

Integrasi Metode Agglomerative Hierarchical Clustering dan Backpropagation Pada Model Peramalan Penjualan

Krisna atma wijaya¹, Daniel Swanjaya²

^{1,2}Teknik Informatika, Fakultas Teknik, Universitas Nusantara PGRI Kediri

E-mail: ¹picerbow60@gmail.com, ²daniel@unpkediri.ac.id

Abstrak – Artificial Neural Network (ANN) merupakan sebuah sistem cerdas yang digunakan untuk mengolah informasi yang merupakan perkembangan dari generalisasi model matematika, salah satu teknik yang sering digunakan untuk peramalan adalah Backpropagation. Banyak penelitian yang telah menggunakan Backpropagation untuk menyelesaikan masalah prediksi, tetapi kualitas peramalan yang didapat belum memuaskan. Penelitian ini bertujuan untuk memodifikasi model peramalan dengan menambahkan proses Clustering data sebelum dilakukannya proses peramalan. Dataset yang digunakan adalah data penjualan kopi sachet per bulan dari PT. Kapal Api. Data penjualan dinormalisasi, kemudian dibentuk menjadi Vektor Fitur dengan panjang tertentu, kemudian dikelompokkan menggunakan Agglomerative Hierarchical Clustering (AHC). Tiap kelompok hasil dari AHC diolah menggunakan Backpropagation sehingga didapat Jaringan yang telah siap digunakan. Jaringan Backpropagation digunakan untuk mendapatkan output, kemudian output tersebut dinormalisasi untuk mendapatkan nilai aktual. Nilai aktual dan nilai nyata dibandingkan untuk mendapatkan nilai Mean Absolute Percentage Error (MAPE). Hasil dari penelitian ini menunjukkan bahwa tingkat heterogenitas data sangat berpengaruh terhadap kualitas pengelompokan dan peramalan yang didapat. Pada penelitian ini nilai MAPE yang diperoleh dari proses peramalan yang mengintegrasikan AHC dan Backpropagation lebih baik dari pada proses peramalan yang hanya menggunakan Backpropagation saja.

Kata Kunci — tuliskan 3 sampai 5 kata menurut urutan alfabet dan dipisahkan dengan tanda koma

1. PENDAHULUAN

Kopi sachet adalah suatu ide baru bagi perusahaan kopi yang mempunyai penjualan yang baik di dalam negeri. Produksi tanaman kopi sendiri di Indonesia cukup luas. Prediksi atau peramalan merupakan salah satu cara efektif untuk memaksimalkan kualitas suatu perusahaan. Prediksi atau peramalan menjadi penting karena dalam susunan suatu rencana, salah satunya berdasar pada suatu prediksi atau peramalan. Oleh karena itu perusahaan wajib mengetahui apa yang akan terjadi dimasa yang akan datang [1]

Peramalan (prediksi) dilakukan dengan harapan dapat membantu perusahaan dalam mengambil keputusan yang tepat. Selain itu peramalan juga dapat membantu meningkatkan penjualan produk. Jaringan Syaraf Tiruan tidak memerlukan model matematis tetapi data dari masalah yang akan diselesaikan. Informasi disampaikan melalui data, dan Jaringan Syaraf Tiruan menyaring informasi tersebut melalui pelatihan. Oleh karena itu, Jaringan Syaraf Tiruan sangat tepat untuk menyelesaikan masalah prediksi penjualan suatu barang. Jaringan syaraf dapat diterapkan pada bidang prediksi. Data masa lalu diasumsikan seperti nilai-nilai fungsi. Jaringan syaraf membangun model fungsi yang menerangkan struktur dari data masa lalu. Fungsi tersebut menggambarkan

ketergantungan nilai data saat ini terhadap nilai data.

Jaringan Syaraf Tiruan memerlukan data dari masalah yang akan diselesaikan. Informasi disampaikan melalui data, dan Jaringan Syaraf Tiruan menyaring informasi tersebut melalui pelatihan. Oleh karena itu, Jaringan Syaraf Tiruan sangat tepat untuk menyelesaikan masalah prediksi penjualan kopi sachet.

Penelitian yang pernah diteliti sebelumnya Agglomerative Hierarchical Clustering Dengan Berbagai Pengukuran Jarak Dalam Mengklaster Daerah Berdasarkan Tingkat Kemiskinan[2]. Implementasi Agglomerative Hierarchical Clustering Pada Data Produksi dan Data Penjualan Perusahaan [3]. Penerapan Data Mining Pada Data Penjualan Sepatu Untuk Membentuk Segmentasi Distributor DI CV. XYZ Menggunakan Metode Clustering [4]. Prediksi Jumlah Penjualan Kredit Sepeda Motor Menggunakan Algoritma Backpropagation [5]. Jaringan Syaraf Tiruan Untuk Prediksi Penjualan Jamur Menggunakan Algoritma Backpropagation [6]. Jaringan Syaraf Tiruan Backpropagation Untuk Peramalan Penjualan Produk Studi Kasus Di Metro Elektronik Dan Furnitur [7].

Berdasarkan latar belakang masalah di atas, kualitas peramalan pada penelitian sebelumnya belum optimal, maka di ajukan

penelitian yang berjudul Integrasi Metode AHC Dan Backpropagation Pada Model Prediksi Penjualan. metode AHC digunakan untuk proses pengelompokan data sebelum proses peramalan. Setelah dilakukan pengelompokan atau sudah mendapatkan hasil berupa klaster, dilakukan proses peramalan dengan menggunakan metode *Backpropagation* untuk tiap kelompok yang terbentuk untuk mendapatkan hasil iterasi. Dengan menggunakan metode *Agglomerative Hierarchical Clustering* (AHC) dan *Backpropagation* ini dapat dijadikan sebagai cara yang efektif dalam peramalan suatu data.

