Journal of Information System, Applied, Management, Accounting and Research
JISAMAR
e-ISSN: 2598-8719 (Online)
p-ISSN: 2598-8700 (Printed)
Vol. 4 No. 3 , Agustus 2020

KOMPARASI METODE *CLUSTERING K-MEANS* DAN *SINGLE LINKAGE* UNTUK PENENTUAN KELOMPOK *AGENT* PADA *CALL CENTER*

Sapriyanti[1], Yan Rianto[2]
Program Magister Ilmu Komputer[1,2]
Program Studi Ilmu Komputer[1,2]
STMIK Nusa Mandiri[1,2]
Sapriyanti.85@gmail.com[1], yan_rianto@nusamandiri.ac.id[2]

ABSTRAK

Pada penelitian ini, dilakukan perhitungan manual dan penerapan menggunakan aplikasi Rapidminer sebagai komparasi *clustering* metode *K-Means* dan *Single Linkage* untuk mengetahui jumlah pengelompokan *agent contact center* pada PT XYZ. Dimana pengelompokan dilakukan berdasarkan standar yang telah ditentukan, sedangkan pengelompokan menggunakan metode *Single Linkage* mengacu pada jarak terdekat antar parameter *Agent*. Dari hasil pengujian didapatkan bahwa pada saat proses pengelompokan menggunakan metode *K-Means* dan metode *Single Linkage*, terdapat perbedaan dalam hal jumlah *Agent* dalam sebuah *cluster*, anggota pada masing masing *cluster*, dan waktu eksekusi untuk membangun sebuah *cluster*. Dalam hal waktu eksekusi untuk membangun sebuah *cluster* dengan 20 *agent contact center*, metode *K-Means* lebih unggul dibanding dengan metode *Single Linkage*. Dengan mencari nilai *centroid distance*, didapatkan bahwa *cluster* yang terbentuk dari metode *K-Means* lebih ideal dibanding metode *Single Linkage*. Dimana jumlah member pada *cluster* dengan metode *K-Means* yaitu C0 (3), C2 (9) dan C3 (8) sedangkan dengan metode *Single Linkage* yaitu C0 (8), C2 (9) dan C3 (3).

Kata kunci: pusat telepon; *K-Means*; metode komparasi; single linkage.

Abstract: *In this study, a manual calculation and application support were used using Rapidminer application as a K-Means and Single Linkage comparison grouping method to determine the number of agents contact center grouping at PT XYZ. Where the grouping is done according to predetermined standards, while grouping using the Single Linkage method is determined at the closest distance between the Agent parameters. From the test results obtained during the grouping process using the K-Means method and the Single Linkage method, it is a difference in the number of Agents in one cluster, members in each cluster, and the implementation time to create a cluster. In terms of execution time to building a cluster with 20 agent contact centers, the K-Mean method is superior to the Single Linkage method. By finding the centroid distance value, a cluster formed by the K-Mean method is more ideal than the Single Linkage method. Where the number of members in the cluster using the K-Means method are C0 (3), C2 (9) and C3 (8) and the Single Linkage method is C0 (8), C2 (9) and C3 (3)..*

Keywords: *Call Center;k-mean;compare method; single linkage.*

I. PENDAHULUAN

Saat ini, kita memasuki era teknologi informasi yang begitu maju dan pesat berkembang, banyak orang yang menginginkan cara pengaksesan informasi yang cepat dan mudah. Salah satu cara yang bisa dilakukan untuk memberikan informasi adalah dengan membangun sebuah sistem *Call Centre* (Telepon terpusat). Untuk menunjang kegiatan operasional perusahaan saat ini, perusahaan sudah harus memiliki Call Centre, dimana fungsi Call Centre tersebut digunakan untuk beberapa hal antara lain pemberian informasi, penanganan kritik dan saran serta untuk melakukan panduan jika terjadi masalah pada perusahaan. Dalam rangka menerapkan *Good Governance* dimana sebuah perusahaan besar memiliki system *Call Center* guna mendukung kegiatan operasional perusahaan .

