut hasil eksperimen metode *Collaborative filtering*:



Gambar 7. Grafik Hasil Eksperimen pada Algoritma k-NN



Gambar 8. Grafik Hasil Eksperimen pada Algoritma k-NN Skala Grafik Diperkecil

Berdasarkan Gambar 8, skala grafik diperkecil sehingga menunjukkan perolehan hasil MAE terkecil pada setiap eksperimen dapat dilihat bahwa eksperimen yang dilakukan menghasilkan nilai MAE terendah pada rasio pembagian data 85:15 dengan k 25%. Dapat disimpulkan bahwa model algoritma pada eksperimen akan lebih baik untuk memprediksi *rating* suatu konten jika memiliki lebih banyak data *training* dalam membentuk model algoritma. Berikut nilai *error* terendah eksperimen pada *Collaborative filtering*:

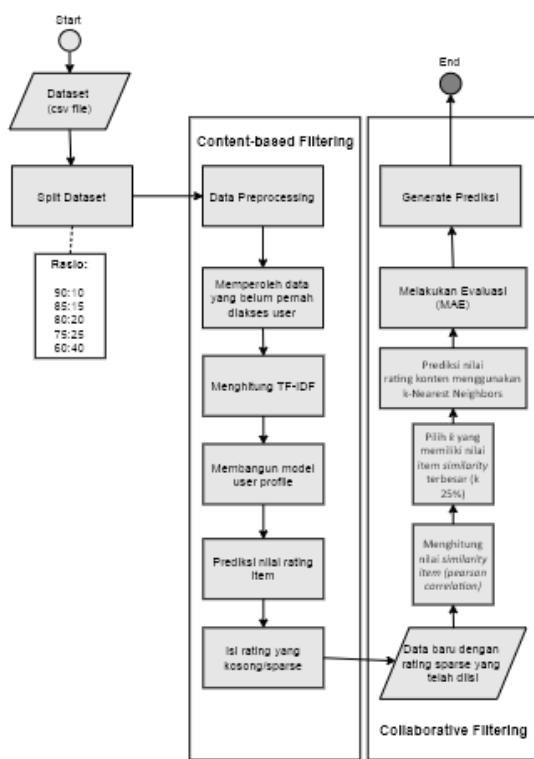
Tabel 2. Hasil Eksperimen *Collaborative Filtering*

| Faktor Pertimbangan | Pembagian Data | Nilai MAE |
|---------------------|----------------|-----------|
| Pemakaian k 100% | 90:10 | 0,487052 |
| Pemakaian k 75% | 85:15 | 0,377969 |
| Pemakaian k 50% | 85:15 | 0,374727 |
| Pemakaian k 25% | 85:15 | 0,374104 |

Berdasarkan Tabel 2, diperoleh eksperimen dengan faktor pertimbangan *k* 25% dengan rasio pembagian data 85:15 memiliki model algoritma terbaik berdasarkan hasil evaluasi nilai MAE terendah dengan nilai 0,3768.

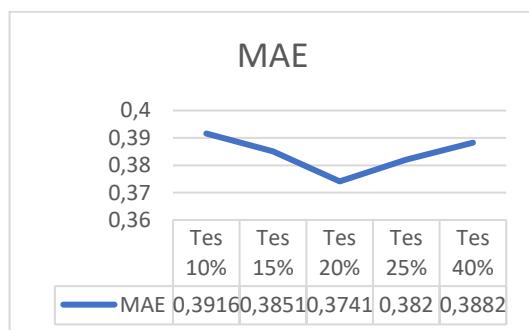
C. Hybrid Filtering

Tahapan-tahapan pada metode *hybrid filtering* dapat dilihat pada *flow* berikut:



Gambar 9. Desain Eksperimen Hybrid Filtering

Pada *hybrid filtering* dilakukan percobaan berdasarkan pembagian (*split*) data yaitu data *training* dan data *test*. Hasil eksperimen teknik *hybrid filtering* yang dilakukan dengan pembagian *dataset* 90:10, 80:20, 85:15, 75:25, dan 60:40 terhadap empat faktor pertimbangan. Berikut hasil nilai MAE yang diperoleh pada eksperimen pertama yaitu memperhitungkan faktor *k* konten terpilih sebanyak 25% terhadap setiap rasio pembagian data.



Gambar 10. Grafik Hasil Eksperimen Hybrid Filtering

Berdasarkan Gambar 10, telah diperoleh nilai MAE pada setiap rasio pembagian data untuk faktor yang mempertimbangkan *k* konten terpilih sebanyak 25%.

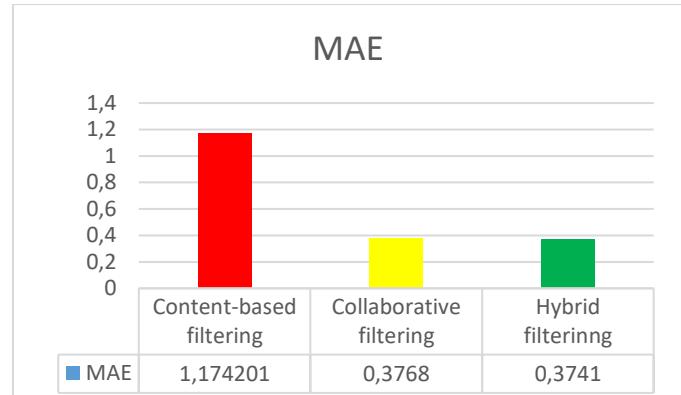
Tabel 3. Hasil Eksperimen Hybrid Filtering

| Rasio Pembagian Data | Nilai MAE |
|----------------------|-----------|
| 90:10 | 0,3916 |
| 85:15 | 0,3851 |
| 80:20 | 0,3741 |
| 75:25 | 0,382 |
| 60:40 | 0,3882 |

Pada eksperimen ini, teknik terbaik terdapat pada rasio pembagian data 85:15 karena memperoleh nilai MAE terendah yaitu 0,3741.

