BAB III

METODOLOGI PENELITIAN

3.1 Gambaran Umum Objek Penelitian

Penelitian ini bertujuan untuk memberikan rekomendasi produk pulsa dan *voucher* berdasarkan kemiripan visual untuk perusahaan atau *end user*. Produk yang diteliti adalah pulsa dan *voucher* yang ditawarkan oleh berbagai penyedia layanan telekomunikasi dan e-*commerce*. Penelitian ini memfokuskan pada produk pulsa dan *voucher* karena keduanya merupakan bagian integral dari layanan telekomunikasi dan perdagangan elektronik yang penting bagi perusahaan dan *end user* dalam menjalankan aktivitas komunikasi dan transaksi online.

Objek penelitian ini mencakup berbagai jenis pulsa dan *voucher* yang ditawarkan oleh penyedia layanan telekomunikasi dan e-*commerce*. Pulsa meliputi berbagai denominasi dan paket yang digunakan untuk mengakses layanan telepon, SMS, dan internet. Sementara itu, *voucher* mencakup berbagai jenis *voucher* yang digunakan untuk pembelian produk dan layanan tertentu, seperti *voucher* diskon, *voucher* hadiah, dan *voucher* langganan.

Data yang digunakan dalam penelitian ini berasal dari perusahaan penyedia layanan telekomunikasi dan e-*commerce* yang dikumpulkan hingga periode Januari 2023. Data ini mencakup informasi tentang berbagai produk pulsa dan *voucher* yang ditawarkan oleh perusahaan tersebut, termasuk gambar produk, deskripsi, dan detail lainnya yang relevan. Data ini merupakan sumber informasi yang penting untuk menganalisis preferensi dan kebutuhan pelanggan serta mengidentifikasi pola pembelian yang ada.

3.2 Metode Penelitian

Penelitian ini termasuk dalam jenis penelitian kuantitatif, dengan pendekatan pengolahan data menggunakan perhitungan rumus dan model matematis [76]. Klasifikasi tersebut didasarkan pada penggunaan data transaksi untuk menghasilkan rekomendasi penjualan menggunakan algoritma *machine learning*.

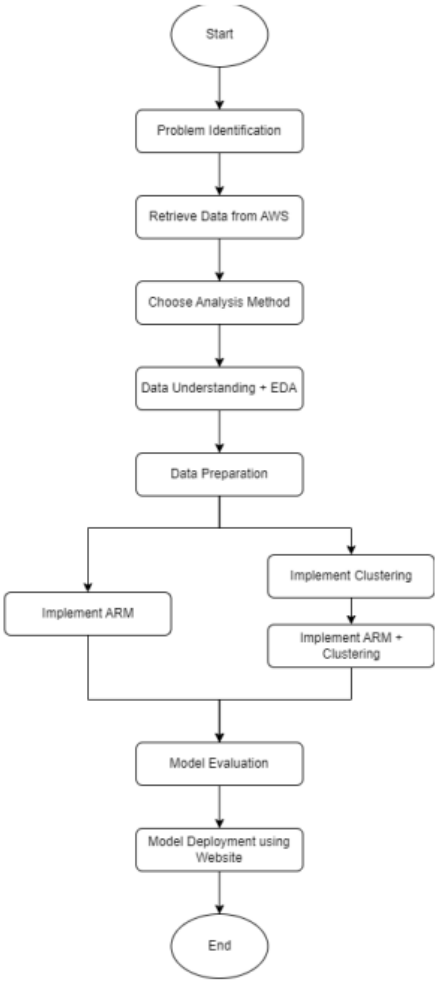
Dalam konteks ini, analisis data transaksi dilakukan untuk mengevaluasi pola pembelian dan mengidentifikasi kesamaan antara transaksi, dengan menggunakan rumus matematika dalam algoritma *machine learning* untuk menghasilkan rekomendasi penjualan. Evaluasi rekomendasi penjualan juga dilakukan berdasarkan perhitungan rumus yang termasuk ke dalam proses kuantitatif. Dengan demikian, penelitian ini menggabungkan elemen analisis data kuantitatif dengan penerapan model matematis dalam algoritma *machine learning* untuk mengoptimalkan strategi penjualan.

3.2.1 Alur Penelitian

Alur penelitian merupakan kronologi prosedural yang dikerjakan oleh seorang peneliti dalam penelitiannya untuk melaksanakan rencana penelitiannya. Alur penelitian dapat membantu dalam menstrukturisasi dan merinci proses utama dalam suatu penelitian. Berdasarkan Gambar 3.1, penelitian ini dimulai dengan identifikasi masalah yang akan diselesaikan melalui *problem indentification*. Berdasarkan hasil identifikasi masalah, ditemukan bahwa terdapat limitasi rekomendasi *sales* pada perusahaan XYZ. Saat ini, sistem rekomendasi *sales* pada XYZ menggunakan analisis data sederhana, yaitu mencari jumlah produk dengan penjualan terbanyak. Produk tersebut yang menjadi bahan pemasaran oleh tim pemasaran. Hal ini tentunya kurang efektif mengingat rekomendasi *sales* tidak hanya berdasar pada kuantitas produk yang terjual, melainkan preferensi masing-masing pelanggan. Generalisasi suatu produk untuk semua konsumen dirasa kurang efektif dalam memberikan rekomendasi.

Berdasarkan permasalah tersebut, dilakukan pengumpulan data transaksi dari *database* XYZ. Data transaksi XYZ disimpan pada sebuah infrastruktur AWS *Athena*. Pengambilan data ini dilakukan dengan melakukan *query* SQL. Data yang diambil adalah data transaksi pada bulan Agustus 2023 dari tanggal 13-20. Data ini mencakup riwayat pembelian produk oleh pelanggan perusahaan selama periode tertentu. Pengumpulan data dilakukan dengan cermat dan menyeluruh untuk memastikan data yang dihasilkan berkualitas dan dapat diandalkan untuk analisis lebih lanjut.