2. METODE PENELITIAN

2.1 Agglomerative Hierarchical Clustering

Hierarchical methods adalah teknik clustering membentuk hirarki atau berdasarkan tingkatan tertentu sehingga menyerupai struktur pohon. Dengan demikian proses pengelompokannya dilakukan secara bertingkat atau bertahap. Biasanya, metode ini digunakan pada data yang jumlahnya tidak terlalu banyak dan jumlah cluster yang akan dibentuk belum diketahui. *Agglomerative* (metode penggabungan) adalah strategi pengelompokan hirarki yang dimulai dengan setiap objek dalam satu *cluster* yang terpisah kemudian membentuk *cluster* yang semakin membesar. Jadi, banyaknya cluster awal adalah sama dengan banyaknya *objek*. Sedangkan *Divisive* (metode pembagian) adalah strategi pengelompokan hirarki yang dimulai dari semua objek dikelompokkan menjadi cluster tunggal kemudian dipisah sampai setiap objek berada dalam cluster yang terpisah [8].

Agglomerative Hierarchical Clustering (AHC) ialah metode *hierarchical clustering* yang menyatukan N buah klaster sampai terbentuk jadi satu klaster tunggal. Metode ini diawali dengan menaruh setiap data sebagai satu klaster tersendiri (*atomic cluster*) setelah itu menyatukan beberapa klaster tersebut menjadi kluster yang lebih besar dan lebih besar lagi sampai akhirnya semua data menjadi satu pada sebuah klaster tunggal [8].

Algoritma *Agglomerative Hierarchical Clustering*.

1. Menghitung Matrik Jarak antara data.
2. Gabungkan dua kelompok terdekat berdasarkan parameter kedekatan yang ditentukan.
3. Perbarui Matrik Jarak antar data untuk merepresentasikan kedekatan diantara kelompok baru dan kelompok yang masih tersisa.

4. Ulangi langkah 2 dan 3 hingga hanya satu kelompok yang tersisa.

a. Average Linkage

Average Linkage adalah jarak rata-rata antara suatu objek. Pengelompokan sebuah klaster dimulai dari tengah atau pasangan suatu objek dengan jarak terdekat atau rata-rata. Pada awalnya harus ditemukan jarak terpendek dalam $D = \{d_{ik}\}$ dan untuk menggabungkan suatu objek yang selaras [9].

$$d_{uv} = \text{average}\{d_{uv}\}, d_{uv} \in D \dots\dots\dots(1)$$

b. Single Linkage

Single Linkage Dalam pengklusteran dengan metode single linkage dilakukan perhitungan jarak terdekat antara dua objek berdasarkan matriks jarak Euclidean yang telah terbentuk. Proses pengklusteran dilakukan berulang-ulang sampai tersisa tiga kluster.

$$d_{uv} = \min \{d_{uv}\}, d_{uv} \in D \dots\dots\dots(2)$$

c. Complete Linkage

Complete Linkage Dalam pengklusteran dengan metode complete linkage dilakukan perhitungan jarak terdekat antara dua objek berdasarkan matriks jarak Euclidean yang telah terbentuk. Proses pengklusteran dilakukan berulang-ulang sampai dengan tersisa tiga kluster.

$$d_{uv} = \max\{d_{uv}\}, d_{uv} \in D \dots\dots\dots(3)$$

2.2 Backpropagation

Backpropagation merupakan suatu bagian dari *Neural Network*. *Backpropagation* merupakan metode pelatihan terawasi (*supervised learning*), dalam artian mempunyai target yang akan dicari. ciri dari *Backpropagation* adalah meminimalkan error pada output yang dihasilkan oleh jaringan dalam metode *Backpropagation*, biasanya digunakan jaringan multilayer.

Backpropagation meliputi tiga fase yaitu: Fase I : Propagasi maju Selama propagasi maju, sinyal masukan x_i dipropagasikan kelapisan tersembunyi menggunakan fungsi aktivasi yang ditentukan. Keluaran dari setiap unit lapisan tersembunyi z_j tersebut selanjutnya dipropagasikan maju lagi kelapisan tersembunyi di atasnya menggunakan fungsi aktivasi yang ditentukan. Demikian seterusnya hingga menghasilkan keluaran jaringan y_k . Berikutnya, keluaran jaringan y_k dibandingkan dengan goal yang

harus dicapai t_k . Selisih $t_k - y_k$ adalah kesalahan yang terjadi. Jika kesalahan ini lebih kecil dari batas toleransi yang ditentukan, maka iterasi dihentikan. Akan tetapi apabila kesalahan masih lebih besar dari batas toleransinya, maka setiap bobot dalam jaringan akan dimodifikasi untuk mengurangi kesalahan yang terjadi.

Fase II : Propagasi mundur Berdasarkan kesalahan $t_k - y_k$, dihitung faktor $\delta_j, k = 1, 2, \dots, m$ yang dipakai untuk mendistribusikan kesalahan di unit y_k ke semua unit tersembunyi yang terhubung langsung dengan y_k . δ_k juga dipakai untuk mengubah bobot yang berhubungan langsung dengan unit keluaran. Dengan cara yang sama, dihitung faktor δ_j disetiap unit di lapisan tersembunyi sebagai dasar perubahan semua bobot yang berasal dari unit tersembunyi di lapisan di bawahnya.

Fase III : Perubahan bobot Setelah semua faktor δ dihitung, semua bobot akan dimodifikasi bersamaan. Perubahan suatu bobot didasarkan atas faktor δ neuron di lapisan atasnya. Sebagai contoh, perubahan bobot yang menuju ke lapisan keluaran didasarkan atas δ_k yang ada di unit keluaran. Ketiga fase tersebut diulang-ulang hingga kondisi penghentian terpenuhi. Secara umum kondisi penghentian diatur oleh jumlah iterasi dan toleransi kesalahan yang telah ditentukan. Iterasi dihentikan jika jumlah iterasi sudah sama dengan jumlah maksimum iterasi yang ditentukan, atau jika kesalahan yang terjadi sudah lebih kecil dari batas toleransi yang diperbolehkan [9].

