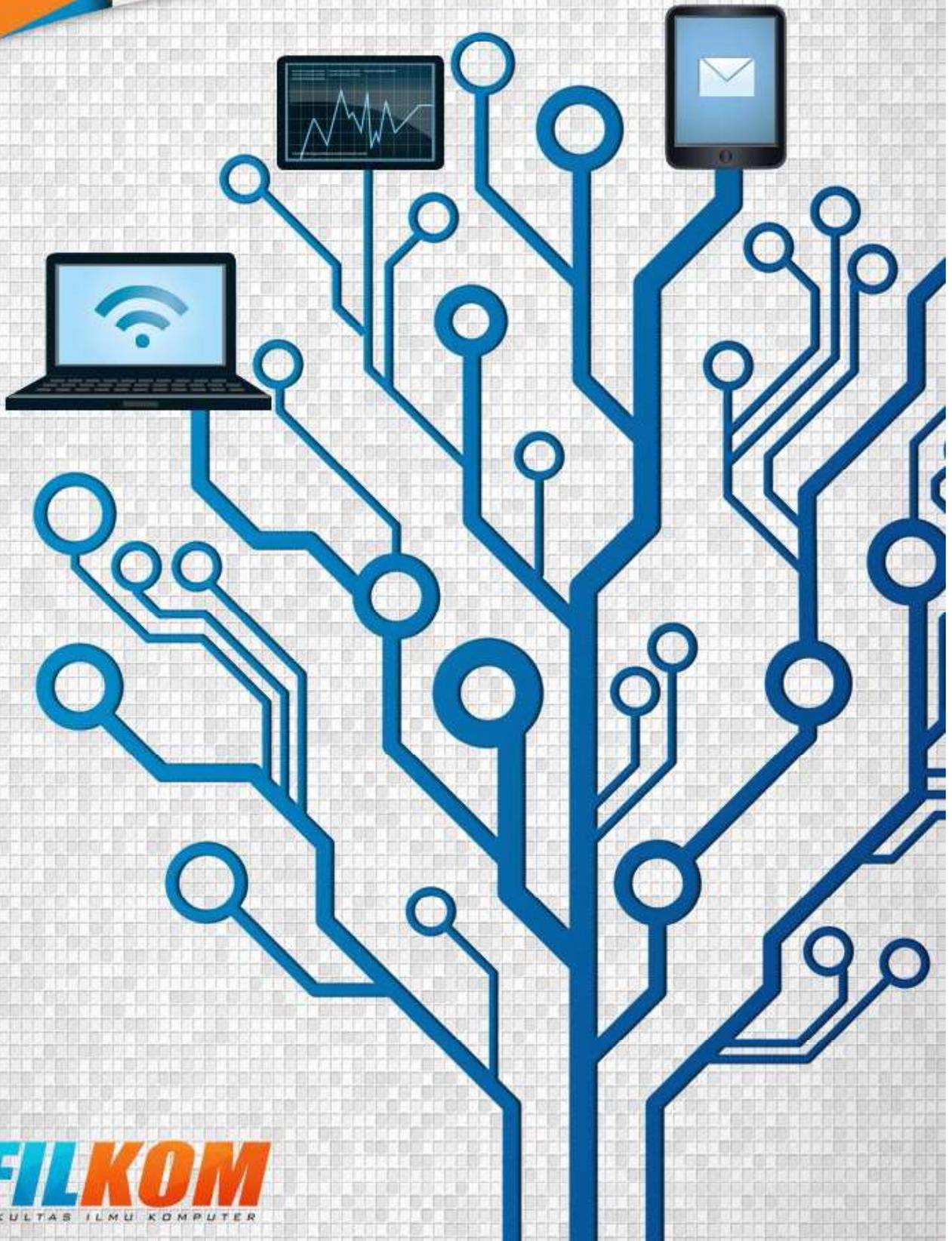


ISSN : 2355-7699

# JURNAL

## TEKNOLOGI INFORMASI & ILMU KOMPUTER

Volume 4 | Nomor 2 | Juni 2017 | Halaman 75-147



# JTIK

## JURNAL TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

Volume 4, Nomor 2, Juni 2017

ISSN 2355-7699

---

JTIK diterbitkan oleh Fakultas Ilmu Komputer (FILKOM), Universitas Brawijaya sejak April 2014. JTIK memuat naskah hasil-hasil penelitian di bidang Teknologi Informasi dan Ilmu Komputer.

### **Ketua Redaksi**

Gembong Edhi Setyawan

### **Ketua Redaksi Pelaksana**

Imam Cholissodin

### **Redaksi Pelaksana**

Candra Dewi

Ahmad Afif Supianto

Muhammad Tanzil Furqon

### **Pelaksana Tata Usaha**

Dwi Nur Indah Lestari

Rieftiyan David Felani

### **Alamat Redaksi dan Tata Usaha**

Jurnal Teknologi Informasi dan Ilmu Komputer  
Fakultas Ilmu Komputer (FILKOM), Universitas Brawijaya  
Jl. Veteran No. 8 Malang, 65145  
Telp./Fax (0341) 577911  
Email: [jtiik@ub.ac.id](mailto:jtiik@ub.ac.id)  
Website: <http://www.jtiik.ub.ac.id>

Redaksi mengundang penulis untuk mengirimkan naskah yang belum pernah diterbitkan di media manapun. Pedoman penulisan naskah terdapat pada bagian belakang jurnal. Naskah yang masuk akan dievaluasi secara *blind-review* oleh Mitra Bestari dan Redaksi Pelaksana.

# J T I I K

## JURNAL TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

Volume 4, Nomor 2, Juni 2017

ISSN 2355-7699

---

Redaksi JTIK mengucapkan terimakasih yang sebesar-besarnya kepada para reviewer yang telah bersedia untuk meluangkan waktunya dalam melakukan review pada naskah-naskah yang masuk di JTIK FILKOM UB.

### MITRA BESTARI

1. Arif Muntasa, Universitas Trunojoyo, Indonesia
2. Agustinus Fritz Wijaya, Universitas Kristen Satya Wacana (UKSW), Indonesia
3. Barlian Henryranu Prasetio, Universitas Brawijaya, Indonesia
4. Budi Rahmadya, Fakultas Teknologi Informasi, Universitas Andalas, Indonesia
5. Budi Darma Setiawan, Universitas Brawijaya, Indonesia
6. Candra Dewi, Universitas Brawijaya, Indonesia
7. Denny Sagita, Universitas Brawijaya, Indonesia
8. Didit Widiyanto, UPN Veteran Jakarta, Indonesia
9. Robbi Rahim, APIKES Imelda Medan, Indonesia
10. Eka Mistiko Rini, Politeknik Negeri Banyuwangi, Indonesia
11. Erick Fernando, STIKOM Dinamika Bangsa Jambi, Indonesia
12. Indri Sudanawati Rozas, UIN Surabaya, Indonesia
13. Heliza Rahmania Hatta, Universitas Mulawarman, Indonesia
14. Muhammad Tanzil Furqon, Universitas Brawijaya, Indonesia
15. Octav M Octaviano Pratama, CEP CCIT Fakultas Teknik Universitas Indonesia, Indonesia
16. Nurfiana Nurfiana, Fakultas Ilmu Komputer, Institut Informatika dan Bisnis Darmajaya, Indonesia
17. Rahimi Fitri, Politeknik Negeri Banjarmasin, Indonesia
18. Ratih Ayuninghemi, Politeknik Negeri Jember, Indonesia
19. Riyanto Sigit, Politeknik Elektronika Negeri Surabaya, Indonesia
20. Titin Pramiyati, UPN "Veteran" Jakarta, Indonesia
21. Wayan Firdaus Mahmudy, Universitas Brawijaya, Indonesia
22. Yeni Anistyasari, Universitas Negeri Surabaya, Indonesia

# JTIK

## JURNAL TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

Volume 4, Nomor 2, Juni 2017

ISSN 2355-7699

### DAFTAR ISI

Analisis Judul Majalah Kawanku Menggunakan Clustering K-Means Dengan Konsep Simulasi Big Data Pada Hadoop Multi Node Cluster <i>Brilliant Aristyo Rahardian, Diva Kurnianingtyas, Dyan Putri Mahardika, Tusty Nadia Maghfira, Imam Cholissodin</i>	75-80
Klasifikasi Sepeda Motor Berdasarkan Karakteristik Konsumen Dengan Metode K-Nearest Neighbour Pada Big Data Menggunakan Hadoop Single Node Cluster <i>Nanda Agung Putra, Ardisa Tamara Putri, Dhimas Anjar Prabowo, Listiya Surtiningsih, Raissa Arniantya, Imam Cholissodin</i>	81-86
Hibridisasi Algoritma Genetika Dengan Variable Neighborhood Search (VNS) Pada Optimasi Biaya Distribusi <i>Asyrofa Rahmi, Wayan Firdaus Mahmudy, Syaiful Anam</i>	87-96
Implementasi Metode K-Nearest Neighbour Dengan Pembobotan TF.IDF.ICF Untuk Kategorisasi Ide Kreatif Pada Perusahaan <i>Rekyan Regasari Mardi Putri, Romario Yudo Herlambang, Randy Cahya Wihandika</i>	97-103
Hybrid K-means Dan Particle Swarm Optimization Untuk Clustering Nasabah Kredit <i>Yusuf Priyo Anggodo, Winda Cahyaningrum, Aprilia Nur Fauziyah, Irma Lailatul Khoiriyah, Oktavianis Kartikasari, Imam Cholissodin</i>	104-110
Pemetaan Data Dan Visualisasi Kedalaman Air Pada Bendungan / Waduk <i>Bayu Rahayudi, Marji Marji</i>	111-116
Desain Protokol Suara Sebagai Pengendali Dalam Smart Home Menggunakan FPGA <i>Barlian Henryranu Prasetio, Dahnia Syauqy</i>	117-121
Sistem Pendukung Keputusan Diagnosis Penyakit Sapi Potong Menggunakan K-Nearest Neighbour (K-NN) <i>Divya Kurnianingtyas, Brilliant Aristyo Rahardian, Dyan Putri Mahardika, Amalia Kartika A., Dwi Angraeni K.</i>	122-126
Analisis Penentuan Lokasi Evakuasi Bencana Banjir Dengan Pemanfaatan Sistem Informasi Geografis Dan Metode Simple Additive Weighting <i>Juliana Andretha Janet Lumban Batu, Charitas Fibriani</i>	127-135
Analisis Dan Perancangan Sistem Informasi Penjualan Berdasarkan Stok Gudang Berbasis Client Server (Studi Kasus Toko Grosir “Restu Anda”) <i>Anthony Anthony, Andeka Rocky Tanaamah, Agustinus Fritz Wijaya</i> PDF	136-147

## ANALISIS JUDUL MAJALAH KAWANKU MENGGUNAKAN *CLUSTERING* K-MEANS DENGAN KONSEP SIMULASI *BIG DATA* PADA HADOOP MULTI NODE *CLUSTER*

Brilliant Aristyo Rahadian<sup>1</sup>, Diva Kurnianingtyas<sup>2</sup>, Dyan Putri Mahardika<sup>3</sup>, Tusty Nadia Maghfira<sup>4</sup>,  
Imam Cholissodin<sup>5</sup>

<sup>1,2,3,4,5</sup>Fakultas Ilmu Komputer Universitas Brawijaya

Email: <sup>1</sup>brilliantarc@gmail.com, <sup>2</sup>divakurnianingtyas@gmail.com, <sup>3</sup>dyanputri.dp@gmail.com,  
<sup>4</sup>tustynadia@gmail.com, <sup>5</sup>imamcs@ub.ac.id

(Naskah masuk: 5 Januari 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Saat ini pembaca *e-magazine* seperti majalah Kawanku semakin marak dan terus berkembang. Sehingga penggunaan data besar sangat dibutuhkan pada majalah Kawanku. Selain itu, dibutuhkan pengkategorian setiap bacaan ke dalam tujuh kategori judul pada majalah Kawanku. Sehingga dibutuhkan suatu pengolahan, pengelompokkan, dan pengkomunikasian antar data teks menggunakan *text mining*. Kombinasi *text mining* dengan *Big Data* dapat menjadi sebuah solusi yang menyediakan cara yang efisien dan reliabel untuk penyimpanan data dan infrastruktur yang efektif. Lalu pengkategorian teks dengan *clustering* K-Means dirasa cukup meskipun menggunakan data besar karena hasilnya memiliki keakuratan yang tinggi. Dari hasil pengujian yang dilakukan, disimpulkan bahwa perbedaan dari banyaknya data tidak mempengaruhi waktu eksekusi karena perbedaan jumlah data yang digunakan tidak terlalu besar.

**Kata kunci:** *text mining, k-means, hadoop, big data, clustering, multi node cluster*

### Abstract

*Nowadays e-magazine reader like Kawanku magazine is increasing more and more. So the use of Big Data is needed in managing e-magazine data in server. In addition, it takes the categorization of each reading into 7 categories of Kawanku magazine. So it takes a processing, grouping, and communicating between the text data using text mining. The combination of text mining with Big Data can be a solution that provides an efficient and reliable way for data storage and effective infrastructure. Then the text categorization with K-Means clustering is enough although using Big Data as a result has a high accuracy. From the results of tests performed, it was concluded that the difference of the number of data does not affect the execution time due to differences in the amount of data used is not too big.*

**Keywords:** *text mining, k-means, hadoop, big data, clustering, multi node cluster*

## 1. PENDAHULUAN

Di era globalisasi seperti saat ini dimana teknologi semakin berkembang, hampir semua aktivitas dilakukan menggunakan internet. Adanya internet memudahkan berbagai kegiatan manusia, terutama komunikasi dan pencarian pengetahuan atau informasi. Penggunaan *e-magazine* di internet menjadi cara baru masyarakat dalam mencari informasi dan pengetahuan. Banyak masyarakat yang beralih menggunakan *e-magazine* karena kemudahan akses informasi yang ditawarkan dan lebih efisien. Selain meningkatnya pengguna *e-magazine* terdapat peningkatan juga pada penggunaan media sosial. Penggunaan sosial media telah meningkat dari tahun 2005 hingga tahun 2015. Pengguna media sosial merupakan pengguna yang berumur antara 18 hingga 29 tahun. Saat ini sebanyak 90% anak muda aktif di media sosial dimana sebelumnya tahun 2005 hanya sebesar 12% (Çakir and Güldamlasioglu, 2016).

Berdasarkan fakta tersebut, banyak *e-magazine* yang memanfaatkan teknologi media sosial seperti Facebook, Twitter, dll dalam membuat akun dari media sosial yang terintegrasi dengan *e-magazine*. Manfaat integrasi dengan media sosial yaitu sebagai wadah untuk berinteraksi antara penerbit dengan pembaca maupun pembaca dengan pembaca lainnya.

Semakin banyaknya pengguna internet berdampak pada banyaknya data berukuran besar yang dihasilkan setiap detik pada web. Dikarenakan meningkatnya pengguna internet terutama *e-magazine* dan media sosial, sejumlah data berukuran besar membanjiri setiap server penyimpanan. Untuk mengolah, mengelompokkan dan mengkomunikasikan antar data teks yang didapatkan dari media sosial dibutuhkan teknik *text mining*. Namun, pemrosesan dan penganalisisan teks berjumlah besar menggunakan metode tradisional dinilai sulit dilakukan karena metode tradisional memiliki sumberdaya dan infrastruktur yang terbatas (Rathore and Shukla, 2015). Dengan adanya pengembangan

teori *Big Data* ditemukan solusi yang menyediakan cara yang efisien dan reliabel untuk penyimpanan data dan infrastruktur yang efektif untuk perhitungan yang efisien. Namun, pada aplikasinya untuk mengenali berkas target yang dicari dari sejumlah data yang besar dibutuhkan metode untuk mengelompokkan data ke dalam beberapa kelompok data yang mudah dikenali. Salah satu metode yang efektif digunakan untuk mengelompokkan data (*clustering*) adalah K-Means.

Sebelumnya terdapat penelitian oleh Rathore dan Shukla (2015) yang mengembangkan metode K-Means untuk mengolah *Big Data* pada 5 data set yang berbeda yaitu *Laustrian* dataset, *Breast cancer* dataset, *Diabetes* dataset, *Iris* dataset, dan *Bupa* dataset. Hasil penelitian menunjukkan bahwa pengembangan metode K-Means yang diusulkan lebih baik daripada K-Means tradisional dengan rata-rata akurasi meningkat sekitar 3.78% sampai 8.94%. selain itu terdapat juga penelitian yang dilakukan oleh Çakir dan Güldamlasioglu (2016) dengan topik *clustering* data teks berukuran besar dari media sosial berbahasa Turki menggabungkan penggunaan sistem Hadoop dengan Spark dan metode *clustering* menggunakan K-Means. Hasil menunjukkan bahwa metode yang digunakan mampu mengelompokkan data berukuran besar dengan benar. Pada penelitian ini, peneliti bermaksud untuk melakukan penelitian tentang analisis judul bacaan pada majalah Kawanku menggunakan metode yang sama dengan penelitian Çakir & Güldamlasioglu yaitu K-Means *clustering* pada *Big Data* environment. Peneliti berharap penggunaan teknologi *Big Data* dan K-Means *clustering* dapat memberikan akurasi yang tinggi untuk proses penentuan judul bacaan pada majalah Kawanku.

## 2. DASAR TEORI

### 2.1 Data Yang Digunakan

Data yang digunakan adalah judul bacaan majalah Kawanku yang dikategorikan dalam 7 kategori judul, dan masing-masing kategori memiliki 10 buah dokumen yang akan digunakan sebagai data latih. Kategori judul yang digunakan yaitu: *seleb* dan *entertainment*, *news*, *playground*, *fashion*, *beauty*, *love*, dan *life*. Berikut ini adalah contoh dataset yang digunakan berdasarkan kategorinya ditunjukkan Tabel 1.

**Tabel 1. Contoh Dataset Setiap Kategori**

Kategori	Dokumen
<i>Seleb &amp; Entertainment</i>	10 Lagu One Direction yang Cocok Untuk Jadi Soundtrack Moment Perpisahan Sekolah
<i>News</i>	5 Fakta Tentang Gajah Yani yang Mati Karena Sakit & Enggak Terawat di Kebun Binatang Bandung

<i>Playground</i>	10 Lagu Barat yang Bisa Bantu Menenangkan Pikiran Stres Saat SBMPTN
<i>Fashion</i>	10 Inspirasi Gaya Memakai Kemeja dari Taylor Swift
<i>Beauty</i>	10 Inspirasi Makeup Flawless Ala Seleb Kpop yang Cocok Untuk Prop atau Wisuda SMA
<i>Love</i>	5 Alasan Kita Harus Mempertahankan Hubungan LDR Dengan Pacar Setelah Pisah Sekolah
<i>Life</i>	5 Cara Tetap Kompak dengan Teman Segeng Setelah Lulus Sekolah

### 2.2 Konsep Big Data

*Big Data* adalah kombinasi teknologi yang bisa memanajemen data yang bervariasi dalam jumlah besar, kecepatan yang tepat, dan memiliki ketepatan saat melakukan analisis dan reaksi. Tiga karakteristik yang dimiliki *Big Data*, yaitu *volume*, *velocity*, dan *variety* (Hurwitz et al., 2013).

Dilihat pada bidang teknologi, banyak manfaat yang diperoleh dalam memproses *Big Data* seperti teknik akses dan penyimpanan data menggunakan *Key-Value Store* (KVS) dan teknik komputasi paralel menggunakan *MapReduce* (Sujana, 2013).

#### 2.2.1 Hadoop

*Hadoop Distributed File System* (HDFS) adalah sistem file terdistribusi yang dirancang untuk mencegah adanya ledakan data dalam jumlah yang sangat besar (*terabyte* atau bahkan *petabyte*) dan memberikan akses informasi *highthroughput*. File disimpan di beberapa mesin untuk memastikan ketahanan terhadap kegagalan dan ketersediaan saat aplikasi dijalankan secara paralel.

#### 2.2.2 Map Reduce

Berdasarkan map LISP dan *reduce* primitif, bisa diciptakan sebagai cara untuk menerapkan pemrosesan paralel tanpa harus berurusan dengan semua komunikasi antara node, dan pembagian tugas (Vaidya, 2012), seperti MPI. Fungsi *MapReduce* terdiri dari menulis, *map*, dan *reduce*. Fungsi *map* digunakan dalam mengambil key, kombinasi nilai output antara nilai menengah dengan nilai key. Fungsi *map* ditulis sedemikian rupa sehingga dapat dijalankan sekaligus dengan membagi tugas-tugas. Fungsi *reduce* digunakan untuk mengambil nilai output dari fungsi *map* lalu menggabungkan nilai-nilai sehingga menghasilkan hasil yang diinginkan dalam file output.

### 2.3 Text Mining

Analisis kata atau kalimat menggunakan proses *text mining*. *Text mining* memiliki dua proses antara

lain *preprocessing* dan pengolahan data (*clustering* atau klasifikasi). Proses *text preprocessing* merupakan tahapan pertama yang dilakukan sebelum input dokumen diolah lebih lanjut menjadi kluster-kluster kalimat. Proses-proses yang dilalui antara lain menghilangkan tanda baca, angka, mengkonversi huruf besar, *tokenizing* (cari kata dalam kalimat), *stemming* (ubah kata ke kata dasar), dan menghapus kata sesuai *stopword* (Çakir and Güldamlasioglu, 2016). Setelah itu proses pengolahan data, hasilnya akan digunakan untuk pengkategorian dengan *clustering* atau klasifikasi.

## 2.4 Clustering Text

Input yang digunakan dalam pembentukan sebuah *cluster* kalimat berasal dari hasil *text processing*. Proses ini mempunyai peranan yang sangat penting dalam meringkas secara otomatis. Setiap topik dalam dokumen harus diidentifikasi secara tepat untuk menemukan kesamaan (*similarity*) dan ketidaksamaan (*dissimilarity*) yang ada pada dokumen sehingga menjamin *good coverage* (Sarkar, 2009).

Faktor koherensi *cluster* digunakan untuk menjamin kualitas hasil ringkasan. Koherensi *cluster* menunjukkan keterkaitan antar kalimat pada masing-masing *cluster* yang terbentuk dalam proses peringkasan banyak dokumen. Tingginya derajat koherensi *cluster* yang sangat sulit dicapai karena memerlukan pengetahuan tentang makna dan struktur kalimat (Sarkar, 2009). Tingginya derajat koherensi dalam sebuah *cluster* dapat dicapai dengan mempertahankan derajat *similarity* antar anggota tetap tinggi (Hammouda and Kamel, 2003).

### 2.4.1 Metode K-Means

Algoritma K-Means dikenal sebagai algoritma yang sangat mudah dalam proses *clustering* data besar. Proses untuk melakukan *clustering* data outlier sangatlah cepat. Selain itu, metode ini juga memiliki kelemahan yaitu dapat dimungkinkan bagi setiap data *cluster* tertentu pada suatu tahapan dapat berpindah ke *cluster* yang lain pada tahapan selanjutnya (Larose et al., 2005).

Berikut ini adalah persamaan dan langkah-langkah perhitungan menggunakan algoritma K-Means (Hung et al., 2005) antara lain:

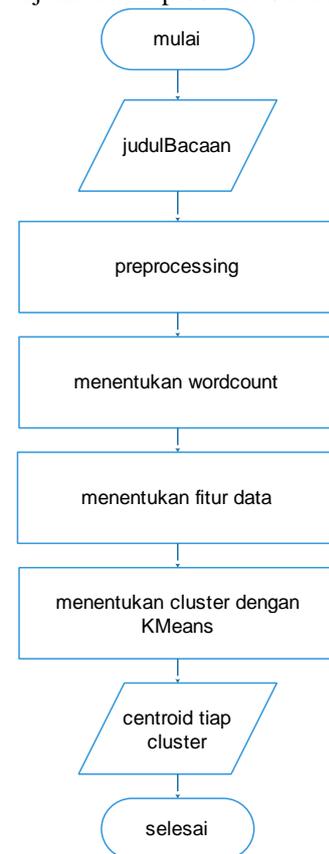
$$v_j = \frac{\sum_{i=1}^n m(c_j|x_i)x_i}{\sum_{i=1}^n m(c_j|x_i)} \text{ for } j = 1, \dots, k \quad (1)$$

- Inisialisasi dataset dari  $n$  titik data  $X = \{x_1, \dots, x_n\}$
- Masukkan jumlah  $k$  *cluster*.
- Inisialisasi *centroid*  $C_j$  untuk  $k$  *cluster* dari sejumlah dataset.
- Letakkan setiap titik pada *cluster* terdekat dengan menggunakan rumus jarak Euclidean.

- Hitung ulang mencari nilai *centroid* dari setiap  $k$  *cluster* dengan jumlah data  $m$  untuk menemukan nilai *centroid*  $C_j$  *cluster* yang baru.
- Proses diulang hingga mendapatkan hasil yang konvergen.

## 3. PERANCANGAN DAN IMPLEMENTASI

Pada penelitian ini terdapat beberapa tahapan yang harus dijalankan untuk menentukan *cluster* dari judul bacaan majalah Kawanku. Berikut ini pada Gambar 1 ditunjukkan alur proses *clustering*.



**Gambar 1 Diagram Alir Clustering K-Means Judul Bacaan pada Majalah Kawanku**

Berdasarkan Gambar 1, proses penelitian ini diawali dengan memberi masukan pada sistem yaitu data judul bacaan yang diambil dari majalah Kawanku secara random sebanyak 40 dokumen. Setelah memasukkan data tersebut, dilakukan proses *preprocessing*. *Preprocessing* pada penelitian ini terdiri dari proses penghilangan tanda baca, simbol, angka, dan kata yang tidak penting. Setelah *preprocessing* selesai dilanjutkan dengan proses *wordcount* yaitu menentukan jumlah setiap kata yang ada pada setiap dokumen. Dari keseluruhan hasil proses *wordcount* diambil 10 kata yang paling sering muncul sebagai fitur data. Selanjutnya proses dilanjutkan dengan proses penentuan *cluster* dengan metode K-Means. Sebelum melakukan perhitungan, ditentukan nilai  $k$  atau jumlah *cluster*. Pada penelitian ini ditentukan jumlah *cluster* sebanyak 4. Hasil akhir

dari penelitian ini yaitu nilai *centroid* dari setiap *cluster*.

Berdasarkan alur penelitian yang sebelumnya telah dijelaskan, berikut ini ditunjukkan potongan kode program proses *clustering* pada Gambar 2.

```
int shouldNextIteration(Configuration conf)
throws IOException{

//Merging output from reducer
FileSystem fs = FileSystem.get(conf);

Path reducerOutput = new
Path("kmeansOutput/out1"); // FOR JAR

Path centroidOld = new
Path("centroid/centroid_"+iterationCount)+
".txt");

Path centroidNew = new
Path("centroid/centroid_"+(iterationCount+1)
+".txt");

FileUtil.copyMerge(fs, reducerOutput, fs, centr
oidNew, false, conf, null);

//Reading new centroid file
String line;

Datapoint[] newCentroid = new
Datapoint[numberOfCentroids];

BufferedReader br = new BufferedReader(new
InputStreamReader(fs.open(centroidNew)));

for(int x=0;x<numberOfCentroids;x++){
line= br.readLine();

String[] centroidString =
line.split("\t");

newCentroid[x]= new
Datapoint(centroidString[0],centroidString[1
]);

System.out.println(centroidString[0]
+ "===VSD NEW===")+
centroidString[1]);
}

br.close();

//Reading old centroid
line="";

Datapoint[] oldCentroid = new
Datapoint[numberOfCentroids];

br = new BufferedReader(new
InputStreamReader(fs.open(centroidOld)));

for(int x=0;x<numberOfCentroids;x++){
line=br.readLine();
String[] centroidString =
line.split("\t");

oldCentroid[x]= new
Datapoint(centroidString[0],centroidString[1
]);

System.out.println(centroidString[0]
+ "===VSD OLD===")+
centroidString[1]);
}
```

```
}

br.close();

//Comparing the change from previous centroid

for(int i = 0; i<numberOfCentroids;i++){

for(int k = 0;k<numberOfAttributes;k++){

if(0.02<Math.abs(newCentroid[i].getVal(k)-
oldCentroid[i].getVal(k))){

shouldContinue = true;

System.out.println("Should continue: "+
shouldContinue);

return 1;
}
else{
shouldContinue = false;

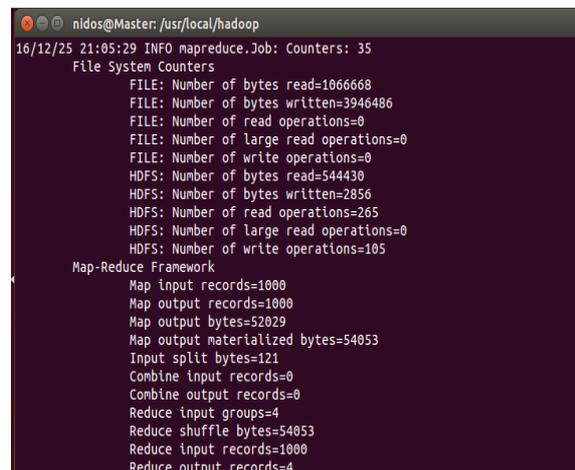
System.out.println("Should continue: "+
shouldContinue);

}
}
}

return 0;
}
```

**Gambar 2. KMeansHadoop.java**

Berikut ini merupakan tampilan keluaran dari sistem proses *run file* yang dijalankan ditunjukkan pada Gambar 3.



```
nidos@Master: /usr/local/hadoop
16/12/25 21:05:29 INFO mapreduce.Job: Counters: 35
File System Counters
  FILE: Number of bytes read=1066668
  FILE: Number of bytes written=3946486
  FILE: Number of read operations=0
  FILE: Number of large read operations=0
  FILE: Number of write operations=0
  HDFS: Number of bytes read=544430
  HDFS: Number of bytes written=2856
  HDFS: Number of read operations=265
  HDFS: Number of large read operations=0
  HDFS: Number of write operations=105
Map-Reduce Framework
  Map input records=1000
  Map output records=1000
  Map output bytes=52029
  Map output materialized bytes=54053
  Input split bytes=121
  Combine input records=0
  Combine output records=0
  Reduce input groups=4
  Reduce shuffle bytes=54053
  Reduce input records=1000
  Reduce output records=4
```

**Gambar 3. Keluaran dari Proses Run File**

Selain itu, berikut tampilan keluaran dari sistem proses *run clustering* yang dijalankan ditunjukkan pada Gambar 4.

```
nidos@Master: /usr/local/hadoop
C1==VSD NEW===0,1,0,0,0.33,0,0.67,0.67,0.33
C2==VSD NEW===0,0,0,0,1,1.5,0,0,0
C3==VSD NEW===0.06,0,0.06,0,0.17,0,0,0,0.11
C4==VSD NEW===1,0.8,0,0.8,0,0.2,0,0,0.2
C1==VSD OLD===0,1,0,0,0.33,0,0.67,0.67,0.33
C2==VSD OLD===0,0,0,0,1,1.5,0,0,0
C3==VSD OLD===0.06,0,0.06,0,0.17,0,0,0,0.11
C4==VSD OLD===1,0.8,0,0.8,0,0.2,0,0,0.2
Should continue: false
```

**Gambar 4. Keluaran Proses Run Clustering**

Keluaran dari sistem menunjukkan hasil dari proses *clustering* yaitu nilai *centroid* dari setiap *cluster* yang dibandingkan dengan nilai *centroid* awal dari setiap *cluster*. Pada proses *clustering* akan terus berulang hingga posisi *centroid* tidak berubah, pada penelitian ini iterasi dilakukan sebanyak 2 kali perubahan *centroid*. Hasil keluaran *centroid* untuk 40 dokumen yang dilakukan adalah ditunjukkan Tabel 2 sebagai berikut.

**Tabel 2. Hasil Keluaran Centroid untuk 40 Dokumen**

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
C1	0	5	0	0	0	0	1.5	0	0	0
C2	0	0	0.57	0	0	0	0	0.64	0.21	0
C3	0	0	0	0	0	3	0	0	0	0
C4	0	0	2	0	0	0	0	0	0	3

Keterangan:

C : Centroid

F : Fitur

Untuk 40 data yang digunakan selanjutnya di klusterkan kedalam 4 kluster dengan menghitung kedekatannya terhadap seluruh pusat *centroid* diatas menggunakan rumus *euclidian*. Berikut ini hasil klustering yang diperoleh ditunjukkan Tabel 3.

**Tabel 3. Hasil Clustering**

Data	Hasil								
1	C1	9	C2	17	C2	25	C2	33	C1
2	C2	10	C2	18	C2	26	C4	34	C2
3	C1	11	C2	19	C2	27	C2	35	C2
4	C4	12	C2	20	C2	28	C2	36	C1
5	C2	13	C2	21	C2	29	C2	37	C1
6	C2	14	C2	22	C2	30	C2	38	C2
7	C2	15	C2	23	C2	31	C2	39	C2
8	C2	16	C4	24	C2	32	C2	40	C3

#### 4. PENGUJIAN DAN ANALISIS

Pengujian ini dilakukan dengan membuat variasi dataset yang berbeda ukuran atau banyaknya data yang digunakan dari keseluruhan data dipecah-pecah menjadi beberapa bagian, misal dataset ke-1 20 data, dataset ke-2 25 data, dataset ke-3 30 data, dataset ke-4 35 data, dan data set ke-4 40 data. Dari variasi data yang digunakan selanjutnya dilihat waktu eksekusi yang diperlukan. Berikut ini hasil dari pengujian yang dilakukan ditunjukkan Tabel 4.

**Tabel 4. Hasil Pengujian Berdasarkan Waktu Eksekusi**

Data	Waktu(detik)
20	13.5
25	13.54
30	13.5
35	13.44
40	13.44

Dari hasil pengujian yang dilakukan perbedaan banyaknya data tidak mempengaruhi waktu eksekusi karena perbedaan jumlah data yang tidak terlalu besar.