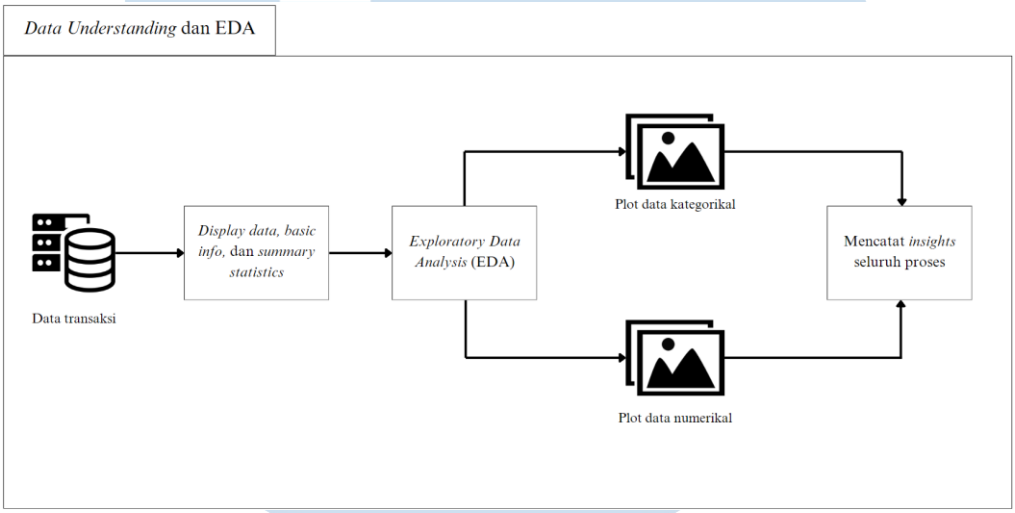
Setelah proses pengumpulan data transaksi perusahaan melalui AWS, dilakukan pemilihan metode analisis. Metode ini akan menjadi landasan proses penelitian ini. Berdasarkan beberapa pro dan kontra metode analisis, digunakan metode CRISP-DM. CRISP-DM cocok digunakan untuk penelitian ini karena tahapan CRISP-DM sampai pada proses *deployment.* Selain itu, CRISP-DM juga mampu disesuaikan dengan berbagai industri, dimana industri bisnis pulsa ini masuk dalam kategori bisnis yang jarang diimplementasikan analisis data.



Gambar 3. 1 Alur penelitian

Setelah melalui tahapan pemilihan metode analisis, dilakukan proses *data understanding* dan *Exploratory Data Analysis* (EDA). Secara lebih rinci, proses *data understanding* dan EDA ditunjukkan pada Gambar 3.2. Proses ini dimulai dengan meng-*import* data yang akan digunakan ke IDE *python*. Selanjutnya, dilakukan penampilan data, *basic info*, dan *summary statistics* untuk

mengetahui karakteristik data. Sebagai upaya memahami data lebih lanjut, dilakukan EDA dengan memvisualisasikan data kategorik dan numerik. Visualisasi dilakukan menggunakan *barplot* dan *countplot*. Seluruh *insights* yang diperoleh dari tahapan ini akan dicatat sebagai landasan untuk menganalisis data. Langkah ini penting untuk mengetahui gambaran umum, karakteristik, dan jenis data yang digunakan.

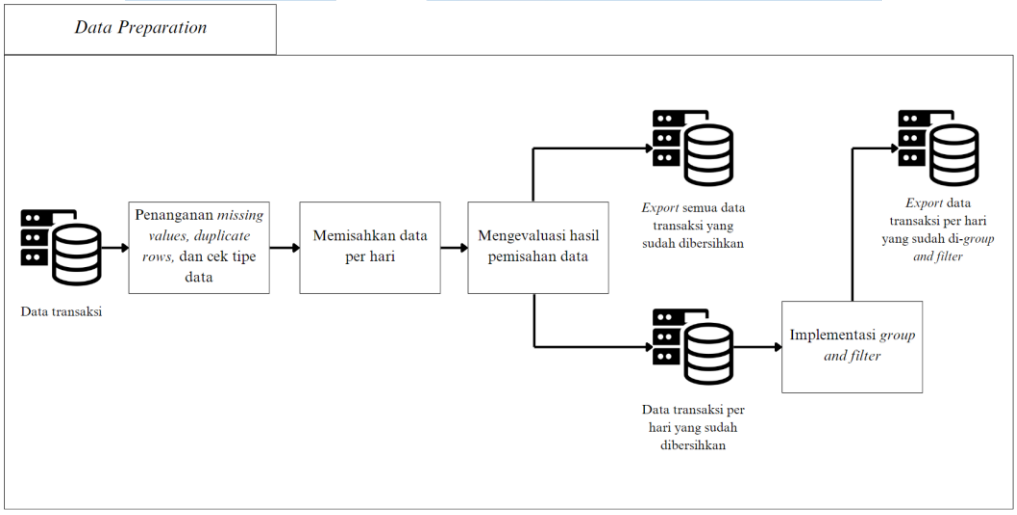


Gambar 3. 2 *Chart* Alur *Data Understanding* dan EDA

Langkah selanjutnya setelah melakukan *data understanding* dan EDA adalah *data preparation*. Tahapan ini diperlukan untuk mentransformasi data agar dapat dianalisis menggunakan algoritma yang diinginkan. Secara lebih rinci, proses *data preparation* ditunjukkan pada Gambar 3.3. Proses ini dimulai dengan meng-*import* data transaksi. Kemudian, dilakukan penanganan *missing values*, *duplicate rows*, dan pengecekan tipe data. Tujuannya agar proses analisis yang dilakukan kedepannya tidak menimbulkan *overfitting* akibat *input* data yang keliru. Selanjutnya, dilakukan pemisahan data per hari. Tujuannya agar transaksi pelanggan per hari dapat dikelompokkan menjadi 1 keranjang belanja. Tahapan ini sangat krusial sebagai *input* algoritma ARM.Saat melakukan pemisahan data, sangat penting untuk mengecek hasil pemisahan data agar data yang dipisahkan tidak bermasalah.