1. Menginisialisasi semua bobot dengan bilangan acak kecil.
2. Meneruskan sinyal masukan ke unit tersembunyi di atasnya.
3. Menghitung semua keluaran di unit tersembunyi $z_j (j = 1, 2, \dots, p)$
4. Algoritma *Backpropagation*
 1. menginisialisasi bobot awal (pilih nilai secara acak yang cukup kecil)
 2. Selama kondisi masih berhenti, maka bernilai salah
5. Tahap perambatan-maju (*Forward Propagation*)
 1. Pada setiap unit input ($x_i, i = 1, 2, 3, \dots, n$) akan menerima sinyal x_i lalu akan meneruskan sinyal tersebut ke semua unit yang terdapat pada lapisan tersembunyi
 2. Pada setiap unit tersembunyi ($z_i, i = 1, 2, 3, \dots, p$) akan menjumlahkan bobot sinyal input dengan menggunakan persamaan 2.
$$z_{in_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \dots \dots \dots (3)$$

3. Lalu menerapkan fungsi aktivasi yang berguna untuk menghitung sinyal outputnya menggunakan persamaan 3.
$$z_i = f(z_{inj}) \dots \dots \dots (4)$$

4. Fungsi aktivasi yang biasa digunakan yaitu fungsi *sigmoid*, kemudian mengirimkan sinyal tersebut kesemua unit *output*

5. Pada setiap unit input ($y_{ki}, k = 1, 2, 3, \dots, m$) akan menjumlahkan bobot sinyal *input* dengan menggunakan persamaan 4.
$$y_{in_k} = w_{0k} + \sum_{i=1}^p z_{iw_{jk}} \dots \dots \dots (5)$$

6. Selanjutnya menerapkan fungsi aktivasi yang berguna untuk menghitung sinyal *output* dengan menggunakan persamaan 5.
$$y_k = f(y_{ink}) \dots \dots \dots (6)$$

6. Tahap perambatan-balik (*Backpropagation*)

1. Pada setiap unit *output* ($y_k, k = 1, 2, 3, \dots, m$) akan menerima pola target yang sesuai dengan pola input pelatihan, kemudian menghitung *error* dengan menggunakan persamaan 6.
$$\delta_k = (t_k - y_k) f'(y_{in_k}) \dots \dots \dots (7)$$

2. F' adalah turunan dari aktivasi, kemudian menghitung koreksi bobot dengan menggunakan persamaan 7.
$$\Delta w_{jk} = a \delta_j z_j \dots \dots \dots (8)$$

3. Selanjutnya menghitung koreksi bias dengan menggunakan persamaan 8.
$$\Delta w_{0j} = a \delta_k \dots \dots \dots (9)$$

4. Selanjutnya akan mengirimkan ke unit-unit yang ada dilapisan tersembunyi ($y_{i,i} = 1, 2, 3, \dots, p$) setelah itu menjumlahkan data input-inputnya (dari unit-unit yang berada pada lapisan yang di kanannya) dengan menggunakan persamaan 9.
$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \dots \dots \dots (10)$$

5. Selanjutnya menghitung informasi *error*, dengan cara dikalikan dari fungsi aktivasinya dengan menggunakan persamaan 10.
$$\delta_j = \delta_{in_j} f'(z - in_j) \dots \dots \dots (11)$$

6. Tahap selanjutnya menghitung koreksi bobot dengan menggunakan persamaan 11.
$$\Delta v_{jk} = a \delta_j x_j \dots \dots \dots (12)$$

7. Lalu menghitung koreksi bias dengan menggunakan persamaan 12.

$$\Delta v_{jk} = a\delta_j \dots \dots \dots (13)$$

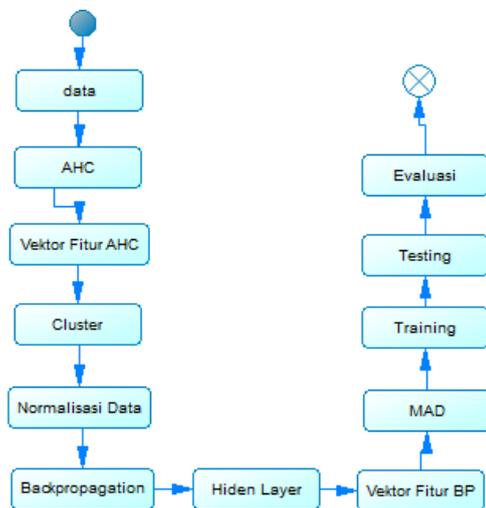
2.3 Perancangan Sistem

Pada bagian ini berisi sebuah perancangan yang ada pada suatu sistem yang akan dibuat untuk menghasilkan suatu sistem yang akan di implementasikan atau suatu sistem yang akan dijalankan. Berikut ini merupakan penjelasan dari perancangan sistem yang akan dibuat :

2.3.1 Activity Diagram

Activity Diagram dapat berguna untuk mendefinisikan rancangan aliran aktivitas atau aliran kerja dalam sebuah sistem yang akan dijalankan. *Activity Diagram* juga digunakan untuk mendefinisikan atau mengelompokkan aluran tampilan dari sistem tersebut. *Activity Diagram* memiliki komponen dengan bentuk tertentu yang dihubungkan dengan tanda panah. Panah tersebut mengarah ke-urutan aktivitas yang terjadi dari awal hingga akhir.

Activity Diagram dapat berfungsi untuk memperlihatkan urutan aktifitas proses pada sistem, membantu memahami proses secara keseluruhan, dan menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses.



