BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Tabel 2.1 merupakan beberapa penelitian terdahulu serta artikel jurnal yang dijadikan acuan dan referensi pada penelitian ini.

Tabel 2. 1 Penelitian Terdahulu

1	Judul	*A Lasso-based Collaborative Filtering Recommendation Model*
	Jurnal	*International Journal of Advanced Computer Science and Applications,* Vol. 13, No. 4, 2022 [15]
	Penulis	H. X. Huynh, L. V. Nguyen, V. Q. Dam, dan N. Q. Phan
	Metode	*User-based collaborative filtering (*recommenderlab*), Item-based collaborative filtering (*recommenderlab*),* lasso-UBCF, lasso-IBC*F, precision.*
	Hasil	Menerapkan algoritma *collaborative filtering* berbasis *user* dan *item* serta *collaborative filtering* dengan pendekatan *Lasso regression* pada *package recommenderlab* menggunakan data *MovieLense.* Nilai presisi yang diperoleh oleh model UBCF_LASSO, UBCF, IBCF_LASSO, dan IBCF secara berutut-turut adalah 0.6, 0.5, 0.5, dan 0.4.
2	Judul	Penerapan Metode *Item-Based Collaborative Filtering* Untuk Sistem Rekomendasi Data MovieLens
	Jurnal	Jurnal Matematika dan Aplikasi, Vol. 9, No. 2, P. 78-83, 2020 (Sinta 5) [16]
	Penulis	Y. V. L. Jaja, B. Susanto, dan L. R. Sasongko
	Metode	*Item-Based Collaborative Filtering*
	Hasil	Menerapkan algoritma *colaborative filtering* berbasis *item* menggunakan *package* recommenderlab pada bahasa pemrograman R. Penelitian berhasil menerapkan metode rekomendasi pada data MovieLens berdasarkan matriks similaritas.
3	Judul	Recommenderlab: *An R Framework for Developing and Testing Recommendation Algorithms*
	Jurnal	ArXiv, 2022 [7]
	Penulis	M. Hahsler
	Metode	*Used-based collaborative filtering* (recommenderlab), *item-based collaborative filtering* (recommenderlab), SVD *approximation* (recommenderlab), *precision*
	Hasil	Menerapkan algoritma *collaborative filtering* berbasis *user* dan *item*, serta SVD *approximation* pada *package recommenderlab* menggunakan data Jester5k. Model yang memperoleh nilai *precision* tertinggi adalah *item-based collaborative filtering* yaitu 0.24.
4	Judul	*An optimally weighted user- and item-based collaborative filtering approach to predicting baseline data for Friedreich's Ataxia patients*
	Jurnal	*Neurocomputing,* Vol. 419, P. 287-294, 2021 (Q1) [8]
	Tahun	2021
	Penulis	Y. Wenbin, W. Zidong, L. Weibo, T. Bo, L. Stanislao, dan L. Xiaohui
	Metode	*Collaborative filtering* berbasis *user*, *item*, dan kombinasi antara *user* dan *item*, PCC, MAE, RMSE.
	Hasil	Menerapkan algoritma *collaborative filtering* berbasis *user*, *item*, dan