#### 5. KESIMPULAN DAN SARAN

Pada penelitian ini telah berhasil menjalankan KMeansHadoop.jar yang terdapat 5 *class* didalamnya yaitu Datapoint, KMeansHadoop, KMeansMapper, KMeansPartitioner dan KMeansReducer. Data *input* yang digunakan adalah data judul bacaan yang diambil dari majalah Kawanku secara random sebanyak 40 dokumen. Data *centroid* awal diambil secara acak dari dokumen *input* yang ada. Dan data output berupa *centroid* akhir (*update*) pada iterasi 2.

Setelah melakukan 5 kali percobaan untuk pengujian banyak data, yaitu sebanyak 20, 25, 30, 35, dan 40 waktu eksekusi program hanya berbeda sedikit artinya perbedaan banyak data tersebut tidak terlalu mempengaruhi waktu eksekusi karena jumlah data yang tidak banyak.

Untuk penelitian selanjutnya diharapkan bisa mengolah teks lebih baik dengan tahapan *pre-processing* yang lengkap sehingga bisa melakukan *clustering* dengan lebih baik. Untuk data *input* yang digunakan lebih banyak dan sangat banyak agar ketika melakukan pengujian bisa terlihat perbedaan waktu eksekusinya.

#### 6. DAFTAR PUSTAKA

ÇAKIR, M. U. & GÜLDAMLASIOGLU, S. Text Mining Analysis in Turkish Language Using Big Data Tools. Computer Software and Applications Conference (COMPSAC),

2016 IEEE 40th Annual, 2016. IEEE, 614-618.

- HAMMOUDA, K. M. & KAMEL, M. S. Incremental document clustering using cluster similarity histograms. *Web Intelligence*, 2003. WI 2003. Proceedings. IEEE/WIC International Conference on, 2003. IEEE, 597-601.
- HUNG, M.-C., WU, J., CHANG, J.-H. & YANG, D.-L. 2005. An Efficient k-Means Clustering Algorithm Using Simple Partitioning. *Journal of Information Science and Engineering*, 21, 1157-1177.
- HURWITZ, J., NUGENT, A., HALPER, F. & KAUFMAN, M. 2013. *Big data for dummies*, John Wiley & Sons.
- LAROSE, E., YEGHIAZARIANS, Y., LIBBY, P., YUCEL, E. K., AIKAWA, M., KACHER, D. F., AIKAWA, E., KINLAY, S., SCHOEN, F. J. & SELWYN, A. P. 2005. Characterization of human atherosclerotic plaques by intravascular magnetic resonance imaging. *Circulation*, 112, 2324-2331.
- RATHORE, P. & SHUKLA, D. 2015. Analysis And Performance Improvement Of K-Means Clustering In Big Data Environment. *Communication Networks (ICCN), 2015 International Conference on*, 43-46.
- SARKAR, K. 2009. Sentence clustering-based summarization of multiple text documents. *International Journal of Computing Science and Communication Technologies*, 2, 325-335.
- SUJANA, A. P. 2013. Memanfaatkan Big Data Untuk Mendeteksi Emosi. *Jurnal Teknik Komputer Unikom-Komputika*, 2, 1-4.
- VAIDYA, M. 2012. Parallel Processing of cluster by Map Reduce. *International journal of distributed and parallel systems*, 3, 167.

## KLASIFIKASI SEPEDA MOTOR BERDASARKAN KARAKTERISTIK KONSUMEN DENGAN METODE K-NEAREST NEIGHBOUR PADA BIG DATA MENGGUNAKAN HADOOP SINGLE NODE CLUSTER

Nanda Agung Putra<sup>1</sup>, Ardisa Tamara Putri<sup>2</sup>, Dhimas Anjar Prabowo<sup>3</sup>, Listiya Surtiningsih<sup>4</sup>, Raissa Arniantya<sup>4</sup>, Imam Cholissodin<sup>6</sup>

<sup>1,2,3,4,5,6</sup>Fakultas Ilmu Komputer Universitas Brawijaya

Email: <sup>1</sup>nandoooo1412@gmail.com, <sup>2</sup>ardisatamar@gmail.com, <sup>3</sup>dhimasanjar@gmail.com, <sup>4</sup>listiyasurtiningsih@gmail.com, <sup>5</sup>rarniantya@gmail.com, <sup>6</sup>imamcs@ub.ac.id

(Naskah masuk: 6 Januari 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Penelitian ini mengusulkan sebuah klasifikasi terhadap sepeda motor berdasarkan karakteristik konsumen. Sepeda motor memiliki beberapa jenis dan merk yang berbeda sehingga menyebabkan banyaknya pilihan yang dimiliki konsumen. Konsumen akan memilih sepeda motor yang diinginkan berdasarkan latar belakang yang berbeda. Pada penelitian ini, Konsumen akan dikelompokkan berdasarkan sepeda motor yang dibeli sehingga penjual dapat mengetahui karakteristik konsumen yang membeli suatu jenis atau merk tertentu. Karakteristik konsumen dapat ditentukan dengan usia, jenis kelamin, pendapatan, status pernikahan dan jumlah anak. Berdasarkan karakteristik tersebut perlu dilakukan pengelompokan untuk menentukan merk sepeda motor. Dalam penelitian ini metode yang digunakan yakni K-Nearest Neighbour (K-NN). K-NN merupakan algoritma yang umum digunakan untuk klasifikasi dan mencari kelas dari data uji dengan mayoritas kelompok yang memiliki jarak terdekat. Dataset yang digunakan dalam penelitian ini yaitu karakteristik konsumen. Uji coba dengan dataset tersebut menghasilkan merk sepeda motor dari data uji yang sudah ditentukan.

**Kata kunci:** *k-nearest neighbor, klasifikasi, k-nearest neighbor classification, sepeda motor.*

### Abstract

*This research proposed a classification of motorcycle based on customer's characteristics. Motorcycles have different type and brand so that customers have many choices. Customer will choose motorcycle which they want to be based on different background. In this study, the customer will be grouped by motorcycle were purchased so that the seller can know characteristics of customers who buy certain type or brand. Characteristics of customers can be determined by age, gender, income, status and number of children. Based on these characteristic, we have to group for specifying motorcycle's type. In this research, the method used K-Nearest Neighbor (K-NN). K-NN algorithm is commonly used for classifying and searching for a group of test data with the majority of the group that has the shortest distance. The dataset used in this project is the final consumer characteristics. Trials with the dataset produce motorcycle brand of test data that has been determined.*

**Keywords:** *k-nearest neighbor, classification, k-nearest neighbor classification, motorcycle.*

## 1. PENDAHULUAN

Sepeda motor adalah kendaraan yang memiliki dua roda dimana mesin bekerja sebagai penggeraknya (Cossalter, 2006). Sepeda motor memiliki 2 roda yang sebaris sehingga sepeda motor dapat tetap stabil pada kecepatan tinggi yang disebabkan oleh gaya giroskopik. Sedangkan kestabilan pada kecepatan rendah bergantung pada pengaturan setang oleh pengendara. Sepeda motor sangat populer di Indonesia karena memiliki harga yang murah dan terjangkau untuk kalangan masyarakat dan biaya operasional yang cukup rendah.

Sepeda motor memiliki beberapa jenis dan merk yang berbeda. Hal ini menyebabkan banyaknya pilihan yang dimiliki konsumen. Konsumen akan memilih sepeda motor yang diinginkan

berdasarkan latar belakang yang berbeda. Maka dari itu mengetahui preferensi konsumen akan sangat menguntungkan penjual. Konsumen akan dikelompokkan berdasarkan sepeda motor yang dibeli sehingga penjual dapat mengetahui karakteristik konsumen yang membeli suatu jenis atau Merk tertentu.

Pengelompokkan dilakukan dengan metode KNN. Algoritma ini bekerja dengan cara mencari kelompok data uji berdasarkan k data tetangga terdekatnya. Algoritma ini menerima masukan berupa parameter dan nilai k.

Pada penelitian yang dilakukan oleh Nouvel (2015), klasifikasi kendaraan roda empat menggunakan metode K-Nearest Neighbour (K-NN) dengan jumlah data sebanyak 14 memiliki tingkat akurasi sebesar 78,57% dan RMSE dari 0,23. Jika

jumlah data sebanyak 1728 data, tingkat akurasi sebesar 95,78%, RMSE 0,19 dan ROC daerah 0.99. Pengujian itu dilakukan untuk membuktikan bahwa tingkat akurasi yang dihasilkan dipengaruhi oleh jumlah data latih yang digunakan. Jika semakin banyak data yang dilatih maka semakin tinggi juga tingkat akurasinya.

Permasalahan tersebut melatarbelakangi peneliti untuk melakukan penelitian mengenai pengelompokan sepeda motor berdasarkan karakteristik konsumen dengan metode K-NN. Penelitian ini akan menghasilkan sistem pengelompokan sepeda motor berdasarkan karakteristik dari konsumen.

## 2. DASAR TEORI

### 2.1 Karakteristik Pembeli

Data yang digunakan adalah data karakteristik pembeli yang diadopsi dari GitHub dengan jumlah data sebesar 2500 data. Data terdiri dari 5 parameter yaitu

1. Usia  
Parameter yang menyimpan usia dari pembeli.
2. Pendapatan  
Parameter yang menyimpan pendapatan per bulan dari pembeli.
3. Status Pernikahan  
Parameter yang menyimpan status pernikahan pembeli. Status pernikahan dibagi menjadi 4 yaitu *Divorce*, *Single*, *Married* dan *Widowed*.
4. Jenis Kelamin  
Parameter yang menyimpan jenis kelamin pembeli. Jenis kelamin dibagi menjadi 2 yaitu *Male* dan *Female*.
5. Jumlah Anak  
Parameter yang menyimpan jumlah anak yang dimiliki oleh pembeli.

Data dari GitHub tersebut diadopsi dengan mengganti kelas data mobil, menjadi kelas data sepeda motor sebagai data simulasi, dengan mempertimbangkan kesetaraan harga mobil dan motor dari yang paling mahal sampai yang murah. Kelas data motor yang digunakan dibagi menjadi 4 kelas antara lain Vario, Mio, Next dan Beat.

### 2.2 Konsep Big Data

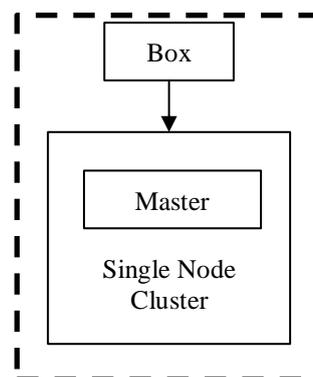
Big data merupakan istilah umum untuk sekumpulan data yang jumlahnya sangat besar dan kompleks sehingga tidak mudah untuk ditangani atau proses hanya dengan metode pemrosesan data biasa. Terdapat tiga masalah utama yang diselesaikan oleh *big data*, antara lain (Pawitra, 2016):

1. *Volume*  
Ukuran data yang disimpan atau diproses.
2. *Velocity*  
Kecepatan membuat data. Kecepatan data dibuat umumnya berbanding lurus dengan volume data.
3. *Variety*  
Keberagaman data yang diolah. Dari segi format maupun struktur data.

Teknologi yang berkaitan dengan big data akan memudahkan proses pengumpulan data-data yang sebelumnya tidak bisa atau sulit untuk dikumpulkan.

#### 2.2.1 Single Node

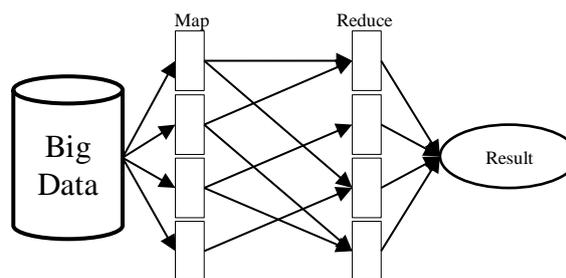
Hadoop *single node* menggunakan 1 mesin / computer saja dalam melakukan prosesnya. Secara default Hadoop dikonfigurasi untuk berjalan pada mode *non-distributed* (berdiri sendiri). Komputer didesain sebagai *master* bukan *slave* sehingga semua proses dilakukan dalam satu mesin seperti yang ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur Single Cluster

#### 2.2.2 Mapreduce

Pada Gambar 2, Mapreduce bertujuan untuk memproses data yang memiliki ukuran yang besar secara terdistribusi dan parallel dalam kluster yang terdiri atas ribuan computer. Dalam prosesnya, mapreduce dibagi menjadi 2 proses yaitu map dan reduce. Map berfungsi dalam pengumpulan informasi dari data-data yang terdistribusi dalam tiap komputer. Keluaran dari proses map akan digunakan dalam proses reduce. Proses reduce berfungsi dalam penggabungan atau pengelompokan berdasarkan kata kunci (Dean & Ghemawat, 2004).



Gambar 2. Cara Kerja MapReduce

## 2.3 KNN

K-Nearest Neighbor (K-NN) merupakan algoritma untuk menentukan kelas objek data uji berdasarkan K objek pada data latih yang terdekat (mirip). Algoritma ini termasuk instance-based learning dan merupakan salah satu teknik lazy learning. Dasar Algoritma K-Nearest Neighbour (Brammer, 2007):

- Tentukan data latih yang paling dekat dengan data uji.
- Kelas yang paling sering muncul dari  $k$  data latih yang terdekat akan dipilih.

### 2.3.1 Inisialisasi

Menentukan parameter  $K$ , dimana  $K$  merupakan jumlah dari tetangga terdekat, nilai  $K$  untuk menguji data uji ditentukan berdasarkan nilai  $K$  optimum pada saat *training*.

### 2.3.2 Alokasi Data

Alokasikan data uji dengan cara menghitung jarak setiap atribut data uji terhadap data latih dengan persamaan (1).

$$d(x_1, x_2) = (x_{11} - x_{21})^2 + (x_{12} - x_{22})^2 \quad (1)$$

dimana  $x_{ij}$  menyatakan koordinat titik  $x_i$  pada dimensi ke- $j$  dan  $d$  merupakan jarak. Setelah menghitung jarak data uji terhadap data latih, urutkan jarak setiap atribut data uji, kelompokkan menurut jarak yang terdekat.

### 2.3.3 Klasifikasi

Klasifikasi kelompok data uji dilakukan dengan cara mengumpulkan kategori  $Y$  (klasifikasi nearest neighbour). Lalu, memilih kategori mayoritas dari  $K$  data yang ditentukan.

## 3. IMPLEMENTASI

### 3.1 Kode Program

Proses ini merupakan proses map dalam *MapReduce* kNN. Proses map sendiri berfungsi dalam pengumpulan informasi data-data yang menjadi input dalam proses kNN. Keseluruhan proses map dapat dilihat dalam Kode Program 1.

```

1  @Override
2  public void map(Object kunci, Text
3  hasil, Context konteks) throws
4  IOException, InterruptedException
5  {
6      String baris = hasil.toString();
7      StringTokenizer token = new
8      StringTokenizer(baris, ",");
9
10     double umur =
11     normalisasi(token.nextToken(), min_umur,
12     max_umur);
13

```

```

14     double penghasilan =
15     normalisasi(token.nextToken(),
16     min_penghasilan, max_penghasilan);
17     String status_pernikahan =
18     token.nextToken();
19     String kelamin = token.nextToken();
20     double anak =
21     normalisasi(token.nextToken(), minanak,
22     maxanak);
23     String motor =
24     token.nextToken();
25     double jarak = totaljarak(umur,
26     penghasilan, status_pernikahan,
27     kelamin, anak, umur2, penghasilan2,
28     status_pernikahan2, kelamin2, anak2);
29
30     KnnMap.put(jarak, motor);
31     if (KnnMap.size() > K)
32     {
33
34         KnnMap.remove(KnnMap.lastKey());
35     }
36 }

```

Penjelasan dari Kode Program 1:

- Baris 1-8 merupakan proses tokenisasi yaitu proses memecah baris menjadi beberapa kata.
- Baris 8-21 merupakan proses inisialisasi parameter. Terdapat beberapa parameter yang perlu dinormalisasi terlebih dahulu yaitu *age*, *income* dan *children*.
- Baris 23-28 merupakan proses perhitungan jarak data uji dan data latih.
- Baris 31 merupakan proses pembuatan *TreeMap* dengan jarak sebagai sebuah *key* dan model sepeda motor sebagai *value*.
- Baris 32-37 memproses *TreeMap* agar hanya memuat  $K$  data. Apabila *TreeMap* memiliki data lebih dari  $K$  data maka akan menghapus data yang tidak diperlukan.

Proses ini merupakan proses reduce dalam *MapReduce* kNN. Proses reduce sendiri berfungsi dalam penggabungan dan pengelompokan berdasarkan *key* atau kata kunci. Keseluruhan proses reduce dapat dilihat pada Kode Program 2.

```

1  public void reduce(NullWritable kunci,
2  Iterable<DoubleString> hasil, Context
3  konteks) throws IOException,
4  InterruptedException
5  {
6      for (DoubleString hsl : hasil)
7      {
8          String motor =
9          hsl.getmotor();
10         double jarak =
11         hsl.gettotaljarak();
12
13         KnnMap.put(jarak,
14         motor);
15         if (KnnMap.size() > K)
16         {
17             KnnMap.remove(KnnMap.lastKey());
18         }
19     }
20 }
21 List<String> baris = new
22 ArrayList<String>(KnnMap.values());

```

```

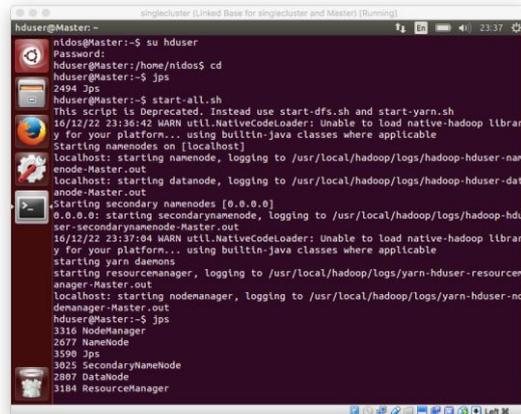
23
24         Map<String, Integer>
25 frekuensi = new HashMap<String,
26 Integer>();
27
28         for(int i=0; i< baris.size();
29 i++)
30         {
31             Integer frek =
32 frekuensi.get(baris.get(i));
33             if(frek == null)
34             {
35
36 frekuensi.put(baris.get(i), 1);
37             } else
38             {
39
40 frekuensi.put(baris.get(i), frek+1);
41             }
42         }
43
44         String MotorMuncul = null;
45         int frekuensi_maks = -1;
46         for(Map.Entry<String,
47 Integer> masukan: frekuensi.entrySet())
48         {
49             if(masukan.getValue() >
50 frekuensi_maks)
51             {
52                 MotorMuncul =
53 masukan.getKey();
54                 Frekuensi_maks =
55 masukan.getValue();
56             }
57         }
58
59         konteks.write(NullWritable.get(),
60 new Text(MotorMuncul));
61     }
62 }
    
```

Penjelasan dari Kode Program 2 :

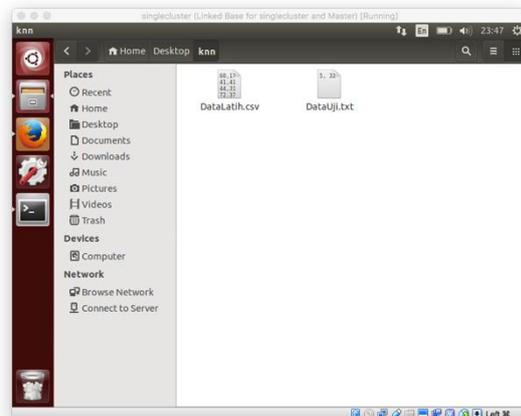
1. Baris 1-20 merupakan proses pembuatan *TreeMap* berdasarkan data dari objek *DoubleString* dengan jarak sebagai *key* dan model sepeda motor sebagai *value* dan mengatur agar *TreeMap* tidak melebihi K data.
2. Baris 21-22 merupakan proses menyimpan nilai dari *TreeMap* dalam sebuah *ArrayList*.
3. Baris 24-25 merupakan proses inialisasi sebuah *HashMap*.
4. Baris 27-41 merupakan proses menghitung banyaknya frekuensi dari model sepeda motor. *HashMap* digunakan untuk menyimpan nilai frekuensi tersebut sebagai *value* dan model sepeda motor sebagai *key*.
5. Baris 43-57 merupakan proses memeriksa *HashMap* untuk memperoleh model sepeda motor yang memiliki frekuensi paling tinggi.
6. Baris 59-62 merupakan proses penyimpanan hasil perhitungan frekuensi tertinggi ke dalam *context*.

### 3.2 Hadoop Single Node

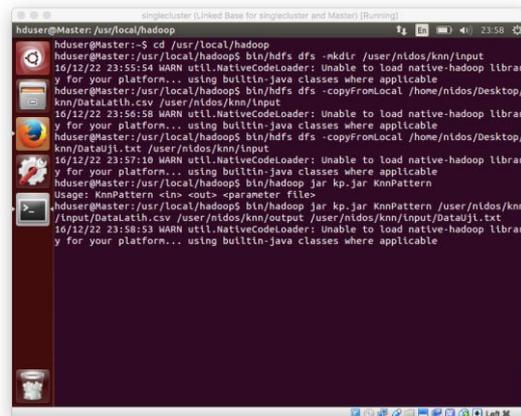
Langkah utama Hadoop Single Node yang perlu dilakukan dapat dilihat pada Gambar 3 hingga 6.



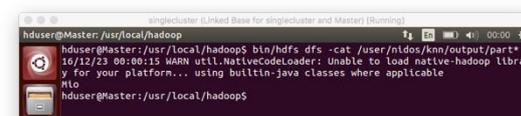
Gambar 3. Jalankan hadoop (start-all.sh)



Gambar 4. Folder Data Latih dan Data Uji



Gambar 5. Proses Klasifikasi



Gambar 6. Contoh Hasil Klasifikasi

Dengan penjelasan sebagai berikut:

- Langkah 1: Pada Gambar 3, masuk sebagai hduser, cek kondisi apakah hadoop telah berjalan atau belum dengan jps, jika belum maka jalankan hadoop dengan start-all.sh, lalu cek kembali kondisi hadoop dengan jps
- Langkah 2: Pada Gambar 4, menyiapkan file data latih dan data uji, dengan data latih berupa file .csv dan data uji berupa file .txt
- Langkah 3: Pada Gambar 5, masuk ke direktori hadoop dengan path /user/local/hadoop, lalu membuat direktori input pada hdfs, dan memasukkan file data latih serta data uji kedalam direktori input yang telah dibuat, selanjutnya cek bentuk format penjalanan programnya dengan bin/hadoop jar kp.jar KnnPattern, kemudian jalankan program dengan masukkan path file data latih, path direktori output, dan path file data uji dengan urutan <data latih> <output> <data uji>, lalu tekan enter dan tunggu program berjalan, dan lihat apakah program berjalan dengan benar tanpa error
- Langkah 4: Pada Gambar 6, menampilkan output hasil program dengan memasukkan path direktori output yang ditentukan pada saat menjalankan program pada langkah ke-3

Berdasarkan Gambar 3 hingga 6, Hadoop Single node dapat berjalan dengan baik dalam melakukan komputasi klasifikasi sepeda motor digunakan kode program KnnPattern.java dan data latih yang didapatkan dari sumber ini ( <https://github.com/matt-hicks/MapReduce-KNN> ) dengan membuat analogi kesetaraan tingkat harga mobil, yang disesuaikan dengan urutan harga motor dalam penelitian ini sebagai kelas, baik untuk data latih maupun data uji.

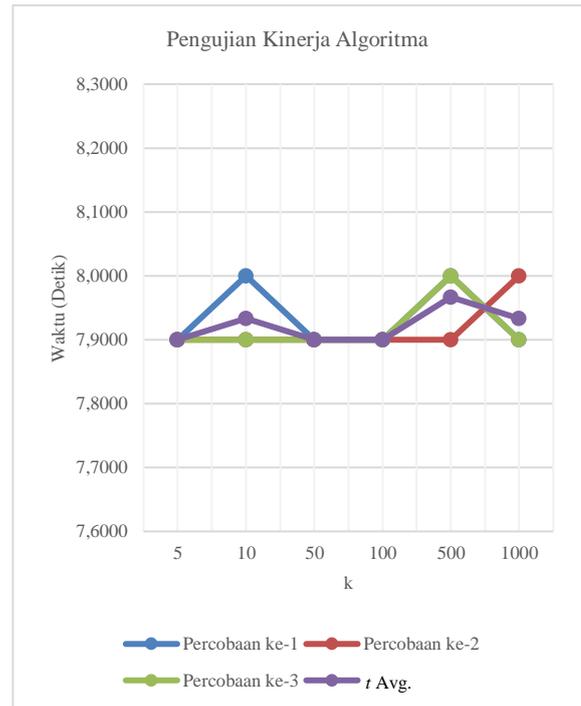
#### 4. PENGUJIAN DAN ANALISIS

##### 4.1 Pengujian Kinerja Algoritma

Pengujian kinerja algoritma dilakukan dengan cara menghitung seberapa cepat waktu yang dibutuhkan untuk mengolah data dengan nilai  $k$  yang bervariasi. Hasil dari pengujian dapat dilihat pada Tabel 1 dan Gambar 7.

Tabel 3.1 Hasil Pengujian Kinerja Algoritma

Nilai $k$	Lama Waktu Komputasi ( $t$ ) Percobaan ke- $i$			$t$ Avg.
	1	2	3	
5	7.9	7.9	7.9	7.9000
10	8.0	7.9	7.9	7.9333
50	7.9	7.9	7.9	7.9000
100	7.9	7.9	7.9	7.9000
500	8.0	7.9	8.0	7.9667
1000	7.9	8.0	7.9	7.9333



Gambar 7. Grafik Hasil Pengujian Kinerja Algoritma

Pengujian ini dilakukan dengan beberapa nilai  $k$  seperti yang dicantumkan pada Tabel 1, dan untuk setiap nilai  $k$ -nya akan dijalankan sebanyak 3 kali percobaan. Dimana hasil akan dilihat adalah lama waktu komputasi dalam satuan detik. Dari Tabel 1 dan Gambar 7, dapat dilihat bahwa lama waktu komputasi dalam ketiga percobaan tidak terlalu berbeda dengan perbedaan waktu hanya 0.1 detik pada beberapa nilai  $k$ , atau dapat dikatakan bahwa program dengan nilai  $k$  sebanyak 5 hingga 1000 dapat berjalan dengan lama waktu yang hampir sama. Namun, walaupun program dapat berjalan dengan waktu yang perbedaannya tidak begitu signifikan, pada klasifikasi atau keluaran tipe kelas untuk beberapa nilai  $k$  memiliki hasil yang berbeda-beda tergantung kelas apa yang menjadi mayoritas dalam nilai  $k$  tersebut. Hal ini menunjukkan juga, bahwa hadoop single node cluster memiliki kemampuan pengolahan data yang cepat dan handal.

#### 5. KESIMPULAN DAN SARAN

Pada implementasi hadoop single node menggunakan metode K-Nearest Neighbour (KNN) pada klasifikasi sepeda motor berdasarkan karakteristik konsumen. Klasifikasi dilakukan dengan menentukan jumlah tetangga terdekat dengan cara menghitung jarak antara data uji dan data latih. Hasil klasifikasi diperoleh dari mayoritas kelas/kategori dalam jumlah tetangga terdekat. Dalam implementasi ini, dilakukan pengujian lama waktu komputasi berdasarkan jumlah tetangga terdekat ( $k$ ) yakni sebanyak 5 hingga 1000 dengan percobaan sebanyak 3 kali dan waktu komputasi

pada tiap percobaan tidak terlalu signifikan dengan perbedaan waktu 0.1 detik. Hasil waktu rata-rata terbaik yaitu 7.9 detik pada nilai  $k$ , masing-masing 5, 50, dan 100.

Adapun saran untuk penelitian selanjutnya, yaitu metode yang digunakan dapat dilanjutkan dengan teknik lain untuk mengetahui perbandingan waktu komputasi terbaik antara hadoop single node cluster dan multi mode cluster, dan juga menambahkan variasi pengujian yang lainnya, misal menggunakan  $k$ -fold cross-validation.

## 6. DAFTAR PUSTAKA

- BRAMMER., 2007. *Principles of Data Mining*. UK: University of Portsmouth.
- COSSALTER, V., 2016. *Motorcycle Dynamics*.
- DEAN, J., & GHEMAWAT, S., 2004. *MapReduce: Simplified Data Processing on Large Clusters*. Google Corp.
- NOUVEL, AHMAD, 2015. Klasifikasi Kendaraan Roda Empat Berbasis KNN. *Jurnal Bianglala Informatika* Vol 3 No 2.
- PAWITRA, P. M., 2016. *Paper Basis Data*. Surakarta.
- S., RUSSEL, & P, NORVIG., 2010. *Artificial Intelligence A Modern Approach*. New Jersey: Pearson Education, Inc.
- X., WU, & V., KUMAR., 2009. *The Top Ten Algorithm in Data Mining*. Chapman and Hall.

## HIBRIDISASI ALGORITMA GENETIKA DENGAN *VARIABLE NEIGHBORHOOD SEARCH* (VNS) PADA OPTIMASI BIAYA DISTRIBUSI

Asyrofa Rahmi<sup>1</sup>, Wayan Firdaus Mahmudy<sup>2</sup>, Syaiful Anam<sup>3</sup>

<sup>1,2</sup>Fakultas Ilmu Komputer, Universitas Brawijaya

<sup>3</sup>Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Brawijaya

<sup>1</sup>asyrofarahmi@gmail.com, <sup>2</sup>wayanfm@ub.ac.id, <sup>3</sup>syaiful@ub.ac.id

(Naskah masuk: 25 Januari 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Proses distribusi dianggap sangat penting bagi perusahaan karena menjadi salah satu faktor yang mempengaruhi perolehan keuntungan. Besarnya biaya yang dikeluarkan serta kompleksnya permasalahan dalam proses distribusi menjadikan permasalahan distribusi sebagai topik yang perlu diteliti lebih mendalam lagi. Karena algoritma genetika (AG) sudah terbukti mampu memberikan solusi terbaik pada berbagai macam permasalahan optimasi dan kombinatorial, maka algoritma ini digunakan untuk menyelesaikan permasalahan distribusi pada penelitian ini. Namun, penerapan GA klasik memiliki kekurangan yaitu belum mencapai titik optimum global sehingga perlu dihibridisasi menggunakan algoritma *variable neighborhood search* (VNS). Algoritma ini dipilih karena selain mencari solusi secara global, algoritma ini juga mencari solusi secara lokal sehingga mampu menutupi kekurangan dari GA. Dengan menggunakan hibridisasi GA dengan VNS maka biaya yang diperoleh adalah 32392960 yang dibuktikan dengan penghematan biaya sebesar 323190 jika dibandingkan dengan GA klasik yaitu 32716150. Namun, dilihat dari waktu komputasi, GA-VNS membutuhkan waktu yang relatif sama dengan GA klasik yaitu 279332 ms (*milisecond*) dan 265091 ms.

**Kata kunci:** *distribusi, algoritma genetika, variable neighborhood search*

### Abstract

*The distribution process is considered importantly for the company as one of the factors that affects profitability. The costs incurred as well as the complexity of the distribution problems makes the distribution problems as a topic that need to be examined more deeply. Since the wide range of combinatorial and optimization problems have been ever solved by using genetic algorithm (GA) well then it is used to resolve the distribution problems in this study. However, the implementation of classical GA has the disadvantage that has not yet reached the global optimum so that needs to be hybridized by using variable neighborhood search (VNS) algorithm. The VNS algorithm has been chosen because its ability either to search the global solutions or local solutions. The local search of VNS algorithm is able to cover the shortage of the GA. By using hybridization of GA with VNS, the cost accrued is 32392960 as evidenced by cost savings of 323190 in comparison with the classical GA is 32716150. However, the computational time of GA-VNS is equal to its classical GA relatively.*

**Keywords:** *distribution, genetic algorithm, variable neighborhood search*

## 1. PENDAHULUAN

Kejelian para pengusaha sangat diuji dalam mengolah strategi perusahaan di tengah maraknya pasar global pada dekade ini. Hal ini ditujukan agar perusahaan mereka tetap memperoleh keuntungan dan masih mampu bersaing dengan perusahaan-perusahaan pesaingnya. Sebagai bagian dari *supply chain*, proses meminimalkan biaya dari proses distribusi merupakan salah satu faktor yang tidak bisa dihindarkan karena proses distribusi dengan cakupan wilayah yang luas membutuhkan biaya yang tidak sedikit. Proses distribusi atau pengiriman barang dari pabrik produksi agar sampai ke konsumen harus melalui beberapa distributor lokal di beberapa wilayah (*multi-level*) seperti *distributor centre, retailer, agen, etc.* (Guo dkk. 2015). Banyaknya kemungkinan jaringan distribusi yang

terbentuk untuk mendapatkan biaya minimal menjadi permasalahan yang perlu diperhitungkan. Penentuan jaringan distribusi akan lebih kompleks ketika suatu perusahaan memproduksi lebih dari satu jenis produk (*multi-produk*) (Gicquel & Minoux 2015; Kim & Lee 2016) selain itu, pemesanan produk oleh konsumen memiliki jumlah yang tidak pasti dengan jumlah jenis produk yang berbeda-beda (Langroodi & Amiri 2016).

Beberapa cara sudah pernah dicoba untuk menyelesaikan permasalahan distribusi salah satunya adalah menggunakan pemrograman matematika dengan memodelkan permasalahan ke dalam integer (Sitek & Wikarek 2012). Pada penelitian tersebut, model distribusi yang diperhitungkan adalah dua level yaitu menggunakan perantara pusat distributor untuk mengirim barang ke konsumen. Pencarian biaya yang minimal dan

optimal diselesaikan menggunakan aplikasi LINGO dan CPLEX sehingga aliran distribusi yang dicari dapat ditemukan yang ditunjukkan oleh beberapa variabel keputusan keluaran dari aplikasi. Keunggulan lainnya adalah menyediakan beberapa kombinasi yang memungkinkan dari analisis kapasitas yang tersedia pada distributor dan jumlah unit transportasi sehingga mendukung keputusan yang tepat dalam permasalahan distribusi. Namun, kelemahannya adalah tidak bisa dilakukan dalam proses nyata karena bentuk standard dari *linear programming* dan ukuran model yang diimplementasikan dalam penelitian ini. Kelemahan lainnya adalah pendekatan ini tidak mendukung proses nalar yang rasional.