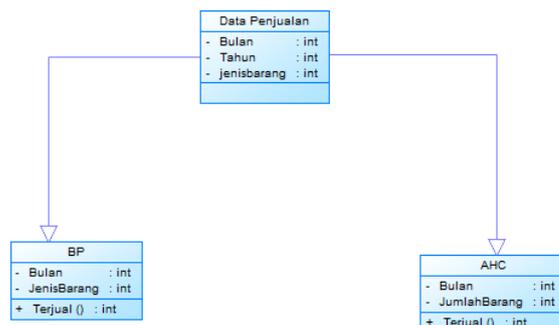
Gambar 2 Proses Activity Diagram

Pada gambar 2 merupakan proses *Activity Diagram* diawali dengan menginput data setelah itu data akan di olah menggunakan metode *AHC* untuk mendapat kan nilai vector fitur setelah itu di proses lagi untuk mendapat kan hasil cluster atau kelompok. Setelah metode *AHC* selesai, untuk tahap selanjutnya data

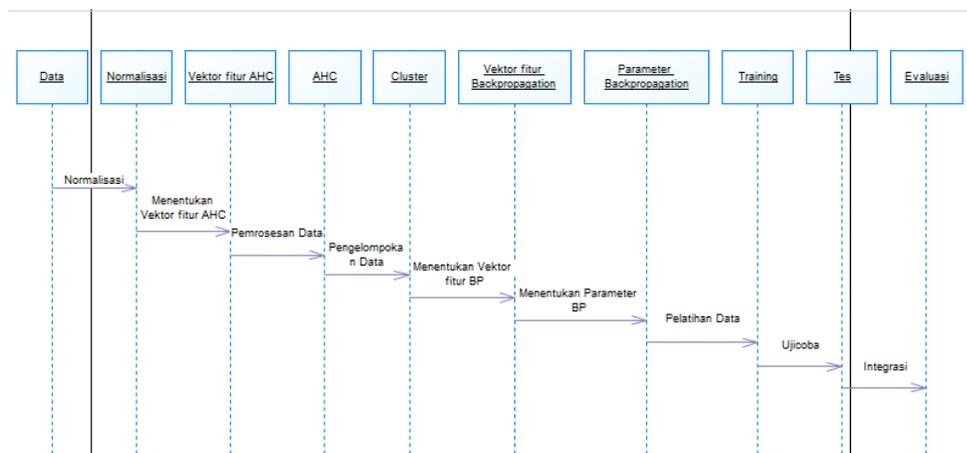
akan di normalisasi setelah data di normalisasi akan di proses dengan algoritma *Backpropagation*. Kemudian data akan di proses untuk menentukan hidden layer. Setelah hidden layer di temukan akan di proses Kembali untuk menentukan vector fitur *Backpropagation*, Kemudian menentukan nilai *Mean Absolute Percentage Error* (MAPE), setelah itu data akan dilatih dan menghasilkan data pengujian. Proses terakhir adalah hasil dari metode *AHC* dan *Bacpropagation*.

2.3.2 Class Diagram

Class diagram adalah jenis diagram struktur statis yang menggambarkan struktur suatu sistem dengan menunjukkan kelas sistem, atributnya, operasi, dan hubungan antar objek. *Class* adalah sebuah spesifikasi yang jika di-instanciasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). Diagram Kelas menggambarkan struktur dan deskripsi *class*, *package* dan *objek* beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. Diagram Kelas memberikan pandangan secara luas dari suatu sistem dengan menunjukan kelas-kelasnya dan hubungan mereka. *Class Diagram* bersifat statis; menggambarkan hubungan apa yang terjadi bukan apa yang terjadi jika mereka berhubungan. *Class diagram* dapat membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling ditemui dalam pemodelan sistem berbasis *object-oriented*. *Class Diagram* memperlihatkan sekumpulan *class*, *interface*, dan *collaborations* dan relasi yang ada di dalamnya. Selama proses analisa, *class diagram* memperhatikan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.



Gambar 3 Proses Class Diagram



Gambar 4 Proses Sequence Diagram

Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat. Kita memodelkan *class diagram* untuk memodelkan *static design view* dari suatu sistem.

Pada gambar 3 dijelaskan bahwa data penjualan yang sudah didapat akan diolah menggunakan metode *AHC* kemudian akan di proses menggunakan *Backpropagation*. Kemudian hasil dari metode *AHC* dan *Backpropagation* tersebut akan diintegrasikan.

2.3.3 Sequence Diagram

Diagram *sequence* merupakan salah satu yang menjelaskan bagaimana suatu operasi itu dilakukan; *message* (pesan) apa yang dikirim dan kapan pelaksanaannya. Diagram ini diatur berdasarkan waktu. Objek-objek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut. Diagram *sequence* menampilkan interaksi antar *objek* dalam dua dimensi. Dimensi vertikal adalah poros waktu, dimana waktu berjalan ke arah bawah. Sedangkan dimensi horizontal merepresentasikan *objek-objek* individual. Tiap *objek* (termasuk *actor*) tersebut mempunyai waktu aktif yang direpresentasikan dengan kolom vertikal yang disebut dengan *lifeline*. Pesan (*message*) direpresentasikan sebagai panah dari satu *lifeline* ke *lifeline* yang lain. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class*. Diagram *sequence* ini biasa

digunakan untuk menggambarkan skenario atau rangkaian langkah – langkah yang dilakukan sebagai respon dari sebuah event untuk menghasilkan output tertentu dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Diagram *sequence* mendeskripsikan bagaimana entitas dalam sistem berinteraksi, termasuk pesan yang digunakan saat interaksi. Semua pesan dideskripsikan dalam urutan dari eksekusi. Diagram *sequence* berhubungan erat dengan diagram *use case*, dimana 1 *use case* akan menjadi 1 diagram *sequence*.

Pada gambar 4 dijelaskan bahwa proses pertama yang dilakukan adalah membaca data kemudian dilakukan normalisasi data sehingga data dapat diolah. Selanjutnya, menentukan Vektor Fitur AHC. Kemudian, data akan diolah menggunakan algoritma AHC sehingga didapatkan hasil berupa cluster (kelompok). Proses kedua yaitu menentukan nilai hidden layer, vektor fitur dan mencari *Mean Absolute Percentage Error* (MAPE). Setelah didapatkan nilai pada proses tersebut kemudian dilakukan training sehingga didapatkan data testing. Setelah itu dilakukan proses evaluasi hasil dan di lakukan integrasi dari hasil kedua metode tersebut.