		kombinasi *user – item* untuk melakukan prediksi *baseline* pada penyakit *Friedreich's ataxia* atau *FRDA.* Algoritma *collaborative filtering* berbasis kombinasi *user – item* dapat memberikan hasil yang lebih baik daripada *collaborative filtering* berbasis *user* maupun *item* sendiri.
5	Judul	*Recommendation Analysis on Item-Based and User-Based Collaborative Filtering*
	Jurnal	*International Conference on Smart Systems and Inventive Technology (ICSSIT), P. 1-4, 2019* [17]
	Penulis	G. Gupta & R. Katarya
	Metode	*User-based collaborative filtering, item-based collaborative filtering, precision*
	Hasil	Menerapkan algoritma collaborative filtering berbasis user, dan item untuk melakukan rekomendasi film menggunakan dataset MovieLens. Model yang memperoleh nilai *precision* terbaik adalah collaborative filtering berbasis user yaitu 0.179.
6	Judul	MuSIF: *A Product Recommendation System Based on Multi-source Implicit Feedback*
	Jurnal	IFIP *International Federation for Information Processing,* Vol. 559, P. 660-672, 2019 (Q4) [9]
	Penulis	Schoinas, I., & Tjortjis, C.
	Metode	*Collaborative Filtering, implicit data*
	Hasil	Menerapkan algoritma *collaborative filtering* menggunakan data *implicit* dari satu sumber dan beberapa sumber. Sumber – sumber data *implicit* tersebut adalah *viewing, searching, purchase*, dan interaksi – interaksi lainnya. Dari hasil evaluasi yang diperoleh, penggunaan model *single source* pada data *implicit* memiliki hasil yang lebih baik daripada penggunaan *multi source.*
7	Judul	Implementation and Evaluation of Movie Recommender Systems Using Collaborative Filtering
	Jurnal	*Journal of Advances in Information Technology*, Vol. 12, No. 3, 2021 [18]
	Penulis	S. Salloum dan D. Rajamanthri
	Metode	*User-based collaborative filtering, modified user-based collaborative filtering, cosine similarity*, RMSE.
	Hasil	Menerapkan algoritma *collaborative filltering* berbasis *user* menggunakan model similaritas *cosine similarity*, dan *modified cosine similarity.* Model yang memperoleh nilai MRSE terbaik adalah *collaborative filtering* berbasis *user* menggunakan *cosine similarity* yaitu 0.99.
8	Judul	*Product Recommendation for e-Commerce System based on Ontology*
	Jurnal	*International Conference on Cybernetics and Intelligent System*, Vol.1, P. 105-109, 2019 [6]
	Penulis	N. M. S. Iswari, Wella, dan A. Rusli
	Metode	*Ontology*
	Hasil	Menerapkan pendekatan *ontology* dalam pemberikan rekomendasi. Hasil menunjukkan bahwa rekomendasi juga dapat memberikan produk yang memiliki kategori diluar ketertarikan *customer.* Hal tersebut dapat memberikan rekomendasi yang lebih bervariasi.
9	Judul	*Exploring the Impact of Similarity Model to Identify the Most Similar Image from a Large Image Database*
	Jurnal	*Journal of Physics: Conference Series,* Vol. 1693, 2020 (Q4) [11]
	Tahun	2020
	Penulis	Y. Chen
	Metode	*Cosine Similarity, Euclidean Distance, Correlation Distance, Manhattan Distance, Chebyshev Distance, Pearson Correlation Coefficient*
	Hasil	Melakukan perbandingan model similaritas untuk mengidentifikasi *similar images.* Dari hasil evaluasi yang diperoleh, *cosine similarity* merupakan

		model similaritas yang memiliki akurasi paling tinggi sebesar 94%, dilanjuti dengan *pearson correlation coefficient* sebesar 93%, *correlation distance* sebesar 83%, *chebyshev distance* sebesar 73%, *manhattan distance* sebesar 64%, dan *euclidean distance* sebesar 62%.
10	Judul	*The stratified K-folds cross-validation and class-balancing methods with high-performance ensemble classifiers for breast cancer classification*
	Jurnal	*Healthcare Analytics*, Vol. 4, 2023 (Q2) [13]
	Penulis	Mahesh T R, Vinoth Kumar V, Dhilip Kumar V, Oana Geman, Martin Margala, & Manisha Guduri
	Metode	*Stratified K-folds cross validation, classification*
	Hasil	Menerapkan pembagian data cross validation dengan teknik stratified K-folds pada pengklasifikasian kanker payudara. Algoritma yang digunakan untuk klasifikasi adalah *logistic regression, support vector machine, k-nearest neighbours, classification and regression tree, naïve bayes, XGBoost,* dan *random forest.* Dengan menerapkan teknik *cross validation* pada tahap evaluasi, algoritma *logistic regression, support vector machine,* dan *naïve bayes* adalah yang terbaik untuk melakukan klasifikasi kanker payudara dengan akurasi sebesar 99,3 %.
11	Judul	*Multi-class fruit-on-plant detection for apple in SNAP system using Faster RCNN*
	Jurnal	*Computers and Electronics in Agriculture*, Vol. 176, 2020 (Q1) [14]
	Penulis	Gao, F., Fu, L., Zhang, X., Majeed, Y., Li, R., Karkee, M., dan Zhang, Q.
	Metode	*Convolutional Neural Network, Mean Average Precision*
	Hasil	Menerapkan algoritma *convolutional neural network* untuk mendeteksi dan mengklasifikasi kelas sebuah apel dari gambar. Penelitian menggunakan metrik pengukuran bernama *mean average precision* dan memperoleh nilai sebesar 0,879 pada sistem yang dibuat.