Sistem *Call Center* yang ada saat ini masih banyak yang menggunakan system PABX analog sebagai sistem switching. Padahal PABX analog

dengan jumlah jalur yang banyak harus dibeli dengan biaya lumayan mahal. Pada penelitian ini dilakukan komparasi antara metode *clustering K-Means* dengan *Single Linkage*, dimana dilakukan penerapan dengan aplikasi Rapidminer sebagai visualisasinya. Perhitungan yang dibuat diolah dengan metode manual, *K-Means* dan *Single Linkage* untuk pengelompokan agent sebagai upaya untuk melakukan monitoring terhadap beban kerja setiap *Agent*[2].

II. METODE DAN MATERI

Fitriyadi pada tahun 2010 telah membuat sebuah layanan informasi kemahasiswaan berbasis IP PBX,dimana sistem yang dibuat menggunakan software Asterisk dan pemrograman PHP AGI[1]. Sulija Hidayati pada tahun 2011 telah membandingkan dua metode analisis *cluster* yaitu metode *Single Linkage* dengan metode *K-Means* serta mengetahui langkah-langkah dari kedua metode tersebut. Data yang dipakai pada penelitian ini untuk proses peng*cluster*an dari kedua metode *Single Linkage* dan metode *K-Means* yaitu empat rasio profitabilitas dari 27 bank yang terdiri dari rasio BEP, ROE, ROA, dan NPM[3].

Nur Rosyid Mubtada'I, dkk pada tahun 2011 telah membuat Analisa Perbandingan *clustering* metode manual dan metode *Single Linkage* untuk menentukan kinerja *Agent* pada *Call Center* berbasis Asterisk for Java. Penelitian ini merupakan pengembangan dari tiga penelitian diatas, dimana pada penelitian ini membuat Perhitungan Manual dari metode *K-Means* dan *Single Linkage* untuk mengetahui pengelompokan *Agent* dengan di implementasikan visualisasinya pada Aplikasi Rapidminer.

2.1. *Call Center*

Call Centre merupakan suatu wadah informasi yang terpusat yang digunakan untuk tujuan menerima dan mengirimkan sejumlah besar permintaan melalui telepon. *Call Centre* dioperasikan oleh sebuah perusahaan sebagai pengadministrasi layanan yang mendukung produk *incoming* dan menyelidiki informasi tentang konsumen. Beberapa panggilan keluar *Call Centre* digunakan untuk telemarketing, dan *debt collection*. Sebagai tambahan untuk *Call Centre*, bahwa penanganan secara kolektif untuk surat, fax, dan email dalam sebuah lokasi lebih sering disebut dengan *contact center*[5].

2.2. Rapidminer

Rapidminer adalah platform perangkat lunak ilmu data yang dikembangkan oleh perusahaan bernama sama dengan yang menyediakan lingkungan terintegrasi untuk persiapan data, pembelajaran mesin, pembelajaran dalam, penambangan teks, dan analisis prediktif. Hal ini digunakan untuk bisnis dan komersial, juga untuk penelitian, pendidikan, pelatihan, *rapid prototyping*, dan pengembangan aplikasi serta mendukung semua langkah dalam proses pembelajaran mesin termasuk persiapan data, hasil visualisasi, validasi model, dan optimasi. RapidMiner dikembangkan pada model inti terbuka. Dengan RapidMiner Studio Free Edition, yang terbatas untuk 1 prosesor logika dan 10.000 baris data, tersedia di bawah lisensi.

2.3. *Call Detail Record (CDR)*

Call Detail Record (CDR) pada teknologi *Voice over IP* (VoIP) berisi data-data tertentu yang berkaitan dengan panggilan-panggilan yang dilakukan. Data-data tersebut diantaranya[2] :

- *Account code* : nomor *account* yang digunakan
- src : nomor identitas pemanggil
- dst : ekstensi tujuan, dll

Pada penelitian ini, CDR digunakan untuk mencatat beberapa parameter untuk menentukan kinerja agent yang meliputi *N Inbound*, dan *Login Time*.