Setelah proses evaluasi pemisahan data selesai, dilakukan dua tahapan untuk *export* data, yaitu *export* semua data dan *export* data per hari. *Export*

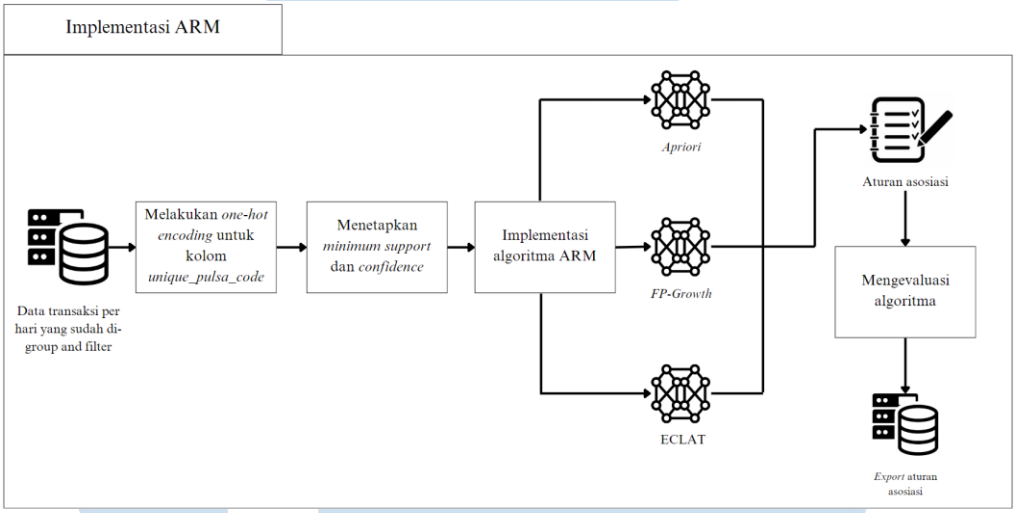
semua data akan digunakan pada tahapan *clustering*, sementara *export* data per hari digunakan untuk proses ARM. Dalam melakukan *export* data per hari, dibutuhkan fungsi *group and filter* untuk melakukan pengelompokkan data transaksi per hari menjadi keranjang belanja. Setelah melakukan *group and filter*, data transaksi per hari dapat di-*export.* Tahapan *data preparation* sangat krusial untuk memastikan konsistensi dan integritas data, serta menghilangkan data yang tidak relevan atau tidak valid yang dapat mengganggu hasil analisis.



Gambar 3. 3 *Chart* Alur *Data Preparation*

Tahapan selanjutnya adalah pengimplementasian ARM pada data yang sudah disiapkan. Secara lebih rinci, proses implementasi ARM ditunjukkan pada Gambar 3.4. Proses ini dimulai dengan meng-*import* data transaksi per hari yang sudah dibersihkan. Selanjutnya, dilakukan *one-hot encoding* pada data tersebut. Tujuannya agar data dapat diimplementasikan dengan algoritma ARM, seperti *Apriori* dan *FP-Growth.* Pada algoritma ECLAT, tahapan ini digantikan dengan proses transformasi data *unique_pulsa_code* menjadi *row*. Setelah persiapan data ini selesai, ditetapkan *minimum support* dan *confidence*. Variabel ini nantinya akan digunakan sebagai parameter algoritma ARM. Kemudian, diimplementasikan algoritma ARM pada data. Semua algoritma yang dibandingkan akan digunakan pada tahap ini. Setelah algoritma diimplementasikan, aturan asosiasi dihasilkan akan dievaluasi berdasarkan

matriks evaluasi. Hasil aturan asosiasi ini akan di-*export* sebagai rekomendasi *sales* untuk perusahaan.

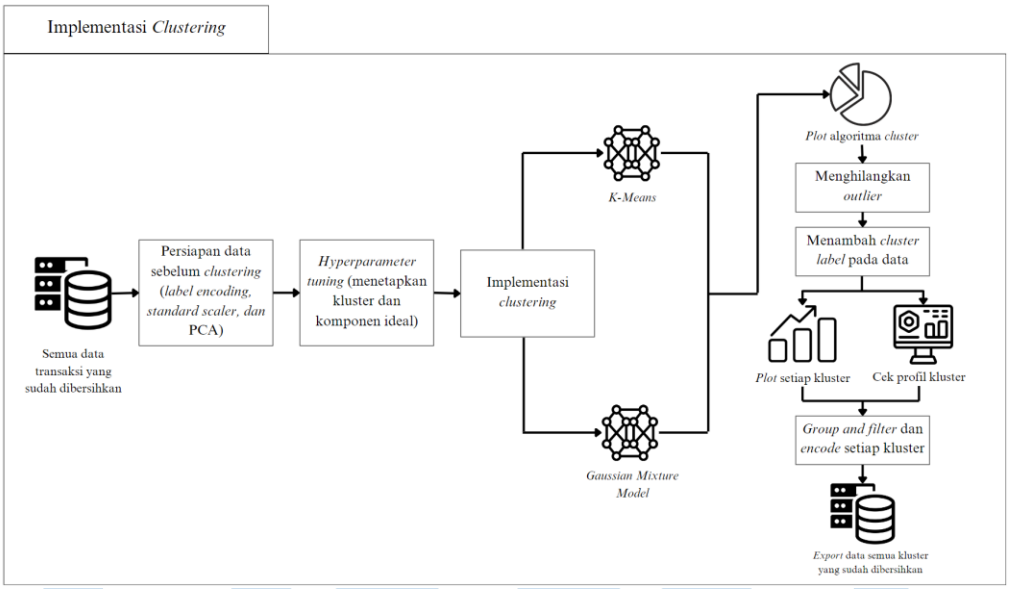


Gambar 3. 4 *Chart* Alur Implementasi ARM

Setelah proses implementasi ARM selesai, dilakukan perbandingan dengan algoritma kombinasi antara ARM dan *clustering.* Dalam mencapai objektif ini, diperlukan implementasi *clustering* pada data terlebih dahulu. Secara lebih rinci, proses implementasi *clustering* ditunjukkan pada Gambar 3.5. Proses ini dimulai dengan meng-*import* semua data transaksi yang sudah dibersihkan. Selanjutnya, dilakukan persiapan data sebelum *clustering.* Beberapa metode yang digunakan antara lain *label encoding, standard scaler,* dan PCA. Tujuan persiapan data ini adalah untuk mengubah data kategorik menjadi numerik, mengurangi bias dan dimensionalitas pada data. Setelah proses persiapan data ini selesai, dilakukan *hyperparameter tuning* untuk memaksimalkan klaster yang dihasilkan. *Hyperparameter* yang dilakukan berbeda-beda tergantung algoritma yang digunakan. Pada algoritma *K-Means*, *hyperparameter tuning* dilakukan menggunakan *elbow method*, CHI, dan DBI. Pada algoritma GMM, dilakukan metode evaluasi *elbow method,* AIC, dan BIC. Proses ini menghasilkan nilai klaster dan komponen optimal.