Terdapat 11 penelitian terdahulu mengenai pembuatan sistem rekomendasi serta teknik evaluasi yang dilakukan. Penelitian 1, 2, dan 3 merupakan penelitian yang membentuk sistem rekomendasi menggunakan model *collaborative filtering* dari *package recommenderlab*. Namun pada penelitian 2, belum ada tahap evaluasi yang dilakukan pada pembentukan model menggunakan *package recommenderlab*. Penelitian 4, 5, dan 7 merupakan penelitian yang membentuk sistem rekomendasi menggunakan model *collaborative filtering* diluar *package recommenderlab.* Namun pada penelitian 4, membentukan sistem rekomendasi *collaborative filtering* berbasis *user*, *item*, dan *hybrid* menggunakan matriks PCC atau *pearson correlation coefficient* menggunakan teknik evaluasi MAE dan RMSE. Kemudian pada penelitian 5 dan 7, pembentukan sistem rekomendasi hanya dilakukan kepada *collaborative filtering* berbasis *user* dan *item* menggunakan matriks *cosine similarity*.

Penelitian ini menggunakan metode yang pernah dilakukan pada penelitian sebelumnya, namun dilakukan pada studi kasus yang berbeda. Penelitian ini mengadopsi penggunaan metode *collaborative filtering* yang dapat hanya menggunakan data *implicit* sesuai dengan penelitian 6, sehingga tidak diperlukan adanya data konten terkait produk. Penelitian ini juga mengikutsertakan sistem rekomendasi yang sedang digunakan oleh perusahaan yaitu model dari *package recommenderlab,* yang juga digunakan pada penelitian 1, 2, dan 3. Selain itu, penggunaan *collaborative filtering* berbasis *user, item,* dan *hybrid* juga mengikuti metode yang digunakan pada penelitian 4, namun menggunakan matriks persamanan yang berbeda. Penggunaan *collaborative filtering* memerlukan nilai similaritas antar *user* dan antar *item* yang dapat dihitung menggunakan beberapa model similaritas. Penelitian ini menggunakan model perhitungan similaritas *cosine similarityi* dan *manhattan distance* yang diadopsi dari peneltian 7 dan 9. Kemudian setiap model yang dibuat pada penelitian ini akan diuji pada tahap evaluasi menggunakan metode *stratified k-fold cross validation* yang dilakukan pada penelitian 10 sehinga seluruh data dapat memperoleh bagian pada data uji dan data latih, serta metrik pengukuran *mean average precision* yang digunakan pada penelitian 11.

2.2 Tinjauan Teori

2.2.1 Farmasi

Menurut Kamus Besar Bahasa Indonesia (KBBI), farmasi merupakan teknik serta teknologi terkait pembuatan, penyimpanan, penyediaan, dan pendistribusian obat [2]. Farmasi merupakan cabang ilmu kesehatan yang penting karena dapat digunakan untuk mencegah, mengurangi gelaja, hingga menyembuhkan penyakit. Industri farmasi merupakan salah satu bidang farmasi yang fokus ke pengembangan, produksi, hingga pengujian obat dalam jumlah yang besar. Industri farmasi merupakan industri padat modal yang harus beroperasi menggunakan teknologi terkini serta selalu melakukan penelitian serta menerapkan regulasi dan kebijakan yang ketat [19].

2.2.2 B2B E-*Commerce*

E-commerce atau *electronic commerce* merupakan proses pemasaran, pembelian, dan penjualan produk yang menggunakan jaringan internet [20]. Pada saat ini, *e-commerce* merupakan salah satu syarat bagi perusahaan untuk dapat bersaing pada bisnis global [21]. Terdapat beberapa jenis *e-commerce* yaitu *Business to Consumer* (B2C), *Business to Business* (B2B), *Consumer to Consumer* (C2C), *Peer-to-Peer* (P2P), dan *Mobile Commerce* (M-Commerce) [22]. *Business to Business* (B2B) *e-commerce* merupakan *e-commerce* yang proses transaksinya dilakukan antar organisasi [23].