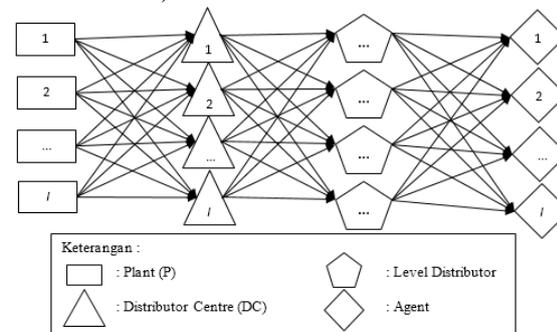
Berbagai permasalahan kombinatorial dengan kompleksnya beberapa *constraint* yang digunakan (Gupta dkk. 2012; Chen & Wang 2011) sudah teruji mampu diselesaikan menggunakan algoritma genetika termasuk permasalahan distribusi (Rahmi dkk. 2016). Permasalahan yang diangkat adalah distribusi banyak (multi) level dengan jenis produk berjumlah satu. Untuk mendapatkan jaringan distribusi yang tepat dengan biaya yang minimal maka beberapa *constraint* utama diperhitungkan. Dibandingkan dengan hasil keluaran *random search*, hasil solusi algoritma genetika yang diperoleh adalah mendekati optimal. Dalam penelitian awal tersebut, algoritma genetika (AG-awal) diimplementasikan dengan menggunakan *extended intermediate* sebagai model *crossover*, *insertion* sebagai model *mutation* dan *roulette-wheel* sebagai model *selection*.

Penelitian lanjutan dari AG-awal adalah menguji permasalahan distribusi yang sama dengan mengimplementasikan model operator algoritma genetika (AG-lanjut) yang berbeda yaitu menggunakan *crossover one-cut-point*, *swapmutation* dan *elitism selection* (Sarwani dkk. 2016). Hasil dapat dilihat dari eksekusi yang dilakukan sebanyak 10 kali untuk mengetahui seberapa jauh pengacakan yang dilakukan algoritma genetika baik AG-awal maupun AG-lanjut dengan hasil *random search*. Berdasarkan nilai *fitness* dari 10 eksekusi, didapatkan bahwa AG-awal dibandingkan *random search* memiliki hasil akhir yang tidak terlalu jauh perbedaannya. Hal ini dikarenakan model operator yang digunakan dalam algoritma genetika memberikan hasil yang konvergensi dini sehingga kurang bereksplorasi dalam mencari solusi sedangkan hasil AG-lanjut dibandingkan *random search* menunjukkan hasil perbedaan yang cukup signifikan. Namun, secara keseluruhan hasil AG-lanjut lebih berkualitas dibandingkan dengan AG-awal sehingga dapat disimpulkan bahwa berdasarkan permasalahan distribusi pada kedua penelitian tersebut (AG-awal dan AG-lanjut), algoritma genetika dengan operator *crossover one-cut-point*, *swapmutation* dan *elitism selection* sangat direkomendasikan.

Meskipun menggunakan model operator yang berbeda, penggunaan algoritma genetika yang masih dasar (Boudjelaba dkk. 2014) memberikan perbedaan yang cukup signifikan pada 10 hasil eksekusi AG-lanjut (Rahmi dkk. 2016). Hal ini menunjukkan bahwa dia tidak secara keseluruhan terjebak dalam konvergensi dini karena masih bisa bereksplorasi lebih jauh dalam mencari solusi. Disamping itu, perolehan hasil tersebut dapat diperbaiki sampai hasil yang diberikan masing-masing eksekusi memiliki perbedaan yang tidak terlalu signifikan sehingga perlu dihibridisasi dengan algoritma pencari global dan pencari lokal seperti *variable neighborhood search* agar bisa memberikan hasil yang lebih baik (Mahmudy dkk. 2014).

## 2. METODOLOGI

Proses distribusi atau pengiriman barang dari pabrik produksi agar sampai ke konsumen harus melalui beberapa distributor lokal di beberapa wilayah (multi-level) seperti *distributor centre, retailer, agen*, etc. (Guo dkk. 2015). Banyaknya kemungkinan jaringan distribusi yang terbentuk untuk mendapatkan biaya minimal menjadi permasalahan yang perlu diperhitungkan. Penentuan jaringan distribusi akan lebih kompleks ketika suatu perusahaan memproduksi lebih dari satu jenis produk (multi-produk) (Gicquel & Minoux 2015; Kim & Lee 2016) selain itu, pemesanan produk oleh konsumen memiliki jumlah yang tidak pasti dengan jumlah jenis produk yang berbeda-beda (Langroodi & Amiri 2016)



Gambar 1. Jaringan Distribusi

Berdasarkan Gambar 1, model distribusi multi-level memiliki banyak kemungkinan solusi untuk membentuk jaringan distribusi yang tepat yang ditandai dengan minimalnya biaya yang dikeluarkan (Sitek & Wikarek 2012; Rahmi dkk. 2016; Sarwani dkk. 2016). Untuk meminimalkan biaya distribusi maka perlu disusun formulasi matematis dalam beberapa variabel untuk pencarian biaya yang minimal beserta beberapa *constraint* utama yang akan mempengaruhi jaringan distribusi (Sitek & Wikarek 2012).

Data pada penelitian ini merupakan data rancangan yang diperoleh proses wawancara dengan beberapa pakar distribusi yang menjadi ahli dan bekerja di perusahaannya masing-masing. Data

disesuaikan dengan permasalahan pada penelitian ini sehingga data yang digunakan adalah data kapasitas stok setiap produk untuk setiap level, kapasitas kendaraan, jumlah kendaraan, biaya tetap kendaraan (Sarwani dkk. 2016; Rahmi dkk. 2016), serta biayadari beberapa produk untuk setiap level distributor.

Agar permasalahan distribusi mampu diselesaikan dengan baik maka perlu dibentuk formulasi matematika baik untuk fungsi tujuan maupun *constraint* yang mengikutinya. Pertama adalah membentuk formulasi untuk menjabarkan fungsi tujuan dari distribusi yaitu biaya yang paling minimal sehingga untuk pencarian biaya, fungsi digunakan ada pada persamaan (1) yang terdiri dari beberapa variabel.

$$z = \sum_{i=0}^I \left[ \sum_{j=0}^J \sum_{k=0}^{K_j} \sum_{r=0}^R \sum_{p=0}^{P_j} ((X_{ijkp} Cvb_{ijp}) + Cf_{ijk}) St_{ijk} \right] \tag{1}$$

dimana  $i \in \{1, 2, \dots, I\}$  adalah level pengiriman distribusi,  $j \in \{1, 2, \dots, J\}$  adalah unit distributor yang berperan sebagai pengirim sedangkan  $r \in \{1, 2, \dots, R\}$  sebagai pemesan.  $p \in \{1, 2, \dots, P\}$  adalah jenis produk yang dipesan dan  $k \in \{1, 2, \dots, K\}$  adalah kendaraan yang dimiliki oleh unit distributor pengirim  $j$  untuk mendistribusikan produknya.  $X_{ijkp}$  merepresentasikan jumlah unit barang yang akan dikirim,  $Cvb_{ijp}$  merupakan biayavariabel untuk setiap produk,  $Cf_{ijk}$  merupakan biaya tetap pengiriman, dan  $St_{ijk}$  merupakan status bahwa distributor pada level  $i$  melakukan pengiriman atau tidak dengan kendaraan  $k$  nya  $\{0,1\}$ .

Setelah mengetahui formulasi matematika dari fungsi tujuan permasalahan distribusi maka selanjutnya adalah membentuk formulasi matematika untuk *constraint* yang harus dipenuhi sehingga solusi yang dihasilkan tidak hanya biaya yang minimal saja namun juga memenuhi *constraint* yang ada sehingga permasalahan distribusi bisa terselesaikan dengan benar.

*Fungsi constraint permintaan order*

*Constraint* pertama ini berhubungan dengan batasan jumlah order dari pemesan sehingga jumlah barang yang dipesan oleh pemesan ( $Or_{irp}$ ) harus sama dengan barang yang dikirimkan nantinya kepada pemesan. Fungsi *constraint* untuk jumlah order ditunjukkan pada persamaan (2).

$$\sum_{j=0}^J \sum_{k=0}^{K_j} \sum_{r=0}^R \sum_{p=0}^{P_j} X_{ijkp} = Or_{irp} \tag{2}$$

*Fungsi constraint kapasitas kendaraan untuk setiap unit distributor*

*Constraint* yang kedua adalah pada kapasitas kendaraan yang digunakan. Suatu barang dikirim ke pemesan maupun level dibawahnya kemudian barang akan diangkut oleh kendaraan. Permasalahannya

adalah kapasitas kendaraan dari setiap distributor pengirim ( $Cp_{ijp}$ ) memiliki batasan kapasitas yang tidak boleh dilanggar. Fungsi *constraint* untuk kapasitas kendaraan ditunjukkan pada persamaan (3).

$$\sum_{j=0}^J \sum_{k=0}^{K_j} \sum_{r=0}^R \sum_{p=0}^{P_j} X_{ijkp} \leq Cp_{ijp} \tag{3}$$

*Fungsi constraint persediaan barang untuk setiap produk pada unit distributor pengirim*

*Constraint* yang terakhir adalah *constraint* untuk ketersediaan stok pada setiap unit distributor pengirim ( $Kcp_{ijkp}$ ). Setiap distribusi selalu memiliki stok ketersediaan barang. Ketika suatu distributor berperan sebagai pengirim, maka dilakukan pengecekan stok bahwa jumlah barang yang akan dikirim tidak boleh melebihi stok. Fungsi *constraint* untuk stok unit distribusi pengirim ditunjukkan pada persamaan (4)

$$\sum_{j=0}^J \sum_{k=0}^{K_j} \sum_{r=0}^R \sum_{p=0}^{P_j} X_{ijkp} \leq Kcp_{ijkp} \tag{4}$$

**3. VARIABLE NEIGHBORHOOD SEARCH (VNS)**

VNS merupakan algoritma yang menerapkan proses pencarian solusi ketetanggaan yang jauh dan dekat sehingga memberikan kemungkinan besar menghasilkan solusi yang unggul. Pencarian solusi pada ketetanggaan yang jauh diartikan sebagai pencarian dengan sistem *random* sehingga bisa menjangkau cakupan masalah dengan skala yang besar. Setelah dilakukan pengacakan dan didapatkan solusi pada suatu area tertentu maka dilanjutkan dengan pencarian ketetanggaan yang dekat yang biasa dikenal sebagai pencarian lokal. Dengan adanya pencarian lokal tersebut, maka solusi paling unggulnya akan ditemukan (Mladenovic dkk. 2016).

Dengan kemampuan yang dimiliki VNS dalam mencari solusi tersebut maka tidak terjebak dalam solusi optimum lokal dan bisa mencapai solusi optimum global menjadi keunggulannya (Castelli & Vanneschi 2014). Gambar 2 merupakan tahapan umum dalam proses VNS (Cheikh dkk. 2015). Algoritma ini dipertimbangkan karena pernah diterapkan untuk menyelesaikan masalah transportasi (Ziebuhr & Kopfer 2016).

**4. HIBRIDISASI ALGORITMA GENETIKA DAN VNS (GA-VNS)**

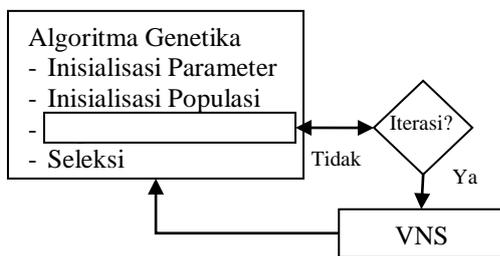
Pada penelitian ini, algoritma genetika (GA) digunakan untuk menyelesaikan permasalahan distribusi multi-level multi-produk. GA merupakan metode yang mengadopsi sifat alami biologi individu pada seleksi alam (Qiongbing 2016) Terdapat beberapa langkah dalam menyelesaikan permasalahan menggunakan GA yaitu dimulai dari melakukan representasi kromosom, reproduksi,

evaluasi dan seleksi. Proses evaluasi pada GA digunakan untuk mengukur seberapa baik sebuah individu untuk dapat memberikan solusi dengan memperhatikan nilai *fitness* sehingga tingginya nilai *fitness* yang dihasilkan oleh sebuah individu maka pada generasi selanjutnya semakin tinggi kemungkinannya terpilih sebagai sebuah solusi (Thakur & Kumar 2016). Arsitektur hibridisasi GA-VNS ditunjukkan pada Gambar 3.

```

PROCEDURE VNS
INPUT :
    xtemp, lgmax
OUTPUT :
    x
x sebagai solusi awal;
Atur ketetanggan dengan lg={1,2,...,lgmax}
lg = 1;
WHILE lg < lgmax do
    xtemp = Acak x
    xlokal = Local(xtemp)
    IF (f(xlokal) > f(x)) then
        x = xlokal
        lg = 1
    ELSE
        lg = lg+1;
    ENDIF
END WHILE
Mengembalikan nilai x sebagai solusi terbaik
    
```

Gambar 2. Pseudocode algoritma VNS



Gambar 3. Arsitektur hibridisasi GA-VNS

Kuantitas dilakukannya proses hibridisasi algoritma genetika adalah dengan persentase 50 berdasarkan iterasi/generasi maksimal yang sudah diinisialisasi sehingga hibridisasi VNS dalam memperbaiki hasil reproduksi dilakukan ketika di pertengahan generasi dan di akhir generasi.

**4.1. Representasi Kromosom dan Inisialisasi Populasi**

Representasi kromosom merupakan suatu proses untuk mengajukan solusi penyelesaian masalah, dimana suatu solusi permasalahan dapat dikodekan kedalam pengkodean yang terdapat pada algoritma genetika (Mahmudy dkk. 2013). Oleh sebab itu, proses representasi kromosom adalah proses awal dan yang paling penting untuk memberikan sebuah solusi permasalahan. Terdapat berbagai macam pengkodean pada algoritma genetika dan pada penelitian ini menggunakan pengkodean berjenis riil (*real-coded*). Struktur

kromosom sebagai solusi dalam proses algoritma genetika memiliki struktur solusi seperti Gambar 4.

Setiap kolom yang terdapat pada solusi merupakan representasi dari jumlah unit barang yang akan dikirim ke distributor pemesan. Setiap kromosom juga dibagi menjadi beberapa segmen. Banyaknya segmen tergantung pada jumlah level yang dibutuhkan berdasarkan pemesanan. Setiap segmen memiliki beberapa subsegmen dan subsegment dari segmen level *i* adalah unit distributor pengirim *j* sesuai dengan level yang ada pada segmen level. Subsegment distributor pengirim memiliki suatu subsegment yaitu kendaraan *v* yang digunakan untuk melakukan pengiriman. Selanjutnya subsegment kendaraan adalah unit distributor yang melakukan pemesanan dan jenis produk yang dipesan *p* menjadi subsegment dari distributor pemesan.

						Level <i>i</i>			...	<i>J</i>	
						Pengirim <i>j</i>			...	<i>J</i>	
						Kendaraan <i>v</i>			...	<i>V</i>	
						Pemesan <i>r</i>			...	<i>R</i>	
<i>P</i>	...	<i>P</i>		<i>P</i>	...	<i>P</i>					
10		111		11		11					

Gambar 4. Struktur Kromosom

Representasi solusi untuk solusi permasalahan distribusi pada Gambar 4 memiliki nilai 10 pada kolom pertama. Arti dari nilai 10 tersebut adalah pada level awal, distributor awal *j* menggunakan kendaraan awal *v* menuju ke distributor pemesan awal *r* mengirimkan produk berjenis *p* awal sebanyak 10 unit produk. Nilai riil pada kolom yang lain juga memiliki makna yang sama sesuai dengan segmen masing-masing. Panjang kromosom didapatkan dari perkalian antara jumlah jenis produk yang dipesan oleh setiap distributor pemesan dan jumlah kendaraan distributor pengirim pada setiap level dan hasil perkalian setiap level dijumlahkan.

**4.2. Fungsi Fitness**

Sebuah kromosom/individu yang dibuat oleh algoritma genetika memiliki nilai fungsi yang digunakan untuk mengukur seberapa baik kromosom tersebut untuk memberikan solusi. nilai fungsi yang digunakan yaitu nilai *fitness*. Pada proses seleksi nilai *fitness* digunakan untuk menentukan kromosom yang dipilih. Semakin besar nilai *fitness* pada kromosom maka semakin besar pula kemungkinan untuk bisa terpilih menjadi sebuah solusi.

Pada penelitian ini fungsi *fitness* digunakan adalah untuk mendapatkan nilai minimum biaya distribusi. sehingga rumus yang digunakan untuk mencari nilai *fitness* ditunjukkan pada Persamaan (5).

$$fitness = \frac{1}{Z} \tag{5}$$

dimana  $Z$  merupakan biaya distribusi yang ditunjukkan pada Persamaan (1) yang dimiliki oleh setiap kromosom

### 4.3. Reproduksi

Tahap reproduksi merupakan proses unik yang dimiliki oleh algoritma genetika karena pembentukan individu/solusi baru berada pada proses ini. Proses reproduksi memiliki 2 operator yaitu pindah silang (*crossover*) dan mutasi yang bertujuan untuk menggali lebih jauh dan lebih dalam untuk mencari individu baru sehingga ditemukannya individu baru yang unggul dalam menyelesaikan permasalahan (Jiao dkk. 2014). Pada masing-masing operator terdapat *crossover rate* ( $cr$ ) pada operator *crossover* dan *mutation rate* ( $mr$ ) pada operator mutasi yang fungsinya adalah sebagai petunjuk berapa jumlah individu baru yang dihasilkan pada setiap proses dalam satu generasi.

#### 4.3.1. Crossover

Salah satu operator pembentuk individu baru dalam proses reproduksi adalah *crossover*. Dikarenakan operator ini membentuk individu baru dari dua individu induk maka hasil yang diperoleh akan beragam karena mewarisi sifat dari dua induk. Jumlah individu baru didapatkan dari perkalian ukuran populasi dan *crossover rate* ( $popSize * cr$ ). Operator *crossover* memiliki beragam model yang bisa dipertimbangkan beberapa diantaranya adalah *one-cut-point*, *two-cut-point*, dan *extended intermediate*.

##### 4.3.1.1. One-cut-point

Model *one-cut-point* merupakan salah satu model *crossover* dengan satu titik pemotongan pada posisi gen. Selanjutnya dilakukan pemotongan posisi gen pada kedua individu induk yang sudah terpilih secara *random* dan disilangkan serangkaian gen sampai pada posisi titik pemotongan gen pada individu induk pertama ke posisi gen pada individu induk kedua (Arora & Arora 2012). Ilustrasi proses *crossover* dengan model *one-cut-point* ditunjukkan pada Gambar 5 dengan perpotongan pada posisi gen ke 5.

$x_1$	10	34	122	...	21	7	51	...	...	...	...	...
$y_1$	30	21	29	...	13	26	4	...	...	...	...	...
$z_1$	30	21	29	...	13	7	51	...	...	...	...	...
$z_2$	10	34	122	...	21	26	4	...	...	...	...	...

Gambar 5. Proses perhitungan *one-cut-point*

dimana  $x \in \{x_1, \dots, x_n\}$  dan  $y \in \{y_1, \dots, y_n\}$  adalah dua individu induk terpilih secara *random* dan  $z \in \{z_1, \dots, z_n\}$  adalah hasil proses *crossover* yang biasa disebut dengan individu baru/anak.

##### 4.3.1.2. Two-cut-point

*Two-cut-point* merupakan pengembangan dari model *one-cut-point* yang menerapkan titik pemotongan pada dua posisi gen secara *random* dalam individu/kromosom sehingga individu baru hasil model ini adalah ditukarnya nilai gen antara kedua posisi terpilih ke individu yang kedua (Arora & Arora 2012). Gambaran model *two-cut-point* dengan posisi gen ke 2 dan 6 diilustrasikan pada Gambar 6.

$x_1$	10	34	122	...	21	7	51	...	...	...	...	...
$y_1$	30	21	29	...	13	26	4	...	...	...	...	...
$z_1$	10	34	29	...	13	26	51	...	...	...	...	...
$z_2$	30	21	122	...	21	7	4	...	...	...	...	...

Gambar 6. Proses perhitungan *two-cut-point*

##### 4.3.1.3. Extended intermediate

Sebagai salah satu model *crossover*, *extended intermediate* merupakan model yang bertujuan untuk mengubah nilai gen berdasarkan bilangan acak  $\alpha$  dengan selisih nilai gen pada kedua individu. (Mahmudy 2015). Persamaan (6) merupakan fungsi pencarian hasil dengan model *extended intermediate*.

$$z_i = x_i + \alpha(y_i - x_i) \tag{6}$$

#### 4.3.2. Mutasi

Operator kedua dalam proses reproduksi yang menjadi ciri khas algoritma genetika adalah mutasi. Berbeda dengan operator *crossover* yang harus menggunakan dua individu induk, operator mutasi hanya menggunakan satu induk saja dalam menghasilkan individu baru. Jumlah individu baru yang dihasilkan berpacu pada hasil perkalian antara ukuran populasi ( $popsize$ ) dan *mutation rate*. Operator mutasi memiliki beberapa model, sebagian diantaranya adalah *swap*, *insertion*, dan *simple random*.

##### 4.3.2.1. Swap

Salah satu model operator mutasi adalah *swap*. Cara kerja model operator mutasi ini adalah menukarkan nilai antar dua posisi gen yang sudah terpilih secara acak (Soni & Kumar 2014).

##### 4.3.2.2. Insertion

Model *insertion* merupakan model yang menyisipkan nilai dari gen posisi tertentu ke posisi gen yang lain yang sudah dibangkitkan secara acak. Setelah disisipkan, posisi gen-gen yang masuk dalam *range* penyisipan akan digeser sampai menempati posisi yang pas dengan jumlah gennya (Soni & Kumar 2014).

### 4.3.2.3. Simple random

Simple random merupakan salah satu model operator mutasi yang membangkitkan bilangan acak  $r$  untuk memperbesar atau memperkecil nilai dari setiap gen setelah dijumlahkan dengan 1. Range  $r$  sendiri berkisar pada -0.1 sampai 0.1. Persamaan (7) merupakan formulasi matematika pada perhitungan simple random.

$$x'_i = x_i(1 + r) \quad i = 1, \dots, n \quad (7)$$

Berdasarkan algoritma yang diusulkan, setelah dilakukan proses reproduksi maka akan memasuki proses kondisi apakah hasil individu baru dari proses reproduksi perlu diperbaiki lagi ke proses hibridisasi menggunakan *variable neighborhood search* (VNS). Penentuan dilakukannya hibridisasi atau tidak, dilihat dari banyaknya iterasi/generasi yang sedang berjalan. Karena persentase dilakukannya proses hibridisasi adalah 50, maka proses hibridisasi dilakukan ketika banyaknya generasi mencapai generasi maksimal (akhir) dan setengah dari generasi yang dideklarasikan di awal. Contoh jika generasi yang dideklarasikan di awal adalah 1000, maka hibridisasi untuk memperbaiki hasil reproduksi dilakukan pada generasi akhir yaitu 1000 dan setengahnya yaitu 500. Hasil perbaikan dari hasil reproduksi inilah yang akan masuk ke dalam proses seleksi.

### 4.4. Seleksi

Langkah selanjutnya adalah proses seleksi untuk memilih individu/kromosom sejumlah ukuran populasi yang akan bertahan menjadi populasi baru pada proses generasi selanjutnya berdasarkan nilai *fitness* yang diperoleh. Ada beberapa jenis model seleksi dalam algoritma genetika seperti *roulette wheely* yang memperhitungkan probabilitas kumulatif nilai *fitness* setiap individu dan *elitism* yang pernah digunakan dalam menyelesaikan permasalahan distribusi multi-level (Rahmi dkk. 2016)(Sarwani dkk. 2016). Model seleksi *elitism* sendiri adalah proses seleksi dengan mengurutkan secara *descending*. Selain *roulette wheely* dan *elitism*, juga terdapat model seleksi *binary tournament*. Model seleksi ini adalah membandingkan nilai *fitness* antara 2 individu yang terpilih secara acak dan individu dengan *fitness* terbesar akan terpilih di populasi baru pada generasi selanjutnya.

## 5. HASIL DAN DISKUSI

Pada tahap ini dilakukan berbagai pengujian untuk menunjang keoptimalan algoritma yang diusulkan seperti yang pertama adalah pengujian parameter yang mencakup pengujian ukuran populasi, banyaknya generasi, nilai  $cr$ , dan nilai  $mr$  berdasarkan nilai rata-rata *fitness* terbaik yang diperoleh. Pengujian selanjutnya adalah pengujian tentang pencarian model operator genetika dari proses reproduksi yang terbaik termasuk didalamnya

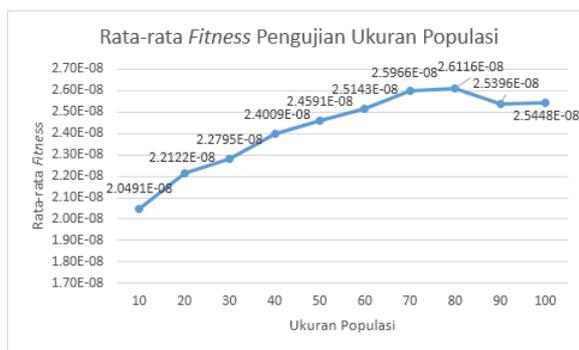
adalah model *crossover* dan mutasi. Setelah dilakukan pengujian parameter dan pengujian pencarian model operator reproduksi yang cocok maka dilakukan analisa dan diskusi seberapa baik model yang diusulkan dalam penelitian ini dalam menyelesaikan permasalahan distribusi multi-level multi-produk.

### 5.1. Pengujian Parameter GA

Pengujian pertama adalah pengujian parameter dari algoritma genetika. Pengujian ini dilakukan untuk memperoleh hasil algoritma genetika terbaik dan stabil dilihat dari rata-rata *fitness*. Nilai rata-rata *fitness* diperoleh dari pengujian satu ukuran dieksekusi sebanyak sepuluh kali dengan asumsi sudah mewakili solusi yang ditawarkan karena mengingat bahwa sifat algoritma genetika yang *stochastic*. Pengujian parameter algoritma genetika yang dimaksud adalah pengujian tentang ukuran populasi, banyaknya generasi, dan kombinasi dari nilai  $cr$  dan  $mr$ .

#### 5.1.1. Ukuran Populasi (*Popsize*)

Penerapan multi-populasi dalam menawarkan beragam solusi membuat algoritma genetika mampu menyelesaikan permasalahan optimasi dalam beberapa bidang. Berdasarkan rata-rata *fitness*, pengujian ukuran populasi bertujuan untuk mengetahui ukuran populasi yang tepat dalam menawarkan solusi sehingga mendapatkan hasil yang optimal. Gambar 7 merupakan hasil pengujian ukuran populasi yang ditunjukkan dengan grafik garis dari rata-rata *fitness*. Parameter awal yang digunakan adalah banyaknya generasi 300, nilai  $cr$  0.5, dan nilai  $mr$  0.5 karena sudah pernah diuji pada penelitian distribusi sebelumnya, begitu juga dengan model *crossover*, *mutation*, dan *selection* yang digunakan (Sarwani dkk. 2016).



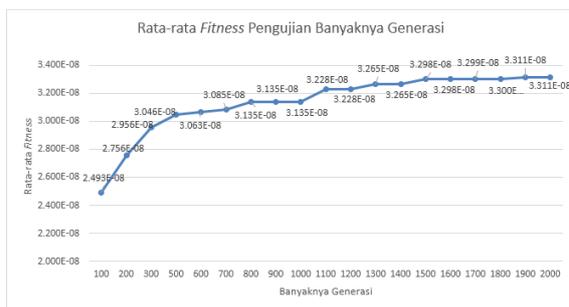
Gambar 7. Pengujian ukuran populasi

Penentuan *popsize* dilihat dari titik awal terjadinya kondisi konvergen berdasarkan grafik garis dari rata-rata *fitness* pada Gambar 7. Kondisi konvergen adalah kondisi ketika pada pengujian yang lebih besar, hasil yang ditunjukkan memiliki perbedaan yang relatif sedikit sehingga

berdasarkan Gambar 7, *popsi* 60 menjadi titik awal terjadinya konvergen. Pada beberapa *popsi* dibawah 60, hasil rata-rata *fitness* pada grafik garis menunjukkan peningkatan hasil yang signifikan. Hal ini disebabkan oleh kecilnya *popsi* memiliki arti sedikitnya solusi yang ditawarkan sehingga masih sulit untuk menemukan solusi yang baik dari sekumpulan solusi tersebut sedangkan pada beberapa *popsi* yang lebih dari 60, hasil rata-rata *fitness* juga mengalami peningkatan sampai pada ukuran 80 meskipun perbedaannya tidak terlalu banyak, namun ketika *popsi* lebih dari 90 terjadi penurunan hasil yang mengartikan bahwa semakin besar *popsi*, peluang solusi yang ditawarkan sudah memiliki solusi terbaiknya sehingga jika diolah kembali, maka tidak lebih baik hasil yang diperoleh. Meskipun *popsi* 90 dan 100 mengalami penurunan, penurunannya tidak akan berbeda jauh dari hasil *popsi* terpilih yaitu 60 (Rahmi dkk, 2016).

**5.1.2. Banyaknya Generasi**

Selanjutnya adalah pengujian parameter banyaknya generasi yang tujuannya adalah mendapatkan banyaknya generasi yang mendekati optimal dengan waktu komputasi yang relatif singkat. Untuk menentukan banyaknya generasi yang cocok sesuai dengan tujuan yang diinginkan adalah dengan memperhatikan banyaknya generasi mana yang menjadi titik dimulainya konvergensi. Konvergensi merupakan kondisi ketika pada banyaknya generasi yang semakin besar solusi yang ditawarkan tidak memiliki selisih yang terlalu jauh. Gambar 8 merupakan grafik hasil pengujian banyaknya generasi dengan ukuran populasi terbaik hasil dari pengujian sebelumnya 60 sedangkan kombinasi *cr* dan *mr* yang digunakan masih sama dengan pengujian sebelumnya.



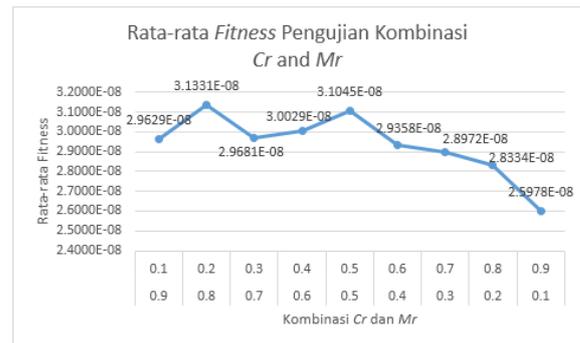
Gambar 8. Pengujian Banyaknya Generasi

Pengujian banyaknya generasi pada Gambar 8 menunjukkan bahwa rata-rata terbaik setiap percobaan mengalami ketidakstabilan meskipun pada generasi tertinggi. Hal ini disebabkan oleh tingkat kompleksitas permasalahan yang cukup tinggi sedangkan berdasarkan nilai *fitness* terbaiknya, titik konvergensi *fitness* berada pada generasi ke 1100. Titik ini dipilih karena perubahan nilai *fitness* terbaik pada generasi yang semakin besar

tidak terlalu memberikan perbedaan yang jauh.

**5.1.3. Kombinasi crossover rate (cr) dan mutation rate (mr)**

Yang terakhir adalah pengujian parameter kombinasi nilai *cr* dan *mr* agar keberagaman anak (individu/solusi baru) hasil proses reproduksi semakin terjaga. Gambar 9 merupakan grafik garis rata-rata *fitness* hasil pengujian nilai *cr* dan *mr* dengan ukuran populasi dan banyaknya generasi hasil pengujian sebelumnya.



Gambar 9. Pengujian kombinasi nilai *cr* dan *mr*

Hasil pengujian pada Gambar 9 menunjukkan bahwa nilai *cr* 0.8 dan *mr* 0.2 merupakan kombinasi yang tepat untuk permasalahan distribusi multi-level multi-produk. Pada kombinasi tersebut solusi yang dihasilkan menunjukkan hasil yang paling baik dengan keberagaman individu yang lebih banyak dilakukan dari proses mutasi.

**5.2. Pengujian Operator Genetika**

Pengujian operator genetika merupakan pengujian untuk menentukan model operator genetika mana yang sesuai dan menghasilkan solusi yang mendekati optimal. Ada dua macam pengujian yang dilakukan berdasarkan operator genetika yaitu pengujian dari model *crossover* dan pengujian model *mutation*.

**5.2.1. Model Crossover**

Pengujian operator genetika pertama adalah pengujian model *crossover*. Tujuannya adalah mencari model *crossover* yang sesuai. Model *one-cut-point*, *two-cut-point*, dan *extended intermediate* adalah *crossover* yang diujikan pada penelitian ini. Tabel 1 adalah proses pengujian yang dilakukan dengan menggunakan parameter algoritma genetika yang terpilih dari hasil pengujian sebelumnya yaitu ukuran populasi 80, banyaknya generasi 800, *cr* 0.3 dan *mr* 0.7.

Berdasarkan pengujian model *crossover* pada Tabel 1 dapat disimpulkan bahwa model *crossover* yang paling baik adalah *two-cut-point*. Hal ini dibuktikan dengan average *fitness* terbesar sehingga

rata-rata biaya (Rupiah) yang dihasilkan paling minimal.