Kemudian, dilakukan implementasi *clustering* pada data yang sudah dipersiapkan. Algoritma yang digunakan untuk *clustering* adalah *K-Means* dan GMM. Hasil akhir dari *clustering* ini akan di-*plot* menggunakan *scatter plot.*

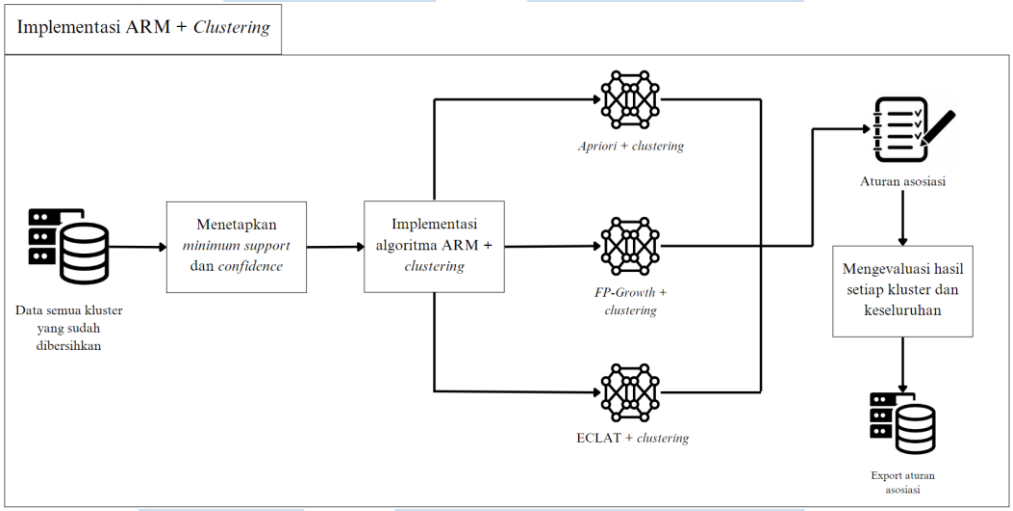
Tujuannya untuk mengetahui bentuk klaster yang dihasilkan. Jika ditemukan *outlier* pada *plot*, dilakukan penghapusan *outlier* pada data. Setelah data bersih dari *outlier*, dilakukan penambahan *cluster label* pada data. Tujuannya agar data dapat dipisahkan berdasarkan klaster yang terbentuk. Dalam memahami karakteristik setiap klaster, dilakukan *plotting* dan *summary statistics* pada setiap klaster. Tahapan terakhir dalam implementasi *clustering* adalah transformasi data yang sudah dipisahkan per klaster dengan metode *group and filter*. Hasil akhirnya akan di-*export* dan digunakan sebagai *input* pada kombinasi algoritma ARM dan *clustering*.



Gambar 3. 5 *Chart* Alur Implementasi *Clustering*

Langkah terakhir dalam proses analisis data penelitian ini adalah kombinasi ARM dan *clustering*. Secara lebih rinci, proses implementasi *clustering* ditunjukkan pada Gambar 3.6. Tahapan awal dalam proses ini adalah meng-*import* data semua klaster yang sudah dibersihkan. Kemudian, proses dilanjutkan dengan menentukan *minimum support* dan *confidence* sama seperti pengimplementasian ARM tunggal. Selanjutnya, dilakukan implementasi algoritma ARM dan *clustering*. Setiap klaster yang terbentuk akan diimplementasikan ARM. Artinya, klaster yang terbentuk dari algoritma *K-Means* dan GMM akan diimplementasikan ARM satu per satu. Hasil implementasi algoritma kombinasi ini adalah aturan asosiasi. Dari hasil aturan

asosiasi ini, dilakukan evaluasi hasil setiap klaster dan keseluruhan kombinasi. Metode ARM tunggal dan kombinasi akan dibandingkan untuk melihat evaluasi pengimplementasian ARM dengan dan tanpa kombinasi. Hasil aturan asosiasinya akan di-*export* sebagai *insight* rekomendasi *sales* untuk perusahaan.



Gambar 3. 6 *Chart* Alur Implementasi ARM dan *Clustering*

Langkah terakhir adalah mengimplementasikan proses dan *insight* hasil analisis ke dalam *website*. *Website* ini akan digunakan oleh tim pemasaran perusahaan untuk membantu optimasi penyebaran newsletter kepada pelanggan. Dengan adanya *insight* yang disajikan secara visual dan terstruktur, diharapkan tim pemasaran dapat mengambil keputusan yang lebih tepat dan efektif dalam menyusun strategi pemasaran serta meningkatkan interaksi dengan pelanggan secara lebih personal dan terarah.

3.2.2 Metode *Data Mining*

Penelitian ini bertujuan untuk mengembangkan model rekomendasi produk dengan memanfaatkan teknik pengolahan data melalui *data mining*. Dalam mencapai tujuan ini, dapat digunakan metode *Market Basket Analysis* (MBA). Dalam metode MBA, terdapat beberapa metode yang umum digunakan, yaitu *Association Rule Mining* (ARM) dan *Collaborative Filtering* Dalam menentukan metode yang cocok untuk menganalisis data transaksi pulsa perusahaan XYZ, dibuatkan tabel perbandingan antara metode ARM dan

Collaborative Filtering. Perbandingan ARM dan *Collaborative Filtering* dapat dilihat pada Tabel 3.1 [77]:

Tabel 3. 1 Perbandingan Metode ARM dan *Collaborative Filtering*

Indikator	ARM	*Collaborative Filtering*
Definisi	Teknik untuk menemukan aturan asosiatif antara item dalam dataset.	Metode rekomendasi berdasarkan kesamaan antara pengguna atau item.
Pendekatan	Analisis data transaksional untuk menemukan pola sering.	Analisis data pengguna atau item untuk menemukan kesamaan.
Jenis Data	Data transaksional, misalnya keranjang belanja.	Data pengguna, misalnya data demografis, jenis kelamin, umur, dll.
Interpretasi Hasil	Mudah diinterpretasikan dalam bentuk aturan seperti "Jika membeli A maka membeli B".	Tergantung pada metode, hasil bisa lebih sulit diinterpretasikan, misalnya vektor kesamaan.
Kebutuhan Data	Memerlukan data transaksional besar.	Memerlukan data pengguna yang cukup untuk menghasilkan rekomendasi yang akurat.