2.2.3 Sistem Rekomendasi

Sistem rekomendasi merupakan sistem yang digunakan untuk memberikan rekomendasi kepada penggunanya dengan harapan rekomendasi yang diberikan dapat memenuhi keinginan serta kebutuhan pengguna [16]. Untuk memberikan rekomendasi, sistem akan melakukan analisis data yang diperlukan berdasarkan kategori sistem rekomendasi yang digunakan. Berikut merupakan beberapa kategori serta data yang diperlukan pada sistem rekomendasi [5][8].

2.2.3.1 *Collaborative Filtering*

Collaborative filtering merupakan sistem rekomendasi yang ditemukan oleh Goldberg pada tahun 1992 [24]. *Sistem rekomendasi ini menggunakan data histori untuk menghasilkan rekomendasi sehingga tidak memerlukan informasi dari objek yang digunakan. Rekomendasi yang dihasilkan pada collaborative filtering akan berdasarkan nilai persamaan atau similarity dari setiap user dan item. Sistem rekomendasi ini memiliki asumsi bahwa sebuah item yang diminati oleh salah satu user juga akan diminati oleh user lainnya dengan pola pembelian yang mirip.*

2.2.3.2 *Content-based*

Content-based merupakan sistem rekomendasi yang menggunakan data atribut dari sebuah *item*. Sistem rekomendasi ini akan memberikan rekomendasi berdasarkan atribut dari sebuah *item* kepada *user* dengan preferensi yang sesuai. Contoh dari sistem rekomendasi *content-based* adalah pemberian rekomendasi lagu pop kepada *user* yang menyukai musin bergenre pop.

2.2.3.3 *Demographic-based*

Demographic-based merupakan sistem rekomendasi yang menggunakan data demografis untuk menghasilkan rekomendasi. Rekomendasi yang diberikan kepada *user* akan berdasarkan preferensi dari *user* yang memiliki usia, jenis kelamin, dan lokasi yang sama. Contoh dari sistem rekomendasi *demographic-based* adalah pemberian rekomendasi kepada seorang perempuan berupa produk yang paling banyak dibeli oleh perempuan pada suatu wilayah.

2.2.3.4 *Hybrid*

Sistem rekomendasi *hybrid* merupakan sistem rekomendasi yang menggabungkan dua atau lebih teknik sistem rekomendasi. Contoh dari penggunaan sistem rekomendasi *hybrid* adalah penggabungan *collaborative filtering* berbasis *user* dan *item.* Pada beberapa penelitian, sistem rekomendasi *hybrid* dikatakan dapat memberikan hasil rekomendasi yang lebih baik daripada sistem rekomendasi lainnya.

2.2.4 *Similarity Model*

Similarity model merupakan perhitungan yang digunakan untuk mencari nilai kemiripan atau similaritas antara 2 identitas. Dalam kasus sistem rekomendasi, *similarity model* digunakan untuk mencari nilai persamaan atau similaritas antara *user* maupun *item*. Terdapat beberapa *similarity model* yang dapat digunakan antara lain *cosine similarity, euclidian distance, correlation*

distance, manhattan distance, chebyshev distance, dan p*earson correlation coefficient* [11]. Setiap *similarity model* memiliki formula atau rumus perhitungan yang berbeda–beda. Nilai hasil dari perhitungan *similarity model* berkisar antara 0 yang berarti tidak ada kemiripan, hingga 1 yang berarti sangat mirip. Berikut merupakan penjelasan dan formula dari beberapa similarity model.