5.2.2. Model Mutasi

Pengujian operator genetika selanjutnya adalah pengujian model mutasi. Tujuannya adalah mencari model mutasi yang cocok dengan permasalahan distribusi sehingga solusi yang ditawarkan mendekati optimal. Hasil pengujian dalam memilih model mutasi yang sesuai ditunjukkan pada **Error! Not a valid bookmark self-reference.** Berdasarkan rata-rata *fitness* dan rata-rata biaya yang dihasilkan, *swap mutation* memberikan solusi yang lebih baik dibandingkan *insertionmutation*.

Tabel 2 merupakan rincian hasil rata-rata *fitness* dengan biaya dari masing-masing model yang dibandingkan. Model mutasi yang diuji dalam penelitian ini adalah *swapmutation* dan *insertionmutation*.

Tabel 1. Model Crossover

no.	model crossover	rata-rata <i>fitness</i>	rata-rata biaya
1	<i>Crossoverone-cut-point</i>	2.54088E-8	39583805
2	<i>Crossovertwo-cut-point</i>	3.07092E-8	32679440
3	<i>Crossoverextended intermediate</i>	2.49052E-8	40537165

Hasil pengujian dalam memilih model mutasi yang sesuai ditunjukkan pada **Error! Not a valid bookmark self-reference.** Berdasarkan rata-rata *fitness* dan rata-rata biaya yang dihasilkan, *swap mutation* memberikan solusi yang lebih baik dibandingkan *insertionmutation*.

Tabel 2. Model Mutasi

no.	model mutasi	rata-rata <i>fitness</i>	rata-rata biaya
1	Mutasi <i>Swap</i>	3.18331E-8	31484625
2	Mutasi <i>Insertion</i>	2.83421E-8	35348305
3	Mutasi <i>Simple random</i>	2.36457E-8	42419490

5.2.3. Model Seleksi

Pengujian operator genetika selanjutnya adalah pengujian model seleksi. Tujuannya adalah mencari model seleksi yang cocok dengan permasalahan distribusi sehingga solusi yang ditawarkan mendekati optimal. Tabel 3 merupakan rincian hasil rata-rata *fitness* dengan biaya dari masing-masing model yang dibandingkan. Hasil model seleksi yang terbaik dibandingkan model lainnya ditunjukkan pada Tabel 3

Hasil pengujian dalam memilih model seleksi yang sesuai ditunjukkan pada Tabel 3. Berdasarkan rata-rata *fitness* dan rata-rata biaya yang dihasilkan, *swapmutation* memberikan solusi yang lebih baik dibandingkan model seleksi yang lain.

Tabel 3. Model Seleksi

no.	model seleksi	rata-rata <i>fitness</i>	rata-rata biaya
1	Seleksi <i>Elitism</i>	3.13905E-8	31994380
2	Seleksi <i>Roulette-wheel</i>	1.15668E-8	86463140
3	Seleksi <i>Binary Tournament</i>	2.81680E-8	94982361

Proses selanjutnya adalah analisa seberapa baik algoritma yang diusulkan. Untuk mendapatkan hasil yang optimal, algoritma yang berbasis genetika bisa menggunakan parameter dan operator genetika hasil pengujian sebelumnya dan kemudian dibandingkan keunggulan solusinya.

Dengan menggunakan parameter dan operator genetika terbaik hasil pengujian sebelumnya maka dilakukan perbandingan hasil antara *random search* (RS), algoritma genetika klasik (GA), dan hibridisasi GA menggunakan *variable neighborhood search* (GA-VNS).

Tabel 4. Hasil Keseluruhan Algoritma

no.	algoritma	rata-rata <i>fitness</i>	rata-rata biaya	rata-rata waktu (ms)
1	RS	2.05077E-8	48784775	1809
2	GA	3.06647E-8	32716150	265091
3	GA-VNS	3.09724E-8	32392960	279332

Berdasarkan Tabel 7, rata-rata waktu dalam *milisecond*(ms) yang ditunjukkan oleh masing-masing algoritma berbeda-beda dan dapat dilihat bahwa waktu yang dihasilkan oleh GA dan GA-VNS relatif sama dan algoritma GA-VNS memberikan hasil yang lebih baik. Meskipun algoritma RS memiliki waktu komputasi yang cepat, hasil yang diperoleh masih jauh selisihnya dibandingkan algoritma lainnya.

6. KESIMPULAN

Permasalahan distribusi multi-level dan multi-produk pada penelitian ini diselesaikan dengan membentuk model matematis dari beberapa variabel yang tujuannya adalah mendapatkan biaya yang minimal. Variabel-variabel tersebut seperti jumlah produk, biaya variabel, kendaraan yang digunakan, biaya tetap, dan status pengiriman. Variabel-variabel yang tersusun secara matematis juga menyelesaikan beberapa *constraint* yang terjadi dalam proses distribusi seperti jumlah produk yang didistribusikan harus sama dengan jumlah order dan tidak melebihi kapasitas maksimal kendaraan maupun jumlah stok yang dimiliki.

Berdasarkan model matematis yang telah tersusun, maka suatu solusi/kromosom permasalahan distribusi dalam penelitian ini terdiri dari serangkaian kolom/gen. Jumlah gen didapatkan dari jumlah gen setiap level yang memesan dimana jumlah gen setiap level diperoleh dari jumlah jenis produk yang dipesan dikalikan dengan kendaraan

yang dimiliki distributor pengirim. Setiap gen dari suatu kromosom berisi bilangan riil yang menunjukkan jumlah produk yang akan didistribusikan. Untuk mengetahui jumlah produk dan jaringan distribusi mana yang terbaik maka diperhitungkan berdasarkan nilai *fitness* yang dihasilkan yang menunjukkan minimalnya biaya yang dikeluarkan dari keseluruhan proses distribusi.

Parameter-parameter optimal algoritma genetika memberikan pengaruh yang besar dalam memberikan solusi terbaik dalam menyelesaikan masalah distribusi. Parameter-parameter tersebut adalah populasi berukuran 60, banyaknya generasi 1100, nilai *cr* 0.8 dan *mr* 0.2. Model *two-cut-point* pada operator *crossover*, model *swapp* pada operator mutasi, dan model seleksi *elitism* menjadi model terbaik dalam memberikan hasil solusi yang lebih baik dalam menyelesaikan masalah distribusi pada penelitian ini.

Berdasarkan parameter dan model optimal tersebut, untuk menyelesaikan masalah distribusi, algoritma GA-VNS mampu memberikan solusi dengan hasil yang mendekati optimal dengan waktu komputasi yang relatif sama dibandingkan dengan algoritma GA klasik yaitu 279332 ms dan 265091 ms. Hasil yang diperoleh dari algoritma GA-VNS juga lebih baik yaitu 32392960 yang dibuktikan dengan penghematan biaya sebesar 323190 jika dibandingkan dengan GA klasik yaitu 32716150. Dibandingkan dengan *random search* (RS), waktu yang dihabiskan oleh algoritma GA-VNS memang lebih lama namun algoritma yang diusulkan tersebut memberikan biaya yang paling minimal dengan penghematan sebesar 16391815.

## 7. REFERENSI

- ARORA, J.S. & ARORA, J.S., 2012. *Chapter 16 – Genetic Algorithms for Optimum Design*, Elsevier Inc.
- BOUDJELABA, K., ROS, F. & CHIKOUCHE, D., 2014. An efficient hybrid genetic algorithm to design finite impulse response filters. *Expert Systems With Applications*, 41(13), pp.5917–5937. Available at: <http://dx.doi.org/10.1016/j.eswa.2014.03.034>.
- CASTELLI, M. & VANNESCHI, L., 2014. Genetic algorithm with variable neighborhood search for the optimal allocation of goods in shop shelves. *Operations Research Letters*, 42(5), pp.355–360. Available at: <http://dx.doi.org/10.1016/j.orl.2014.06.002>.
- CHEIKH, M. DKK., 2015. A variable neighborhood search algorithm for the vehicle routing problem with multiple trips. *Electronic Notes In Discrete Mathematics*, 47, pp.277–284. Available at: <http://dx.doi.org/10.1016/j.endm.2014.11.036>.
- CHEN, Z.-Q. & WANG, R.-L., 2011. Solving the m-way graph partitioning problem using a genetic algorithm. *Ieej Transactions On Electrical And Electronic Engineering*, 6(5), pp.483–489. Available at: <http://doi.wiley.com/10.1002/tee.20685>.
- GICQUEL, C. & MINOUX, M., 2015. Multi-product valid inequalities for the discrete lot-sizing and scheduling problem. *Computers And Operation Research*, 54, pp.12–20. Available at: <http://dx.doi.org/10.1016/j.cor.2014.08.022>.
- GUO, H., WANG, X. & ZHOU, S., 2015. A Transportation Problem with Uncertain Costs and Random Supplies. *International Journal Of E-Navigation And Maritime Economy*, 2, pp.1–11. Available at: <http://www.sciencedirect.com/science/article/pii/S2405535215000558>.
- GUPTA, A. DKK., 2012. A Comparative Study of Three Echelon Inventory Optimization using Genetic Algorithm and Particle Swarm Optimization. *International Journal Of Trade, Economics And Finance*, 3(3), pp.205–208.
- JIAO, R. DKK., 2014. A Multistage Multiobjective Substation Siting and Sizing Model Based on Operator-Repair Genetic Algorithm. *Ieej Transactions On Electrical And Electronic Engineering*, 9, pp.S28–S36.
- KIM, S.H. & LEE, Y.H., 2016. Synchronized production planning and scheduling in semiconductor fabrication. *Computers & Industrial Engineering*, 96, pp.72–85. Available at: <http://dx.doi.org/10.1016/j.cie.2016.03.019>.
- LANGROODI, R.R.P. & AMIRI, M., 2016. A system dynamics modeling approach for a multi-level, multi-product, multi-region supply chain under demand uncertainty. *Expert Systems With Applications*, 51, pp.231–244. Available at: <http://dx.doi.org/10.1016/j.eswa.2015.12.043>.
- MAHMUDY, W.F., 2015. Optimization of Part Type Selection and Machine Loading Problems in Flexible Manufacturing System Using Variable Neighborhood Search. *Iaeng International Journal Of Computer Science*, 42(3).
- MAHMUDY, W.F., MARIAN, R.M. & LUONG, L.H.S., 2014. Hybrid Genetic Algorithms for Part Type Selection and Machine Loading Problems with Alternative Production Plans in Flexible Manufacturing System. *Ecti Transaction On Computer And Information Technology*, 8(1), pp.80–93.
- MAHMUDY, W.F., MARIAN, R.M. & LUONG, L.H.S., 2013. Modeling and Optimization of Part Type Selection and Loading Problem in Flexible Manufacturing System Using Real Coded Genetic Algorithms. *International*

*Journal Of Electrical, Computer, Electronics And Communication Engineering*, 7(4), pp.251–260.

- MLADENOVIC, N., UROSEVIC, D. & PEREZ-BRIT, D., 2016. Variable Neighborhood Search for Minimum Linear Arrangement Problem. *Yugoslav Journal Of Operations Research*, 26(1), pp.3–16.
- QIONGBING, Z., 2016. A New Crossover Mechanism for Genetic Algorithms with Variable-length Chromosomes for Path Optimization Problems. *Expert Systems With Applications*, 60, pp.183–189. Available at: <http://dx.doi.org/10.1016/j.eswa.2016.04.005>.
- RAHMI, A., SARWANI, M.Z. & MAHMUDY, W.F., 2017. Genetic Algorithms for Optimization of Multi-Level Product Distribution. *Accepted to International Journal Of Intelligent Engineering & Systems*.
- SARWANI, M.Z., RAHMI, A. & MAHMUDY, W.F., 2017. An Adaptive Genetic Algorithm for Cost Optimization of Multi-Stage Supply Chain. *Accepted to Journal Of Telecommunication, Electronic And Computer Engineering*.
- SITEK, P. & WIKAREK, J., 2012. Mathematical programming model of cost optimization for supply chain from perspective of logistics provider. *Management And Production Engineering Review*, 3(2), pp.49–61.
- SONI, N. & KUMAR, T., 2014. Study of Various Mutation Operators in Genetic Algorithms. *International Journal Of Computer Science And Information Technology*, 5(3), pp.4519–4521.
- THAKUR, M. & KUMAR, A., 2016. Electrical Power and Energy Systems Optimal coordination of directional over current relays using a modified real coded genetic algorithm : A comparative study. *International Journal Of Electrical Power And Energy Systems*, 82, pp.484–495. Available at: <http://dx.doi.org/10.1016/j.ijepes.2016.03.036>.
- ZIEBUHR, M. & KOPFER, H., 2016. Solving an integrated operational transportation planning problem with forwarding limitations. *Transportation Research Part E: Logistics And Transportation Review*, 87, pp.149–166. Available at: <http://dx.doi.org/10.1016/j.tre.2016.01.006>.

## IMPLEMENTASI METODE K-NEAREST NEIGHBOUR DENGAN PEMBOBOTAN TF.IDF.ICF UNTUK KATEGORISASI IDE KREATIF PADA PERUSAHAAN

Romario Yudo Herlambang<sup>1</sup>, Rekyan Regasari Mardi Putri<sup>2</sup>, Randy Cahya Wihandika<sup>3</sup>

<sup>123</sup>Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>romarioyudo@gmail.com, <sup>2</sup>rekyan.rmp@ub.ac.id, <sup>3</sup>rendicahya@ub.ac.id

(Naskah masuk: 4 Februari 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Ide kreatif/inovasi merupakan hal yang dibutuhkan perusahaan dalam pengembangan sebuah individu, kelompok ataupun perusahaan pada teknologi seperti pada masa ini. Pengembangan ide kreatif berpengaruh pada peningkatan kinerja perusahaan. Pada kebanyakan kasus, pengelompokan ide tersebut harus dikelompokkan dengan kecocokan tema yang diusung untuk mempermudah proses pencarian. Oleh sebab itu dibutuhkan suatu sistem yang mampu bekerja secara otomatis untuk mengelompokkan ide tersebut. Kemungkinan salah satu teknik pembobotan yang digunakan adalah dengan menggunakan *TF.IDF.ICF*, yang telah mengalami pengembangan dari metode sebelumnya. *TF.IDF.ICF* tidak dapat digunakan sendiri melainkan harus ada metode perhitungan jarak seperti *Cosine Similarity* dan metode klasifikasi lain seperti *KNN* dapat dipakai ke semua atribut. Aplikasi ini nantinya akan diterapkan pada perusahaan PJB Paiton sebagai studi kasus dan ide kreatif yang dikategorikan, dituliskan dalam Bahasa Indonesia. Aplikasi ini akan melakukan beberapa tahap pemrosesan seperti *tokenizing* yaitu pemisahan kalimat menjadi tiap kata, *filtering* yang merupakan penghapusan *stopwords*, *stemming*, *cosine similarity* dan *KNN* yang masing-masing metode digunakan untuk perhitungan jarak dan proses perhitungan klasifikasi. Dari hasil pengujian yang telah dilakukan, sistem mampu menghasilkan akurasi terbaik sebesar 93% menggunakan dengan nilai *k* sebesar 1 menggunakan presentase data uji sebanyak 50 akan menghasilkan klasifikasi ideal.

**Kata kunci:** *ide, kelas, cosine, KNN.*

### Abstract

*Creative ide is one thing that needed by the company for group development or even the company itself. The development of creative ideas has a big influence on improving corporate performance. On most cases, the clasification of the idea must be grouped based on the similarity of the theme that submitted to simplify the searching process. Therefore we need a system that could work automatically to classify the idea. Probably, one weighting techniques that used is TF.IDF.ICF that already been developed from the method before. TF.IDF.ICF cant be used alone. there must be another method that used before, such as cosine similarity for distance calculation method and KNN for classification method in order TF.IDF.ICF can be used by all atributes. This application will be focused on the PJB company's creative idea and these ideas will be in indonesian language. This application will do a few processing steps such as, tokenizing for breaking sentence into words, filtering which is elimination of stopwords, stemming, cosine similarity, and KNN. each method used for distance calculation and classification calculation process. From the testing result that has been done, the system could produce the best accuracy as big as 93% by using the value of K as big as 1 using the precentage of test data as big 50 produce the ideal classification.*

**Keywords:** *idea, class, cosine, KNN*

## 1. PENDAHULUAN

PT Pembangkitan Jawa Bali (PJB) adalah anak perusahaan PT PLN (persero) yang didirikan pada tanggal 3 Oktober 1995, sedangkan Pembangkit Listrik Tenaga Uap (PLTU) Unit Pembangkitan Paiton ini berdiri pada 15 Maret 1993. Pada tanggal 3 Oktober 1995 terdapat restrukturisasi pada PT PLN (persero) dan melahirkan dua anak perusahaan yakni PT PLN Pembangkitan Tenaga Listrik Jawa Bali I dan II yang disebut PJB I dan II. Karena PT PJB anak perusahaan PT PLN maka PT PJB harus

mengikuti aturan dari PT PLN yang salah satunya adalah kontes ide kreatif.

Kontes ide kreatif yang dikonteskan pada PT PLN Tiap Tahun. Maka dari itu PT PJB harus mengirimkan 5 ide kreatif pertahun yang akan ditandingkan, untuk hal itu dalam penerimaan berbagai ide dan solusi kreatif terkait berbagai permasalahan maupun peningkatan efisiensi kerja harus diklasifikasikan sesuai kategorinya. Kategori yang biasa ditandingkan untuk menemukan solusi-solusi masalah adalah T untuk Teknik dan NT untuk Non Teknik. Teknik adalah kategori yang membahas seputar mesin, alat berat, dll, sedangkan untuk Non Teknik adalah kategori yang ruang lingkup

bahasannya umum, misal pengadaan jus buah pada pagi hari dan sore sebelum jam kerja habis. Masalah yang muncul adalah tingkat pengkategorian yang dimasukkan oleh peserta adalah terjadinya beberapa kesalahan dalam pengkategorian teks ide kreatif.

Permasalahan ini hampir sama dengan penelitian yang pernah dilakukan oleh Sriram dkk pada tahun 2010, yang melakukan klasifikasi teks dari *twitter* untuk meningkatkan penyaringan informasi. Sehingga, dalam mengatasi permasalahan kategori terhadap pengkategorian ide kreatif, peneliti mengusulkan proses klasifikasi dimana sistem dapat mengolah data teks menjadi kategori-kategori yang diinginkan. Salah satu metode dalam melakukan pemrosesan teks adalah *TF.IDF.ICF*.

*TF.IDF.ICF* salah satu metode pembobotan dalam mencari tingkat kemiripan teks. Metode ini terdiri dari 3 komponen, *TF* untuk menghitung jumlah kata pada suatu dokumen, *IDF* untuk jumlah frekuensi tiap kata pada dokumen dan *ICF* adalah jumlah frekuensi tiap kata berdasarkan kelas, lalu ketiganya akan dikalikan untuk membentuk *TF.IDF.ICF*. Berdasarkan konsep yang digunakan tersebut, dikembangkan pemodelan dokumen untuk melakukan pencarian terhadap dokumen yang dibutuhkan. Dokumen akan direpresentasikan sebagai vektor atau biasa disebut *Vector Space Model* dengan cara menghitung besarnya sudut yang terbentuk antara dua vektor dan kemudian diurutkan dari data yang memiliki besar sudut yang terkecil hingga yang terbesar yang menandakan urutan data hasil pengurutan dari yang paling relevan hingga yang tidak relevan yaitu dengan *Cosine Similarity*. Namun setelah data tersebut selesai dihitung dengan *Cosine Similarity* maka akan diklasifikasikan yang salah satu metode yang dapat diterapkan adalah metode *KNN*.

Berdasarkan latar belakang yang telah dijelaskan, maka pada penelitian ini akan mengimplementasikan *TF.IDF.ICF* pada pembobotan kata dan *KNN* pada klasifikasi untuk pengkategorian ide.

## 2. LANDASAN TEORI

### 2.1. Text Mining

Klasifikasi ide kreatif menjadi beberapa kategori adalah satu contoh aplikasi dari *text mining*. Beberapa contoh lain aplikasi dari *text mining* adalah *text summarization*, *text categorization*, *document clustering*, *language identification*, *ascribing authorship* (Mustafa. dkk. 2009). *Text mining* atau dengan sebutan lain seperti *intelligent text analysis*, *text data mining*, atau *knowledge discovery in text* secara sederhana dapat diartikan sebagai proses penemuan pola yang sebelumnya tidak terlihat pada dokumen teks atau sumber tertentu (Manning. dkk. 2009). Dalam pemrosesan teks/*text mining* biasanya awal dilakukannya dengan melakukan *Preprocessing*, beberapa diantaranya yaitu dengan

*Tokenizing* atau pemisahan suatu dokumen/kalimat menjadi tiap kata/term, *Filtering* yaitu dengan penghapusan kata penghubung, kata sambung, dll dari hasil *Tokenizing* tiap dokumen, dan *Stemming* atau pemisahan imbuhan pada kata menjadi kata dasar. Lalu setelah proses diatas terjadi maka harus ada pembobotan yang dilakukan dengan metode *TF.IDF.ICF* yang lalu dihitung jaraknya dengan *Cosine Similarity* dan pengklasifikasian salah satunya adalah *KNN*.

### 2.2. Preprocessing

*Preprocessing* adalah proses awal yang dilakukan agar metode pembobotan kata dapat diterapkan, beberapa metode antara lain: *tokenizing*, *filtering* dan *stemming*.

#### 2.2.1 Tokenizing

Proses pemecahan kalimat yang terdapat pada sebuah dokumen menjadi kata. Setiap kata yang didapat biasanya dalam *text mining* disebut dengan *term/token*. Metode ini adalah serangkaian metode dalam proses *preprocessing*.

#### 2.2.2 Filtering

*Filtering* atau biasa disebut *stop-word removal* adalah proses pada *preprocessing* yang berguna untuk menghilangkan kata sambung, kata penghubung atau kata umum lainnya. *Filtering* ini berguna untuk mengurangi perhitungan dari kata yang tidak seharusnya tidak mewakili kelas apapun.

#### 2.2.3. Metode Stemming Arifin dan Setiono

Algoritma *stemming* akan mengolah tiap kata yang dimasukkan untuk diproses dan menghasilkan kata dalam bentuk dasarnya. Langkah-langkah yang dilakukan dalam algoritma *stemming* bahasa Indonesia Arifin dan Setiono adalah sebagai berikut (Arifin. dkk. 2001) :

Pemeriksaan yang dilakukan pada keseluruhan probabilitas bentuk kata. Setiap kata diasumsikan memiliki 2 awalan atau prefika dan 3 akhiran atau sufiks (Arifin. dkk. 2001). Sehingga memiliki bentuk sebagai berikut:

Prefiks 1 + Prefiks 2 + Kata dasar + Sufiks 3 + Sufiks 2 + Sufiks 1

Jika terdapat perbedaan rangkaian kata atau kekosongan pada susunan kata seperti diatas maka cukup ditambahkan x untuk prefix dan xx untuk sufiks pada bagian yang kosong (Arifin. dkk. 2001).

Pemotongan dilakukan dengan catatan AW mewakili awalan, AK adalah akhiran, dan KD untuk kata dasar secara berurutan sebagai berikut :

- AW I, hasilnya tersimpan pada p1 (prefiks 1)
- AW II, hasilnya tersimpan pada p2 (prefiks 2)

- c. AK I, hasilnya tersimpan pada s1 (sufiks 1)
- d. AK II, hasilnya tersimpan pada s2 (sufiks 2)
- e. AK III, hasilnya tersimpan pada s3 (sufiks 3)

Pada setiap tahap pemotongan di atas diikuti pemeriksaan pada kamus apakah hasil pemotongan sudah berada dalam bentuk dasar. Jika pemeriksaan berhasil dilakukan maka proses dinyatakan selesai sehingga tidak perlu melanjutkan proses pemotongan imbuhan lainnya (Arifin. dkk. 2001). Contoh pemenggalan kata "membuatkannya" . Berikut Langkahnya :

- a. Pengecekan kata dalam kamus  
Ya : Success  
Tidak : lakukan pemotongan AW I  
Kata = buatkannya
- b. Pengecekan kata dalam kamus  
Ya : Success  
Tidak : lakukan pemotongan AW II  
Kata = buatkannya
- c. Pengecekan kata dalam kamus  
Ya : Success  
Tidak : lakukan pemotongan AK I  
Kata = buat
- d. Pengecekan kata dalam kamus  
Ya : Success  
Tidak : lakukan pemotongan AK II  
Kata = buat
- e. Pengecekan kata dalam kamus  
Ya : Success  
Tidak : lakukan pemotongan AK III. Dalam hal ini AK III tidak ada, sehingga kata tidak diubah.  
Kata = main
- f. Pengecekan kata dalam kamus  
Ya : Success  
Tidak : "Kata tidak ditemukan"

Jika sampai pada pemotongan AK III belum juga ditemukan kata yang sama dalam kamus, maka perlu dilakukan kombinasi (Arifin. dkk. 2001). KD yang dihasilkan dikombinasikan dengan imbuhan-imbuhan dalam 12 konfigurasi seperti dibawah ini:

- a. KD
- b. KD + AK III
- c. KD + AK III + AK II
- d. KD + AK III + AK II + AK I
- e. AW I + AW II + KD
- f. AW I + AW II + KD + AK III
- g. AW I + AW II + KD + AK III + AK II
- h. AW I + AW II + KD + AK III + AK II + AK I
- i. AW II + KD
- j. AW II + KD + AK III
- k. AW II + KD + AK III + AK II
- l. AW II + KD + AK III + AK II + AK I

Sebenarnya kombinasi a, b, c, d, h, dan l sudah diperiksa pada tahap sebelumnya, karena kombinasi ini merupakan hasil pemotongan bertahap (Arifin. dkk. 2001). Dengan demikian, kombinasi yang masih perlu dilakukan yaitu (e, f, g, i, j, dan k).

Pemeriksaan 12 kombinasi dibutuhkan, sebab adanya fenomena *overstemming* pada algoritma pemotongan imbuhan. Kelemahannya berdampak pada pemotongan bagian kata yang sebenarnya adalah milik kata dasar itu sendiri dan kebetulan mirip dengan salah satu jenis imbuhan yang sudah ada. Dengan 12 kombinasi itu, pemotongan yang sudah terlanjur tersebut dapat dikembalikan sesuai strukturnya (Arifin. dkk. 2001).

### 2.3. TF.IDF.ICF

Nilai *Term Frekuensi (TF)* didapat dari jumlah setiap term dalam setiap dokumen yang ada.

*Inverse Document Frequency (IDF)* adalah proses yang dilakukan saat pembobotan kata. Bertujuan untuk mencari bobot pada setiap kata. *IDF* didefinisikan sebagai

$$Idf_i = \log \left( \frac{n_d}{df_i} \right) \quad (1)$$

Keterangan :

$Idf_i$  = Inverse document frequency untuk tiap term/kata.

$\log$  = Operasi matematika yang merupakan kebalikan dari operasi pangkat.

$n_d$  = Jumlah dokumen yang ada pada data training.

$df_i$  = Jumlah dokumen yang memiliki kata tersebut.

*Inverse Class Frequency (ICF)* adalah proses yang dilakukan saat pembobotan kata. Bertujuan untuk mencari bobot pada setiap kata. *ICF* didefinisikan sebagai

$$Icf_i = \log \left( \frac{n_c}{cf_i} \right) \quad (2)$$

Keterangan :

$Icf_i$  = Inverse class frequency untuk tiap term/kata.

$\log$  = Operasi matematika yang merupakan kebalikan dari operasi pangkat.

$n_c$  = Jumlah kelas yang ada pada data training.

$cf_i$  = Jumlah kelas yang memiliki kata tersebut.

dimana  $cf_i$  merupakan frekuensi dokumen dari term  $i$  atau sama dengan jumlah dokumen yang mengandung term  $i$  dan  $n$  adalah total dokumen di dalam database. *Log* digunakan untuk memperkecil pengaruh relative untuk  $tf_{ij}$ .

Bobot  $w_{ij}$  dihitung menggunakan ukuran *TF.IDF.ICF* (*term frequency-inversed document frequency-inversed class frequency*) didefinisikan sebagai

$$w_{ij} = tf_{ij} \times idf_i \times icf_i \quad (3)$$

Keterangan :

- $w_{ij}$  = Hasil perkalian dari  $tf_{ij}$ ,  $idf_i$  dan  $icf_i$ .
- $tf_{ij}$  = Frekuensi kemunculan kata/term pada dokumen.
- $idf_i$  = *Inverse document frequency* untuk tiap *term*/kata.
- $icf_i$  = *Inverse class frequency* untuk tiap *term*/kata.

**2.4 VSM**

*Vector Space Model* (VSM) mempresentasikan setiap dokumen yang terdapat pada *TF.IDF.ICF* ke dalam vektor multidimensi. Dimensi dari vektor sesuai dengan jumlah setiap dokumen dan jumlah jarak antara dokumen dengan tiap *query* tersebut untuk dibentuk pada suatu ruang vektor.

**2.5 Cosine Similarity**

Salah satu ukuran kemiripan teks yang populer digunakan pada VSM untuk pencarian dokumen adalah *cosine similarity* (Krzysztof J. Cios. Dkk. 2007).

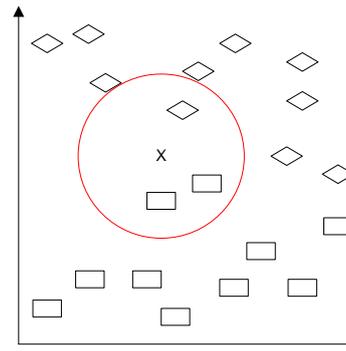
Konsep dari *cosine similarity* yaitu menghitung nilai cosinus sudut antara dua vektor yaitu jika diberikan dokumen yang diwakili oleh vektor  $d_j$  dan query  $q$ , dan term  $t$  yang diekstrak dari database, maka nilai *cosine similarity* didefinisikan sebagai

$$\cos(\vec{d}_j, \vec{d}_k) = \vec{d}_j \cdot \vec{d}_k \tag{4}$$

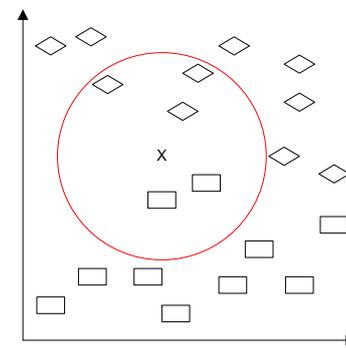
Sudut yang merentang antara vektor  $d_j$  dan query  $q$  akan menghasilkan sudut yang jika semakin kecil sudut diantara kedua vektor  $d_j$  dan query  $q$ , maka akan semakin tinggi derajat kesamaan. Cosinus dari sudut tersebut merupakan koefisien yang dapat mewakili kemiripan antara vektor  $d_j$  dan query  $q$ .

**2.6 KNN**

Algoritma *K-Nearest Neighbour* (KNN) adalah sebuah metode klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Data pembelajaran dideskripsikan dengan atribut *numeric* n-dimensi. Setiap data pembelajaran merepresentasikan sebuah titik, dalam ruang n-dimensi. Jika sebuah data query yang labelnya tidak diketahui dimasukkan, maka *K-Nearest Neighbour*nya akan mencari k buah data pembelajaran yang jaraknya paling dekat dengan data *query* dalam ruang n-dimensi. Jarak antar data *query* dengan data pembelajaran dihitung dengan cara mengukur jarak antara titik yang merepresentasikan data *query* dengan semua titik yang merepresentasikan data pembelajaran. Pada kasus ini menggunakan rumus *Cosine Similarity* sebagai rumus perhitungan jarak.



Gambar 1. Cara kerja metode KNN dengan k adalah 3



Gambar 2. Cara kerja metode KNN dengan k adalah 5

Seperti pada Gambar 1 dan Gambar 2 X adalah data latih yang sebagai pusat untuk mencari kemiripan sedangkan □ dan ◇ adalah data yang dicari kemiripan dengan data X, pada Gambar 1 dijelaskan bahwa jika menggunakan k = 3, maka kemiripan dari data X adalah yang ada pada lingkaran merah, hampir sama dengan Gambar 1, pada Gambar 2. didalam lingkaran adalah data yang memiliki kemiripan dengan data X namun dengan k = 5.

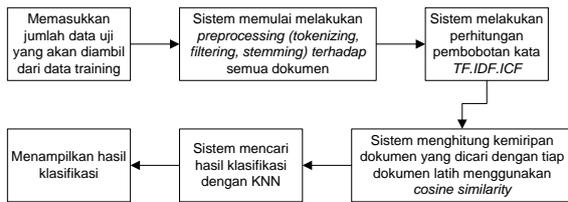
**2.6 Standar Deviasi**

Standar deviasi yaitu perbedaan nilai sampel terhadap rata-rata. Nilai sampel yakni sedikit dari jumlah keseluruhan objek yang diamati. Didefinisikan sebagai

$$\sqrt{\frac{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}{n(n-1)}} \tag{2.5}$$

**3. PERANCANGAN**

Penelitian ini merupakan salah satu bentuk penelitian implementatif, dimana penelitian ini akan menghasilkan program yang akan membantu mengklasifikasikan ide yang mirip.



Gambar 3. Diagram Blok Program

#### 4. HASIL UJI COBA DAN ANALISA

Pada bab ini membahas tentang tahapan pengujian dan analisis dari hasil implementasi algoritma *TF.IDF.ICF* untuk pengklasifikasian ide kreatif menjadi kelas Teknik dan Non Teknik.