2.4 Mean Absolute Percentage Error

Mean Absolute Percentage Error atau lebih sering disebut dengan MAPE adalah perhitungan yang digunakan untuk menghitung rata-rata pada prosentase kesalahan yang mutlak. MAPE digunakan untuk mengevaluasi metode peramalan menggunakan jumlah dari kesalahan-kesalahan.

MAPE mengukur ketepatan peramalan dengan merata-rata kesalahan.

Untuk mengukur tingkat keakuratan peramalan menggunakan metode MAPE dihitung menggunakan persamaan 14.

$$MAPE = \frac{\sum \left| \frac{y_i - \hat{y}_i}{y_i} \right|}{n} \times 100\% \dots\dots\dots (14)$$

Keterangan

- y : Nilai Hasil Aktual
- \hat{y} : Nilai Hasil Prediksi
- n : Jumlah Data

3. HASIL DAN PEMBAHASAN

Penelitian ini bertujuan untuk membuat model peramalan atau prediksi penjualan dengan mengintegrasikan metode *Agglomerative Hierarchical Clustering (AHC)* dan *Backpropagation*. Untuk mencapai tujuan tersebut penulis menggunakan data penjualan Kopi Sachet per bulan dari PT.Kapal Api tahun 2016 sampai dengan 2020. Adapun proses *integrasi* tersebut menggunakan langkah-langkah sebagai berikut :

3.1 Sumber Data

Dalam penelitian ini, dataset didapat dari PT. Kapal Api yang merupakan rekap data penjualan bulanan Bolu Pisang selama 5 tahun dari tahun 2016 sampai dengan 2020. Pada dataset ini terdapat 9 varian Kopi Sachet, yaitu Arabika, Robusta dan Blend. Masing-masing varian memiliki 3 jenis kemasan yaitu 250 Gr, 500 Gr dan 1 Kg. Tabel 1 merupakan contoh data rekap penjualan tahun 2016.

Tabel 3. Data Penjualan Kopi Arabika Sachet, Tahun 2016.

| Bulan | 250gr | 500gr | 1000gr |
|-------|-------|-------|--------|
| Jan | 5542 | 911 | 1564 |
| Feb | 1541 | 832 | 2254 |
| Mar | 1565 | 1527 | 1231 |
| Apr | 664 | 5231 | 4312 |
| Mei | 2431 | 2191 | 3433 |
| Jun | 1547 | 2264 | 974 |
| Jul | 4121 | 2766 | 843 |
| Ags | 557 | 4376 | 1523 |
| Sep | 2274 | 1236 | 1434 |
| Okt | 2324 | 621 | 535 |
| Nov | 816 | 1769 | 965 |

| | | | |
|-----|------|-----|------|
| Des | 1453 | 618 | 2123 |
|-----|------|-----|------|

Tabel 4. Data Penjualan yang telah dinormalisasi.

| Bulan | 250gr | 500gr | 1000gr |
|-------|-------|-------|--------|
| Jan | 0,900 | 0,222 | 0,328 |
| Feb | 0,259 | 0,209 | 0,472 |
| Mar | 0,263 | 0,319 | 0,259 |
| Apr | 0,118 | 0,900 | 0,900 |
| Mei | 0,401 | 0,423 | 0,717 |
| Jun | 0,260 | 0,434 | 0,206 |
| Jul | 0,672 | 0,513 | 0,178 |
| Ags | 0,101 | 0,766 | 0,320 |
| Sep | 0,376 | 0,273 | 0,301 |
| Okt | 0,384 | 0,176 | 0,114 |
| Nov | 0,143 | 0,357 | 0,204 |
| Des | 0,245 | 0,176 | 0,445 |

Tabel 5. Contoh *Vektor Fitur* Kopi Arabika kemasan 250Gr, dengan panjang *vektor*, M = 5.

| No | X ₁ | X ₂ | X ₃ | X ₄ | X ₅ |
|----|----------------|----------------|----------------|----------------|----------------|
| 1 | 0,900 | 0,259 | 0,263 | 0,118 | 0,401 |
| 2 | 0,259 | 0,263 | 0,118 | 0,401 | 0,260 |
| 3 | 0,263 | 0,118 | 0,401 | 0,260 | 0,672 |
| 4 | 0,118 | 0,401 | 0,260 | 0,672 | 0,101 |
| 5 | 0,401 | 0,260 | 0,672 | 0,101 | 0,376 |
| 6 | 0,260 | 0,672 | 0,101 | 0,376 | 0,384 |
| 7 | 0,672 | 0,101 | 0,376 | 0,384 | 0,143 |
| 8 | 0,101 | 0,376 | 0,384 | 0,143 | 0,245 |
| 9 | 0,376 | 0,384 | 0,143 | 0,245 | 0,291 |
| 10 | 0,384 | 0,143 | 0,245 | 0,291 | 0,277 |
| 11 | 0,143 | 0,245 | 0,291 | 0,277 | 0,138 |
| 12 | 0,245 | 0,291 | 0,277 | 0,138 | 0,142 |

3.2 Praproses

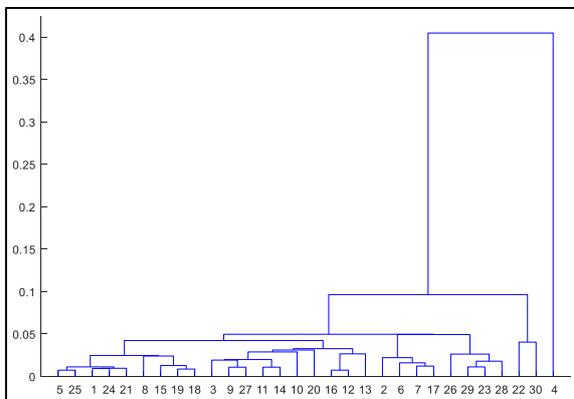
Tahap pre-processing atau praproses data merupakan proses untuk mempersiapkan data mentah sebelum dilakukan proses lain. Sebelum data penjualan diolah, data tersebut perlu ditransformasi atau *normalisasi* menggunakan metode min-max (persamaan 15) sehingga domain data penjualan menjadi 0,1 hingga 0,9, seperti tabel 2.