Pada penelitian ini, data yang digunakan adalah data historis transaksi dan tidak memiliki informasi data pengguna. Berdasarkan Tabel 3.1, ARM memiliki karakteristik yang lebih cocok untuk penelitian ini dibandingkan *Collaborative Filtering*. Hal ini tercermin pada jenis data yang digunakan, dimana ARM berfokus pada data transaksi perusahaan, sedangkan *Collaborative Filtering* berfokus pada data pengguna. Selain itu, metode ARM memiliki interpretasi hasil yang lebih mudah dipahami, sehingga memudahkan tim pemasaran dalam memahami hasil analisis.

Tujuan lainnya penelitian ini adalah menerapkan kombinasi ARM dan *clustering.* Dilakukan perbandingan antara beberapa algoritma *clustering*, seperti *K-Means*, *Gaussian Mixture Model*, dan *Spectral Clustering* untuk mencari algoritma yang paling cocok untuk penelitian ini. Hal ini merujuk pada perbandingan antara ketiga algoritma tersebut pada penelitian yang dilakukan oleh Husein, A., dkk [78]. Perbandingan algoritma *clustering* dapat dilihat pada Tabel 3.2:

Tabel 3. 2 Perbandingan Algoritma *Clustering*

Indikator	*K-Means*	GMM	Spectral Clustering
Prinsip Kerja	Membagi data ke dalam klaster berdasarkan centroid yang diminimalkan jaraknya dari semua titik dalam cluster.	Mengasumsikan data berasal dari beberapa distribusi *Gaussian* dan menemukan parameter distribusi tersebut untuk membentuk klaster.	Menggunakan eigenvektor dari matriks kesamaan data untuk mengurangi dimensi data dan mengelompokkan data dalam ruang yang dikurangi tersebut.
Kinerja *Dataset*	Cepat dan efisien untuk data besar.	Relatif cepat, tetapi lebih kompleks daripada K-Means.	Kurang efisien pada data yang sangat besar.
Parameter Tambahan	Jumlah klaster.	Jumlah klaster, parameter distribusi *Gaussian.*	Jumlah klaster, matriks kesamaan *(similarity matrix).*
Metode *Clustering*	Partisi (berbasis *centroid*)	*Model-Based* (berbasis probabilitas)	*Graph-Based* (berbasis grafik)

Berdasarkan Tabel 3.2, *K-Means* dan GMM lebih cocok untuk penelitian ini dari segi efisiensi mengolah data dalam jumlah yang besar. Hal ini dikarenakan algoritma *Spectral Clustering* memerlukan komputasi eigenvektor yang mahal. Penelitian ini menggunakan data yang besar dan kompleks, sehingga lebih cocok untuk mengimplementasikan *K-Means* dan GMM dibandingkan *Spectral Clustering*.

Ada beberapa kerangka kerja yang dapat digunakan untuk mendukung pelaksanaan proses *data mining*, seperti CRISP-DM (*Cross-Industry Standard Process for Data Mining*), SEMMA (*Sample, Explore, Modify, Model, Assess*), dan KDD (*Knowledge Discovery in Databases*). Perbandingan setiap kerangka kerja ditunjukkan pada Tabel 3.3:

Tabel 3. 3 Perbandingan Kerangka Kerja

Indikator	CRISP-DM	SEMMA	KDD
Tahapan	• *Business Understanding* • *Data Understanding* • *Data Preparation* • *Modeling* • *Evaluation* • *Deployment*	• *Sample* • *Explore* • *Modify* • *Model* • *Assess*	• Pemahaman Domain • Seleksi Data • Preprocessing Data • Transformasi Data

Indikator	CRISP-DM	SEMMA	KDD
			• *Data Mining* • Interpretasi dan Evaluasi Hasil
Kelebihan	• Pendekatan sistematis dan terstruktur • Memiliki dokumentasi yang kuat untuk setiap tahap • Dapat disesuaikan dengan berbagai industri	• Fokus pada analisis data yang lebih dalam dan eksploratif • Mudah diimplementasikan menggunakan perangkat lunak SAS • Cocok untuk analisis data yang tidak terlalu kompleks	• Pendekatan yang komprehensif untuk penemuan pengetahuan dari data • Memperhatikan tahapan pemahaman domain dan interpretasi hasil • Dapat menangani berbagai jenis data dan masalah di berbagai domain
Kekurangan	• Dukungan terhadap kolaborasi tim dan pengimplementasian *big data* tidak ada • Proses yang cenderung linear sehingga kurang fleksibel dalam menghadapi perubahan.	• Proses terbatas pada analisis, tidak masuk ke lingkup *deployment*. • Lebih cocok untuk analisis data sederhana	• Proses terbatas pada analisis, tidak masuk ke lingkup *deployment*. • Metode lama dan kurang cocok untuk implementasi analisis data modern. • Tidak dapat diterapkan pada analisis *big data.*

Berdasarkan Tabel 3.3 mengenai perbandingan CRISP-DM, SEMMA, dan KDD, CRISP-DM adalah metode *data mining* yang paling relevan untuk penelitian ini. Metode CRISP-DM memiliki tahap *deployment* yang sangat penting, dimana hasil dari analisis data dapat diimplementasikan ke dalam lingkungan *production* dengan lebih terstruktur dan terukur. Hal ini sesuai dengan tujuan akhir dari penelitian ini, yaitu mengimplementasikan rekomendasi penjualan ke dalam sistem perusahaan. Selain itu, CRISP-DM menawarkan pendekatan yang terstruktur dan sistematis, memastikan bahwa setiap tahapan dalam proses *data mining* dapat dijalankan dengan baik dan efisien [79].

Selain itu, metode CRISP-DM juga memiliki sifat yang iteratif. Hal ini merupakan hal yang penting mengingat data yang digunakan dalam penelitian ini terus bertambah setiap hari. Dengan pendekatan yang iteratif, penelitian dapat dilakukan secara fleksibel menyesuaikan analisis dengan perubahan dan penambahan data yang terjadi seiring waktu tanpa mengganggu keseluruhan proses. Hal ini memungkinkan untuk terus meningkatkan kualitas hasil analisis seiring dengan berkembangnya data yang tersedia. Berikut adalah proses implementasi CRISP-DM pada penelitian ini:

3.2.2.1 *Business Understanding*

Dalam tahap ini, fokus utama penelitian adalah memahami secara mendalam permasalahan bisnis yang dihadapi oleh perusahaan XYZ terkait dengan penyebaran newsletter produk kepada pelanggan. Melalui analisis yang teliti, telah teridentifikasi bahwa strategi penyebaran newsletter saat ini bersifat general dan tidak terarah, menyebabkan hasil pemasaran tidak mencapai potensi maksimal. Dengan pemahaman yang jelas mengenai kendala ini, penelitian dapat bergerak ke arah menyusun solusi yang lebih tepat sasaran.