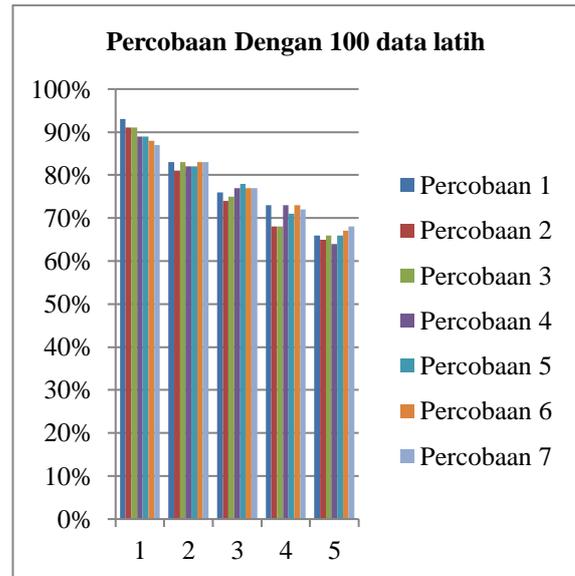
##### 4.1 Hasil dan Analisis Pengaruh Variasi K Dan Jumlah Data Latih

Pada uji coba ini, dilakukan pengujian terhadap pengaruh data latih terhadap akurasi. Data yang diuji adalah data ide kreatif PT PJB UP Paiton tahun 2014 yang 50 data menjadi data uji yang dipilih secara acak pada awal percobaan sehingga data uji yang digunakan selalu dan sisa dari data uji akan menjadi data latih yang kesemua data latih. Setiap percobaan dengan jumlah data latih yang sama akan diambil jarak 10 data latih dengan percobaan sebelumnya atau melakukan irisan terhadap percobaan sebelumnya. Percobaan dilakukan sebanyak 27 kali dengan catatan untuk setiap percobaan menggunakan jumlah k sebanyak 1-5. Jumlah data latih berbeda, mulai dari 100 data latih dengan 7 kali percobaan, 80 data latih dengan 9 kali percobaan dan 60 data latih dengan 11 kali percobaan. Seperti dijelaskan pada Tabel 1.

Tabel 1. Tabel percobaan

Percobaan ke-	Jumlah data latih	Akurasi dalam persentase				
		Jumlah k				
		1	2	3	4	5
1	100	93	83	76	73	66
2		91	81	74	68	65
3		91	83	75	68	66
4		89	82	77	73	64
5		89	82	78	71	66
6		88	83	77	73	67
7		87	83	77	72	68

Uji pengaruh data latih yang pertama dilakukan dengan menggunakan 100 data latih dengan 7 percobaan awal dengan menggeser id sebanyak 10 id tiap kali percobaan pada Tabel 1 yang hasilnya bisa diilustrasikan dalam Gambar 4.



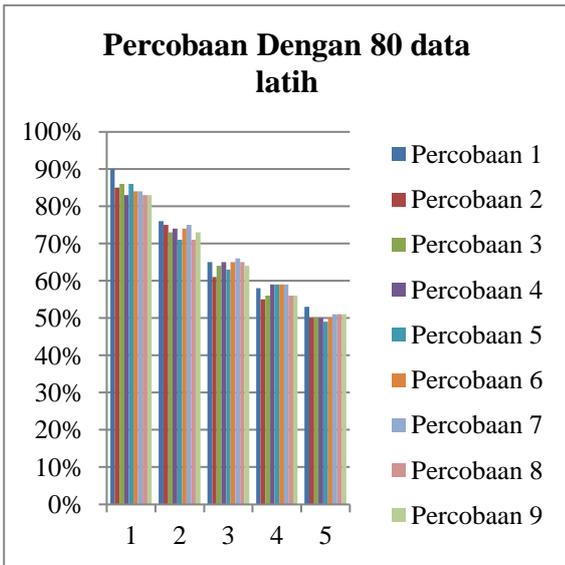
Gambar 4. Hasil pengujian dengan jumlah data latih sebanyak 100

Pada Gambar 4, percobaan dilakukan sebanyak 7 kali dengan menggunakan k mulai dari 1 – 5 pada setiap percobaan. Hasil uji coba pertama digambarkan dengan menggunakan garis Percobaan 1 dan hasil percobaan kedua digambarkan dengan menggunakan garis Percobaan 2 dan begitu seterusnya. Dengan semakin besar jumlah k maka akan semakin menurunkan tingkat akurasi. Dengan 100 data latih variasi data terlihat kecil dengan jarak akurasi (maks-min) terjauh adalah 6% dengan k = 1 dan k = 4. Pada k = 2 terjadi jarak akurasi(maks-min) terkecil yaitu 2%.

Tabel 2. Tabel percobaan

Percobaan ke-	Jumlah data latih	Akurasi dalam persentase				
		Jumlah k				
		1	2	3	4	5
1	80	90	76	65	58	53
2		85	75	61	55	50
3		86	73	64	56	50
4		83	74	65	59	50
5		86	71	63	59	49
6		84	74	65	59	50
7		84	75	66	59	51
8		83	71	65	56	51
9		83	73	64	56	51

Pada uji coba kedua dengan menggunakan 80 data latih dengan 9 percobaan mulai dari percobaan 1 - 9 pada Tabel 2 yang hasilnya bisa diilustrasikan dalam Gambar 5.



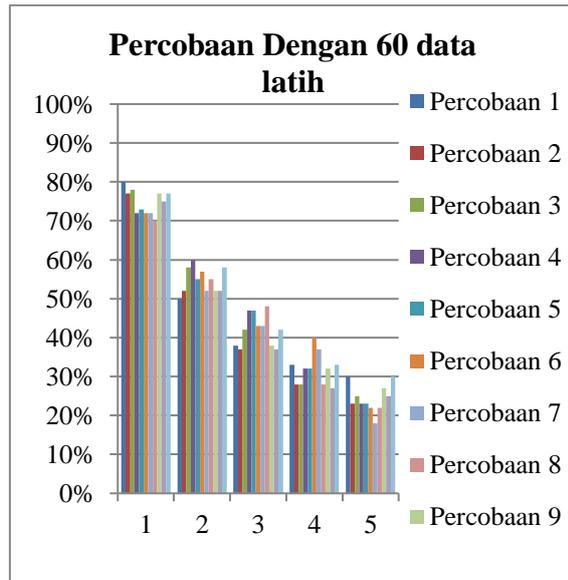
Gambar 5. Hasil pengujian dengan jumlah data latih sebanyak 80

Pada Gambar 5 uji coba dilakukan sebanyak 9 kali dengan menggunakan k mulai dari 1 – 5 pada setiap percobaan. Hasil uji coba kedua digambarkan dengan menggunakan garis Percobaan 1 dan hasil percobaan kedua digambarkan dengan menggunakan garis Percobaan 2 dan begitu seterusnya seperti pada percobaan dengan data latih sebanyak 80. Dengan semakin besar jumlah k maka akan semakin menurunkan tingkat akurasi. Dengan 80 data latih variasi data terlihat lumayan kecil dengan jarak akurasi (maks-min) terjauh adalah 7% dengan k = 1. Pada k = 4 dan 5 terjadi jarak akurasi(maks-min) terkecil yaitu 4%.

Tabel 3. Tabel percobaan

Percobaan ke-	Jumlah data latih	Akurasi dalam persentase				
		Jumlah k				
		1	2	3	4	5
1	60	80	50	38	33	30
2		77	52	37	28	23
3		78	58	42	28	25
4		72	60	47	32	23
5		73	55	47	32	23
6		72	57	43	40	22
7		72	52	43	37	18
8		70	55	48	28	22
9		77	52	38	32	27
10		75	52	37	27	25
11		77	58	42	33	30

Pada uji coba terakhir dengan menggunakan 60 data latih dengan 11 percobaan mulai dari percobaan 1 - 1 pada Tabel 6.3 yang hasilnya bisa diilustrasikan dalam Gambar 6.3.



Gambar 6. Hasil pengujian dengan jumlah data latih sebanyak 60

Pada Gambar 6 uji coba dilakukan sebanyak 11 kali dengan menggunakan k mulai dari 1 – 5 pada setiap percobaan. Hasil uji coba terakhir ini digambarkan dengan menggunakan garis Percobaan 1 dan hasil percobaan kedua digambarkan dengan menggunakan garis Percobaan 2 dan begitu seterusnya seperti pada percobaan dengan data latih sebanyak 60. Dengan semakin besar jumlah k maka akan semakin menurunkan tingkat akurasi. Dengan 60 data latih variasi data terlihat kecil dengan jarak akurasi(maks-min) terjauh adalah 13% dengan k = 4. Pada k = 1 dan 2 terjadi jarak akurasi(maks-min) kecil yaitu 10%.

Berdasarkan 3 pengujian pada Tabel 1, Tabel 2 dan Tabel 3, variabel yang memiliki pengaruh variasi data adalah jumlah data latih, semakin banyak data latih akan semakin kecil jarak variasi data. Mulai dari akurasi pada 60 data latih sampai 100 data latih selalu ada peningkatan akurasi, namun dengan keterbatasan data latih hanya yang dapat dijadikan data latih hanya 100 data. Jumlah k juga berpengaruh terhadap akurasi yaitu jumlah k berbanding terbalik dengan akurasi. Semakin banyak jumlah k, maka akurasi semakin kecil. Untuk itu penggunaan klasifikasi KNN pada kasus ini sebaiknya tidak digunakan, hanya perlu 1NN saja sudah dapat diambil kesimpulan pasti mengenai klasifikasinya.

### 6.2 Pengujian Standar Deviasi

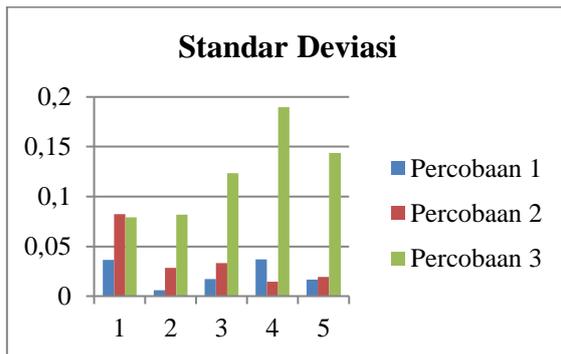
Pengujian standar deviasi dilakukan untuk mengetahui penyebaran data akurasi pada masing-masing k sehingga pada k yang memiliki akurasi terbaik.

Tabel 4. Tabel Standar Deviasi

Percobaan ke-	Jumlah data latih	Standar deviasi * 100				
		Jumlah k				
		1	2	3	4	5

1	100	3,6 5	0,6 2	1,72	3,69	1,68
2	80	8,2 1	2,8 5	3,30	1,48	1,95
3	60	7,9 3	8,1 9	12,3 7	18,9 5	14,3 9

Pada Tabel 4 adalah standar deviasi tiap data latih dengan jumlah data latih sama dari Tabel 1, Tabel 2 dan Tabel 3 berdasar persamaan 5 dengan patokan jumlah k.



Gambar 7. Hasil pengujian standar deviasi

Hasilnya penyebaran data akurasi pada masing-masing k sehingga pada k yang memiliki akurasi terbaik yaitu k = 1, dinyatakan valid karena dengan variasi skenario uji coba, akurasi yang dihasilkan tidak memiliki persebaran data yang tinggi.

Tabel 5. Tabel Standar Deviasi

Perco baan ke-	Jum lah data lati h	Standar deviasi dalam persentase				
		Jumlah k				
		1	2	3	4	5
1	100	2,08 7917	3,17 7666	4,69 0372	7,76 6456	5,92 898

Pada Tabel 5 didapatkan standar deviasi dari semua percobaan dengan berpatokan pada jumlah k. Disimpulkan bahwa dengan jumlah k = 1 akan menghasilkan standar deviasi terendah, dan k = 4 adalah standar deviasi tertinggi. Hasil ini memperkuat bahwa dengan 1NN atau KNN dengan k = 1 sudah cukup untuk mendapat hasil maksimal.

## 5. KESIMPULAN

Berdasarkan hasil uji coba parameter *TF.IDF.ICF* pada permasalahan pengelompokan ide kreatif adalah sebagai berikut :

1. Algoritma *TF.IDF.ICF* dapat menyelesaikan permasalahan pada pengelompokan ide kreatif pada PT PJB UP Paiton, hasil akurasi yang didapat memiliki rata-rata kemiripan yang paling tinggi adalah 90% dengan k = 1.

2. Hasil parameter terbaik pada hasil pengujian ini adalah sebagai berikut.

- Jumlah k : 1
- Jumlah data latih : 100
- Percobaan : 1
- Jumlah data uji : 50
- Akurasi rata-rata : 90%

3. Pada bab pengujian dijelaskan dengan k = 1 sudah bisa didapat standar deviasi terkecil, jika jumlah k bertambah standar deviasi akan semakin tinggi. Standar deviasi pada variasi data dengan ketentuan sebagai berikut.

- Jumlah k : 1
- Jumlah data uji : 50
- Standar deviasi : 2,087917

## 6. DAFTAR PUSTAKA

ARIFIN, A. Z. & NOVAN, S. A. 2001. Klasifikasi Dokumen Berita Kejadian Berbahasa Indonesia dengan Algoritma *Single Pass Clustering*. ITS, Surabaya

CIOS, K. J., PEDRYCZ, W., SWINIARSKI, R.W. & KURGAN, L.. 2007. *Data Mining A Knowledge Discovery Approach*. Springer.

MANNING, C., RAGHAVAN, P. & SCHÜTZE, H. 2009. *An Introduction to Information Retrieval*. Cambridge University Press.

MUSTAFA, A., AKBAR, A. & SULTAN, A. 2009. *Knowledge Discovery using Text Mining: A Programmable Implementation on Information Extraction and Categorization*. *International Journal of Multimedia and Ubiquitous Engineering* Vol. 4, No. 2, April, 2009.

SRIRAM, B., FUHRY, D., DEMIR, E., FERHATOSMANOGLU, H. & DEMIRBAS, M. 2010. *Short Text Classification in Twitter to Improve Information Filtering*. *International ACM SIGIR Conference on Research and Development in Information Retrieval*.

## HYBRID K-MEANS DAN PARTICLE SWARM OPTIMIZATION UNTUK CLUSTERING NASABAH KREDIT

Yusuf Priyo Anggodo<sup>1</sup>, Winda Cahyaningrum<sup>2</sup>, Aprilia Nur Fauziyah<sup>3</sup>, Irma Lailatul Khoiriyah<sup>4</sup>,  
Oktavianis Kartikasari<sup>5</sup>, Imam Cholissodin<sup>6</sup>

<sup>123456</sup>Fakultas Ilmu Komputer, Universitas Brawijaya, Indonesia

Email: <sup>1</sup>anggodyusuf1950@gmail.com, <sup>2</sup>winda.cahyaningrum1@gmail.com, <sup>3</sup>Nfapril7@gmail.com,  
<sup>4</sup>irma.lailatul31@gmail.com, <sup>5</sup>oktakartikasari5@gmail.com, <sup>6</sup>imamcs@ub.ac.id

(Naskah masuk: 20 Februari 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Kredit merupakan suatu pendapatan terbesar bagi bank. Akan tetapi, bank harus selektif dalam menentukan nasabah yang dapat menerima kredit. Permasalahan ini menjadi semakin kompleks karena ketika bank salah memberikan kredit kepada nasabah dapat merugikan, selian itu banyaknya parameter penentu dalam penentuan nasabah yang kredit. *Clustering* merupakan salah satu cara untuk dapat menyelesaikan permasalahan ini. K-Means merupakan metode yang simpel dan populer dalam menyelesaikan permasalahan *clustering*. Akan tetapi, K-Means murni tidak dapat memberikan solusi optimum sehingga perlu dilakukan *improve* untuk mendapatkan solusi optimum. Salah satu metode optimasi yang dapat menyelesaikan permasalahan optimasi dengan baik adalah *particle swarm optimization* (PSO). PSO sangat membantu dalam proses *clustering* dengan melakukan optimasi pada titik pusat tiap *cluster*. Untuk meningkatkan hasil yang lebih baik pada PSO ada beberapa *improve* yang dilakukan. Pertama penggunaan *time variant inertia* untuk membuat nilai *w* atau *inertia* dinamis tiap iterasinya. Kedua melakukan kontrol kecepatan partikel atau *velocity clamping* untuk mendapatkan posisi terbaik. Selian itu untuk mengatasi konvergensi dini dilakukan *hybrid* PSO dengan *random injection*. Hasil pengujian menunjukan *hybrid* PSO K-Means memberikan hasil terbesar dibandingkan K-Means dan PSO K-Means, dimana *silhouette coefficient* dari K-Means, PSO K-Means, dan *hybrid* PSO K-Means masing-masing 0.57343, 0.792045, 1.

**Kata kunci:** Kredit, Clustering, PSO, K-Means, Random Injection

### Abstract

Credit is a biggest revenue for the bank. However, banks have to be selective in deciding which clients can receive the credit. This issue is becoming increasingly complex because when the bank was wrong to give credit to customers can do harm, that the large number of selian deciding parameter in determining customer credit. Clustering is one way to be able to resolve this issue. K-Means is a simple and popular method in solving clustering. However, the K-Means pure can not provide optimum solutions so that needs to be done to get the optimum solution to improve. One method of optimization that can solve the problems of optimization with particle swarm optimization is good (PSO). PSO is very helpful in the process of clustering to perform optimization on the central point of each cluster. To improve better results on PSO there are some that do improve. The first use of time variant inertia to make the value of each dynamic inertial *w* or iterasinya. Both control the speed of the particle velocity or clamping to get the best position. Selian it to overcome premature convergence do hybrid PSO with random injection. The results of this research provide the optimum results for solving clustering of customer credits. The test results showed the hybrid PSO K-Means provide the greatest results than K-Means and PSO K-Means, where the silhouette's of the K-Means, PSO K-Means, and hybrid PSO K-Means respectively 0.57343, 0.792045, 1.

**Keywords:** Credit, Clustering, PSO, K-Means, Random Injection

## 1. PENDAHULUAN

Kredit menurut undang-undang No.10/1998 pasal 1 (11) tentang Perubahan Atas undang-undang No.7/1992 tentang Perbankan merupakan kesepakatan antara pihak penyedia jasa keuangan dengan pihak lain atas penyediaan uang atau tagihan yang akan dikembalikan sesuai dengan kesepakatan waktu dan bunga yang telah disepakati bersama (YLBHI, 2007). Kredit timbul dari adanya

kepercayaan antara pemberi kredit dengan penerima kredit bahwa penerima kredit dapat memenuhi perjanjian atau kesepakatan bersama. Penjaminan keamanan transaksi dilakukan dengan melakukan pemilihan nasabah yang akan melakukan kredit secara selektif. Setiap bank memiliki parameter untuk menentukan nasabah yang dapat memperoleh kredit.

Penentuan pemberian kredit pada nasabah menjadi satu masalah yang cukup kompleks karena disatu sisi kredit dapat memberi masukan bank yang

cukup besar sedangkan disisi lain dapat merugikan jika nasabah tidak dapat membayar tagihan kredit. Resiko pemberian kredit pada nasabah yang kurang tepat menyebabkan bank tidak memperoleh pendapatan bunga, selain itu jumlah tagihan harus dikompensasi dengan jumlah modal sehingga modal yang ada akan berkurang dan mungkin habis (Suharli dan Oktorina, 2005). Hal tersebut dapat melanggar ketentuan *Capital Adequacy Ratio*. Jika Bank memiliki modal yang cukup, maka tagihan tersebut akan dicadangkan sesuai dengan peraturan Bank Indonesia No. 7/2/PBI/2005. Hal tersebut berpengaruh terhadap dana yang dapat ditempatkan dan menghasilkan bagi bank (Suhardi, 2006).

Ada berbagai pendekatan untuk menyelesaikan permasalahan tersebut, salah satunya adalah *clustering*. Ketepatan dalam membentuk model akan memberikan hasil yang maksimal (Anggodo dan Mahmudy, 2016). K-Means merupakan metode *clustering* yang populer dan bersifat tanpa kendali (Karimov dan Ozbayglu, 2015). Pembentukan model dengan K-Means dari data yang besar dapat memaksimalkan hasil pengelompokan (MacQueen, 1967). Tujuan utama K-Means adalah melakukan *clustering* objek ke *cluster* atau kelompok. Pertama adalah menentukan jumlah *k* yang mempresentasikan jumlah *cluster* yang akan dibentuk, lalu menentukan titik pusat atau *centroid* untuk setiap *cluster* dan dilakukan pengelompokan. Akan tetapi, karena penentuan titik pusat dilakukan secara acak pada tahapan pertama dan mengambil nilai rata-rata untuk langkah selanjutnya memberikan hasil yang kurang maksimal dan diperoleh konvergensi optimum lokal (Niknma dan Amiri, 2010). Permasalahan tersebut dapat diatasi dengan melakukan optimasi nilai titik pusat sehingga didapatkan titik pusat yang paling maksimal untuk melakukan perhitungan K-Means.

Ada berbagai metode meta-heuristik yang dapat digunakan untuk mengatasi permasalahan optimasi (Anggodo *et al*, 2017). PSO merupakan salah satu metode yang dapat melakukan optimasi nilai titik pusat K-Means dengan hasil yang baik (Kuo *et al*, 2011). Penggunaan PSO dapat diterapkan untuk melakukan optimasi nilai titik pusat dengan mempresentasikan solusi permasalahan dalam bilangan real (Alam *et al*, 2015). PSO terinspirasi dari kecerdasan individu dan social dari burung (Kennedy dan Eberhart, 1995). Pada penelitian Liu *et al* (2014) penggunaan PSO dapat melakukan *clustering* segmentasi citra digital. Sedangkan pada permasalahan kompleks penggunaan PSO juga sangat membantu dalam melakukan *clustering* (Li *et al*, 2017; Armano dan Farmani, 2016; Anggodo dan Mahmudy, 2017). Berdasarkan penelitian sebelumnya ditunjukkan penggunaan PSO sangat membantu untuk melakukan optimasi nilai *centroid* sehingga didapatkan tingkat kesalahan *clustering* yang rendah. Penelitian *clustering* nasabah kredit sebelumnya pernah diajukan oleh Wahyuni *et al*

(2016) dengan menggunakan *hybrid* K-Means dan PSO. Akan tetapi, pada penelitian sebelumnya hanya digunakan 2 parameter pada penelitian ini digunakan 13 parameter untuk menentukan seorang nasabah layak menerima kredit atau tidak. Selain itu juga dilakukan penanganan konvergensi dini untuk mendapat solusi terbaik menggunakan *random injection* (Mahmudy, 2015).

Fokus pertama penelitian ini adalah melakukan *clustering* nasabah kredit berdasarkan 13 parameter. Kedua untuk mendapatkan hasil *clustering* yang maksimal dan mencegah konvergensi optimum lokal dilakukan optimasi titik pusat menggunakan PSO. Ketiga penggabungan *random injection* dan PSO untuk penanganan konvergensi dini dan mendapatkan solusi terbaik.

## 2. ANALISIS NASABAH KREDIT

Kredit merupakan suatu pembiayaan yang diberikan bank kepada nasabah untuk jangka pendek (Sheny dan Williams, 2017). Pinjaman yang diberikan untuk keperluan sehari-hari rumah tangga, keperluan membeli rumah, motor, dan lain sebagainya yang bersifat konsumtif. Walaupun kredit bukanlah bantuan likuiditas atau kewajiban perusahaan memenuhi kewajiban jangka pendek akan tetapi kredit juga digunakan untuk keperluan produkti. Berdasarkan hal tersebut kredit menjadi sumber pendapatan paling besar bagi bank, disatu sisi juga memiliki resiko yang besar (Suhardi, 2006). Selain itu pada penelitian Albuquerque *et al* (2015) menunjukkan hasil prediksi nasabah kredit dari tahun ketahun semakin meningkat sehingga bank harus selektif dalam mengambil keputusan untuk mendapatkan pendapatan yang besar dan resiko yang kecil. Pada penelitian ini akan digunakan data dari UCI Dataset. Pada penelitian ini akan digunakan 4121 data dengan 13 parameter lain pekerjaan, status, pendidikan, pernah kredit, kredit rumah, pinjaman bank yang merupakan parameter tergolong kategori. Sedangkan pemasaran sebelumnya, umur, rata-rata gaji dalam tiga bulan, indek konsumsi nasabah dalam satu bulan, indek kepercayaan nasabah dalam satu bulan, rata-rata *euribor* 3 bulan untuk harian, dan nomer perkerja dalam 4 bulan merupakan parameter tergolong numerik. Tabel 1 menunjukkan data nasabah kredit.

**Table 1.** Data nasabah kredit

No	g1	g2	g3	g4	g5	g6	g7	g8	...	g13
1	30	2	2	3	1	2	1	2	...	5099.1
2	39	8	3	4	1	1	1	2	...	5191
-	-	-	-	-	-	-	-	-	...	...
-	-	-	-	-	-	-	-	-	...	...
2120	18	0	0	0	0	0	0	1	...	4963.6
2121	88	11	3	7	2	2	2	3	...	5228.1

Dimana parameter nasabah kredit menunjukkan g1, g2, g3, g4, g5, g6, g7, g8, g9, g10, g11, g12, g13 secara berurut-urut adalah umur, pekerjaan, status,

pendidikan, pernah kredit, kredit rumah, pinjaman bank, pemasaran sebelumnya, rata-rata gaji dalam tiga bulan, indeks konsumsi nasabah dalam satu bulan, indeks kepercayaan nasabah dalam satu bulan, rata-rata *euribor* 3 bulan untuk harian, dan nomor pekerja dalam 4 bulan.

### 3. K-MEANS

Prosedur untuk dalam melakukan *clustering* K-Means sangatlah simpel dan mendapatkan hasil yang terbaik (MacQueen, 1967). Pertama melakukan insialisasi nilai  $k$  yang merupakan jumlah *cluster*, lalu menentukan titik pusat untuk tiap *cluster* secara acak. Setiap objek akan diatur untuk dimasukkan ke *cluster* tertentu, dimana objek-objek dalam satu *cluster* memiliki kemiripan. Penentuan kemiripan objek dengan titik pusat *cluster* diukur dengan *Euclidean Distance* (D) yang ditunjukkan pada persamaan 1. Sebuah objek akan masuk ke *cluster* yang memiliki nilai D terkecil.

$$D(x_2, x_1) = \sqrt{\sum_{j=1}^p (x_{2j} - x_{1j})^2}, \quad (1)$$

Dimana,

$p$  = Dimensi data

$x_1$  = Posisi titik 1

$x_2$  = Posisi titik 2

Tahapan algoritma *clustering* K-Means sebagai berikut (Karimi dan Guerro-Zapata, 2015):

1. Insialisasi nilai  $k$  *cluster* dan *centroid* tiap *cluster* secara acak.
2. Menentukan setiap objek masuk *cluster* dengan jarak terdekat berdasarkan nilai *Euclidean Distance*.
3. Melakukan perhitungan ulang nilai *centroid* tiap *cluster* dengan persamaan 2.

$$V_j = \frac{1}{n_j} \sum_{p=0}^n \text{data}_p, \quad (2)$$

dimana  $V_j$  merupakan nilai *centroid* dari *cluster*  $j$ .  $n_j$  merupakan jumlah objek dalam *cluster*  $j$ .  $\text{data}_p$  menunjukkan vektor data ke  $p$ .

4. Ulangi langkah ke 2 sampai nilai *centroid* tidak berubah atau sudah melampaui iterasi tertentu.

### 4. PARTICLE SWARM OPTIMIZATION

PSO merupakan metode optimasi bersifat stotastik yang merepresentasikan solusi permasalahan dalam bentuk partikel (Kennedy dan Eberhart, 1995). Partikel dibangkitkan secara acak sebanyak  $n$  partikel, tiap partikel terdiri dari beberapa dimensi posisi  $x_i$  dan kecepatan  $v_i$ . Tiap partikel akan diukur seberapa besar nilai fitness yang diberikan dengan persamaan 3.

$$F = f(x) \quad (3)$$

Dimana  $f(x)$  adalah nilai dari *silhouette coefficient*, yaitu suatu metode untuk mengukur objek yang

terbentuk dalam *cluster*. Semakin besar nilai *silhouette coefficient* maka semakin bagus data yang dikelompokkan. Setiap partikel akan “terbang” menjelajahi ruang pencarian solusi. Perpindahan dari satu posisi ke posisi lainnya sangat dipengaruhi oleh kecepatan tiap partikel, untuk mendapatkan posisi terbaik diperlukan formulasi kecepatan yang dinamis menggunakan persamaan 4 (Eberhart dan Shi, 2000).

$$v_i^{t+1} = w \cdot v_i^t + c_1 \cdot r_1 \cdot (p_i - x_i) + c_2 \cdot r_2 \cdot (p_{gi} - x_i), \quad (4)$$

dimana  $v_i$  menunjukkan nilai kecepatan untuk dimensi partikel ke  $i$  sampai ke  $n$ ,  $t$  menunjukkan waktu iterasi,  $w$  merupakan nilai vektor inerti yang nilainya didapatkan secara dinamis menggunakan persamaan 5 (Ratnaweera *et al*, 2004).  $p_i$  adalah posisi terbaik yang pernah didapatkan untuk tiap partikel, sedangkan  $p_{gi}$  adalah posisi terbaik yang pernah dicapai oleh keseluruhan partikel.  $c_1$  dan  $c_2$  berurut adalah konstanta *cognitive* dan *social*, yang mana pada penelitian ini bernilai 1.  $r_1$  dan  $r_2$  adalah bilangan yang dibangkitkan secara acak antara  $[0,1]$ . Setelah didapatkan kecepatan maka akan dilakukan *update* posisi menggunakan persamaan 6.

$$w = (w_{max} - w_{min}) \frac{(Iterasi-t)}{Iterasi} + w_{min}, \quad (5)$$

$$x_i^{t+1} = x_i + v_i^{t+1}, \quad (6)$$

Pada perpindahan posisi partikel PSO biasanya terlalu cepat sehingga tidak dapat menemukan solusi optimum. Hal tersebut dapat diatasi dengan melakukan kontrol kecepatan atau *velocity clamping* (Marini dan Walzcak, 2015). Mekanisme kontrol kecepatan dengan melakukan kondisi untuk kecepatan tiap partikel menggunakan persamaan 7.

$$\begin{aligned} &\text{if } (v_{ij}^{t+1} > v_j^{\max}) \text{ maka } v_{ij}^{t+1} = v_j^{\max} \\ &\text{if } (v_{ij}^{t+1} < v_j^{\min}) \text{ maka } v_{ij}^{t+1} = v_j^{\min}, \end{aligned} \quad (7)$$

sedangkan nilai  $v_j^{\max}$  dibangkitkan menggunakan persamaan 8 dan  $v_j^{\min}$  merupakan nilai negatif dari  $v_j^{\max}$ .

$$v_j^{\max} = k \frac{(x_{j,\max} - x_{j,\min})}{2}, \quad k \in (0, 1], \quad (8)$$

Siklus perhitungan nilai kecepatan  $v_i$  dan *update* posisi  $x_i$  akan terus diulangi sampai iterasi selesai. Ketika iterasi sudah selesai, partikel terbaik keluar sebagai solusi optimum.

### 5. RANDOM INJECTION

*Random injection* merupakan suatu metode yang digunakan untuk penanganan konvergensi dini. *Random injection* pertama diajukan untuk menangani permasalahan konvergensi dini pada algoritma genetika (Mahmudy *et al*, 2013).

Permasalahan yang sering terjadi pada PSO adalah ketika iterasi tertentu, nilai solusi yang diberikan tiap partikel sama sebelum tercapai solusi optimum. Pada penelitian Wahyuni *et al* (2016)

penggunaan *hybrid* PSO masih belum memberikan hasil yang solusi optimum karena belum dilakukan penanganan konvergensi dini. Berbagai macam penanganan konvergensi dini salah satunya *random injection*, pada penelitian Mahmudy (2015) penggunaan *random injection* dapat meningkatkan hasil akurasi dari pada penggunaan PSO saja. Mekanisme *random injection* sangat sederhana, dengan memasukan  $p$  partikel acak pada  $q$  iterasi. Penentuan nilai  $p$  partikel dan  $q$  iterasi yang terbaik harus dilakukan pengujian terlebih dahulu.

**6. SILHOUETTE COEFFICIENT**

*Silhouette coefficient* merupakan suatu metode yang digunakan untuk mengukur seberapa baik objek diletakan dalam *cluster* (Wahyuni et al, 2016). Metode ini merupakan penggabungan dari metode *cohesion* dan *separation* (Anggara et al, 2016) yang digunakan untuk melakukan perhitungan nilai *fitness* pada tiap partikel dalam PSO. Tahapan perhitungan *silhouette coefficient* ditunjukkan pada persamaan 9 sampai 12:

1. hitung jarak rata-rata  $i$  titik *centroid* ke semua objek dalam satu *cluster*.

$$a(i) = \frac{1}{|A|} \sum_{j \in A, j \neq i} C d(i,j), \tag{9}$$

Dimana  $j$  merupakan objek dalam *cluster*  $A$  dan  $d(i, j)$  merupakan jarak objek  $i$  ke  $j$ .

2. hitung jarak rata-rata  $i$  titik *centroid* ke semua objek *cluster* lain, lalu ambil nilai terkecil.

$$d(i,C) = \frac{1}{|A|} \sum_{j \in C} d(i,j), \tag{10}$$

3. Dimana  $d(i,C)$  merupakan jarak objek  $i$  ke objek *cluster* lain.

$$b(i) = \min_{C \neq A} d(i,C), \tag{11}$$

4. hitung nilai *silhouette coefficient* untuk tiap *centroid*.

$$s(i) = \frac{b(i)-a(i)}{\max(a(i),b(i))}, \tag{12}$$

Nilai rata-rata *silhouette coefficient* atau  $s(i)$  menunjukkan seberapa tepat data dikelompokan,

**7. HYBRID PSO K-MEANS**

*Siklus Hybrid PSO K-Means Clustering*

Langkah 1: inisialisasi

- Inisialisasi parameter PSO seperti  $n, w_{max}, w_{min}, C1, C2$ .
- inisialisasi partikel yang merepresentasikan nilai *centroid* tiap *cluster* dan kecepatan.

Langkah 2: *update*

- *update* posisi tiap partikel dan hitung nilai *fitness*nya menggunakan K-Means
- Update posisi terbaik untuk tiap partikel.
- Update posisi terbaik keseluruhan partikel.

Langkah 3: *random injection*

- Memasukan  $p$  partikel yang dibangkitkan

secara acak untuk tiap  $q$  iterasi.