2.2.4.1 *Cosine Similarity*

Cosine similarity merupakan model perhitungan persamaan yang menggunakan konsep derajat kosinus [25]. Rumus 2.1 merupakan *formula* atau rumus untuk menghitung nilai *cosine similarity* antara dua identitas. Sim (AB) merupakan nilai similaritas dari identitas A dan identitas B. $A \cdot B$ merupakan perkalian antara nilai pada identitas A dan identitas B. $\|A\| \cdot \|B\|$ merupakan nilai mutlak dari perkalian antara nilai mutlak dari identitas A dan identitas B.

$$Sim(AB) = \frac{A \cdot B}{\|A\| \cdot \|B\|}$$

Rumus 2. 1 *Cosine Similarity* [25]

2.2.4.2 *Manhattan Distance*

Manhattan distance merupakan model perhitungan persamaan berdasarkan jarak yang menggunakan konsep selisih mutlak [26]. Berikut merupakan *formula* atau rumus untuk menghitung nilai *manhattan distance* antara dua identitas. $d(x, y)$ merupakan jarak *manhattan* dari nilai A dan nilai B. $\sum_{i=1}^{n}$ merupakan penjumlahan dari pengulangan dari i=1 hingga i=n. $|xi - yi|$ merupakan nilai mutlak dari selisih nilai x pada urutan i dan nilai y pada urutan i.

$$d(x, y) = \sum_{i=1}^{n} |xi - yi|$$

Rumus 2. 2 *Manhattan Distance* [27]

2.2.4.3 *Euclidean Distance*

Euclidean distance merupakan model perhitungan persamaan berdasarkan jarak yang menggunakan konsep ruang *euclidian* atau teori *pythagoras* [28]. Rumus 2.3 merupakan *formula* atau rumus untuk menghitung nilai *euclidian distance* antara dua identitas. $d(i, j)$ merupakan jarak *euclidean* dari identitas i dan j. $\sum_{k=1}^{n}$ merupakan penjumlahan dari pengulangan dari k=1 hingga k=n. $(x_{ik} - x_{jk})^2$ merupakan nilai kuadrat dari selisih nilai identitas i pada urutan ke k dan nilai identitas j pada urutan ke k.

$$d(i, j) = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2}$$

Rumus 2. 3 *Euclidean Distance* [28]

2.3 *Framework*, Algoritma, dan Metode Evaluasi

2.3.1 CRISP-DM *Framework*

CRISP-DM atau *cross-industry standard process for data mining* merupakan sebuah proses dari *framework* data mining yang bersifat *industry independent* [29]. CRISP-DM tidak hanya berfokus pada proses data mining, tetapi juga menambahkan beberapa tahap teknis seperti *business understanding* atau pemahaman bisnis [29]. Terdapat 6 tahapan yang dilakukan pada CRISP-DM, yaitu sebagai berikut.

1) *Business understanding*

Business understanding merupakan tahap pertama dari CRISP-DM. Tahap *business understanding* meliputi pencarian permasalahan yang ada beserta solusinya hingga menentukan *requirement* apa saja yang diperlukan dalam penelitian.

2) *Data Understanding*

Data understanding merupakan tahap kedua dari CRISP-DM. Tahap *data understanding* meliputi pengambilan hingga memahami data yang diperlukan untuk penelitian. Pemahaman data dapat dilakukan dengan analisa statistik maupun visualisasi.

3) *Data Preparation*

Data preparation merupakan tahap ketiga dari CRISP-DM. Tahap *data preparation* digunakan untuk mempersiapkan data yang telah diperoleh pada tahap sebelumnya, agar kemudian dapat dilatih pada model. *Data preparation* meliputi pembersihan data, penyeleksian data, transformasi data, hingga pembagian data menjadi data uji dan data latih.

4) *Modeling*

Modeling merupakan tahap keempat dari CRISP-DM. Tahap *modeling* merupakan tahap pembuatan model menggunakan algoritma dan *formula* yang sudah ditentukan pada tahapan sebelumnya.

5) *Evaluation*

Evaluation merupakan tahap kelima dari CRISP-DM. Tahap *evaluation* dilakukan untuk mengevaluasi perfoma dari model–model yang sudah dibangun. Tahap ini akan menilai apakah model yang dibuat sudah memiliki hasil yang baik atau tidak. Apabila model masih belum dapat memberikan hasil yang baik, maka proses data mining perlu dilakukan kembali dengan menggunakan model lainnya.

6) *Deployment*

Deployment merupakan tahap terakhir dari CRISP-DM. Tahap *deployment* digunakan untuk menyebarkan model ataupun hasil yang sudah dibangun. *Deployment* dalam pengelolaan data dapat berupa *dashboard* visualisasi

data, laporan, hingga API atau *application programming interface* yang dapat dipanggil oleh program lainnya.