D. Pembahasan

Berdasarkan hasil eksperimen yang telah dilakukan pada masing-masing metode, diperoleh nilai MAE terendah pada metode *content-based filtering* 1,174201 dengan rasio pembagian data 90:10, nilai MAE terendah pada metode *Collaborative filtering* 0,3768 dengan rasio pembagian data 85:15 dan nilai MAE terendah pada metode *hybrid filtering* 0,3741 dengan rasio pembagian data 80:20.



Gambar 11. MAE Terendah Setiap Metode

Berdasarkan Gambar 11, diperoleh nilai MAE terendah pada metode *hybrid filtering*. Dibandingkan dengan hasil rekomendasi menggunakan *Collaborative filtering*, hasil dari *hybrid filtering* lebih optimal karena dilukannya pengisian *rating* yang kosong. Metode *content-based filtering* dengan menghasilkan prediksi *rating* pada seluruh *user*, sehingga *rating* memiliki distribusi merata dimana hal ini dapat membantu metode *Collaborative filtering* untuk membuat rekomendasi.

IV. KESIMPULAN

Berdasarkan hasil yang telah diperoleh, Kesimpulan yang dapat disimpulkan selama penggerjaan makalah ilmiah adalah sebagai berikut.

- Metode *hybrid filtering* yaitu kombinasi metode *content-based filtering* dan *Collaborative filtering* dapat diterapkan pada sistem rekomendasi pariwisata yaitu dengan cara mengisi *rating* yang kosong (*sparse*) menggunakan metode *content-based filtering* selanjutnya dihitung *similarity item* menggunakan *pearson correlation*, memilih sebanyak *k* konten berdasarkan urutan *item similarity*, menghitung prediksi *item* yang akan direkomendasikan kepada *user*.
- Pada penelitian ini, *hybrid filtering* terbukti cukup akurat dan lebih baik dalam memberikan rekomendasi dengan hasil eksperimen dengan MAE terendah adalah 0,3741 dengan *k* 25% dengan perbandingan metode *content-based filtering* yang menghasilkan nilai MAE yaitu 1,174201 dan metode *Collaborative filtering* yang menghasilkan nilai MAE 0,3768 dengan *k* 25%.
- Permasalahan *sparsity data* pada *dataset* yang mempengaruhi hasil rekomendasi pada metode

Collaborative filtering dapat diatasi menggunakan metode *content-based filtering* yang melakukan prediksi *rating* dan mengisi *rating* yang kosong (*sparse*). Pengisian *data sparse* terbukti lebih efektif berdasarkan nilai MAE yang diperoleh pada metode *hybrid filtering* lebih baik dibandingkan metode *Collaborative filtering* yang menggunakan *data sparse* dalam melakukan rekomendasi.

Dalam penelitian ini, saran yang dapat diberikan penulis untuk pengembangan penelitian selanjutnya terkait melakukan variasi pada kombinasi *hybrid filtering* dengan metode dasar lainnya seperti *knowledge-based* yang mampu mengatasi permasalahan *coldstart problem* yang belum dapat diatasi oleh metode *hybrid filtering* (*content-based filtering* dan *Collaborative filtering*) yang dilakukan penulis dan memilih *model based* dalam metode *Collaborative filtering* yang mampu memberikan waktu komputasi lebih cepat dan optimal dari pada *memory based* yang membutuhkan waktu komputasi lebih lama.

DAFTAR PUSTAKA

- [1] Kementerian Pariwisata. (2019, October 1). *Laporan Akuntabilitas Kinerja Kementerian Pariwisata*. Retrieved from Kementerian Pariwisata RI: <https://www.kemenpar.go.id/post/laporan-akuntabilitas-kementerian-pariwisata-lakip-tahun-2018>
- [2] Isinkaye, F., Folajimi, Y., & Ojokoh, B. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*(16), 261–273.
- [3] Xiang, Z., & Gretzel , U. (2010, April). Role of social media in online travel information search. *Tourism Management*. (S. Page, Ed.) *Tourism Management*, 31(2), 179-188
- [4] Burke, R. (2007). *The Adaptive Web, Chapter Hybrid Web Recommender systems, Lecture Notes in Computer Science* (Vol. 432). Springer, Berlin, Heidelberg, Germany.
- [5] Aggarwal, C. C. (2016). *Recommender systems*. Yorktown Heights, NY, USA: Springer International Publishing.
- [6] Kleef, N. v. (2014). *Making people matches using Supervised*. Enschede, Netherlands: University of Twente.
- [7] Hu, Y., Yang, Y., Li, C., Wang, Y., & Li, L. (2016). A hybrid genre-based personalized. *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*. China: IEEE.
- [8] Liu, S., Dong, Y., & Chai, J. (2016). Research of Personalized News Recommendation System Based on Hybrid. *2016 2nd IEEE International Conference on Computer and Communications*. Beijing, China: IEEE.
- [9] Uluyagmur, M., Cataltepe, Z., & Tayfur, E. (2012). Content-based Movie Recommendation Using Different Feature Sets. *Proceedings of the World Congress on Engineering and Computer Science 2012 Vol I. 1*, pp. 1-5. San Francisco, USA: Springer.
- [10] Kamath, S. (2016). A Personalized *Recommender system* Using Machinelearning Based Sentiment Analysis Over Social Data. *2016 IEEE Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*. Bhopal, India: IEEE.
- [11] Jurafsky, D., & Martin, J. H. (2019). *Speech and Language Processing (An Introduction to Natural Language Processing)* (Third Edition draft ed.). Amerika Serikat: Prentice Hall.