Langkah 4: kondisi berhenti

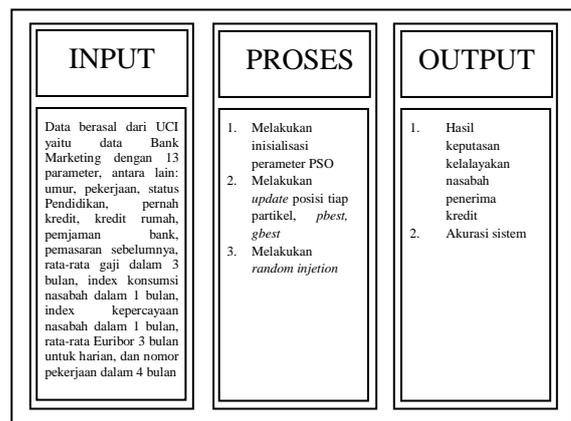
- Jika iterasi belum selesai lakukan langkah 2 dan sudah berhenti.

**Gambar 1.** Siklus metode *hybrid* PSO K-Means

Gambar 1 menunjukkan siklus metode *hybrid* PSO K-Means dalam melakukan *clustering* data nasabah kredit. Salah satu hal yang cukup berpengaruh terhadap hasil *clustering* adalah representasi partikel PSO dalam melakukan optimasi nilai *centroid*. Penelitian ini representasi partikel dalam bentuk kategori numerik dan interger sesuai dengan batasan masing-masing parameter. Terdapat dua baris dan 13 kolom, dimana baris merupakan nilai *centroid* tiap *cluster* dan kolom merupakan dimensi partikel yang merepresentasikan nilai dari parameter yang ditunjukkan pada Tabel 2 dan Gambar 2 menunjukkan alur kerja sistem.

**Table 2.** Representasi partikel

g1	g2	g3	g4	g5	g6	g7	g8	g9	...	g13
35	7	0	1	1	0	1	2	0.9	...	4335.7
47	8	0	4	0	1	1	2	-1.2	...	3049.9



**Gambar 2.** Alur kerja sistem

**8. METODOLOGI**

Proses pengelompokan nasabah kredit berdasarkan 13 parameter pada penelitian ini, akan dimaksimalkan menggunakan metode *hybrid* PSO K-Means. Langkah pertama K-Means adalah membangkitkan titik pusat secara acak. Pada penelitian ini terdapat 2, yaitu “diterima” untuk 1 dan “ditolak” untuk 2. Tabel 3 menunjukkan titik pusat dari 2 klas.

**Table 3.** Representasi partikel

No	g1	g2	g3	g4	g5	g6	g7	g8	g9	...	g13
1	35	7	0	1	1	0	1	2	0.9	...	4335.7
2	47	8	0	4	0	1	1	2	-1.2	...	3049.9

Langkah selanjutnya adalah menghitung kedekatan setiap dengan *centroid* 1 dan 2. Semisal terdapat

data (30, 5, 1, 2, 1, 1, 1, 0, 0.5, 40, -15, 1.2, 3555), sehingga diperoleh jarak dengan kedua *centroid* menggunakan persamaan 1.

$$D(\text{data}, c_1) = \sqrt{\begin{matrix} (35-30)^2 + (7-5)^2 + (0-1)^2 \\ + (1-2)^2 + (1-1)^2 + (0-1)^2 \\ + (1-1)^2 + (2-0)^2 + (0.9-0.5)^2 \\ + (16.428-40)^2 + (-27.2+15)^2 \\ + (1.367-1.2)^2 + (4335.7-3555)^2 \end{matrix}}$$

$$D(\text{data}, c_1) = 780.1748$$

Dengan cara yang sama sehingga diperoleh nilai  $D(\text{data}, c_2) = 507.4336$ . Karena jarak ke *centroid* 2 lebih pendek maka data tersebut dimasukkan ke dalam *cluster* 2. Langkah selanjutnya adalah melakukan penentuan *cluster* untuk data lainnya, lalu dilakukan perhitungan titik pusat menggunakan persamaan 2 sampai nilai titik pusat tidak berubah atau iterasi berakhir.

Pada PSO proses pertama adalah pembangkitan nilai partikel secara acak sesuai dengan batasan masing dimensi dalam partikel. Tabel 4 menunjukkan hasil pembangkitan nilai partikel. Selain itu inisialisasi nilai kecepatan tiap dimensi pada partikel dengan nilai 0.

**Table 4.** Inisialisasi partikel awal

No.	g1	g2	g3	g4	g5	g6	g7	g8	g9	...	g13
1	35	7	0	1	1	0	1	2	0.9	...	4335.7
	47	8	0	4	0	1	1	2	-1.2	...	3049.9
2	25	5	1	3	0	0	1	1	0.5	...	2000.7
	45	3	0	2	1	1	1	0	1	...	5000.9
-	-	-	-	-	-	-	-	-	-	-	-
20	34	5	0	2	1	0	1	1	-0.5	...	5000.7
	50	8	0	1	0	0	0	1	0.4	...	3000.9

Beberapa parameter yang dilakukan inisialisasi diawal seperti jumlah partikel, jumlah iterasi, *cognitif*,  $w_{max}$ , dan  $w_{min}$ . Langkah selanjutnya adalah menghitung nilai *fitness* dari setiap partikel, dimana nilai *fitness* diperoleh dari *silhouette coefficient* pada persamaan 9-12. Tabel 5 menunjukkan hasil perhitungan *fitness* untuk setiap partikel.

**Table 5.** Hasil perhitungan nilai *fitness*

Partikel	<i>Fitness</i>
1	0.2345
2	0.3214
...	...
20	0.1414

Setelah didapatkan nilai *fitness* untuk tiap partikel maka dilakukan upload *pbest* dan *gbest* yaitu nilai *fitness* terbaik pada tiap partikel dan keseluruhan partikel. Selanjutnya adalah tahapan melakukan perubahan kecepatan untuk tiap dimensi pada partikel menggunakan persamaan 4. Akan tetapi, bangkitkan terlebih dahulu nilai  $w$  menggunakan persamaan 5 dengan nilai iterasi 4,  $c_1 = 1$ ,  $c_2 = 1$ ,  $r_1 = 1$ , dan  $r_2 = 1$ .

$$w = (0.9 - 0.4) \frac{(10-1)}{10} + 0.4 = 0.96$$

$$v_{2,1}^2 = 0.96 \cdot 0 + 1.1 \cdot (25 - 30) + 1.1 \cdot (25 - 30) = -10$$

Pada partikel 2 dimensi 1 diperoleh nilai kecepatan -10, dengan cara yang sama maka diperoleh kecepatan pada dimensi tiap partikel yang ditunjukkan pada Tabel 6.

**Table 6.** Kecepatan baru

No.	g1	g2	g3	g4	g5	g6	g7	g8	g9	...	g13
1	-10	2	1	0	0	0	1	1	0.3	...	435
	12	3	1	2	0	1	1	2	0.5	...	304
2	14	4	0	1	1	0	1	0	0.7	...	207
	20	4	0	1	0	1	1	0	0.8	...	500
-	-	-	-	-	-	-	-	-	-	-	-
20	10	2	0	1	1	0	1	1	0.2	...	500
	23	3	0	1	1	0	0	1	0.4	...	300

Setelah didapatkan kecepatan maka langsung adalah melakukan perubahan posisi menggunakan persamaan 6.

$$x_{2,1}^2 = 25 + 14 = 39.$$

Dengan cara yang sama maka diperoleh nilai posisi baru yang ditunjukkan pada Tabel 7.

**Table 7.** Posisi baru

No.	g1	g2	g3	g4	g5	g6	g7	g8	g9	...	g13
1	25	9	1	1	1	0	2	3	1.2	...	477
											0.7
	59	11	1	6	0	2	2	4	-0.7	...	335
											3.9
2	39	9	1	4	1	0	2	1	1.2	...	220
											7.7
	65	7	0	3	1	2	2	0	1.8	...	550
											0.9
-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-
20	44	7	0	3	2	0	2	2	-0.3	...	550
											0.7
	73	11	0	2	1	0	0	2	0.8	...	330
											0.9

Proses PSO akan terus dijalankan sampai iterasi berakhir, *gbest* keluar sebagai solusi yang diberikan.

## 9. HASIL

Proses pengujian digunakan untuk mengetahui metode yang menghasilkan solusi terbaik. K-Means dan PSO merupakan metode statistik atau bersifat *random* sehingga akan dilakukan simulasi sebanyak 5 kali dan hasil rata-rata akurasi dan waktu komputasi akan dibandingkan.

*Clustering* nasabah bank akan diujikan sebanyak 4121 *record* nasabah. Data tersebut akan dikelompokkan ke *cluster* 1 (“diterima”) atau *cluster* 2 (“ditolak”). Parameter terbaik PSO yang akan digunakan adalah iterasi = 10, populasi = 10,  $v_{min} = -0.6$ ,  $v_{max} = 0.6$ ,  $w_{max} = 0.9$ ,  $w_{min} = 0.4$ . Sedangkan parameter *random injection* terbaik adalah  $p = 0.3$ ,  $q = 2$ . Hasil pengujian yang telah didapatkan ditunjukkan pada Tabel 8.

**Table 8.** Hasil perbandingan

Algoritma	Rata-rata Akurasi	Rata-rata Waktu komputasi (s)
K-Means	0.57343	0.58
PSO K-Means	0.792045	1.55
Hybrid PSO K-Means	1.00	1.67

Berdasarkan hasil pengujian parameter menunjukkan bahwa penggunaan *hybrid* PSO K-Means memberikan hasil akurasi yang maksimal sehingga metode pada penelitian ini dapat menentukan *cluster* nasabah kredit dengan sangat baik.

**10. KESIMPULAN**

*Clustering* permasalahan nasabah kredit telah diselesaikan secara baik menggunakan *hybrid particle swarm optimization* K-Means. Serangkaian pengujian menunjukkan bahwa penggunaan *hybrid* PSO K-Means dengan *random injection* memberikan hasil yang mendekati solusi optimum dan waktu komputasi yang relatif cepat, yaitu kurang dari 2 menit untuk jumlah data yang besar. *Hybrid* PSO K-Means memberikan hasil solusi yang lebih baik dibandingkan dengan PSO K-Means dan K-Means. Penggunaan *hybrid* PSO K-Means dapat melakukan *clustering* data dengan benar sebesar 80%.

Pada penelitian ini pada iterasi ke 10 sudah mengalami konvergensi secara global untuk keseluruhan partikel, hal ini karena sebagian besar parameter merupakan tipe data kategori yang jumlah kategorinya tidak lebih dari 5. Untuk mendapatkan hasil yang lebih baik lagi perlu dilakukan seleksi parameter. Salah satu cara seleksi parameter dengan merepresentasikan seleksi parameter kedalam segmen baru dalam partikel PSO (Novitasari *et al*, 2016).

**DAFTAR PUSTAKA**

ANGGODO, Y. P. & MAHMUDY, W. F., 2016. Peramalan butuhan hidup minimum menggunakan automatic clustering dan fuzzy logical relationship. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 3, no. 2, pp. 94-102.

ANGGODO, Y. P. & MAHMUDY, W. F., 2017. Automatic clustering and optimized fuzzy logical relationships for minimum living needs forecasting. *Journal of Environmental Engineering and Sustainable Technology (JEEST)*, vol. 4, no. 1, pp. 1-7.

ANGGODO, Y. P., ARIYANI, A. K., ARDI, M. K., & MAHMUDY, W. F. 2017. Optimization of Multi-Trip Vehicle Routing Problem with Time Windows using Genetic

Algorithm. *Journal of Environmental Engineering and Sustainable Technology* vol. 3, no.2, pp. 92-97.

ANGGARA, M., SUJIANI, H. & NASUTION, H., 2016. Pemilihan Distance Measure Pada K-Means Clustering Untuk Pengelompokan Member Di Alvaro Fitness, *Jurnal Sistem dan Teknologi Informasi (JustIN)*, vol. 1, no. 1, pp. 1-6.

ALAM, S., DOBBIE, G. & REHMAN, S. S., 2015. Analysis of particle swarm optimization based hierarchical data clustering approaches, *Swarm and Evolutionary Computation*, vol. 25, pp. 36-51.

ARMANO, G. & FARMANI, M. R., 2016. Multiobjective clustering analysis using particle swarm optimization. *Expert Systems With Applications*, vol. 55, pp. 184-193

EBERHART, R.C. & SHI, Y., 2000. Comparing inertia weight and constriction factors in particle swarm optimization. *IEEE Congress on Evolutionary Computation*, 16-19 Juni, San Diego, CA, US.

KARAMI, A. & GUERRERO-ZAPATA, M., 2015. A fuzzy anomaly detection system based on hybrid PSO-Kmeans algorithm in content-centric networks. *Neurocomputing*, vol. 149, no. PC, pp. 1253-1269,

KARIMOV, J. & OZBAYOGLU, M., 2015. *Clustering Quality Improvement of k-means Using a Hybrid Evolutionary Model*, *Procedia Comput. Sci.*, vol. 61, pp. 38-45.

KENNEDY, J. & EBERHART, R., 1995. Particle swarm optimization. *IEEE International Conference on Neural Networks*, 27 Nov. – 1 Dec., Perth, Western, Australia, vol. 4, pp. 1942-1948.

KUO, R. J., WANG, M. J. & HUANG, T. W., 2011. An application of particle swarm optimization algorithm to clustering analysis. *Soft Comput.*, vol. 15, no. 3, pp. 533-542.

LIU, L., JIAO, L., ZHAO, J., SHANG, R. & GONG, M., 2017. Quantum-behaved discrete multi-objective particle swarm optimization for complex network clustering. *Pattern Recognition*, vol. 63, pp. 1-14.

LIU, R., CHEN, Y., JIAO, LICHENG. & LI, Y., 2014. A particle swarm optimization based simultaneous learning framework for clustering and clustering. *Pattern Recognition*, vol. 47, pp. 2143-2152.

MACQUEEN, J., 1967. Some methods for clustering and analysis of

multiobservations. *Proceedings of the fifth symposium on mathematical statistic and probability*, 21 Juni – 18 Juli, California, USA, vol. 1, pp. 281-297.

- MAHMUDY, W. F., 2015. Improved particle swarm optimization untuk menyelesaikan permasalahan part type selection dan mechine loading pada flexible manufacturing system (FMS). *Konf. Nas. Sist. Inf. (KNSI)*, Universitas Klabat, Airmadidi, Minahasa Utara, Sulawesi Utara, 26-28 Februari, pp. 1003–1008.
- MAHMUDY, W. F., MARIAN, R. M. & LUONG, L. H. S. 2013. Optimization of part type selection and loading problem with alternative production plans in flexible manufacturing system using hybrid genetic algorithms – Part 1: modelling and representation. *5th International Conference on Knowledge and Smart Technology (KST)*, Chonburi, Thailand, pp. 75-80.
- MARINI, F. DAN WALCZAK, B., 2015., Particle swarm optimization (PSO). A tutorial. *Chemometrics and Intelligent Laboratory Systems*, vol. 149, pp. 153-165.
- NIKNAM, T. & AMIRI, B., 2010. An efficient hybrid approach based on PSO, ACO, and K-Means for cluster analysis. *Applied soft computing*, vol. 10, pp. 183-197.
- NOVITASARI, D., CHOLISSODIN, I. & MAHMUDY, W. F., 2016. Hybridizing PSO with SA for optimizing SVR applied to software effort estimation. *TELKOMNIKA*, vol. 14, no. 1, pp. 245-253.
- SHENOY, J. & WILLIAMS, R., 2017. Trade credit and the joint effect of supplier and customer financial characteristics. *J. Finan. Intermediation*, vol. 29, pp. 68-80.
- SUHARLI, M. & OKTORINA, M., 2005. Memprediksi tingkat pengembalian investasi pada equity securities melalui rasio profitabilitas, likuiditas, dan utang pada perusahaan publik di Jakarta. *Semen. Nas. Akunt. VII*, pp. 288-296.
- RATNAWEERA, A., HALGAMUGE, S. K. DAN WATSON, H. C., 2004. Self-organizing hierarchical particle swarm optimization with time-varying acceleration coefficients, *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240-255.
- SUHARDI, G., 2006. Resiko dalam pemberian kredit perbankan. *Jurnal Hukum Projustitia*, vol. 24, no. 1, pp. 96-111.
- WAHYUNI, I., AULIYA, Y. A., RAHMI, A. & MAHMUDY, W. F., 2016. Clustering nasabah bank berdasarkan tingkat likuiditas menggunakan hybrid particle swarm optimization. *Jurnal Ilmiah Teknologi dan Informasi ASIA (JITIKA)*, vol. 10, no. 2, pp. 24-33.
- YLBHI, 2007. *Panduan Bantuan Hukum di Indonesia: Pedoman Anda Memahami dan Menyelesaikan Masalah Hukum*. Jakarta: Yayasan Obor Indonesia. <https://books.google.co.id/books?id=Y1oghffVI2cC>.

## PEMETAAN DATA DAN VISUALISASI KEDALAMAN AIR PADA BENDUNGAN / WADUK

Bayu Rahayudi<sup>1</sup>, Marji<sup>2</sup>

<sup>12</sup>Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>ubay1@ub.ac.id, <sup>2</sup>marji@ub.ac.id

(Naskah masuk: 01 Maret 2017, diterima untuk diterbitkan: 07 Mei 2017)

### Abstrak

Penelitian dilakukan untuk melakukan pemetaan data dan visualisasi kedalaman air suatu waduk. Hasil dari pemetaan dan visualisasinya digunakan untuk mengetahui kontur endapan dan volume waduk. Pemetaan dilakukan dengan memprediksi titik-titik yang tidak diketahui kedalaman airnya berdasarkan titik-titik yang diketahui kedalaman airnya. Metode prediksi yang digunakan pada penelitian ini adalah metode interpolasi IDW (*Inverse Distance Weighting*), yaitu perhitungan rata-rata n buah titik, dimana titik-titik terdekat memiliki bobot pengaruh yang lebih besar dibandingkan titik-titik hasil pengukuran yang lebih jauh. Digunakannya metode ini bertujuan agar proses prediksi yang dilakukan dapat dilakukan dengan cepat, karena prediksi dilakukan untuk banyak titik-titik lain yang belum diketahui kedalaman airnya. Dari hasil penelitian diperoleh hasil bahwa jumlah titik yang paling baik digunakan untuk memprediksi titik yang tidak diketahui kedalaman airnya adalah sebanyak 5 buah titik terdekat. Data yang digunakan sebagai data percobaan pada penelitian ini adalah data pengukuran kedalaman air di waduk atau bendungan Wonorejo tahun 2004 dan tahun 2005. Hasil dari penelitian ini adalah suatu perangkat lunak yang digunakan untuk melakukan pemetaan dan memvisualisasikan kedalaman air di suatu waduk atau bendungan.

**Kata kunci:** *visualisasi, pemetaan, interpolasi, inverse distance weighting (idw)*

### Abstract

*Prediction has been widely used in every research and field. In water level visualization system, prediction can be used for measuring water level at the dam. From the measurement, can be known the dam contour and also its water volume. The prediction can be done by measuring water level at several certain points, and used the points to predict another unmeasure points. Prediction metode used in this research is inverse distance weighting (IDW) interpolation, which were using greater weighting on neaerer point than farther point. From this research known that using 5 nearest point will result optimum prediction. This metode is used in order to get faster result because of thousand prediction will be done. Data used in this research are water level measurement data of Wonorejo dam in 2004 and 2005. The result of this research is a program that can be used to map and visualize water level of a dam.*

**Keywords:** *visualization, mapping, interpolation, inverse distance weighting (idw)*

## 1. PENDAHULUAN

Prediksi data telah banyak digunakan dalam berbagai bidang. Pada bidang ekonomi, prediksi dapat digunakan untuk memprediksi pertumbuhan ekonomi di masa mendatang. Di bidang demografi, prediksi data digunakan untuk memprediksi pertumbuhan penduduk di suatu wilayah tertentu. Demikian pula di bidang pengairan, prediksi data dapat digunakan untuk memprediksi kedalaman air di suatu waduk atau bendungan. Dari pengukuran kedalaman air tersebut dapat diketahui kontour dasar waduk atau bendungan tersebut, dan pada akhirnya dapat pula diketahui volume airnya.

Bendungan atau waduk adalah sarana vital yang salah satunya digunakan sebagai pengatur dan pengendali debit dan volume air yang mengalir di suatu daerah. Volume air yang dapat ditampung di suatu bendungan atau waduk dipengaruhi oleh

tingginya endapan dan sedimentasi yang terjadi di dasarnya. Jika endapan dan sedimentasi di dasar bendungan atau waduk tersebut cukup banyak/tinggi, maka volume air yang dapat ditampung oleh bendungan atau waduk tersebut semakin berkurang, sehingga diperlukan tindakan lebih lanjut, misalnya dengan pengerukan endapan atau sedimentasi tersebut. Tinggi/banyaknya endapan atau sedimentasi di suatu bendungan atau waduk dapat diketahui dengan melakukan pengukuran kedalaman air di beberapa tempat.

Data kedalaman air yang diperoleh dari hasil pengukuran di suatu bendungan atau waduk, bukanlah data yang berukuran kecil, karena data kedalaman air tersebut diperoleh dari pengukuran terhadap beberapa puluh atau beberapa ratus titik pengukuran. Analisa terhadap kedalaman air, yang meliputi analisa ketinggian endapan atau sedimentasi pada dasar bendungan atau waduk, cukup sulit

dilakukan jika hanya dengan melihat data-data pengukuran yang telah diperoleh. Dengan melakukan visualisasi dan pemetaan terhadap data kedalaman air yang diperoleh akan memudahkan petugas atau tenaga ahli yang bekerja untuk melakukan analisa terhadap kondisi yang terjadi pada bendungan atau waduk tersebut, misalnya dengan mengetahui volume air bendungan atau waduk, dan perkiraan terhadap ketinggian endapan atau sedimentasi yang terdapat di tempat atau area tertentu.

Penggunaan data pengukuran untuk melakukan pemetaan atau perhitungan belumlah mencukupi, karena untuk membuat pemetaan dan visualisasi data yang baik diperlukan data dalam jumlah besar. Jika dilakukan pengukuran untuk semua titik yang diperlukan, maka akan diperlukan waktu, tenaga dan biaya yang besar. Untuk itu diperlukan prediksi terhadap titik-titik yang tidak diketahui dengan memanfaatkan titik-titik hasil pengukuran. Metode prediksi yang digunakan dapat digunakan adalah metode interpolasi IDW (*inverse distance weighting*). Pada tahun 1999 metode ini digunakan untuk melakukan prediksi kandungan suatu tanah berdasarkan atas kandungan tanah yang ada di sekitarnya (Kravchenko, 1999).

### 1.1 Pengertian Visualisasi Dan Pemetaan

Visualisasi didefinisikan sebagai suatu metode untuk merepresentasikan suatu data atau permasalahan ke dalam format grafik atau bentuk gambar yang mudah untuk dipahami. Sedangkan pemetaan adalah salah satu bentuk visualisasi, yaitu dengan merepresentasikan suatu data ke dalam suatu grafik dengan klasifikasi, misalnya peta. Salah satu bidang yang banyak berhubungan dengan visualisasi dan pemetaan adalah sistem informasi geografis (SIG) atau Geographic Information System (GIS) (Kang-tsung, 2006).

Roth (2015) melakukan penelitian dengan membandingkan beberapa perangkat lunak visualisasi pemetaan sistem air, untuk mengetahui perangkat lunak yang paling efisien di dalam melakukan visualisasi data dan perhitungan prediksi bencana yang dapat terjadi.

### 1.2 Prediksi Data Spasial

Prediksi data spasial adalah sebuah proses penentuan nilai yang belum diketahui atau nilai di masa mendatang menggunakan data yang telah dimiliki. Prediksi data spasial secara umum merupakan sebuah metode prediksi yang dapat menyatukan berbagai macam ketergantungan spasial. Proses ini digunakan dalam berbagai bidang seperti eksplorasi minyak, pertambangan, dan polusi air. Data yang tersedia untuk bidang ini berupa data pada lokasi-lokasi tertentu (seperti stasiun pengukuran di atas tanah maupun pengukuran dengan jarak tertentu

di udara). Tujuan pemanfaatan proses prediksinya adalah untuk memprediksi nilai pada lokasi tertentu yang tidak mengalami pengukuran. Lokasi yang tidak diukur biasanya ditampilkan dalam bentuk grid biasa dan prediksi data digunakan untuk melakukan plot permukaan ataupun pembuatan peta garis batas (*contour map*).

Proses prediksi data yang dilakukan untuk area yang tidak mengalami pengukuran atau tidak memiliki data disebut sebagai *spatio temporal prediction* (Manolopoulos, et.al, 2005). Prediksi ini dapat dilakukan dengan menggunakan berbagai macam algoritma, dan yang paling sederhana dan mudah digunakan adalah dengan menggunakan interpolasi.

### 1.3 Interpolasi

Metode prediksi dengan interpolasi merupakan metode dasar yang banyak digunakan untuk melakukan pengolahan data seperti fungsi, kurva, dan proses serta perluasan informasi ke sebuah domain yang lebih luas. Perluasan ini hanya berupa perkiraan, dan ketepatannya tergantung pada berbagai faktor yang ada dalam proses perkiraan tersebut. Interpolasi sebagai sebuah alat dapat dilihat juga sebagai sebuah kasus untuk perkiraan fungsi penyebaran data. Proses perkiraan ini menggunakan batasan-batasan tertentu, sehingga sebuah perkiraan dapat dihasilkan dari interpolasi ketika batasan-batasan yang ada mengizinkan.

### 1.4 Interpolasi Inverse Distance Weighting (IDW)

*Interpolasi Inverse Distance Weighting* (IDW) adalah sebuah proses interpolasi yang berdasarkan asumsi bahwa nilai yang tersimpan pada lokasi terdekat, dengan lokasi yang akan dicari nilainya, memiliki pengaruh yang lebih besar terhadap nilai interpolasi dari pada nilai yang lokasinya lebih jauh (Armes, 2006; Kravchenko, 1999). Metode ini menjadikan jarak dari nilai yang diketahui berbanding terbalik dengan pengaruh yang diberikan. Ketepatan nilai yang dihasilkan tergantung pada jumlah nilai yang digunakan dan jarak nilai yang dicari. Interpretasi IDW secara matematika didefinisikan seperti pada persamaan (1) :

$$Z^*(x_0) = \sum_{i=1}^n w_i * Z(x_i) \quad (1)$$

dimana  $Z^*(x_0)$  adalah nilai yang dicari untuk lokasi  $x_0$ ,  $Z(x_i)$  adalah nilai-nilai di sekitar  $Z^*(x_0)$  yang digunakan sebagai data perhitungan,  $W_i$  merupakan bobot jarak masing-masing titik terhadap nilai yang dicari, dan  $n$  merupakan jumlah titik yang akan digunakan (*referensi*). Jumlah nilai yang digunakan sebagai referensi bisa berjumlah tetap ataupun bervariasi. Dari persamaan (1) dapat diketahui bahwa

data yang digunakan sebagai referensi harus diberi bobot. Pembobotan untuk setiap data yang digunakan dapat dihitung dengan menggunakan persamaan (2) :

$$w_i = \frac{\frac{1}{d_i^p}}{\sum_{i=1}^n \frac{1}{d_i^p}} \quad (2)$$

dimana  $w_i$  adalah bobot untuk nilai ke- $i$ , dimana merupakan jarak dari lokasi nilai ke- $i$  dengan lokasi nilai yang dicari, dan  $p$  merupakan nilai parameter eksponensial untuk jarak titik ke  $i$ .

## 2. METODE PENELITIAN

Metode penelitian yang dilakukan meliputi proses pencarian dan perolehan data, dilanjutkan dengan perancangan basis data, kemudian dilakukan perancangan perangkat lunak. Data yang digunakan pada penelitian ini adalah data kedalaman air di suatu waduk (selama dua periode, 2004 dan 2005) yang diperoleh dari data kedalaman air waduk yang terdapat di salah satu bendungan atau waduk di propinsi Jawa Timur (bendungan Wonorejo). Data yang digunakan adalah data akhir (yang telah ada) yang diperoleh dari instansi terkait (Jasa Tirta), dan tidak dengan melakukan pengukuran kedalaman air secara mandiri. Data akhir tersebut adalah data hasil survey lapangan yang dilakukan secara berkala oleh petugas di bendungan atau waduk tersebut (Anonym, 2004).

Struktur basis data yang digunakan mengacu kepada struktur umum yang digunakan pada basis data geospasial, yaitu data spasial (keruangan) yang akan menyimpan data posisi (jarak vertikal atau  $y$ , dan jarak horisontal atau  $x$ ) dari titik-titik pengukuran, dan data atribut (deskripsi) yang akan menyimpan data kedalaman air pada titik-titik pengukuran yang telah ditentukan. Posisi titik-titik pengukuran (nilai  $x$  dan  $y$ ) dihitung dari suatu titik referensi pengukuran yang telah ditentukan, dimana nilai  $x$  adalah jarak horisontal titik pengukuran ke titik referensi, dan nilai  $y$  adalah jarak vertikal titik pengukuran ke titik referensi yang digunakan. Setiap data-data pengukuran akan disimpan ke dalam tabel, sehingga ketika ingin melakukan visualisasi dan pemetaan data berdasarkan pada data atau periode pengukuran tertentu, dapat dilakukan dengan membaca tabel yang sesuai.

Setelah basis data didesain dan algoritma prediksi yang digunakan telah ditentukan, kemudian dilakukan desain *user interface* dan proses interaksi yang diperlukan antara pengguna dengan sistem informasi yang dibuat. Proses interaksi yang dapat terjadi antara lain interaksi pemasukan data, manipulasi dan analisa data, dan interaksi untuk mendapatkan informasi dari hasil pemrosesan data (output). Interaksi pemasukan data, dibuat agar pengguna dapat memasukkan data-data pengukuran, yang

meliputi lokasi pengukuran (nilai  $x$  dan  $y$  pengukuran dari titik referensinya), dan data kedalaman airnya. Agar proses penginputan data mudah dilakukan, maka proses inputnya dibuat dalam bentuk tabel (tabular). Proses pemetaan dan visualisasi data dapat dilakukan bila proses input data telah selesai dilakukan.

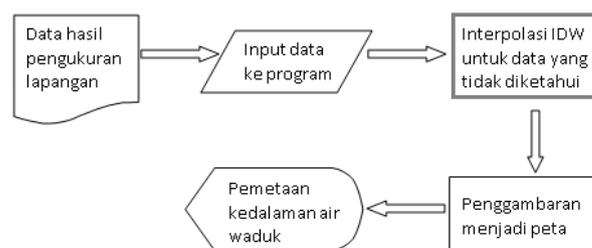
## 3. HASIL DAN PEMBAHASAN

### 3.1 Deskripsi Sistem

Sistem yang akan dibangun adalah sistem yang dapat melakukan penyajian data spasial ke dalam bentuk peta atau gambar dua dimensi. Data spasial yang digunakan adalah data spasial titik (*point*) yang berupa kedalaman air di suatu waduk hasil pengukuran lapangan.

### 3.2 Diagram Alir Sistem

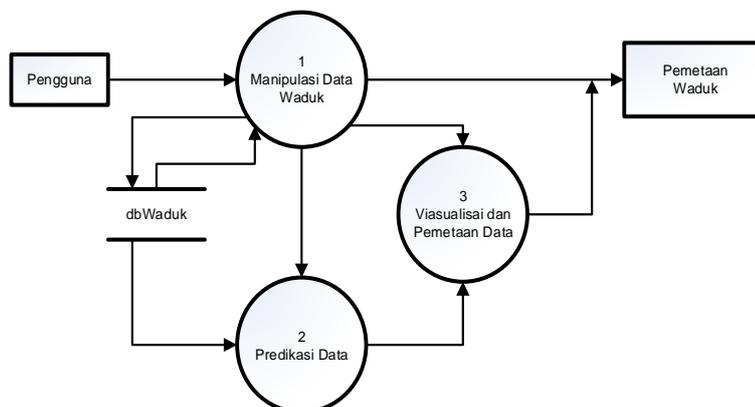
Diagram alir sistem pada penelitian ini dapat dilihat pada gambar 4.1.



Gambar 1. Diagram alir dari sistem yang dibuat.

Diagram tersebut menjelaskan tentang proses prediksi sampai dengan pemetaan dari sistem yang dibuat.

Diagram alur data level 0 dari sistem yang dibuat dapat dilihat pada Gambar 2. Pada diagram alur data level 0 dijabarkan bahwa sistem memiliki tiga proses utama, yaitu manipulasi data, prediksi data dan visualisasi data. Pada proses manipulasi data dan prediksi data akan melibatkan media penyimpanan, yang dalam hal ini adalah database dbwaduk.



Gambar 2. Diagram level 0 dari sistem yang dibuat.

### 3.3 Pembuatan Basis Data

Proses prediksi data dan pemetaan kedalaman air waduk yang dilakukan dalam penelitian ini memerlukan data waduk serta data kedalaman air hasil pengukuran lapangan. Data waduk yang digunakan antara lain nama waduk, lokasi waduk, daya tampung maksimum dan efektif dari waduk.

Dapat pula ditambahkan keterangan-keterangan lain tentang waduk sebagai informasi pelengkap. Sedangkan data kedalaman air yang digunakan adalah data lokasi dalam bentuk absis, ordinat, dan elevasi air.

Berdasarkan kebutuhan data tersebut, maka dibuat tabel yang dapat menampung data-data tersebut. Setelah dilakukan analisa kebutuhan, maka dibentuk beberapa tabel, yaitu tabel waduk, tabel lokasi\_point, tabel cross\_point, tabel ukur\_lapang, tabel spatial\_data\_saver. Tabel-tabel ini digunakan untuk menyimpan seluruh data hasil pengolahan data lapangan yang akan dijadikan data acuan untuk proses prediksi dan pemetaan kedalaman air waduk

Tabel waduk menyimpan data yang berkaitan dengan keterangan waduk seperti nama waduk, lokasi waduk, tahun peresmian, level air, dan daya tampung waduk. Tabel lokasi\_patok menyimpan data lokasi patok yang digunakan sebagai titik acuan untuk pengukuran melintang. Tabel ini menyimpan nama patok, lokasi patok, dan ketinggian patok. Tabel cross\_point menyimpan data pasangan patok yang digunakan untuk pengukuran melintang kedalaman air waduk. Tabel ini berisi nama cross point, patok awal, patok akhir dan jarak antar patok. Tabel ukur\_lapang menyimpan data kedalaman air dan lokasi pengukuran hasil pengukuran lapangan. Data ini meliputi data cross point pengukuran, tahun pengukuran, jarak ukur dari patok awal, dan kedalaman air. Tabel ini digunakan untuk menyimpan dokumentasi data hasil pengukuran lapangan. Tabel penyimpanan data spasial (spatial data saver) menyimpan data spasial hasil pengolahan data lapang yang telah diubah dalam bentuk titik x,y,z dan tahun pengukuran. Data spasial tersebut merepresentasikan lokasi pengukuran dan kedalaman air. Struktur dan relasi antar tabel dapat digambarkan seperti pada Gambar 3.