2.4. Parameter Penentu Kinerja *Agent*

Beberapa parameter penentu kinerja agent. Meliputi[5]:

- *N-Inbound*
  N-inbound adalah jumlah panggilan yang diterima selama selang waktu tertentu.

$$Ntotal = n + Ntolak \quad \textbf{(1)}$$

N_{total} : jumlah seluruh panggilan yang masuk
n : jumlah panggilan yang diterima(N)
N_{tolak} : jumlah panggilan yang ditolak

- *Login Time.*
  Login Time adalah waktu yang dipergunakan oleh *Agent* untuk login.
  Dimana:
  t : Login Time
  Lo : waktu logout
  Li : waktu login
  Is : lama waktu istirahat

Nilai total dari kedua parameter tersebut adalah :

$$t = L0 - Li - ls \qquad (2)$$

2.5. *Clustering* dengan Metode *K-Means*

K-Means adalah suatu metode penganalisaan data atau metode *Data Mining* yang melakukan proses pemodelan tanpa supervisi *(unsupervised)* dan merupakan salah satu metode yang melakukan pengelompokan data dengan sistem partisi[6]. Metode *k-means* berusaha mengelompokkan data yang ada ke dalam beberapa kelompok, dimana data dalam satu kelompok mempunyai karakteristik yang sama satu sama lainnya dan mempunyai karakteristik yang berbeda dengan data yang ada di dalam kelompok yang lain. Dengan kata lain, metode ini berusaha untuk meminimalkan variasi antar data yang ada di dalam suatu *cluster* dan memaksimalkan variasi dengan data yang ada di *cluster* lainnya.

Objective function yang berusaha diminimalkan oleh k-means adalah:

$$J(U,V) = \sum_{k=1}^{N}\sum_{i=1}^{c} a_{ik}\, D(x_{k,}v_i)^2 \qquad (3)$$

dimana:
U : Matriks keanggotaan data ke masing-masing *cluster* yang berisikan nilai 0 dan 1
V : Matriks centroid/rata-rata masing-masing *cluster*
N : Jumlah data
c : Jumlah *cluster*
a_{ik} : Keanggotaan data ke-k ke *cluster* ke-i
x_k : data ke-k
v_i : Nilai centroid *cluster* ke-i

Prosedur yang digunakan dalam melakukan optimasi menggunakan k-means adalah sebagai berikut:

Step 1. Tentukan jumlah *cluster*
Step 2. Alokasikan data ke dalam *cluster* secara random
Step 3. Hitung centroid/rata-rata dari data yang ada di masing-masing *cluster*.
Step 4. Alokasikan masing-masing data ke centroid/rata-rata terdekat
Step 5. Kembali ke Step 3, apabila masih ada data yang berpindah *cluster* atau apabila perubahan nilai centroid, ada yang di atas nilai threshold yang ditentukan atau apabila perubahan nilai pada objective function yang digunakan, di atas nilai threshold yang ditentukan.

Centroid/rata-rata dari data yang ada di masing-masing *cluster* yang dihitung pada Step 3. didapatkan menggunakan rumus sebagai berikut:

$$v_{ij=} \sum_{N_i}^{k=0} \left(\frac{x_{kj}}{N_i}\right) \qquad (4)$$

dimana:
i,k : indeks dari *cluster*
j : indeks dari variabel
v_{ij} : centroid / rata-rata *cluster* ke-i untuk variabel ke-j
x_{kj} : nilai data ke-k yang ada di dalam *cluster* tersebut untuk variabel ke-j
N_i : Jumlah data yang menjadi anggota *cluster* ke-i

Sedangkan pengalokasian data ke masing-masing *cluster* yang dilakukan pada Step 4. dilakukan secara penuh, dimana nilai yang memungkinkan untuk a_{ik} adalah 0 atau 1. Nilai 1 untuk data yang dialokasikan ke *cluster* dan nilai 0 untuk data yang dialokasikan ke *cluster* yang lain. Dalam menentukan apakah suatu data teralokasikan ke suatu *cluster* atau tidak, dapat dilakukan dengan menghitung jarak data tersebut ke masing-masing centroid/rata-rata masing-masing *cluster*. Dalam hal ini, a_{ik} akan bernilai 1 untuk *cluster* yang centroidnya terdekat dengan data tersebut, dan bernilai 0 untuk yang lainnya.