$$X_{new} = 0.1 + 0.8 \times \frac{x_{old-min}}{max-min} \dots\dots\dots (15)$$

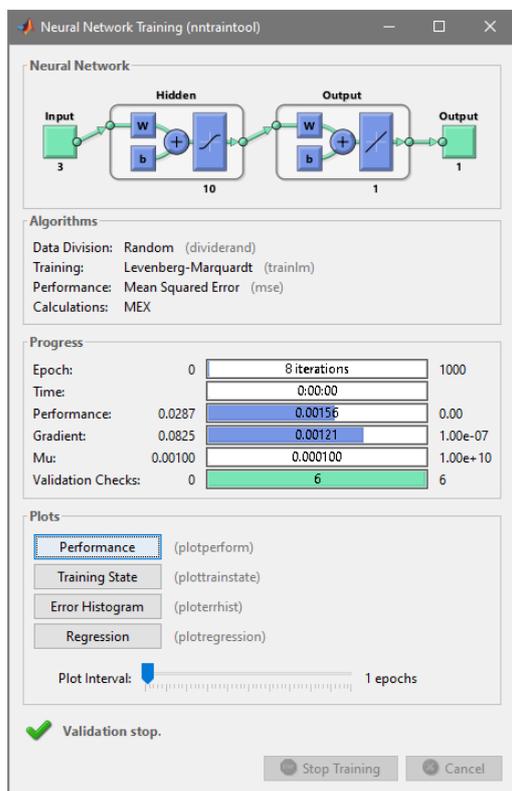
Setelah proses transformasi selesai, perlu dibuat *Vektor Fitur*, dimana data penjualan didistribusikan ke bentuk *vektor* dengan panjang M. Tabel 3 adalah *Vektor Fitur* dengan M = 5, dimana kolom ke-1 berisi penjualan bulan ke-1,

kolom ke-2 berisi penjualan bulan ke-2, hingga kolom ke-5 yang berisi penjualan bulan ke-5.

Pada penelitian ini masing-masing varian dan kemasan Kopi Sachet dibuat Vektor Fiturnya, sehingga terdapat 6 Vektor Fitur.



Gambar 5. Dendrogram pengelompokan AHC



Gambar 6. Tampilan Neural Network Training Backpropagation salah satu eksperimen

3.3 Proses Pengelompokan

Vektor Fitur yang telah dibuat masing-masing dikelompokkan menggunakan metode *Agglomerative Hierarchical Clustering* (AHC). Pada penelitian ini penulis menggunakan Matlab 2017a sebagai aplikasi bantuan, dengan fungsi bawaan dari Matlab yaitu *linkage*, dengan parameter utama jarak antar data, dan parameter opsional metode perhitungan jarak antar cluster.

Jarak antar data dihitung dengan fungsi bawaan Matlab yaitu *pdist*. metode perhitungan jarak antar cluster yang disediakan oleh Matlab adalah *average, centroid, complete, median, single, ward* dan *weighted*.

Urutan proses pengelompokan data oleh Matlab disajikan dalam bentuk *dendrogram*. *Dendrogram* adalah representasi visual dari langkah-langkah dalam analisis *cluster* yang menunjukkan bagaimana *cluster* terbentuk dan nilai koefisien jarak pada setiap langkah. Angka disebelah kanan adalah obyek penelitian, dimana obyek-obyek tersebut dihubungkan oleh garis dengan obyek yang lain sehingga pada akhirnya akan membentuk satu *cluster*. Gambar 5 adalah contoh dendrogram yang dihasilkan oleh fungsi bawaan Matlab, dimana sumbu x adalah nomor data atau *cluster* dan sumbu y adalah jarak antar *cluster*.

3.4 Proses Peramalan

Setelah data dikelompokkan menggunakan SOM, berikutnya untuk setiap kelompok dilakukan proses pembentukan jaringan *Backpropagation*. Pada proses pelatihan jaringan *Backpropagation*, masukan yang digunakan adalah *Vektor Fitur* kolom X_1, X_2, \dots, X_{M-1} dan Targetnya adalah X_M . Misal jika $M = 5$, maka masukannya adalah X_1, X_2, X_3, X_4 dan targetnya adalah X_5 .

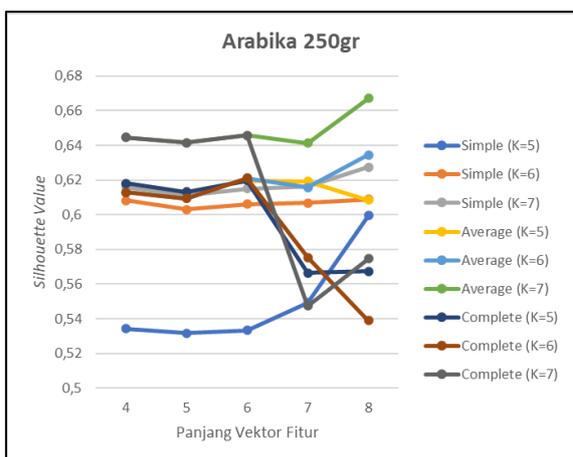
Pada penelitian ini Matlab 2017a digunakan sebagai alat bantu, dimana untuk membangun jaringan *Backpropagation* digunakan fungsi bawaan dari matlab yaitu *feedforwardnet*, dengan parameter banyaknya *hidden layer* yang akan digunakan untuk proses pelatihan. Fungsi *train* untuk pelatihan jaringan yang telah diinisiasi, yaitu data latih. Fungsi *sim* untuk mendapatkan keluaran dari masukan yang diberikan. Gambar 6 adalah tampilan *Neural Network Training* hasil dari pelatihan jaringan *Backpropagation* salah satu eksperimen. Setelah proses pelatihan jaringan selesai, berikutnya adalah proses evaluasi menggunakan *Mean Absolute Percentage Error* (MAPE). Karena domain keluaran dari jaringan *Backpropagation* adalah $[0, 1]$, antara 0 hingga 1, maka perlu dilakukan transformasi data menggunakan persamaan 14 untuk mendapatkan nilai nyatanya, proses ini disebut sebagai denormalisasi. Setelah hasil nyata didapatkan, berikutnya dibandingkan dengan nilai yang sesungguhnya sehingga didapat kualitas peramalan menggunakan metode MAPE.