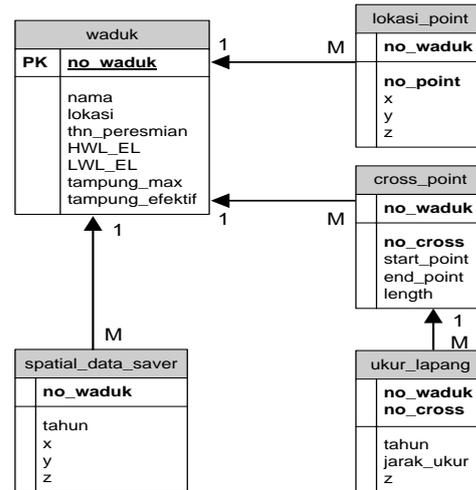
### 3.4 Prediksi Data

*Inverse Distance Weigthing* (IDW) merupakan metode interpolasi yang memanfaatkan bobot dan nilai dari masing-masing data referensi yang berada di sekitar titik yang dicari untuk menentukan nilai yang tidak diketahui tersebut. Langkah-langkah yang dilakukan untuk melakukan prediksi adalah sebagai berikut :

1. Menentukan lokasi titik yang belum memiliki nilai kedalaman air.
2. Menentukan nilai-nilai di sekitar titik yang dicari dengan jumlah tertentu.

3. Menentukan jarak dan bobot dari masing-masing data referensi menggunakan persamaan bobot untuk IDW.
4. Penentuan bobot ini menggunakan parameter eksponensial dengan nilai 2.
5. Setelah bobot dari masing-masing data referensi diperoleh, proses prediksi data dilaksanakan.

Hasil prediksi kemudian dipetakan ke dalam bentuk gambar melalui proses pemetaan.



Gambar 3. Struktur dan relasi antar tabel.

Data kedalaman air waduk hasil pengukuran lapangan berupa jarak lokasi pengukuran dengan titik acuan dan kedalaman air. Data ini dikonversi menjadi data spasial berbentuk titik (*point*) dengan x,y untuk lokasi dan z untuk kedalaman air. Data spasial hasil konversi digunakan sebagai data referensi untuk prediksi kedalaman air di seluruh area waduk.

Prediksi data kedalaman air dilakukan dengan menggunakan sejumlah titik referensi tertentu. Evaluasi dilakukan terhadap sejumlah titik referensi yang berbeda untuk mengetahui tingkat ketepatan dan tingkat kesalahan prediksi data. Jumlah titik referensi yang dievaluasi yaitu 5, 10, 15, 20, 25 dan 30 dengan parameter eksponensial 2.

### 3.5 Pemetaan Data

Pemetaan data merupakan bagian yang memanfaatkan proses prediksi data. Nilai-nilai yang dihasilkan dari proses prediksi kemudian ditampilkan ke layar dengan memanfaatkan library grafik dari OpenGL dengan memanfaatkan obyek point dan triangel. Proses pemetaan ini dilakukan untuk seluruh area waduk untuk penggambaran data dengan jarak tertentu.

Proses pemetaan hasil prediksi data tersebut dapat dijelaskan dalam langkah-langkah sebagai berikut :

1. Penentuan jarak pemetaan setiap data untuk mendapatkan detail data dan gambar.

2. Penentuan jumlah titik referensi untuk proses prediksi
3. Dilakukan prediksi untuk semua titik yang diperlukan untuk proses pemetaan dan menyimpannya ke dalam tabel sementara (*spatial data saver*)
4. Gambar ke dalam bentuk triangle untuk semua titik pengukuran dan perhitungan yang diperoleh.
5. Pengulangan proses pemetaan jika diinginkan tingkat ketelitian yang berbeda

Hasil prediksi data dengan menggunakan jumlah titik referensi dengan tingkat kesalahan terkecil dipetakan menggunakan fasilitas pustaka OpenGL. Proses ini dilakukan untuk menampilkan data hasil prediksi dalam bentuk peta dua dimensi menggunakan primitif *triangle strip*. Setiap data kedalaman air direpresentasikan dengan warna untuk menunjukkan nilai kedalaman air dalam peta. Area perairan waduk dipetakan dengan warna biru untuk air dalam dan biru muda untuk air dangkal sedangkan area di luar perairan dipetakan dengan warna hijau dan kuning.

### 3.6 Evaluasi Hasil Pemetaan Data

Evaluasi dilakukan dengan membandingkan hasil prediksi dengan menggunakan n jumlah titik referensi yang berbeda-beda, dengan nilai n adalah 5, 10, 15, 25 dan 30. Berdasarkan hasil evaluasi tersebut, nilai kesalahan terkecil untuk prediksi yang diuji yaitu terdapat pada penggunaan 5 data referensi terdekat dengan menggunakan data kedalaman air waduk.

Ternyata semakin besar jumlah data referensi yang digunakan menyebabkan tingkat kesalahan semakin besar sehingga ketepatan prediksi data berkurang. Hal ini disebabkan semakin banyaknya data yang berpengaruh dan nilai yang berpengaruh semakin bervariasi. Hasil prediksi dengan menggunakan data referensi semakin banyak menghasilkan nilai yang semakin berbeda dengan nilai sebenarnya.

Hasil prediksi dan pemetaan dengan menggunakan 5 buah titik referensi dapat dilihat pada gambar 4. Warna kuning (terang) menunjukkan daerah yang sangat dangkal, sedangkan yang berwarna biru tua (gelap) menunjukkan daerah yang dalam. Dari hasil prediksi, pemetaan dan visualisasi tersebut dapat diketahui kondisi daerah yang memiliki area yang dangkal dan dalam, dan juga daerah-daerah yang memiliki endapan yang tinggi.

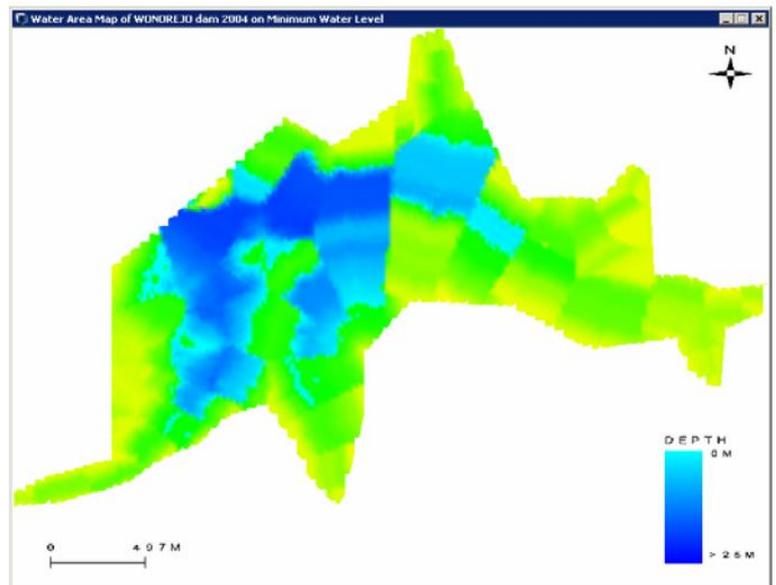
## 4. KESIMPULAN

Dari penelitian yang telah dilakukan dapat dibuat kesimpulan sebagai berikut :

1. Pemetaan dan visualisasi dapat dilakukan terhadap hasil pengukuran lapangan dari

kedalaman air dari suatu waduk, sehingga dapat diperoleh hasil analisis yang lebih baik dan lebih mudah untuk dipahami.

2. Di dalam melakukan pemetaan dan visualisasi data kedalaman air suatu waduk perlu dilakukan prediksi terhadap titik-titik yang tidak dilakukan pengukuran terhadapnya. Untuk itu dapat digunakan metode prediksi interpolasi yang sederhana. Digunakannya metode ini karena proses prediksi dapat dilakukan dengan cepat, mengingat titik-titik yang diprediksi jumlahnya cukup banyak.
3. Metode Interpolasi *Inverse Distance Weighting* (IDW) berdasarkan tingkat ketepatan prediksi data dapat digunakan untuk memprediksikan data spasial yang berupa data kedalaman air waduk. Hasil prediksi data yang berupa data lokasi dan kedalaman air dapat ditampilkan dalam bentuk peta dua dimensi menggunakan metode grafik OpenGL dengan primitif *triangle strip*. Jumlah data referensi yang dapat digunakan dalam metode ini yaitu sebanyak 5 data referensi dengan data referensi yang bersifat linear.
4. Semakin besar jumlah data referensi yang digunakan, semakin besar tingkat kesalahan yang dihasilkan sehingga ketepatan prediksi data berkurang.



Gambar 4. Hasil prediksi dan pemetaan dengan menggunakan 5 titik referensi.

## 5. DAFTAR PUSTAKA

- ANONYM, 2004, Laporan Echo Sounding Waduk Wonorejo, Divisi Jasa Air dan Sumber Air, Perum Jasa Tirta I, Malang.

- ARMES, C.J., 2006, Spatial Prediction of Soil Penetration Resistance using Geographic Information Systems, Centre of Excellence – GIS Laboratory, Lincoln University, Missouri.
- KRAVCHENKO AND G. BULLOCK, 1999, Comparative Study of Interpolation Method for Mapping Soil Properties, *Agron Journal*, 91: 393-400.
- KANG-TSUNG, C., 2006, Introduction to Geographic Information Systems, McGraw-Hill.
- MANOLOPOULOS, Y., PAPADOPOULOS, A. N., AND VASSILAKOPOULOS, M. GR., 2005, Spatial Databases: Technologies, Techniques, and Trends, Idea Group Publishing.
- ROTH RE, C QUINN, AND D HART, 2015, The competitive analysis method for evaluating water level visualization tools, In: A. Vondrakova, J. Brus, and V. Vozenilek (eds) *Modern Trends in Cartograph Lecture Notes in Geoinformation and Cartography*, Chapter 19: 241-256

## DESAIN PROTOKOL SUARA SEBAGAI PENGENDALI DALAM SMART HOME MENGUNAKAN FPGA

Barlian Henryranu Prasetio<sup>1</sup>, Dahnia Syauqy<sup>2</sup>

<sup>1,2</sup> Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>barlian@ub.ac.id, dahnia87@ub.ac.id<sup>2</sup>

(Naskah masuk: 9 Maret 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Smart home adalah suatu sistem yang menggunakan komputer dan teknologi informasi untuk mengendalikan peralatan yang terdapat di rumah seperti jendela dan lampu. Sistem dapat berupa sistem control sederhana hingga sistem yang kompleks. Komputer/mikrokontroler yang berbasis jaringan internet/ethernet dilengkapi dengan sistem cerdas dan sistem otomasi sehingga mampu membuat rumah menjadi bekerja secara otomatis. Banyak perangkat komputer/mikrokontroler yang dapat diimplementasikan sebagai pengendali dalam smart home. Sistem pengendali smart home pada penelitian ini menggunakan Xilinx xpartan-3e yang mengendalikan peralatan dalam rumah melalui jaringan LAN (Local Area Networking). Sistem pengendali ini berkomunikasi menggunakan broadcast voice pada jaringan lokal. Sistem Pengendali ini dirancang untuk dapat mengirimkan paket sinyal suara (*voice*) dari masukan microphone dan kemudian mengirimnya menggunakan protokol ethernet dalam jaringan lokal rumah menggunakan FPGA. FPGA ini diprogram untuk mengirimkan dan mengkodekan paket data, mengkonversi data digital menjadi data analog untuk dapat mengendalikan peralatan dalam rumah. Dari hasil pengujian simulasi menggunakan ISim, terlihat bahwa sistem bekerja secara realtime.

**Kata kunci:** *smart home, suara, FPGA, pengendalian*

### Abstract

*Smart home is a system that uses computers and information technology to control home-like equipment such as windows and lights. The system can be a simple control system to a complex system. Computer / microcontroller based on internet/ethernet network equipped with intelligent system and automation system so as to make home to work automatically. Many computer devices / microcontrollers that can be implemented as a controller in the smart home. Smart home control system in this study using Xilinx xpartan-3e that controls the equipment in the house through LAN (Local Area Networking). This control system communicates using broadcast voice on the local network. The Controller System is designed to be able to transmit a voice signal packet from the microphone input and then send it using the ethernet protocol in the home local network using the FPGA. The FPGA is programmed to transmit and encode data packets, converting digital data into analog data to be able to control the equipment in the home. From the simulation test results using ISIM, it is seen that the system works in realtime.*

**Keywords:** *smart home, voice, FPGA, control*

## 1. PENDAHULUAN

Smart home yang sering disebut sebagai rumah pintar atau eHome adalah suatu rumah yang memiliki sistem otomatis yang sangat canggih untuk mengontrol peralatan rumah seperti pencahayaan dan suhu, peralatan multi-media, memantau dan mengaktifkan alarm serta membuka dan menutup jendela atau pintu dan banyak fungsi lainnya. Sebuah rumah pintar tampak "cerdas" karena sistem komputer yang dapat memantau banyak aspek kehidupan sehari-hari. Misalnya, sebuah lemari es yang dapat mengisi persediaannya sendiri, menyarankan menu, merekomendasikan alternatif yang sehat, dan memesan makanan. Sistem rumah pintar bahkan mungkin mengurus membersihkan

kotak sampah kucing dan menyiram tanaman (Bregman D, 2010).

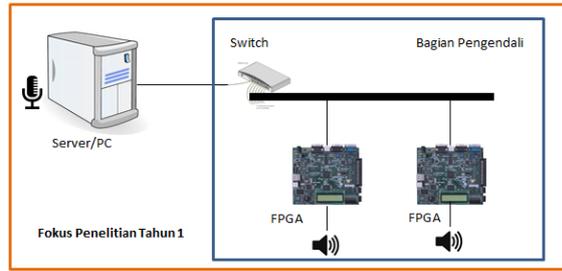
Hampir semua sistem smart home terhubung ke sebuah komputer server, sehingga komputer menjadi sistem monitor dan kendali yang terpusat. Sistem ini biasanya dibekali dengan sistem keamanan berupa sistem autentikasi username dan password (Arifiyanto F etal, 2014). Dengan sistem yang terpusat, pengendalian sistem menjadi lebih efisien dan mudah. Namun jika ditinjau dari segi keamanan, sistem terpusat akan menjadi suatu kelemahan. Sistem autentikasi username dan password menjadi sistem autentikasi yang rawan terbongkar. Sistem autentikasi smart home menggunakan keunikan manusia (suara) akan menjadi gerbang pengendali dalam komunikasi teknologi konvergen.

Jika kita tinjau dari sisi infrastruktur dari sistem yang kompleks akan membutuhkan infrastruktur yang banyak dan mahal. Sebenarnya kebutuhan infrastruktur untuk sistem komunikasi/informasi ini hanya membutuhkan koneksi antar komputer. Namun pada penelitian yang pernah dilakukan, implementasi secara nyata menjadi tidak praktis dan cukup mahal karena harus membeli banyak komputer untuk penerima audio yang bersifat tidak real-time. Sebuah perangkat alternatif yang dapat digunakan untuk menerima paket suara dan pengendali audio eksternal (speaker) adalah Field Programmable Gate Array (FPGA) (Perkins C, 2003). FPGA lebih murah dibandingkan dengan komputer dan hemat daya dan mampu bekerja secara real time (fpga4fun, 2014). Selain itu, dengan menggunakan FPGA untuk menerima dan mengendalikan speaker akan memungkinkan sistem ini bekerja secara independen menggunakan sisa bandwidth jaringan utama (Xilinx, 2011).

Penelitian ini, bertujuan untuk mendesain sistem komunikasi/informasi berbasis Ethernet dengan menggunakan PC dan FPGA Xilinx Spartan-3E. Dengan menggunakan PC pada bagian input dan FPGA bagian penerima, penelitian ini diharapkan akan mampu menunjukkan prinsip-prinsip komunikasi data (VoIP) dan menjelaskan pemahaman perangkat lunak dan perangkat keras yang diimplementasikan sebagai aplikasi embedded sistem untuk komunikasi data. Jaringan menggunakan Ethernet 802.3 standar (IEEE, 2012) yang memungkinkan banyak host untuk mengirim dan menerima data melalui jaringan twisted pair. Penelitian ini memiliki tiga bagian: server/transmitter (PC), backbone LAN, dan penerima (FPGA).

**2. BLOK DIAGRAM SISTEM**

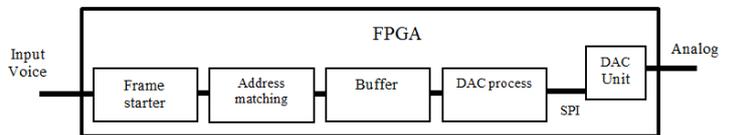
Pada bagian pengendali, sistem ini menggunakan komputer pribadi (PC) yang terhubung ke Local Area Network yang berfungsi sebagai server pusat. Sebuah mikrofon, terhubung ke jack mikrofon dari audio card PC (on board) berfungsi sebagai input. Modul ini menampilkan informasi seperti nama host dari server PC, alamat IP, port dan alamat tujuan (Cisco, 2001) dan total ukuran buffer. FPGA terhubung ke LAN melalui konektor RJ45 (fpga4fun, 2013), bertindak sebagai penerima. FPGA ini akan menghasilkan suara yang nantinya akan berfungsi sebagai pengendali peralatan yang berada dalam rumah. Blok diagram sistem secara keseluruhan dapat dilihat pada gambar 1.



Gambar 1. Blok Diagram Sistem

**3. METODE PENELITIAN**

Paket data suara dikirim pada modul pengendali melalui konektor RJ45 onboard. Modul ini dibagi menjadi empat bagian seperti yang ditunjukkan pada Gambar 2: SFD (Frame Starter), AM (Address Matching), Buffer, dan DAC (Digital Analog Converter).



Gambar 2. Blok Diagram Modul Pengendali

Paket data diterima melalui konektor RJ45. Setelah diterima, modul pengendali akan memeriksa apakah data tersebut valid atau tidak. Jika valid, maka akan mengaktifkan FS.

FS mendeteksi start bit pada frame ethernet. paket ethernet terdiri dari 7 bit. Menurut standar ethernet 802.3, LSB dikirim terlebih dahulu. Untuk memastikan fungsi ini berjalan maka diberikan sebuah counter yang menghitung sebanyak 7 siklus. Setelah FS mendeteksi paket data telah terkirim semua, maka FS mengaktifkan AM.

AM memeriksa apakah informasi paket ditujukan pada modul pengendali. Pencocokan alamat pengirim meliputi alamat MAC, tipe ethernet, jenis protokol, alamat IP dan alamat port. Jika informasi cocok, Buffer menyimpan data dalam paket. Buffer menerima data setiap 4 bit tiap siklus clock sehingga membutuhkan 2 siklus untuk dapat menyimpan data dalam 1 blok RAM. Proses ini berlanjut sampai counter mencapai batas tertentu. Setelah counter mencapai batas tersebut, maka buffer berubah status menjadi disable dan mengaktifkan DAC.

Setelah data disimpan, DAC mengubah data digital menjadi analog untuk memicu speaker eksternal untuk dapat menghasilkan sinyal suara analog. Pseudocode modul FPGA ditunjukkan pada code 1.

**Code 1. Pseudocode Modul Pengendali**

```

FS CODE
if STATE_FS = '1' then
if RX_DATA = "1010" then
COUNT_FS <= COUNT_FS + 1;
elsif RX_DATA = "1011" then
    
```

```

if COUNT_FS = 15 then
ENABLE <= '1';
RESET_FS <= '1';
end if;
COUNT_FS <= 0;
else
COUNT_FS <= 0;
end if;
else
ENABLE <='0';
RESET_FS <= '0';
end if;
-----
AM CODE
MAC Code
if STATE_MAC = '1' then
if COUNT_MAC = 13 then
if RX_MAC = FIX_MAC then
MATCH_MAC <= '1';
end if;
RESET_MAC <= '1';
COUNT_MAC <= 1;
else
MATCH_MAC <= '0';
RESET_MAC <= '0';
RX_MAC <= (others => '0');
end if;

Ethernet Code
if STATE_ETYPE = '1' then
if RECOUNT_ETYPE = '0' then
if COUNT_ETYPE < 11 then
COUNT_ETYPE <= COUNT_ETYPE+1;
else
RECOUNT_ETYPE <= '1';
COUNT_ETYPE <= 1;
end if;
else
if COUNT_ETYPE = 5 then
if RX_ETYPE = FIX_ETYPE then
MATCH_ETYPE <= '1';
end if;
RESET_ETYPE <= '1';
COUNT_ETYPE <= 1;
RECOUNT_ETYPE <= '0';
else
COUNT_ETYPE <= COUNT_ETYPE + 1;
end if;
end if;
else
MATCH_ETYPE <= '0';
RESET_ETYPE <= '0';
RX_ETYPE <= (others => '0');
end if;

Protocol Code
if STATE_PRO = '1' then
if RECOUNT_PRO = '0' then
if COUNT_PRO < 17 then
COUNT_PRO <= COUNT_PRO+1;
else
RECOUNT_PRO <= '1';
COUNT_PRO <= 1;
end if;
else
if COUNT_PRO = 3 then
if RX_PRO = FIX_PRO then
MATCH_PRO <= '1';
end if;
RESET_PRO <= '1';
COUNT_PRO <= 1;

```

```

RECOUNT_PRO <= '0';
end if;
end if;
else
MATCH_PRO <= '0';
RESET_PRO <= '0';
RX_PRO <= (others => '0');
end if;

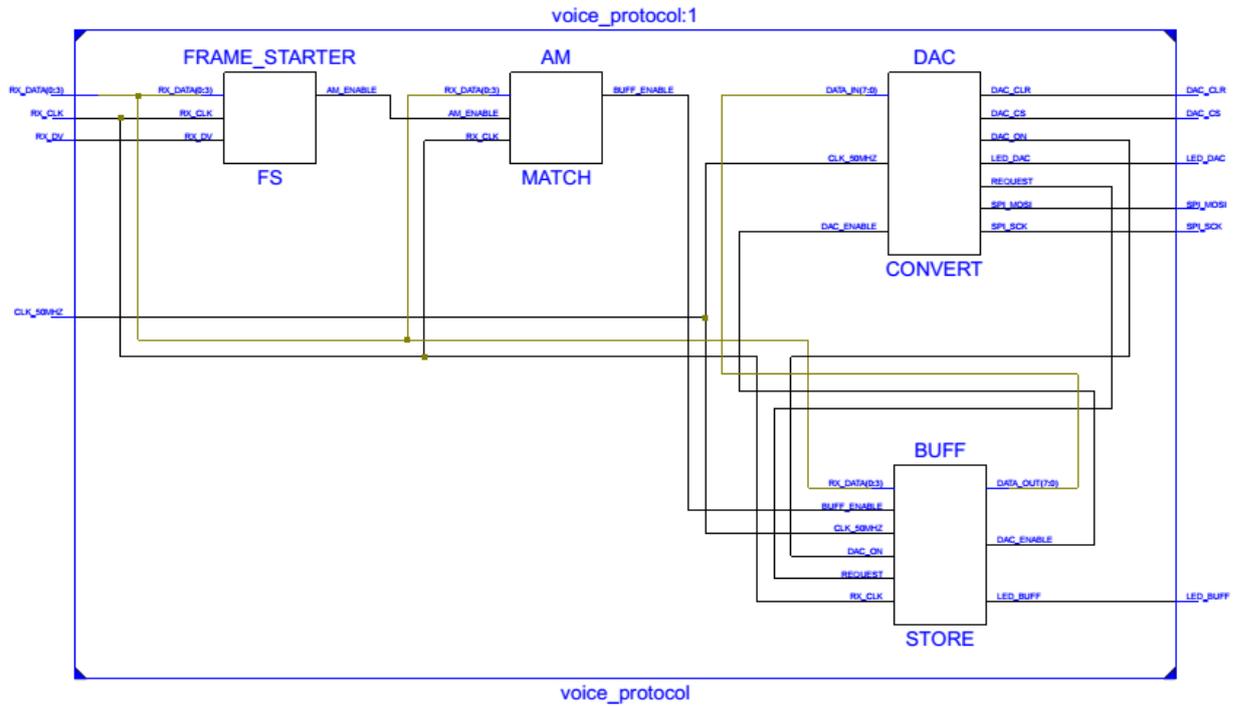
IP Code
if STATE_IP = '1' then
if RECOUNT_IP = '0' then
if COUNT_IP < 11 then
COUNT_IP <= COUNT_IP + 1;
else
RECOUNT_IP <= '1';
COUNT_IP <= 1;
end if;
else
if COUNT_IP = 9 then
if RX_IP = FIX_IP then
MATCH_IP <= '1';
end if;
RESET_IP <= '1';
COUNT_IP <= 1;
RECOUNT_IP <= '0';
end if;
else
MATCH_IP <= '0';
RESET_IP <= '0';
RX_IP <= (others => '0');
end if;
end if;

Port Code
if STATE_PORT = '1' then
if RECOUNT_PORT = '0' then
if COUNT_PORT < 3 then
COUNT_PORT <= COUNT_PORT + 1;
else
RECOUNT_PORT <= '1';
COUNT_PORT <= 1;
end if;
else
if COUNT_PORT = 5 then
if RX_PORT = FIX_PORT then
MATCH_PORT <= '1';
end if;
RESET_PORT <= '1';
COUNT_PORT <= 1;
RECOUNT_PORT <= '0';
end if;
else
MATCH_PORT <= '0';
RESET_PORT <= '0';
RX_PORT <= (others => '0');
end if;
end if;

```

#### 4. HASIL DAN PEMBAHASAN

Modul Pengendali dibangun menggunakan Xilinx Spartan-3E dan diprogram dalam VHDL menggunakan Xilinx ISE 8.2i. Rangkaian RTL modul FPGA ditunjukkan pada gambar 4.



Gambar 4. Rangkaian RTL Modul Pengendali

Gambar 4 menunjukkan bahwa sistem terdiri atas frame starter (FS), Address Matching (AM), Buffer (BUFF) and Digital to Analog Converter (DAC). Deskripsi RTL Schematics dapat dilihat pada table 1 sedangkan rangkuman penggunaan FPGA ditunjukkan pada table 2.

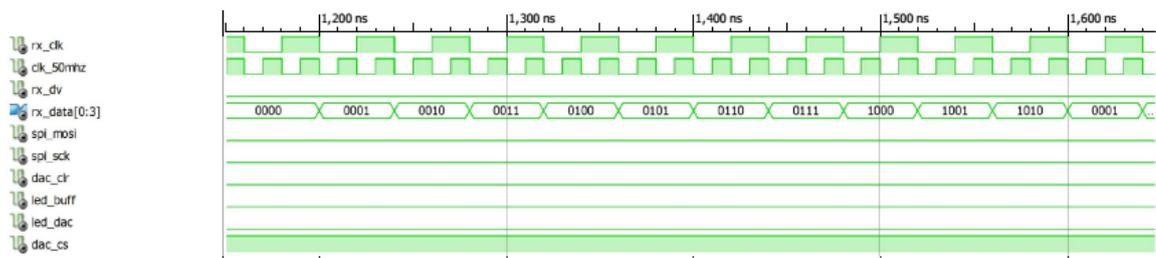
Tabel 1 Deskripsi RTL Schematics

komponen: frame_starter	
masukan:	keluaran:
<ul style="list-style-type: none"> <li>rx_data(0:3)</li> <li>rx_clk</li> <li>rx_dv</li> </ul>	<ul style="list-style-type: none"> <li>am_enable</li> </ul>
komponen: AM	
masukan	keluaran:
<ul style="list-style-type: none"> <li>rx_data(0:3)</li> <li>am_enable</li> <li>rx_clk</li> </ul>	<ul style="list-style-type: none"> <li>buff_enable</li> </ul>
komponen: BUFF	
masukan:	keluaran:
<ul style="list-style-type: none"> <li>rx_data(0:3)</li> <li>buff_enable</li> <li>clk_50mhz</li> <li>dac_on</li> <li>request</li> <li>rx_clk</li> </ul>	<ul style="list-style-type: none"> <li>data_out(7:0)</li> <li>dac_enable</li> <li>led_buff</li> </ul>
komponen: DAC	
masukan:	keluaran:
<ul style="list-style-type: none"> <li>data_in(7:0)</li> <li>clk_50mhz</li> <li>dac_enable</li> </ul>	<ul style="list-style-type: none"> <li>dac_clr</li> <li>dac_cs</li> <li>dac_on</li> <li>led_dac</li> <li>request</li> <li>spi_mosi</li> <li>spi_sck</li> </ul>

penggunaan logika	digunakan	tersedia	penggunaan
jumlah flip-flop	485	9,312	5%
jumlah luts 4 masukan	683	9,312	7%
jumlah slice yang digunakan	575	4,656	12%
total jumlah luts 4 masukan	929	9,312	9%
jumlah ramb16s	16	20	80%
jumlah bufgmuxs	4	24	16%

Program VHDL di simulasikan pada ISim (O.87xd). Simulasi ISim ditunjukkan pada gambar 5.

Tabel 2. Rangkuman Penggunaan FPGA



Gambar 5. Simulasi Sistem keseluruhan menggunakan ISim

## 5. KESIMPULAN

Sistem smart home ini berbasis suara melalui Ethernet. Sistem menggunakan PC sebagai server dan Xilinx Spartan-3E sebagai penerima. Sistem mengirimkan sinyal audio dari PC melalui LAN ke FPGA dan mereproduksi sinyal menggunakan speaker audio eksternal. Berdasarkan pengujian yang dilakukan pada sistem, suara dapat berhasil dikirim ke FPGA. Dari hasil pengujian simulasi menggunakan ISim, terlihat bahawa sistem bekerja secara realtime. Selain itu, menurut device utilization dapat dilihat prosentase penggunaan memori terlihat cukup besar yakni mencapai 80%.

## 6. DAFTAR PUSTAKA

ARIFIYANTO F ETAL, 2014, Perancangan Prototype Web-Based Online Smart Home Controlled By Smartphone, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro.

BREGMAN D, 2010, Smart Home Intelligence - The eHome that Learns, International Journal of Smart Home, Vol.4, No.4.

CISCO, 2001, IP Multicast Technology Overview, [http://www.cisco.com/c/en/us/td/docs/ios/solutions\\_docs/ip\\_multicast/White\\_papers/mcst\\_ovr.html#wp1009849](http://www.cisco.com/c/en/us/td/docs/ios/solutions_docs/ip_multicast/White_papers/mcst_ovr.html#wp1009849)

FPGA4FUN, 2013, How Ethernet works, <http://www.fpga4fun.com/10BASE-T1.html>

FPGA4FUN, 2014, Receiving Packets, <http://www.fpga4fun.com/10BASE-T4.html>

IEEE, 2012, IEEE Standard for Ethernet, IEEE Computer Society

PERKINS C, 2003, RTP Audio and Video for the Internet, Addison Wesley

XILINX, 2011, Spartan-3E Starter Kit Board User Guide, UG230 (v1.0), [www.xilinx.com/support/documentation/boards\\_and\\_kits/ug230.pdf](http://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf)

## SISTEM PENDUKUNG KEPUTUSAN DIAGNOSIS PENYAKIT SAPI POTONG MENGUNAKAN *K- Nearest Neighbour (K- NN)*

Divia Kurnianingtyas<sup>1</sup>, Brillian Aristyo Rahardian<sup>2</sup>, Dyan Putri Mahardika<sup>3</sup>, Amalia Kartika A.<sup>4</sup>, Dwi Angraeni K.<sup>5</sup>

<sup>1,2,3,4,5</sup>Fakultas Ilmu Komputer, Universitas Brawijaya

Email: <sup>1</sup>divakurnianingtyas@gmail.com, <sup>2</sup>brillianarc@gmail.com, <sup>3</sup>dyanputri.dp@gmail.com,  
<sup>4</sup>kartikaamalia95@gmail.com, <sup>5</sup>dwiangraenik59@gmail.com

(Naskah masuk: 15 Maret 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Industri peternakan merupakan salah satu industri yang penting dalam bidang penyediaan nutrisi makanan sehingga pertumbuhan produk ternak bisa menciptakan suatu ancaman kesehatan masyarakat dimana menyebabkan permasalahan kesehatan. Kurangnya pengetahuan peternak sapi potong mengenai berbagai penyakit yang menyerang serta solusi penanganan salah satu alasan manajemen kesehatan ternak dirasa cukup menyulitkan beberapa peternak. Pengembangan sistem pendukung keputusan dengan *K-Nearest Neighbour (K-NN)* untuk mendiagnosis penyakit ini. Data 11 jenis penyakit dapat dikenali oleh sistem pendukung keputusan dan 20 jenis gejala yang dapat dikenali oleh sistem. Hasil pengujian keakuratan 325 data latih dan 11 data uji telah menghasilkan tingkat akurasi 100% dengan nilai  $k = 3$ .