2.6. *Cluster*ing dengan Metode *Single Linkage*

Journal of Information System, Applied, Management, Accounting and Research
e-ISSN: 2598-8719 (Online)
p-ISSN: 2598-8700 (Printed)
Vol. 4 No. 3 , Agustus 2020

Pada penelitian ini, sebagai pembanding dengan metode manual (standart) maka sistem yang dibuat juga ditambahi dengan metode *Single Linkage*. Salah satu alasan pemilihan metode *Single Linkage* adalah metode ini merupakan salah satu bentuk metode *clustering* yang hierarchical sehingga seluruh data akan masuk ke dalam *cluster*. Karena seluruh data masuk dalam *cluster* maka hasil pengelompokan (*clustering*) yang dihasilkan bisa dibandingkan dengan metode manual (standart). Metode pengelompokan ini menggunakan obyek yang paling dekat atau paling sama antar objek satu dengan yang lain untuk dikelompokkan. Untuk menggunakan algoritma ini ada beberapa langkah yang harus dilakukan yaitu[2][3][4]:

1. Mulai dengan N *cluster*, setiap *cluster* mengandung entiti tunggal dan sebuah matriks simetrik dari jarak (similarities) D = {dik} dengan tipe NxN.
2. Cari matriks jarak untuk pasangan *cluster* yang terdekat (paling mirip). Misalkan jarak antara *cluster* U dan V yang paling mirip adalah duv.
3. Gabungkan *cluster* U dan V. Label *cluster* yang baru dibentuk dengan (UV). Update entries pada matrik jarak dengan cara :
a. Hapus baris dan kolom yang bersesuaian dengan *cluster* U dan V.
b. Tambahkan baris dan kolom yang memberikan jarakjarak antara *cluster* (UV) dan *cluster-cluster* yang tersisa.
4. Ulangi langkah 2 dan 3 sebanyak (N-1) kali. (Semua objek akan berada dalam *cluster* tunggal setelah algoritma berahir). Catat identitas dari *cluster* yang digabungkan dan tingkat-tingkat (jarak atau similaritas) di mana penggabungan terjadi[5].

Dalam pengerjaan menggunakan metode ini, ada beberapa perhitungan yang harus dilakukan diantaranya:
1. Mencari nilai rata-rata tiap parameter

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \qquad (5)$$

Dimana :
$\bar{x}$: nilai rata-rata parameter
xi : nilai parameter
n : jumlah *Agent*

2. Mencari nilai standar deviasi tiap parameter.
Untuk mencari nilai standart deviasi pada tiap parameter yang ada dapat menggunakan rumus sebagai berikut :

$$std(x) = \frac{\sqrt{\sum_{i=1}^{n} xi - \bar{x}}}{n-1} \qquad (6)$$

Dimana :
$std(x)$: nilai standarisasi parametert
x_i : nilai parameter
$\bar{x}$: nilai rata-rata parameter

3. Mencari nilai standar untuk masing-masing *Agent*.

$$Z_i = \frac{xi - \bar{x}}{std(x)} \qquad (7)$$

Dimana :
Z_I : nilai standar masing-masing *Agent*
x_i : nilai parameter
$\bar{x}$: nilai rata-rata parameter
$std(x)$: nilai standarisasi parameter

Setelah mendapat nilai normalisasi, maka dilakukan perhitungan menggunakan Euclidean Distance.

$$d_{rs} = \sum_{i=0}^{k} (x_{ri} - x_{si}) \qquad (8)$$

Dimana :
r : *Agent* ke 1-9, dimana s= *Agent* ke r+1
k : jumlah *Agent*.
d_{rs} : jarak antara *Agent* r dengan s
x_{ri} : nilai parameter *Agent* r ke-1 s/d 9
x_{si} : nilai parameter *Agent* r+1

Untuk memberikan label pada *cluster* yang ditentukan sesuai dengan standarisasi yang ada, maka digunakan rumus berikut.