3.5 Pengujian

Pada penelitian ini dilakukan beberapa pengujian, yaitu :

3.5.1 Pengujian pengelompokan menggunakan AHC dengan Metode *Silhouette*

Metode *Silhouette* menggabungkan dua metode yaitu metode *cohesion* yang berfungsi untuk mengukur seberapa dekat relasi antara objek dalam sebuah cluster, dan metode *separation* yang berfungsi untuk mengukur seberapa jauh sebuah cluster terpisah dengan cluster lain.



Gambar 7. Grafik perbandingan nilai Silhouette Kopi Arabika kemasan 250 Gr.

Tabel 6. Perbandingan Pengelompokan semua varian & kemasan Kopi.

| Varian & Kemasan | Silhouette value | Panjang Vektor | Banyak Cluster | Metode |
|------------------|------------------|----------------|----------------|----------|
| Arabika 250gr | 0,6672 | 8 | 7 | Average |
| Arabika 500gr | 0,6609 | 4 | 6 | Single |
| Arabika 1Kg | 0,6668 | 4 | 5 | Single |
| Robusta 250gr | 0,6825 | 7 | 7 | Complete |
| Robusta 500gr | 0,6494 | 6 | 6 | Single |
| Robusta 1Kg | 0,6626 | 8 | 7 | Average |
| Blend 250gr | 0,6650 | 6 | 6 | Single |
| Blend 500gr | 0,6831 | 6 | 7 | Average |
| Blend 1Kg | 0,6545 | 8 | 6 | Average |

Pada penelitian ini data yang akan dikelompokkan adalah *Vektor Fitur* yang merupakan data penjualan yang disusun dengan pola terurut, dimana panjang *Vektor* (M) ditentukan secara acak. Banyak kelompok (K) pada AHC ditentukan secara acak, namun memiliki batasan yang ditentukan dengan formula Sturgess.

Pengujian ini bertujuan untuk mendapatkan nilai M dan K serta metode perhitungan jarak antar cluster yang tepat pada proses pengelompokan menggunakan SOM, untuk semua varian & kemasan Kopi. Metode perhitungan jarak antar cluster yang digunakan adalah *single*, *average* dan *complete*. Domain nilai K

yang digunakan 5 hingga 7 (formula *Sturgess*). Domain nilai M yang digunakan adalah 4 hingga 8. Gambar 7 adalah Grafik perbandingan nilai *Silhouette* untuk Kopi Arabika kemasan 250 Gr, dari grafik tersebut didapat nilai *Silhouette* terbaik didapat dari Metode Average, M = 8 dan K = 7 dengan nilai 0.6672, untuk varian lainnya dilakukan perbandingan serupa dan didapat hasil seperti tabel 6, dimana parameter pengelompokan tiap varian & kemasan berbeda-beda.

3.5.2 Pengujian proses peramalan menggunakan *Backpropagation* dengan Metode MAPE

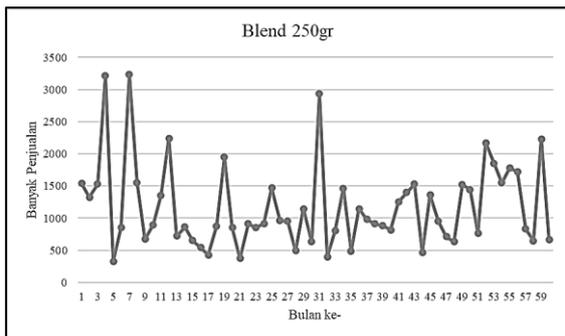
Proses peramalan pada penelitian ini menggunakan metode *Backpropagation*, dengan data masukannya adalah *Vektor Fitur* hasil pengelompokan *Agglomerative Hierarchical Clustering* (AHC) tiap varian & kemasan Kopi. Atribut *Vektor Fitur* dibagi menjadi 2, yaitu input dan target, dimana inputnya adalah X_1, X_2, \dots, X_{M-1} dan targetnya adalah X_M , dimana M adalah panjang vektor fitur.

Proses pengujian ini, diawali dengan proses pembentukan jaringan pada setiap kelompok hasil dari AHC, kemudian dilakukan denormalisasi sehingga didapat nilai nyatanya. Berikutnya perhitungan nilai MAPE dengan membandingkan hasil dari *Backpropagation* dan nilai yang sesungguhnya.

Nilai MAPE dari tiap kelompok dari tiap varian & kemasan seperti pada tabel 7. Prosentase Error tertinggi didapat dari peramalan penjualan kopi Blend 250 Gr, sebesar 8.75 %, hal ini disebabkan oleh tingkat *heterogenitas* data penjualan dari Kopi Blend 250 Gr yang terlalu tinggi dengan rata-rata 1178,5 dan nilai Standart Deviasi sebesar 653.294, seperti terlihat pada gambar 8.