2.3.2 SEMMA *Framework*

SEMMA merupakan sebuah framework data mining yang dikembangkan oleh institusi SAS pada tahun 1997 [17]. Terdapat 5 tahapan pada SEMMA yaitu sample, explore, modify, model, dan assess yang memiliki fokus untuk membuat model dari data mining. Sample merupakan tahapan yang melakukan pengumpulan data serta informasi yang dibutuhkan. Explore merupakan tahap pencarian kumpulan data yang berhubungan dengan topik data mining. Modify merupakan tahap modifikasi data seperti mengelompokkan variabel. Model merupakan tahap pembuatan model menggunakan data yang telah dimodifikasi. Access merupakan tahap melakukan penilaian berupa evaluasi pada model [30].

2.3.3 *Collaborative Filtering Algorithm*

Collaborative Filtering merupakan salah satu metode yang digunakan pada sistem rekomendasi. *Collaborative filtering* dapat menghasilkan rekomendasi berdasarkan pola seperti pembelian *customer* tanpa harus menggunakan informasi *exogenous* dari *user* maupun *item* [31]. Terdapat beberapa teknik pada *collaborative filtering* yaitu *collaborative filtering* berbasis *user, item,* dan *hybrid.*

2.3.3.1 *User-based Collaborative Filtering*

Collaborative filtering berbasis *user* merupakan *collaborative filtering* yang menggunakan nilai similaritas antar *user* berdasarkan *item* yang dibeli [32]. Rumus 2.4 merupakan *formula* atau rumus yang digunakan pada *collaborative filtering* berbasis *user*. $\bar{\pi}^{UCF}_{Um}(i_n)$ merupakan nilai rekomendasi sebuah *item* n kepada *user* m menggunakan algoritma *collaborative filtering* berbasis *user*. $\bar{\pi}(u_m)$ merupakan nilai rata–rata dari keseluruhan *item* yang dibeli *user* m.

$\sum_{Allsimilarusers}$ merupakan penjumlahan dari pengulangan seluruh *user* selain m terhadap *user* m. $Aff(\pi(u_m),\pi(u_{m'}))$ merupakan nilai similaritas *user* m dan *user* m'. $(\pi_{um}(i_{n'}) - \overline{\pi}(u_{m'})$ merupakan selisih dari nilai *item* n pada *user* m' dengan nilai rata-rata dari keseluruhan *item* yang dibeli *user* m'.

$$\overline{\pi}^{UCF}_{Um}(i_n) = \overline{\pi}(u_m) + \frac{\sum_{Allsimilarusers} Aff(\pi(u_m),\pi(u_{m'})) * (\pi_{um}(i_{n'}) - \overline{\pi}(u_{m'})}{\sum_{Allsimilarusers} |Aff(\pi(u_m),\pi(u_{m'}))|}$$

Rumus 2. 4 *Used-based Collaborative Filtering* [10]

2.3.3.2 *Item-based Collaborative Filtering*

Collaborative filtering berbasis *item* merupakan *collaborative filtering* yang menggunakan similaritas antar *item* berdasarkan *user* yang membeli [32]. Rumus 2.5 merupakan *formula* atau rumus yang digunakan pada *collaborative filtering* berbasis *item* [10]. $\overline{\pi}^{ICF}_{Um}(i_n)$ merupakan nilai rekomendasi sebuah *item* n kepada *user* m menggunakan algoritma *collaborative filtering* berbasis *item.* $\sum_{Allsimilaritems}$ merupakan penjumlahan dari pengulangan seluruh *item* selain n terhadap *item* n. $Aff(\pi(i_n),\pi(i_{n'}))$ merupakan nilai similaritas *item* n dan *item* n'. $\pi_{um}(i_{n'})$ merupakan nilai *item* n pada *user* m'.

$$\overline{\pi}^{ICF}_{Um}(i_n) = \frac{\sum_{Allsimilaritems} Aff(\pi(i_n),\pi(i_{n'})) * \pi_{um}(i_{n'})}{\sum_{Allsimilaritems} |Aff(\pi(i_n),\pi(i_{n'}))|}$$

Rumus 2. 5 *Item-based Collaborative Filtering* [10]