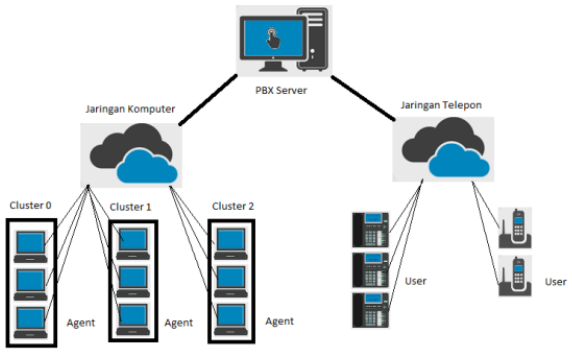
$$total = \sqrt{Xa^2 + Xb^2} \qquad (9)$$

Dimana :

total : nilai akhir
x_a : nilai parameter pertama
x_b : nilai parameter kedua.

III. PEMBAHASA DAN HASIL

Pada tahap ini, metode *K-Means* dan *Single Linkage* digunakan untuk pengelompokan agent sebagai upaya untuk mengetahui beban kerja terhadap agent. Proses perhitungan manual dan dengan di implementasikan kedalam aplikasi Rapidminer. Blok Diagram PBX sistem dapat dilihat pada Gambar 1.



Gambar 1. Blok Diagram PBX

Cara kerja dari sistem ini adalah saat ada panggilan masuk dan diterima oleh *Agent*, maka panggilan tersebut akan dicatat dan disimpan pada Database. Data yang disimpan berupa jumlah panggilan yang dihitung tiap panggilan masuk yang diterima *Agent*, rata-rata lama panggilan yang digunakan oleh *Agent*, lama waktu idle *Agent*, dan lama waktu Login dari *Agent* terhadap server.

Data-data tersebut disimpan dalam tabel yang nanti bisa dilihat setiap saat. Untuk melakukan penilaian performansi dari kinerja *Agent*, digunakan metode *K-Means Cluster*ing dan *Single Linkage*. Dari penelitian yang dibuat, diambil sampel data performansi *Agent* dengan 2 paramater yaitu N inbound dan login time seperti yang terlihat pada Tabel 1. Kemudian untuk perhitungan manual metode *K-Means* dapat dilihat pada Tabel 2. Setelah dilakukan 3 iterasi didapatkan hasil pengelompokan *Agent* yang optimal sesuai metode.

Dibawah ini adalah contoh dari data kinerja pada masing-masing agent *Call Center* dengan jumlah panggilan yang ditangani dan lama login pada aplikasi *Call Center*.

Tabel 1. Sampel data *performance* kinerja *Agent*

Agent	***Jumlah Panggilan***	***Waktu Login***
Hasbi	335	504921
Ais	326	513678
Beni	348	503246
Tasya	360	527757
Daus	269	530861
Asri	303	501095
Ayu	260	326976
Khusnul	290	445545
Dani	328	537828
Jai	276	360000
Joko	293	453124
Dilan	309	367990
Maya	284	378910
Aji	270	400000
Artika	301	402500
Sari	266	375000
Dewi	333	349690
Luna	343	555008
Ita	367	532165
Pontas	350	453678

Dari 20 data sampel di atas, dengan menggunakan metode *K-Means* dengan K=3 akan di dapatkan 3 buah *cluster* pada hasil dari iterasi yang dilakukan. Untuk proses perhitungan manual *K-Means* bisa diliat pada Tabel 2. Menggambarkan nilai untuk masing jarak antar parameter ke centroid dari masing-masing *cluster*:

Journal of Information System, Applied, Management, Accounting and Research
e-ISSN: 2598-8719 (Online)
p-ISSN: 2598-8700 (Printed)
Vol. 4 No. 3 , Agustus 2020

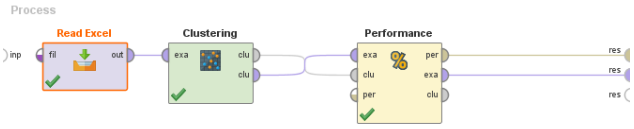
Avg. within centroid distance_*cluster*_1: -299620927.556
Avg. within centroid distance_*cluster*_2: -553512651.922

Tabel 2. Iterasi 1 metode *K-Means*

C1	C2	C3
50087.00	3826.13	8757.00
41330.00	12583.02	0.00
51762.00	2151.47	10432.02
27251.01	26662.06	14079.04
24147.11	29766.02	17183.09
53913.01	0.00	12583.02
228032.02	174119.01	186702.01
109463.01	55550.00	68133.01
17180.01	36733.01	24150.00
195008.01	141095.00	153678.01
101884.01	47971.00	60554.01
187018.00	133105.00	145688.00
176098.01	122185.00	134768.01
155008.02	101095.01	113678.01
152508.01	98595.00	111178.00
180008.02	126095.01	138678.01
205318.00	151405.00	163988.00
0.00	53913.01	41330.00
22843.01	31070.07	18487.05
101330.00	47417.02	60000.00

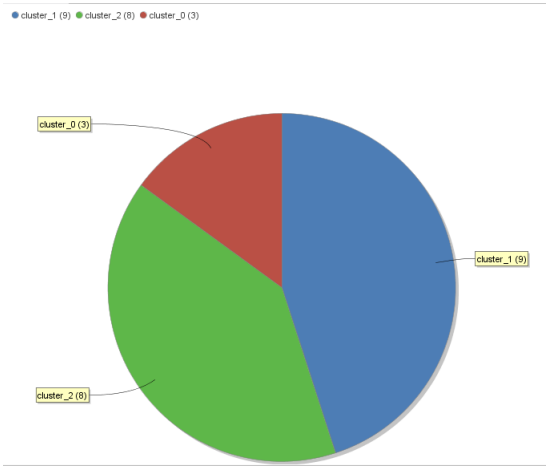
Dari data pada Tabel 2. didapatkan untuk *Cluster* 0 (C1) = 2, *Cluster* 1(C2) = 14 dan *Cluster* 2(C3) = 4. Kemudian setelah itu dilakukan proses iterasi selanjutnya hingga membentuk pusat *cluster* baru yang sama jumlah member *cluster* dengan iterasi sebelumnya.

Pada penelitian ini, dengan metode *K-Means* didapatkan 3 iterasi dari jumlah maksimal 10 iterasi yang dilakukan. Kemudian setelah itu dilakukan proses implementasi dengan menggunakan aplikasi Rapidminer dengan memasukkan data training ke aplikasi yang berisi jumlah panggilan dan waktu login pada Table 1 ke dalam aplikasi untuk pengolahan. Berikut adalah proses pada Rapidminer :



Gambar 2. Proses *K-Means* pada Rapidminer

Dari hasil perhitungan aplikasi di dapatkan data sebagai berikut :



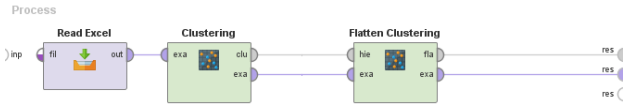
Gambar 3. Hasil proses Rapidminer

Sedangkan untuk perhitungan berdasarkan jarak dengan pusat *cluster* (centroid) sebagai berikut :