Tabel 7. Nilai MAPE peramalan tiap varian & Kemasan Kopi

| Varian & Kemasan | MAPE (%) |
|------------------|----------|
| Arabika 250gr | 2,72 |
| Arabika 500gr | 4,57 |
| Arabika 1000gr | 7,91 |
| Robusta 250gr | 5,93 |
| Robusta 500gr | 8,62 |
| Robusta 1000gr | 8,57 |
| Blend 250gr | 8,75 |
| Blend 500gr | 6,15 |
| Blend 1000gr | 5,25 |



Gambar 8. Grafik data penjualan Kopi Blend 250 Gr.

Tabel 8. Nilai MAPE permalan menggunakan *Backpropagation* untuk tiap varian & kemasan kopi.

| Varian & Kemasan | MAPE (%) |
|------------------|----------|
| Arabika 250gr | 59.86 |
| Arabika 500gr | 44.22 |
| Arabika 1000gr | 48.74 |
| Robusta 250gr | 69.26 |
| Robusta 500gr | 46.31 |
| Robusta 1000gr | 51.29 |
| Blend 250gr | 43.88 |
| Blend 500gr | 45.42 |
| Blend 1000gr | 44.13 |

3.5.3 Perbandingan model peramalan menggunakan *Backpropagation* dengan model peramalan *AHC* dan *Backpropagation*

Pada penelitian sebelumnya sistem peramalan hanya menggunakan *Backpropagation* dan kualitas peramalan yang didapan belum memuaskan. Pada penelitian ini sistem peramalan dimodifikasi, sehingga sebelum dilakukan peramalan menggunakan *Backpropagation*, data dikelompokkan terlebih dahulu sehingga data awal yang heterogen dapat dijadikan lebih homogen. Tujuan perbandingan ini adalah untuk mengetahui apakah modifikasi sistem peramalan dengan mengintegrasikan *AHC* dan *Backpropagation* lebih baik dari sistem peramalan yang hanya menggunakan *Backpropagation* saja.

Pada proses peramalan yang hanya menggunakan *Backpropagation* saja didapat nilai MAPE untuk tiap varian & kemasannya seperti pada tabel 8.

Prosentase *Error* yang didapat sangat tinggi jika dibandingkan dengan nilai MAPE dari proses peramalan yang mengintegrasikan *AHC* dan *Backpropagation*. Hal ini membuktikan bahwa modifikasi sistem peramalan yang dilakukan telah berhasil memperbaiki kualitas peramalan sistem yang sebelumnya.

4 SIMPULAN

Berdasarkan hasil penelitian dan pembahasan tentang Integrasi Metode *Agglomerative Hierarchical Clustering* (*AHC*) dan *Backpropagation* pada Model Peramalan Penjualan dapat ditarik kesimpulan sebagai berikut :

1. Integrasi metode *AHC* dan *Backpropagation* pada model peramalan penjualan mampu memperbaiki kualitas peramalan dari sistem permalan sebelumnya yang hanya menggunakan metode *Backpropagation* saja.
2. Pada proses pengelompokan didapat kombinasi Metode perhitungan jarak antar *cluster*, panjang Vektor Fitur (*M*) dan banyak kelompok (*K*) yang terbaik seperti tabel 6, dimana nilai *Silhouette Coefficient* terbaik yang didapat adalah 0.6831, untuk pengelompokan data penjualan Kopi Blend 500 Gr.
3. Pada proses peramalan didapat nilai *Mean Absolute Precentage Error* (*MAPE*) dari tiap Varian & Kemasan seperti pada tabel 7. Nilai *MAD* ini lebih baik dari pada sistem yang hanya menggunakan *Backpropagation* untuk sistem peramalannya, yang memiliki nilai *MAPE* seperti pada tabel 8.

5 SARAN

Penelitian Integrasi Metode *Agglomerative Hierarchical Clustering* (*AHC*) dan *Backpropagation* pada Model Peramalan Penjualan hanya menggunakan 1 kali percobaan pada arsitektur jaringan *Backpropagation* (proses peramalan), sehingga untuk penelitian selanjutnya dapat menggunakan lebih banyak percobaan agar dapat memperoleh model yang lebih baik.

DAFTAR PUSTAKA

- [1] P. Konsumsi, M. Universitas, T. Kopi, B. Instan, and S. Saji, "Jiia, volume 1, no. 4, oktober 2013," vol. 1, no. 4, pp. 326–333, 2013.
- [2] E. Suherman, "Agglomerative Hierarchical Clustering Dengan Berbagai Pengukuran Jarak Dalam Mengklaster Daerah Berdasarkan Tingkat Kemiskinan," vol. 5, no. 1, pp. 978–979, 2019.

- [3] E. ARISKA, "Implementasi Agglomerative Hierarchical Clustering Pada Data Produksi Dan Data Penjualan Perusahaan," pp. 4–16, 2018.
- [4] A. M. B. Ririn Patmawati Sari, "PENERAPAN DATA MINING PADA DATA PENJUALAN SEPATU MENGGUNAKAN METODE CLUSTERING Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)," vol. 5, no. 2, 2016.
- [5] F. Dristyan, "Prediksi Jumlah Penjualan Kredit Sepeda Motor Menggunakan Algoritma Backpropagation," *Semin. Nas. R.*, vol. 1, no. 1, pp. 185–190, 2018.
- [6] Y. D. Lestari, "Jaringan syaraf tiruan untuk prediksi penjualan jamur menggunakan algoritma backpropagation," *J. ISD*, vol. 2, no. 1, pp. 40–45, 2017.
- [7] Welnof Satria, "Jaringan Syaraf Tiruan Backpropagation Untuk Peramalan Penjualan Produk (Studi Kasus Di Metro Electronic Dan Furniture)," *Djtechno J. Inf. Technol. Res.*, vol. 1, no. 1, pp. 14–19, 2020.
- [8] N. Pitaloka, "Pengelompokan Data Menggunakan Hierarchical Clustering (AHC)," 2009.
- [9] M. Barrimi *et al.*, "濟無No Title No Title," *Encephale*, vol. 53, no. 1, pp. 59–65, 2013, [Online]. Available: <http://dx.doi.org/10.1016/j.encep.2012.03.001>.