2.3.3.3 *Hybrid Collaborative Filtering*

Collaborative filtering berbasis *hyrbid* merupakan *collaborative filtering* yang menggabungkan *collaborative filtering* berbasis *user* dan *item* [5]. Rumus 2.6, 2.7, dan 2.8 merupakan rumus atau *formula* yang digunakan pada *collaborative filtering* berbasis *hybrid* [10].

$$\bar{\pi}^{HCF}_{Um}(i_n) = \max(\bar{\pi}_{\wedge}(u_m(i_n)); \min(\bar{\pi}_{\vee}(u_m(i_n)), 1 - h(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n))))$$

Rumus 2. 6 *Hybrid Collaborative Filtering* (i) [10]

Dimana,

$$\bar{\pi}(u_m(i_n)) = \frac{\min(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n))}{h(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n))}$$

$$\pi_{\vee}(u_m(i_n)) = \max(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n))$$

$$h(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n)) = 1 - inc(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n)) = 1 - \max(\bar{\pi}^{UCF}_{Um}(i_n) \otimes \bar{\pi}^{ICF}_{Um}(i_n))$$

Rumus 2. 7 *Hybrid Collaborative Filtering* (ii)

Setelah disederhanakan, formula atau rumus yang digunakan pada hybrid collaborative filtering dapat dilihat pada Rumus 2.8. $\bar{\pi}^{HCF}_{Um}(i_n)$ merupakan nilai rekomendasi sebuah *item* n kepada *user* m menggunakan algoritma *collaborative filtering* berbasis *hybrid.* $\min(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n))$ merupakan nilai minimum antara nilai rekomendasi sebuah *item* n kepada *user* m menggunakan algoritma *collaborative filtering* berbasis *user* dan *item*. $\max(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n))$ merupakan nilai maksimum antara nilai rekomendasi sebuah *item* n kepada *user* m menggunakan algoritma *collaborative filtering* berbasis *user* dan *item*

$$\bar{\pi}^{HCF}_{Um}(i_n) = \frac{\min(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n))}{\max(\bar{\pi}^{UCF}_{Um}(i_n), \bar{\pi}^{ICF}_{Um}(i_n))}$$

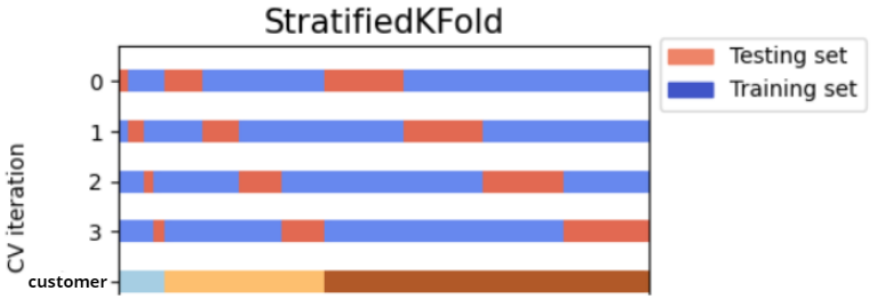
Rumus 2. 8 *Hybrid Collaborative Filtering* (iii)

2.3.4 *Evaluation Method*

Evaluation method merupakan metode yang digunakan untuk mengevaluasi model–model dengan tujuan memperoleh model terbaik. Terdapat dua metode yang digunakan pada penelitian ini yaitu *stratified k-fold cross validation* serta metrik pengukuran *mean average precision.*

2.3.4.1 *Stratified K-Folds Cross Validation*

Cross validation merupakan teknik evaluasi model–model pada *machine learning* dengan asumsi bahwa data yang digunakan pada pelatihan dan pengujian model adalah data yang berbeda [33]. Terdapat beberapa tipe *cross validation* seperti *k-fold, repeated k-fold, leave one out, stratified k-fold, group k-fold,* dan lain–lain [34]. Gambar 2.1 merupakan *stratified k-folds cross validation* yang membagi dataset sebanyak k kali dan tetap memastikan bahwa setiap pembagian memiliki distribusi kelas yang sama [13].



Gambar 2. 1 *Stratified K-Fold Cross Validation*

Penggunaan *stratified k-fold cross validation* dapat membagi data produk yang dibeli oleh setiap *customer* menjadi data latih dan data uji. Simulasi pembagian data pada menggunakan *stratified k-fold cross validation* dapat dilihat pada Tabel 2.2. Tabel tersebut merupakan simulasi yang dilakukan pada *customer* dengan pembelian produk A, B, C, D, dan E.