3.2.2.2 *Data Understanding*

Pada langkah ini, penelitian fokus untuk memahami struktur dan komposisi data yang ada. Data mengenai pelanggan dan pembelian produk dikumpulkan dari database perusahaan XYZ. Analisis mendalam dilakukan untuk memahami karakteristik data dan menentukan kemungkinan analisis yang dapat dilakukan. Hal ini penting agar proses selanjutnya dapat memanfaatkan data secara efisien dan efektif.

3.2.2.3 *Data Preparation*

Setelah memahami data yang ada, langkah selanjutnya adalah mempersiapkan data untuk proses analisis. Ini melibatkan pengambilan data dari *database* perusahaan dan impor ke dalam lingkungan pengembangan. Selain itu, *data cleansing* dilakukan untuk menghilangkan data yang tidak lengkap atau tidak valid. Standardisasi data juga dilakukan jika diperlukan, untuk memastikan konsistensi dan kualitas data yang digunakan dalam proses analisis.

3.2.2.4 *Modeling*

Implementasi model-model seperti *Apriori*, ECLAT, dan FP-*Growth* menggunakan bahasa pemrograman *Python* menjadi fokus pada tahap ini. Dengan data yang telah disiapkan sebelumnya, parameter-parameter model disesuaikan dan performa masing-masing model diuji. Proses ini memungkinkan peneliti untuk mengeksplorasi dan membandingkan efektivitas model-model yang berbeda dalam menghasilkan rekomendasi penjualan yang lebih tepat sasaran.

3.2.2.5 *Evaluation*

Tahap evaluasi menjadi krusial untuk menentukan model terbaik yang akan digunakan dalam meningkatkan strategi penjualan. Model-model yang dihasilkan dari tahap *modeling* dievaluasi menggunakan metrik seperti *support*, *confidence*, dan *lift*. Dengan analisis yang teliti terhadap hasil evaluasi, peneliti dapat menentukan model yang paling sesuai untuk diterapkan dalam lingkungan bisnis perusahaan XYZ.

3.2.2.6 *Deployment*

Setelah model terbaik dipilih melalui tahap evaluasi, langkah selanjutnya adalah mendeploy model tersebut ke *website*. Proses implementasi ini bertujuan untuk memastikan bahwa tim pemasaran dapat dengan mudah mengakses dan memanfaatkan hasil insight dari analisis data yang telah dilakukan. Dengan menyajikan hasil analisis dalam *website*, tim pemasaran dapat dengan cepat mengidentifikasi pola pembelian yang

relevan, membuat keputusan yang lebih cerdas, dan mengarahkan strategi pemasaran dengan lebih tepat sasaran.

Proyek ini menggunakan *Laravel* sebagai *framework back-end* untuk mengelola logika bisnis dan interaksi dengan *database*. *Laravel* adalah *framework* PHP yang sangat populer dan kuat, yang menyediakan berbagai fitur untuk memudahkan pengembangan aplikasi web, termasuk autentikasi, routing, dan ORM (*Object-Relational Mapping*). Selain itu, proyek ini menggunakan *MinIO* sebagai alat untuk menyimpan file *Excel*. *MinIO* adalah sistem penyimpanan objek sumber terbuka yang kompatibel dengan *Amazon* S3, yang memungkinkan penyimpanan dan pengambilan file dalam skala besar dengan efisiensi tinggi. *Framework laravel* dipilih untuk menyesuaikan *framework* yang digunakan oleh perusahaan XYZ saat ini.

Pada bagian *front-end*, proyek ini menggunakan *Blade* sebagai *framework templating engine*. *Blade* adalah bagian dari Laravel dan menyediakan cara yang mudah dan intuitif untuk membangun tampilan web yang dinamis. Dengan *Blade*, pengembang dapat mencampurkan kode PHP ke dalam tampilan HTML tanpa perlu meninggalkan kesatuan sintaks HTML. Dengan menggunakan *Blade*, pembuatan tampilan web menjadi lebih efisien dan terstruktur.

Selain itu, proyek ini juga menjalankan skrip *Python* di dalam proyek *Laravel*. Hal ini memungkinkan pengembang untuk memanfaatkan kemampuan *Python* dalam melakukan tugas-tugas tertentu, seperti pemrosesan data atau interaksi dengan layanan pihak ketiga, sambil tetap menjaga kesatuan proyek dalam lingkungan *Laravel*. Kombinasi antara Laravel sebagai *back-end framework*, *Blade* sebagai *framework front-end*, dan penggunaan skrip *Python* menunjukkan pendekatan yang holistik dalam pengembangan aplikasi web yang kuat dan terstruktur.

3.3 Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan adalah pengambilan data dari histori transaksi perusahaan XYZ dengan menggunakan metode *random sampling*.

Proses ini dilakukan dengan mengambil sebagian data transaksi yang mencakup rentang waktu tertentu, misalnya beberapa hari terakhir, sebagai representasi dari keseluruhan histori transaksi perusahaan. Penggunaan sampel data ini memungkinkan untuk mengurangi kompleksitas dan waktu yang dibutuhkan dalam proses pengambilan dan analisis data, sambil tetap mempertahankan representativitas informasi dari data asli. Data transaksi perusahaan diambil dari AWS *Athena*, yaitu *data warehouse* perusahaan untuk menyimpan data riwayat transaksi. Pengambilannya menggunakan *query* SQL dengan memanfaatkan kolom yang sudah dipartisi agar proses *query* lebih cepat.

Setelah memilih rentang waktu yang relevan, data transaksi dipilih secara acak menggunakan metode *random sampling*. Proses pengambilan data ini dilakukan dengan memperhatikan kepentingan variabilitas dan representativitas data yang dipilih. Dengan demikian, teknik pengambilan data dari histori transaksi perusahaan XYZ menggunakan sampel data dari data *real* memungkinkan untuk mendapatkan wawasan yang cukup akurat tentang pola pembelian pelanggan dan tren bisnis tanpa harus memproses seluruh volume data transaksi yang ada.