Performance Vector:
Avg. within centroid distance: -358299489.902
Avg. within centroid distance_*cluster*_0: -13766744.889
Avg. within centroid distance_*cluster*_1: -299620927.556
Avg. within centroid distance_*cluster*_2: -553512651.922

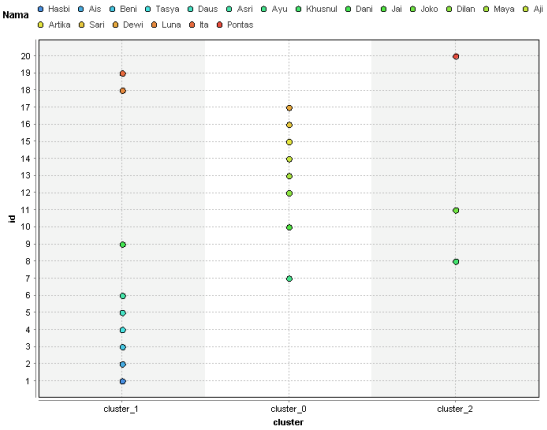
Untuk selanjutnya dilakukan proses perhitungan manual dan penerapan aplikasi untuk metode *Single Linkage* dengan data training yang sama dan diterapkan dengan aplikasi Rapidminer. Adapun untuk proses perhitungan metode *Single Linkage* dapat dilihat pada Lampiran 1. Karena dilakukan perhitungan jarak dari D1 hingga ke D20 dilampirkan pada halaman terakhir untuk

memudahkan pengecekan hasil perhitungan. Lalu data yang sama diterapkan dengan aplikasi Rapidminer sebagai berikut :



Gambar 4. Proses *Single Linkage* pada Rapidminer

Dari proses Gambar 4 tersebut maka setelah di proses menghasilkan data sebagai berikut :



Gambar 5. Hasil proses Rapidminer untuk Single Lingkage.

Untuk jumlah *cluster* sesuai dengan hasil tersebut adalah sebagai berikut :

Cluster 0: 8 items
Cluster 1: 9 items
Cluster 2: 3 items
Total number of items: 20

IV. KESIMPULAN

Dari hasil penelitian ini didapatkan perbedaan jumlah *Agent* pada metode *clustering K-Means* dengan *Single Linkage* yang secara tidak langsung memiliki kemiripan dalam komposisi jumlah *Agent*. Untuk lebih memastikan dengan data training yang sama bisa juga dicoba dengan menggunakan metode *clustering* lainnya untuk melihat perbedaan maupun persamaan pada jumlah *Agent* pada setiap *cluster* nya.

Untuk penelitian ke depan, penulis akan mencoba menggabungkan dan mengkomparasi antara metode *clustering* dengan *classification* seperti *Naïve Bayes*, KNN atau C45 untuk mendapatkan hasil *clustering Agent* yang lebih optimal optimal guna menunjang kinerja dan beban kerja setiap *Agent*.

REFRENSI

[1] Fitriyadi. "Implementasi Screen Pop pada Layanan informasi Mahasiswa Berbasis CTI", Surabaya., 2010.
[2] M.S.Pramono, "Estimasi Jumlah Kelompok pada Analisis Kelompok Metode *Single Linkage* dan KMeans Melalui Beberapa Metode Penentuan Jumlah Kelompok", ITS, 2007.
[3] S.Hidayati,"Perbandingan Analisis *Cluster* dengan Metode *Single Linkage* dan *K-Means*", UIN Sunan Kalijaga Jogjakarta, 2011.
[4] Nur Rosyid Mubtada'I, Mike Yuliana, dan Beni Ilham Priyambodo. "Analisa Perbandingan *Cluster*ing Metode Manual dan Metode *Single Linkage* untuk Menentukan Kinerja *Agent* pada *Call Center*", The 13th Industrial Electronics Seminar 2011.
[5] PT. XYZ, "*Call Center* PT. XYZ",2018.
[6] *K-Means*.(2011). (https://yudiagusta. wordpress.com/k-means/), diakses 28 Agustus 2018.