Tabel 2. 2 Simulasi *Stratified K-Fold Cross Validation*

Customer 1: A, B, C, D, E					
Split	Split 1	Split 2	Split 3	Split 4	Split 5
Pelatihan	B, C, D, E	A, C, D, E	A, B, D, E	A, B, C, E	A, B, C, D
Pengujian	A	B	C	D	E

2.3.4.2 *Mean Average Precision @* K

Mean average precision @ k merupakan teknik untuk menghitung presisi yang dikenalkan oleh H&M pada kompetisi *H&M Personalized Fashion Recommendations* di situs Kaggle [35]. Penggunaan *mean average precision @* k dapat menghilangkan penalti untuk memberikan rekomendasi sebanyak k kepada *customer* yang tidak membeli item sebanyak k. Dalam kasus penelitian ini, hal tersebut dapat menghindari pengurangan nilai presisi pada *customer* yang membeli kurang dari 10 produk. Semakin tinggi nilai *mean average precision* yang diperoleh, semakin baik rekomendasi yang diberikan [14].

Rumus 2.9 merupakan *formula* atau rumus yang digunakan untuk menghitung nilai *mean average precision @* k. $MAP@10$ merupakan nilai *mean average precision* yang diperoleh. $\frac{1}{U}\sum_{u=1}^{U}$ merupakan pembagian dari jumlah nilai *average precision @* k dari seluruh *customer* dengan jumlah *customer*. $\frac{1}{\min(m,10)}\sum_{k=1}^{\min(n,10)} P(k)\times rel(k)$ merupakan nilai *average precision @* k yang diperoleh dari pembagian nilai p(k)*rel(k) dengan jumlah produk yang direkomendasikan.

$$MAP@10 = \frac{1}{U}\sum_{u=1}^{U}\frac{1}{\min(m,10)}\sum_{k=1}^{\min(n,10)} P(k)\times rel(k)$$

Rumus 2. 9 *Mean Average Precision*

2.4 Tools

2.4.1 BigQuery

BigQuery merupakan tools milik Google yang digunakan untuk *data warehouse*. Dengan menggunakan BigQuery, pengguna dapat melakukan kueri pada data berukuran besar dengan cepat yang diberikan oleh Google [36]. BigQuery merupakan *tools* yang mudah digunakan karena memnggunakan SQL sebagai bahasa untuk melakukan kuerinya. Pemrosesan kueri pada BigQuery dapat dilakukan dengan cepat karena BigQuery menggunakan pemrosesan pararel yang membagi pemrosesan data berkuantitas besar menjadi beberapa unit pemrosesan.

2.4.2 Python

Python merupakan bahasa pemrograman yang dapat digunakan untuk proyek pengelolaan data, *machine learning*, *data scraping*, hingga pengembangan aplikasi. Python dibuat oleh Guido van Rossum pada tahun 1990-an di Stitching Mathematisch Centrum (CWT) Belanda [37]. Selama satu dekade terakhir, Python lebih banyak digunakan untuk pembelajaran *machine learning* [38]. Hal tersebut dapat dilihat dari banyaknya *libraries* untuk pengelolaan data yang tersedia pada bahasa pemrograman Python.

Terdapat beberapa kelebihan dari penggunaan bahasa pemrograman Python, yaitu:

1. *Python* merupakan bahasa pemrograman yang mudah untuk dipelajari serta dipahami dikarenakan memiliki sintaks yang sederhana [39].
2. *Python* merupakan bahasa pemrograman *open-source* yang memiliki banyak *libraries* untuk digunakan secara gratis [40].

2.4.3 FastAPI

FastAPI merupakan sebuah kerangka kerja website yang digunakan untuk membuat *application programming interface* atau API menggunakan bahasa pemrograman Python [41]. *Application programming interface* atau API sendiri merupakan media yang digunakan untuk memfasilitasi komunikasi

antara klien dan server sehingga dapat saling bertukar informasi [42]. FastAPI umumnya digunakan pada pembuatan API proyek *data science.* FastAPI dapat digunakan dengan menginstall *package* bernama fastapi dan uvicorn.