3.3.1 Populasi dan Sampel

Populasi data dalam konteks ini adalah seluruh histori transaksi yang dimiliki oleh perusahaan XYZ sejak awal beroperasi hingga saat ini. Ini mencakup semua transaksi yang terjadi di semua cabang atau outlet perusahaan dan mencakup berbagai jenis produk yang dijual. Populasi pada data transaksi merupakan jumlah keseluruhan data pada *data warehouse* perusahaan XYZ untuk transaksi *prepaid*. Data ini dapat diakses pada tabel *general_report_prepaid* melalui AWS Athena. Tabel ini memiliki data transaksi prepaid dari tahun 2023-2024. Jumlah populasi pada penelitian ini tidak diketahui dikarenakan pencarian jumlah populasi memerlukan *query cost* yang besar.

Sementara itu, sampel data adalah *subset* dari populasi data yang dipilih untuk dianalisis. Dalam pengambilan sampel ini, dipilih transaksi-transaksi yang terjadi di perusahaan XYZ. Sampel ini mewakili sebagian kecil dari

populasi yang lebih besar dan diharapkan mencerminkan karakteristik umum dari populasi tersebut. Sampel yang digunakan penelitian ini adalah data transaksi perusahaan dari tanggal 13-20 Agustus 2023. Pengambilan pada tanggal ini digunakan dalam rangka menguji penelitian pada *dataset* yang lama, untuk mengurangi resiko keamanan data baru jika terjadi kebocoran data selama penelitian dilakukan. Penetapan tanggal 13-20 Agustus 2023 dilandasi dari permintaan perusahaan sebagai studi kasus penelitian ini.

3.3.2 Periode Pengambilan Data

Periode pengambilan data untuk analisis mencakup rentang waktu 13-20 Agustus 2023. Artinya, data transaksi yang diambil dari histori perusahaan XYZ mencakup semua transaksi yang terjadi mulai dari 13-20 Agustus 2023. Data yang digunakan termasuk jenis data primer. Teknik yang digunakan untuk mengambil sampel dari populasi adalah *random sampling*. Metode *random sampling* merupakan metode statistik dalam pengambilan sampel dimana setiap orang dalam suatu populasi mempunyai peluang yang sama untuk dipilih menjadi sampel. Pemilihan teknik *random sampling* dilakukan berdasarkan diskusi dengan perusahaan bahwa setiap data memiliki peran yang sama pentingnya dalam menganalisis data transaksi. Sehingga, penerapan *random sampling* sesuai dengan skenario ini, dimana setiap data memiliki kesempatan yang sama untuk dipilih menjadi sampel.

Dalam kasus ini, pemilihan rentang waktu pengambilan data diambil secara acak dengan durasi 8 hari. Pemilihan rentang waktu ini dilandaskan pada studi kasus perusahaan yang ingin menguji algoritma rekomendasi menggunakan data dalam rentang waktu singkat, yaitu 8 hari. Selain itu, pemilihan waktu 8 hari dikarenakan keterbatasan infrastruktur *hardware* peneliti karena data transaksi ini memiliki jumlah data yang sangat besar, kurang lebih 200 ribu data per hari. Selain itu, *query* data selama 8 hari dinilai lebih efisien dibandingkan pengambilan data dengan rentang waktu yang lebih lama. Hal ini dikarenakan setiap *query* yang dilakukan pada AWS Athena memiliki biaya (*query cost*).

Proses pengambilan data ini memungkinkan untuk menganalisis tren dan pola pembelian pelanggan selama periode tersebut, sehingga memungkinkan perusahaan untuk membuat keputusan yang lebih informasional berdasarkan informasi yang akurat dan terkini. Dalam konteks ini, pengambilan data dari tentang waktu tersebut dianggap sudah memberikan gambaran yang cukup komprehensif tentang aktivitas transaksi perusahaan selama beberapa tahun terakhir. Hal ini memungkinkan untuk mendapatkan wawasan yang lebih mendalam tentang preferensi pelanggan, performa produk, serta tren bisnis yang mungkin mempengaruhi strategi pemasaran dan penjualan perusahaan kedepannya. Dengan demikian, periode pengambilan data menjadi penting untuk memastikan analisis yang lebih terperinci dan relevan dengan kondisi aktual perusahaan.

3.4 Implementasi Model

Implementasi model pada penelitian ini harus menggunakan bahasa pemrograman yang kompatibel dengan masalah analisis data. *Python* dan *R* merupakan bahasa pemrograman yang paling populer jika dilihat dari penggunaannya untuk analisis data. Perbandingan antara *Python* dan *R* ditunjukkan pada Tabel 3.4 [80]:

Tabel 3. 4 Perbandingan *Python* dan R

Indikator	***Python***	***R***
Ketersedian *Library*	*Python* memiliki berbagai *library* dan paket yang kuat untuk analisis data seperti *Pandas*, *NumPy*, dan *Scikit-learn*. Selain itu, terdapat juga *library* visualisasi seperti *Matplotlib* dan *Seaborn.*	*R* juga memiliki koleksi paket dan *library* yang kaya seperti *dplyr*, *ggplot2*, dan *caret*. *R* dikenal memiliki keunggulan dalam visualisasi data dengan ggplot2.
Usability	*Python* dianggap lebih mudah dipelajari bagi pemula karena sintaksnya yang sederhana dan mudah dimengerti. Selain itu, *Python* juga memiliki dokumentasi yang baik dan banyak tutorial *online*.	*R* cenderung lebih kompleks dalam sintaksnya, terutama bagi pemula. Namun, *R* memiliki komunitas yang aktif dan ramah, serta banyaknya sumber belajar yang tersedia.
Performa	*Python* umumnya dianggap lebih cepat dalam eksekusi kode daripada *R*, terutama untuk pemrosesan data yang besar.	*R* bisa menjadi lambat dalam pemrosesan data besar karena beberapa faktor, meskipun dengan menggunakan paket yang tepat, performa dapat ditingkatkan.

Indikator	*Python*	*R*
Analisis Statistik	*Python* memiliki alat analisis statistik yang kuat seperti *SciPy* dan *StatsModels*.	*R* memiliki lebih banyak paket statistik dan algoritma analisis statistik yang tersedia secara *native*, sehingga sering menjadi pilihan utama bagi para ahli statistik
Komunitas dan Dukungan	*Python* memiliki komunitas yang besar dan aktif, dengan dukungan yang luas dari berbagai industri dan domain.	Meskipun komunitas R lebih kecil daripada *Python*, komunitasnya tetap sangat aktif terutama di bidang akademis dan penelitian.

Dalam konteks analisis data dan data science, penggunaan *Python* menjadi pilihan yang lebih menguntungkan dibandingkan dengan *R* atas beberapa alasan yang signifikan. Pertama, *Python* menawarkan fleksibilitas yang luar biasa dalam pengembangan aplikasi dan proses analisis data berkat ekosistem library dan paket yang kuat seperti Pandas, NumPy, dan Scikit-learn. Keberagaman pustaka ini memungkinkan para praktisi data untuk mengakses berbagai alat dan teknik analisis, dari pemrosesan data hingga pembuatan model machine learning, semuanya dalam satu bahasa pemrograman. Selain itu, *Python* juga populer di luar dunia data science, sehingga para profesional dapat memanfaatkan keterampilan yang mereka peroleh dalam berbagai bidang teknologi.

Selanjutnya, kemudahan pembelajaran dan penggunaan *Python* membuatnya menjadi pilihan yang lebih ramah bagi pemula maupun pengguna yang memiliki latar belakang pemrograman yang beragam. Sintaksis *Python* yang sederhana dan mudah dimengerti, bersama dengan dokumentasi yang luas dan banyaknya sumber belajar online, memungkinkan para pengguna untuk dengan cepat memahami dan mulai bekerja dengan bahasa ini. Hal ini berdampak positif pada efisiensi dan produktivitas tim, terutama dalam lingkungan kerja yang mungkin memiliki anggota dengan tingkat keterampilan yang berbeda.

Selain itu, performa yang relatif lebih cepat dalam eksekusi kode *Python*, terutama dalam pemrosesan data yang besar, memberikan keunggulan tambahan bagi *Python* sebagai pilihan utama dalam analisis data. Dalam lingkungan dimana kecepatan dan efisiensi sangat penting, *Python* memberikan kinerja yang stabil dan andal. Dengan demikian, keseluruhan ekosistem *Python* dalam hal fleksibilitas,

kemudahan penggunaan, dan performa yang baik menjadikannya pilihan yang unggul untuk proyek-proyek analisis data dan data science di berbagai industri dan domain.

3.5 Variabel Penelitian

Penelitian menggunakan metode ARM merupakan penelitian kuantitatif. Penelitian ini melibatkan analisis data untuk mencari dan menemukan pola dan hubungan antara variabel satu dengan yang lainnya dalam kumpul data. Tujuan utama penelitian ini adalah mencari hubungan antar *item* dalam sebuah data transaksi. Penelitian ini mengimplementasikan algoritma dan teknik statistik sehingga masuk ke dalam kategori penelitian kuantitatif. Selain itu, penelitian ini juga memanfaatkan algoritma *clustering* untuk dikombinasikan dengan ARM. Hal ini membuat variabel penelitian yang digunakan menjadi semakin meluas. Adapun beberapa variabel penelitian yang digunakan dalam mengimplementasikan ARM dan *clustering* adalah sebagai berikut:

3.5.1 *Association Rule Mining*

3.5.1.1 *Pulsa Code*

Pulsa code merupakan produk yang dibeli oleh pelanggan. *Pulsa code* berisi nama-nama produk. Produk-produk di dalamnya sangat variatif, mulai dari pulsa, data, voucher, game, dan lain lain. *Pulsa code* ini yang nantinya digunakan sebagai *input item* pada penelitian ARM. Untuk penelitian menggunakan *Apriori* dan *FP-Growth*, *pulsa code* ini akan di encode menjadi *true or false* agar dapat diproses. Pada algoritma ECLAT, setiap *pulsa code* akan ditransformasi menjadi kolom. *Pulsa code* merupakan variabel yang sangat penting dalam penelitian ARM.

3.5.1.2 *Transaction HP*

Transaction HP merupakan variabel yang digunakan untuk mengidentifikasi nomor pelanggan yang membeli produk pulsa. Dalam penelitian transaksi pulsa, setiap *transaction ID* hanya memiliki 1 produk. Hal ini dikarenakan proses belanja produk pulsa harus dilakukan satu per satu untuk setiap produk. Oleh karena itu, peneliti harus menerapkan *group*

and filter terhadap *transaction HP* bukan *transaction ID.* Jika terdapat *Transaction HP* yang muncul lebih dari 1 kali di dalam 1 hari yang sama, maka setiap *pulsa code* untuk masing-masing transaksi akan dikelompokkan dan dianggap sebagai 1 keranjang belanja. *Transaction HP* nantinya akan digunakan sebagai variabel untuk mengidentifikasi keranjang belanja yang unik.

3.5.2 *Clustering*

3.5.2.1 *Pulsa Price*

Pulsa price dalam konteks *clustering* digunakan untuk membagi data ke dalam klaster berdasarkan *pulsa price*. Tujuannya agar klaster dengan *pulsa price* kategori rendah, menengah, dan atas. Hal ini tentunya membantu tim pemasaran dalam memahami karakteristik pelanggan mereka secara lebih komprehensif.

3.5.2.2 *Pulsa Type*

Pulsa type dalam konteks *clustering* digunakan untuk membagi data ke dalam klaster berdasarkan *pulsa type* yang paling sering dibeli pelanggan di masing-masing klaster. Tujuannya agar tim pemasaran dapat mengetahui klaster pelanggan yang cenderung membeli tipe produk tertentu. Tim pemasaran dapat mengidentifikasi pelanggannya dengan baik dan strategi pemasaran dapat disesuaikan untuk setiap klaster.

3.5.2.3 *Supplier*

Supplier digunakan untuk mengetahui *supplier* mana yang paling sering menyediakan layanan/produk tertentu di setiap klaster. Dengan informasi ini, tim pemasaran dapat mengetahui *supplier* mana yang cocok untuk menyediakan layanan untuk klaster tertentu.

3.5.2.4 *Operator Name*

Operator name digunakan untuk mengetahui jenis operator yang paling mendominasi di masing-masing klaster. Tim pemasaran dapat mempromosikan jenis operator tertentu kepada pelanggan di masing-

masing klaster. Tujuannya agar promosi produk dari supplier dapat lebih fokus dan memberikan *output* penjualan yang lebih besar.

3.5.2.5 *Pulsa Code*

Pulsa code dalam konteks *clustering* digunakan untuk mengetahui produk pulsa apa yang cenderung dibeli oleh klaster pelanggan tertentu. Hal ini tentunya dapat mempermudah tim *pemasaran* dalam membuat *newsletter* terhadap setiap klaster pelanggan.