**Kata kunci:** penyakit sapi potong, sistem pendukung keputusan, *K-Nearest Neighbour*

### Abstract

*The livestock industry is one of the important industries in the field of providing food nutrition so that the growth of livestock products can create a public health threat which causes health problems. Lack of knowledge of beef cattle ranchers on various diseases that attack and one way of managing livestock health is considered quite difficult for some breeders. K-Nearest Neighbor (K-NN) Supporting Coding to diagnose the disease. 11 types of disease data can be recognized by decision support systems and 20 types of symptoms that can be recognized by the system. The results of the test accuracy of 325 training data and 11 test data have resulted in 100% accuracy with  $k = 3$ .*

**Keywords:** *cattle cow disease, decision support system, K-Nearest Neighbour*

## 1. PENDAHULUAN

Industri peternakan merupakan salah satu industri yang penting dalam bidang penyediaan nutrisi makanan. Hewan ternak merupakan masalah penting dalam memproduksi sumber pangan hewani. Pertumbuhan pada produk ternak yang sangat pesat merupakan titik kritis untuk menciptakan suatu ancaman kesehatan masyarakat dimana akan menyebabkan permasalahan kesehatan seperti penyakit zoonosis (Mahmoud et al., 2015). Usaha dalam menjaga kesehatan ternak dengan mengontrol secara rutin sehingga dapat mencegah sedini mungkin penyakit menyerang ternak dan memberikan penanganan yang sesuai dengan penyakit ternak merupakan salah satu contoh cara manajemen kesehatan pada ternak sehingga dapat meningkatkan produksi dengan mengoptimalkan hasil produksi dan memaksimalkan keuntungan.

Kurangnya pengetahuan peternak sapi potong mengenai berbagai penyakit yang menyerang hewan ternaknya serta solusi penanganan terhadap penyakit tersebut merupakan salah satu alasan proses manajemen kesehatan pada ternak dirasa cukup menyulitkan beberapa peternak (Swai et al., 2010).

Selain itu, dampak kerugian dapat dirasakan dari biaya pengobatan yang dikeluarkan sangatlah besar meskipun belum ada jaminan kesembuhan pada hewan ternak. (Mdetele et al., 2015). Permasalahannya pengeluaran yang besar sangat mempengaruhi perekonomian (Pohlmeier and Van Eenennaam, 2009). Selain menyebabkan kerugian langsung bagi perekonomian ternak, hal itu juga menyebabkan kerugian tidak langsung dalam hal perdagangan, dampak yang mungkin lebih tinggi dari kerugian langsung (Mdetele et al., 2015). Dalam hal ini peternak sulit untuk memutuskan diagnosis penyakit yang di derita

sapi potong mereka, sehingga diperlukan sistem pendukung keputusan untuk membantu para peternak memutuskan diagnosis. Membangun sistem didasarkan dari data pasien yang disimpan ke dalam database. Sistem pendukung keputusan membantu memperkuat dalam mengambil keputusan, tetapi bisa juga tidak mempengaruhi keputusan yang diambil oleh user akan membantu mempercepat penganalisisan jenis penyakit pada hewan ternak dengan memberikan informasi terkait jenis penyakit yang menjangkit hewan ternak dan solusi penanganannya.

Penggunaan metode K-NN dalam melakukan prediksi pada sistem pendukung keputusan dengan mencari jarak terpendek antar data. Jarak tersebut akan dievaluasi berdasarkan nilai K pada data latihnya. Berdasarkan pengujiannya yang dilakukan secara manual, sistem menghasilkan data yang baik.

## 2. PENYAKIT SAPI POTONG

Sebuah peternakan biasanya berada di daerah terpencil, dengan kondisi sulit untuk mendiagnosis penyakit. Umumnya, sulit untuk mendiagnosis penyakit dengan cepat dan akurat. Tapi peternakan bisa mendiagnosis penyakit hewan dengan cepat dan akurat (Larsen-Freeman and Long, 2014).

Mastitis dan kepincangan merupakan penyakit yang paling sering muncul dan penyakit yang merugikan pada industri peternakan, dalam halekonomi dan kesejahteraan hewan (Miekley et al., 2012).

Jenis Penyakit sapi potong yang digunakan dalam program ini berjumlah 11 penyakit, yaitu Abses, Ascariasis, BEF, Bloat, Endometritis, Entritis, Mastitis, Omphaltis, Pneumonia, Retensio, dan Scabies. Masing masing penyakit mempunyai cara penanggulangan masing-masing. Data penyakit diperoleh dari penelitian (Tyas et al., 2015)

## 3. K-NN

*K-Nearest Neighbor* (KNN) memiliki prinsip kerja dengan mencari jarak terdekat antara K tetangga (*neighbour*) dengan data yang akan di evaluasi terdekat dalam data latih. Algoritma KNN memiliki cara kerja yang sederhana berdasarkan nilai jarak terpendek sehingga dapat ditentukan hasil KNN-nya. Langkah-langkah algoritma KNN:

1. Tentukan parameter K
2. Hitung jarak antara data uji dengan seluruh data latih.
3. Jarak tersebut diurutkan secara *descending*
4. Tentukan jarak terdekat sampai urutan K
5. Cari kelas yang sesuai dengan jarak yang telah diurutkan
6. Cari jumlah kelas terbanyak lalu klasifikasikan data tersebut.

Berikut rumus KNN di Persamaan (1)

$$d_i = \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2}$$

(1)

Keterangan:

$x_1$  = Data *training*

$x_2$  = Data *testing*

$d$  = Jarak

$i$  = Variabel Data

$p$  = Dimensi Data

### 3.1 Siklus Algoritma

- a. Menentukan Nilai k

Langkah pertama yaitu dengan menentukan jumlah tetangga terdekat atau nilai k. Pada penelitian ini diberikan nilai  $k = 3$ .

- b. Menghitung Jarak Euclidean Data Latih

Selanjutnya yaitu menghitung nilai jarak antara data uji dan masing-masing data data latih sebanyak 325 data menggunakan rumus Persamaan (1). Data uji 8 (U8) ditunjukkan detail gejala pada Tabel 1:

**Tabel 1. Gejala Data Uji**

Data	Penyakit	G1	G2	G3	G4	G5
8	OMPHA-LITIS	1	0	0	0	1
		G6	G7	G8	G9	G10
		0	0	0	0	0
		G11	G12	G13	G14	G15
		0	0	0	0	1
		G16	G17	G18	G19	G20
		0	0	0	1	0

Hasil perhitungan dapat dilihat pada Tabel 2

**Tabel 2. Perhitungan Jarak Euclidean**

DATA	U8
L1	1.414213562
L2	2.828427125
L3	2.828427125
L4	2.828427125
...	...
L325	2.44948974

c. Pengurutan Data berdasarkan nilai  $k$

Langkah selanjutnya yaitu mengurutkan data berdasarkan jarak terpendek pada masing-masing data uji terhadap semua data latih. Hasil pengurutan dapat dilihat pada Tabel 3.

**Tabel 3. Pengurutan dan Pengambilan Data Berdasarkan Nilai  $k$**

DATA	U1
L296	0
L297	0
L1	1.41421356

d. Evaluasi

Langkah selanjutnya yaitu menjumlahkan setiap kelas kategori dan hasil terbesar akan dipilih menjadi rekomendasi keputusan. Hasil penentuan kelas pada masing masing data uji ditunjukkan Tabel 4.

**Tabel 4. Penentuan Kelas Data Uji**

Data Uji	$k$	Penyakit
U8	$k = 3$	Omphalitis

#### 4. MANAJEMEN DATA

##### 4.1 Sumber Data

Data pada sistem pendukung keputusan diagnosis penyakit sapi potong dengan metode KNN ini sebagai berikut:

a. Sumber Data Internal

- Daftar Penyakit

Data penyakit yang digunakan pada penelitian ini berjumlah 11 data penyakit yang diperoleh dari penelitian (Tyas et al., 2015). Data penyakit sapi potong ditunjukkan Tabel 5.

**Tabel 5. Daftar Penyakit**

No	Nama Penyakit
----	---------------

1	Abses
2	Ascariasis
3	Bef
4	Bloat
5	Endometritis
6	Enteritis
7	Mastitis
8	Ophalitis
9	Pneumonia
10	Retensio
11	Scabies

- Daftar Gejala

Data gejala yang digunakan pada penelitian ini berjumlah 20 data gejala yang diperoleh dari penelitian (Tyas et al., 2015). Data gejala ditunjukkan pada Tabel 6.

**Tabel 6. Daftar Gejala**

Kode Gejala Klinis	Nama Gejala
G1	Bengkak
G2	Bulu Berdiri
G3	Bulu Kusam
G4	Bulu Rontok
G5	Demam
G6	Diare
G7	Diare Berdarah
G8	Gatal-Gatal
G9	Keluar Ingus
G10	Keluar Lendir pada Vulva
G11	Kembung
G12	Kulit Kasar
G13	Kurus
G14	Mamae Keras
G15	Nafsu Makan Berkurang
G16	Pincang
G17	Plasenta Menjuntai Keluar
G18	Sesak Nafas
G19	Munculnya Belatung pada Puser
G20	Vulva Berbau Busuk

- Data

Masukkan sistem pendukung keputusan berupa data. Data keseluruhan sebanyak 336 data terdiri dari 325 data latih dan 11 data uji

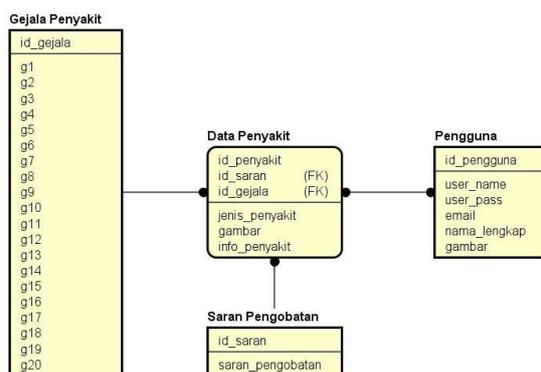
yang diperoleh dari penelitian (Tyas et al., 2015).

#### b. Sumber Data Personal

Data personal yang digunakan dalam sistem pendukung keputusan diagnosis penyakit sapi potong menggunakan metode KNN ini berasal dari pakar yang berasal dari Poskeswan Kab. Nganjuk yang disimpan ke dalam database.

### 4.2 Struktur Basis Data

Struktur basis data sistem pendukung keputusan diagnosis penyakit sapi potong menggunakan metode KNN dapat ditunjukkan Gambar 1.



**Gambar 1. Struktur Basis Data**

Dari Gambar 1 dapat dilihat bahwa sistem pendukung keputusan diagnosis penyakit sapi potong menggunakan metode KNN ini menggunakan 4 tabel utama untuk menyimpan data, yaitu tabel Gejala Penyakit, tabel Data Penyakit, tabel Saran Pengobatan, dan tabel Pengguna.

## 5. PENGUJIAN

### 5.1 Pengujian Akurasi

Uji coba tingkat akurasi ini dilakukan seberapa besar K-NN hasil klasifikasi dalam memberikan rekomendasi diagnosis penyakit sapi. Uji akurasi diperoleh dari persentase kebenaran dengan cara membandingkan jumlah data yang di uji benar dengan banyaknya data keseluruhan sehingga diperoleh suatu Persamaan (2) untuk menghitung nilai akurasi sebagai berikut:

$$AkurasiK - NN = \frac{jumlahdataujibenar}{jumlahsemuadatauji} \times 100\% \quad (2)$$

Contoh pengujian data ke-8 dengan k=3 ditunjukkan pada Tabel 7 berikut:

**Tabel 7. Pengujian Data Ke-1**

DATA	U8	Nama Penyakit
L296	0	Omphalitis
L297	0	Omphalitis
L1	1.41421356	Abses

Data Uji 8 hasil yang diperoleh dari metode K-NN sama dengan hasil yang tersimpan ke dalam database, maka nilai akurasi sebesar 1. Sebaliknya jika hasilnya tidak sama, maka nilai akurasi diberikan nilai 0.

Setelah melakukan pengujian keseluruhan, maka perhitungan akurasi dengan k=3 menggunakan Persamaan (2)

$$AkurasiK - NN = \frac{11}{11} \times 100\% = 100\%$$

Dari hasil perhitungan akurasi di atas, dapat diketahui akurasi 11 data uji dengan k=3 sebesar 100%.

## 6. KESIMPULAN DAN SARAN

### 6.1 Kesimpulan

Sistem pendukung keputusan untuk melakukan diagnosis penyakit sapi potong dengan memberikan masukan berupa gejala pada ternak. Setiap gejala klinis yang dimasukkan ke dalam sistem mempunyai nilai sehingga dapat dibandingkan antara data uji dan data latih menggunakan metode K-NN. Kesimpulan diagnosis diperoleh dari nilai kelas yang memiliki keanggotaan terbesar.

Pengujian sistem yang dilakukan adalah pengujian akurasi. Pengujian akurasi pada 11 data uji memiliki akurasi 100% menggunakan nilai k=3.

### 6.2 Saran

Sistem pendukung keputusan diagnosis penyakit sapi potong memiliki akurasi 100% sehingga tidak memerlukan saran untuk penelitian selanjutnya.

## DAFTAR PUSTAKA

- LARSEN-FREEMAN, D. & LONG, M. H. 2014. *An introduction to second language acquisition research*, Routledge.
- MAHMOUD, H. A., EL HADAD, H. M., MOUSA, F. A. & HASSANIEN, A. E. Cattle Classifications System Using Fuzzy K-Nearest Neighbor Classifier. *Informatics, Electronics & Vision (ICIEV)*, 2015 International Conference on, 2015. IEEE, 1-5.
- MDETELE, D., KASANGA, C., SETH, M. & KAYUNZE, K. 2015. Socio-economic

Impact of Foot and Mouth Disease in Wildlife-Livestock Interface and Non-Interface of Tanzania. *World*, 5, 31-35.

MIEKLEY, B., TRAULSEN, I. & KRIETER, J. 2012. Detection of mastitis and lameness in dairy cows using wavelet analysis. *Livestock Science*, 148, 227-236.

POHLMEIER, B. & VAN EENENNAAM, A. 2009. Potential Effects Of Biotechnology On Animal Health And Well-Being. *Applications of Animal Biotechnology in Animal Health*, 1-5.

SWAI, E. S., SCHOONMAN, L. & DABORN, C. 2010. Knowledge And Attitude Towards Zoonoses Among Animal Health Workers And Livestock Keepers In Arusha And Tanga, Tanzania. *Tanzania Journal of Health Research*, 12, 272-277.

TYAS, R. D. O., SOEBROTO, A. A. & FURQON, M. T. 2015. Pengembangan Sistem Pakar Diagnosa Penyakit Sapi Potong Dengan Metode Fuzzy K-Nearest Neighbour. *Journal of Enviromental Engineering and Sustainable Technology*, 2, 58-66.

## ANALISIS PENENTUAN LOKASI EVAKUASI BENCANA BANJIR DENGAN PEMANFAATAN SISTEM INFORMASI GEOGRAFIS DAN METODE *SIMPLE ADDITIVE WEIGHTING*

(Studi Kasus : Kota Surakarta)

Juliana Andretha Janet Lumban Batu<sup>1</sup>, Charitas Fibriani<sup>2</sup>

<sup>1</sup>Program Studi S1 Sistem Infomasi, Universitas Kristen Satya Wacana

<sup>2</sup>Program Studi S2 Sistem Infomasi, Universitas Kristen Satya Wacana

Email: <sup>1</sup>682013074@student.uksw.edu, <sup>2</sup>charitas.fibriani@staff.uksw.edu

(Naskah masuk: 29 Maret 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Kota Surakarta merupakan kota dengan ancaman banjir yang cukup besardan kebanyakan diantaranya merupakan genangan, mengingat bahwa kondisi geografis kota Surakarta yaitu salah satunya dikelilingi oleh bantaran sungai bengawan solo. Tingginya curah hujan sewaktu-waktu dapat menyebabkan aliran sungai tidak dapat menahan debit air sehingga menyebabkan meluapnya air dan menyebabkan banjir. Upaya mitigasi dengan cara memberi peringatan dini kepada masyarakat dirasa dapat menjadi salah satu upaya dalam menekan angka kerugian sebelum bencana banjir datang. Dengan pemanfaatan Sistem Informasi Geografis serta metode *Simple Additive Weighting* dalam menentukan lokasi evakuasi bencana banjir maka diharapkan ketika banjir datang maka masyarakat kota Surakarta dapat memiliki persiapan lebih dini. Hal ini juga dapat membantu pihak pemerintahan terkhususnya pihak Badan Penanggulangan Bencana Daerah (BPBD) dalam melakukan evakuasi ketika banjir datang.

**Kata kunci:** *sistem informasi geografis, simple additive weighting, mitigasi, lokasi evakuasi, surakarta.*

### Abstract

*Surakarta is a city with a fairly high threat of flooding and most of them are puddles in some areas, considering that one of the geographical conditions of Surakarta was said that Surakarta surrounded by riverbank of bengawan solo. The river flow could not hold the water flow whenever heavy rainfall come and could possibly causing an overvlow and cause of flooding. The attempts of mitigation by giving early warning to the public is considered to be one attempts to reduce the number of losses before the flood came. With the use of Geographic Information Systems and Simple Additive weighting method to determine the great location of shelter then the people of the city of Surakarta could have early preparation before the floods come as it expected. This attempts can also help the government especially to the Badan Penanggulangan Bencana Daerah (BPBD) in attempt of the evacuation when the flood came.*

**Keywords:** *geographic information system, simple additive weighting, mitigation, shelter, surakarta.*

## 1. PENDAHULUAN

Banjir adalah salah satu bencana alam yang disebabkan oleh beberapa faktor alam. Faktor alam yang dimaksud diantaranya dapat berupa curah hujan yang tinggi, kelerengan tanah yang lebih rendah dari permukaan air laut maupun dikarenakan tanggul ataupun aliran sungai yang tidak mampu menahan debit air hujan. Rendahnya kesadaran manusia dalam menjaga lingkungan juga dapat dijadikan salah satu pemicu terjadinya banjir termasuk di Indonesia. (Rosyidie, 2013).

Kota Surakarta sendiri saat ini banjir sudah menjadi catatan penting di 21 kelurahan dan terdapat 6 sungai yang harus diwaspadai berdasarkan pemaparan dari Badan Penanggulangan Bencana Daerah (BPBD) kota Surakarta. Seperti pada tanggal 28-30 November 2016 banjir melanda beberapa

wilayah di Surakarta salah satu diantaranya di kelurahan Sangkrah dan kelurahan Semanggi, Pasar Kliwon yang dikarenakan luapan sungai Bengawan Solo yang akhirnya mengharuskan warga di lokasi tersebut untuk menempati pengungsian di daerah bantaran sungai Bengawan Solo. ([www.surakarta.go.id](http://www.surakarta.go.id)).

Menurut Ketua Peneliti Balitek Daerah Aliran Sungai (DAS) Bengawan Solo yaitu Endang Savitri, beliau mengatakan bahwa banjir luapan yang terjadi dikarenakan rusaknya atau kurang berfungsinya wilayah resapan air terutama di daerah hulu (Wonogiri hingga Ngawi). Iklim yang saat ini tidak menentu juga dapat dijadikan salah satu alasan terjadinya banjir dikarenakan curah hujan yang tidak pada masanya sehingga kurangnya mitigasi pada saat musim yang tidak termasuk musim penghujan seperti pada banjir di Kota Surakarta pada bulan Juni 2016 silam. Cepatnya rambatan air dan lambannya

informasi antisipasi bencana banjir dapat menyebabkan kurang maksimalnya penekanan dampak dari banjir itu sendiri. Salah satu tindakan antisipasi yang dapat dilakukan sebelum bencana banjir datang adalah dengan menentukan lokasi evakuasi atau tempat singgah untuk pertolongan bencana banjir dan di sebarkan kepada masyarakat luas demi untuk mempercepat proses evakuasi korban bencana sehingga dapat meminimalisir kerugian dari banjir itu sendiri. Hal tersebut didukung oleh pemaparan Gatot Sutanto selaku Kepala Pelaksana Harian BPBD Surakarta yang menyebutkan bahwa penanganan banjir yang terjadi di kota Surakarta pada bulan Juni 2016 masih kurang terantisipasi baik masyarakat maupun pemerintah. (Solopos, 2016).

Berkaitan dengan pemaparan latar belakang tersebut, maka akan dimanfaatkan Sistem Informasi Geografis (SIG) dengan menggunakan fungsi *Overlay* dan *Buffer* serta dilanjutkan dengan perhitungan menggunakan metode *Simple Additive Weighting* (SAW) yang diharapkan dapat menentukan lokasi evakuasi terbaik agar dapat dijadikan pertimbangan serta membantu instansi terkait terkhususnya BPBD dalam rangka meningkatkan mitigasi banjir di kota Surakarta.

Output yang nantinya akan dihasilkan adalah hasil analisa yang berupa peta informasi lokasi evakuasi saat banjir tiba di kota Surakarta.

## 2. KAJIAN PUSTAKA

Penelitian terdahulu yang digunakan sebagai salah satu acuan penelitian ini adalah berjudul “Penentuan Jalur Evakuasi, Tempat Evakuasi Sementara (TES) Beserta Kapasitasnya Di Kota Pariaman Dengan Sistem Informasi Geografis (SIG)” yang ditulis oleh Dini Purbani (2015). Penelitian ini dibuat untuk menentukan Jalur Evakuasi, Tempat Evakuasi Sementara (TES) Beserta Kapasitasnya dengan pemanfaatan Sistem Informasi Geografis (SIG) dalam rangka melakukan usaha mitigasi di kota Pariaman yang merupakan daerah pesisir Sumatera Barat yang dekat dengan lokasi sumber gempa dan tsunami. Usaha mitigasi tersebut yaitu berupa pembuatan Tempat Evakuasi Sementara (TES) di daerah pesisir yang dapat dijangkau dengan cepat oleh masyarakat ketika tsunami datang. Penentuan jalur tercepat juga dianalisa mengingat pentingnya efektifitas jangkauan menuju lokasi evakuasi dikarenakan tsunami yang datang dengan cepat. Penelitian ini bertujuan untuk mengupayakan mitigasi diatas dengan pemanfaatan SIG. Secara konseptual penelitian terdahulu memiliki beberapa kesamaan dengan penelitian ini diantaranya memiliki kesamaan dalam upaya mitigasi, pemanfaatan SIG sebagai metode pengolahan data spasial untuk pengambilan keputusan terkait lokasi evakuasi. Namun disisi lain memiliki perbedaan diantaranya pada penelitian terdahulu lebih terfokus kepada

daerah yang terklasifikasi sebagai daerah rawan bencana sedangkan pada penelitian ini dilakukan dikelurahan kota Surakarta. Perbedaan lainnya adalah penelitian terdahulu sudah melakukan perhitungan terkait jalur tercepat yang dapat dilewati, namun penelitian ini masih terfokus kepada penentuan lokasi evakuasi saja.

Penelitian lainnya berjudul “Pemetaan Wilayah Risiko Bencana Banjir Kabupaten Kudus Berdasarkan Aspek Ancaman, Kerentanan, dan Kapasitas Berbasis Sistem Informasi Geografis” yang bertujuan untuk menganalisis dan memetakan daerah resiko banjir di Kabupaten Kudus yang nantinya digunakan sebagai acuan untuk melakukan tindakan mitigasi yang diantaranya yaitu penanggulangan dan pengurangan resiko banjir di Kabupaten Kudus. Penelitian ini memanfaatkan SIG dalam poses pemetaan resiko banjir. Hal yang sama yang dilakukan pula dengan penelitian ini ialah analisis menggunakan analisis spasial yang diantaranya yaitu *overlay* dan *buffer* dalam pengambilan keputusan. Penelitian terdahulu ini dilakukan pula penentuan parameter terkait apa saja yang berkaitan dengan kerentanan bencana banjir, konsep serupa dilakukan dalam penelitian ini yaitu menentukan parameter sesuai dengan hal-hal yang berkaitan dengan kesesuaian lokasi evakuasi banjir. (Chernovita, 2013).

Banjir adalah salah satu bencana alam yang mana terjadi dikarenakan meluapnya volume air yang mengakibatkan meluapnya air ke daratan maupun suatu daerah. (UU Nomor 24 Tahun 2007). Meluapnya debit air dapat dikarenakan tingginya curah hujan sehingga tidak dapat dibendung oleh sungai maupun tanggul.

Ahli hidrologi banjir di Indonesia membagi banjir menjadi 3 jenis diantaranya: akibat dari peluapan sungai yang mana aliran air tidak dapat menampung jumlah debit air yang kemudian menyebabkan air akan turun ke daerah dataran banjir. Kemudian terdapat banjir lokal yang terjadi akibat jumlah debit air yang tinggi dan kondisi tanah yang sulit melakukan penyerapan air. Jenis terakhir ialah banjir akibat pasang surut air laut yang terjadi karena tinggi permukaan air laut meningkat yang menyebabkan kondisi aliran air pada muara sungai berbeda seperti saat kondisi laut surut dikarenakan aliran sungai menjadi lambat sehingga aliran sungai tidak dapat menahan debit air dan akan mengalir ke luar sungai.

Kerugian yang dapat disebabkan oleh bencana banjir sendiri dapat berupa terancamnya keselamatan masyarakat baik secara fisik maupun psikis serta kerugian material lain. (Chernovita, 2013).

Perlunya upaya mitigasi banjir untuk mencegah atau mengurangi resiko dari bencana tersebut. Mitigasi sendiri adalah upaya penekanan dampak bencana dengan upaya meniadakan masyarakat

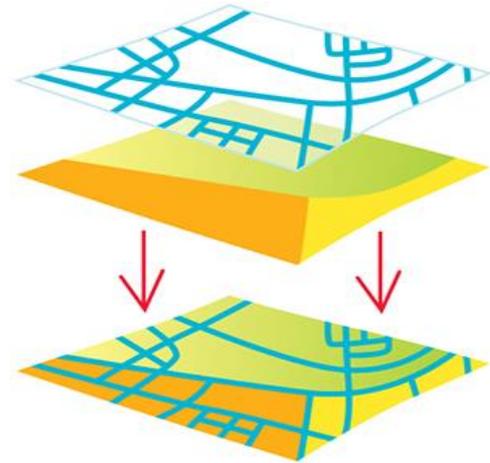
akan bahaya bencana. (Undang-undang Republik Indonesia Nomor 2 Tahun 2007 Tentang Penanggulangan Bencana). Kegiatan mitigasi tersebut dapat dilakukan dengan melalui pelaksanaan penataan ruang, pengaturan pembangunan, pembangunan infrastruktur, tata bangunan dan penyelenggaraan pendidikan, penyuluhan, dan pelatihan baik secara konvensional maupun modern. (Undang-undang Republik Indonesia Pasal 47 Nomor 2 Tahun 2007 Tentang Penanggulangan Bencana). Salah satu upaya yang dapat dilakukan dalam tahap kesiapsiagaan sebelum terjadinya banjir ialah dengan menyiapkan tempat pengungsian yang diperkirakan aman dari banjir dan mampu menampung masyarakat yang akan mengungsi bila banjir terjadi. (Ilmu Geografi, 2016).

Dengan melakukan mitigasi bencana maka diharapkan dapat mengurangi tingkat resiko bencana yaitu seperti memperkecil ancaman bencana, mengurangi kerentanan kawasan terancam dan meningkatkan kapasitas kawasan terancam.

Sistem Informasi Geografis merupakan suatu sistem dengan basis komputer yang dapat mengolah data geografis yaitu dengan memasukkan data, mengolah data (menyimpan dan mengambil data), manipulasi dan analisis data, serta keluaran sebagai hasil akhir (*output*). Hasil akhir (*output*) akan menjadi salah satu pengambil keputusan dalam kasus tertentu terkait geografis. (Chernovita, 2013).

SIG dapat dibagi menjadi dua klasifikasi yaitu manual dan otomatis. Letak perbedaannya dapat terlihat dari teknis pengelolaannya yang mana pada manual pengerjaannya yaitu dengan menggabungkan data berupa peta, media lembar transparan untuk tumpang susun (*overlay*) seperti pada Gambar 1. Seluruh data tersebut dapat diolah dan dianalisis secara manual tanpa menggunakan komputer. Berbeda halnya dengan sistem otomatis yang dapat mengolah data yang telah melalui proses digitasi. sumber data dapat berupa citra satelit maupun foto udara yang telah terdigitasi serta peta tematik yang telah terdigitasi pula. (Tanaamah, 2013).

Analisis overlay sendiri merupakan operasi spasial yang terkomputerisasi yang terjadi pada dua atau lebih *layer* peta yang telah terkoordinat. Dalam fungsi *geoprocessing*, titik perpotongan geometris dapat digunakan untuk menggabungkan, memodifikasi maupun memperbaharui fitur pada hasil dataset peta. (ESRI).



Gambar 1. *Overlay Map Layer* (ESRI)

Metode *Simple Additive Weighting* (SAW) dapat pula disebut sebagai metode penjumlahan terbobot. Pada dasarnya metode SAW digunakan untuk mencari penjumlahan terbobot dari *rating* kinerja pada setiap alternatif di setiap kriteria yang ada. Alternatif yang ada dihauskan untuk melalui poses normalisasi matriks keputusan ( $X$ ) dengan berdasarkan skala yang berasal dari bobot masing-masing atribut. Berikut merupakan langkah penyelesaian dengan menggunakan metode SAW :

1. Menentukan  $C_i$  yang merupakan kriteria saja yang akan dijadikan acuan dalam pengambilan keputusan.
2. Menentukan *rating* kecocokan setiap alternatif pada setiap kriteria.
3. Membuat matriks keputusan berdasarkan ( $C_i$ ), kemudian kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut keuntungan ataupun atribut biaya) sehingga diperoleh matriks ternormalisasi  $R$ . rumus yang digunakan untuk melakukan normalisasi adalah seperti Persamaan (1) berikut:

$$R_{ij} = \begin{cases} \frac{X_{ij}}{\text{Max}_i X_{ij}} & \text{Jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\text{Min}_i X_{ij}}{X_{ij}} & \text{Jika } j \text{ adalah atribut biaya (cost)} \end{cases} \quad (1)$$

Dengan keterangan:

- $R_{ij}$  = *Rating* kinerja ternormalisasi  
 $\text{Max}_{ij}$  = nilai maksimal dari setiap baris dan kolom  
 $\text{Min}_{ij}$  = nilai minimal dari setiap baris dan kolom

$X_{ij}$  = baris dan kolom dari matriks

4. Hasil akhir diperoleh dari proses perangkangan yaitu penjumlahan dari perkalian matriks

ternormalisasi R dengan vektor bobot sehingga diperoleh nilai terbesar yang dipilih sebagai alternatif terbaik (Ai) sebagai solusi.  $r_{ij}$  adalah rating kinerja ternormalisasi dari alternatif Ai pada atribut  $C_j$ ;  $i = 1, 2, \dots, m$  dan  $j = 1, 2, \dots, n$ . Nilai preferensi untuk setiap alternatif ( $V_i$ ) diberikan sebagai rumus pada Persamaan(2) berikut :

$$V_i = \sum_{j=1}^n W_j r_{ij} \tag{2}$$

Dengan keterangan:

$V_i$  = nilai akhir dari alternatif

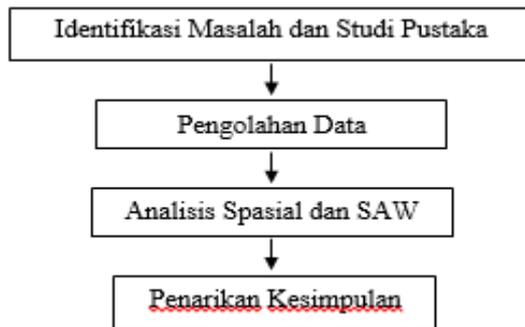
$W_j$  = bobot yang telah ditentukan

$R_{ij}$  = normalisasi matriks

Nilai  $V_i$  yang lebih besar mengindikasikan bahwa alternatif Ai lebih terpilih.

### 3. METODOLOGI

Pada bab ini akan menjelaskan mengenai metode penelitian ini dilaksanakan. Gambar 4 dibawah ini akan menjelaskan tahapan metode penelitian dalam pencapaian tujuan dari penelitian.



Gambar 2. Tahapan Penelitian

#### 3.1 Identifikasi Masalah dan Studi Pustaka

Tahapan penelitian ini dimulai dengan mengidentifikasi masalah yang terjadi di Kota Surakarta yang mana dilakukan dengan metode wawancara kepada pihak BPBD Surakarta mengenai permasalahan banjir beserta upaya mitigasinya. Hasil wawancara tersebut diperoleh temuan-temuan serta batasan dari penelitian ini. Hasil wawancara tersebut menjelaskan mengenai kriteria Kota Surakarta secara geografis dalam hal bencana banjir serta variabel apa saja yang dapat menjadikan suatu lokasi dikatakan layak untuk dijadikan lokasi *shelter* sesuai dengan kriteria secara geografis kota Surakarta.

Studi pustaka dilakukan untuk penggunaan literatur yang berkaitan dengan penelitian ini diantaranya studi pustaka mengenai banjir, mitigasi, sistem informasi geografis, serta kriteria dari kota Surakarta terkait bencana banjir.

#### 3.2 Pengolahan Data

Terdapat dua jenis data dalam penelitian ini diantaranya data primer dan sekunder. Data primer didapatkan melalui wawancara dengan pihak BPBD terkait dengan kriteria kota Surakarta secara geografis mengenai bencana banjir serta persyaratan suatu lokasi dikatakan layak sebagai lokasi evakuasi bencana banjir. Persyaratan tersebut diantaranya berupa :

- Aman/relatif aman dari kemungkinan terjadi bencana serupa.
- Letaknya strategis mudah dijangkau oleh umum dengan dukungan komunikasi dan akses jalan, serta transportasi memungkinkan.
- Dekat dengan sumber air dan tersedianya MCK.
- Tidak bermasalah dalam penggunaan dan pemanfaatan atas status tanah.
- Mendekatkan kepada fasilitas umum dan social lainnya.

Persyaratan tersebut masih harus disesuaikan dengan kondisi riil dari Kota Surakarta agar dapat dimodelkan untuk pengambilan data sekunder selanjutnya. Sedangkan data sekunder yang digunakan terdiri dari beberapa peta yang diterjemahkan melalui data primer. Data tersebut adalah berupa Peta Administrasi, Peta Bencana Banjir, Peta Tata Guna Lahan, Peta Curah Hujan Kota Surakarta. Peta tersebut telah memiliki atribut yang berisi *field* data terkait informasi sesuai dengan masing-masing peta. Data-data tersebut digunakan untuk pengambilan keputusan dalam *overlay* dan metode SAW.

#### 3.3 Analisis Spasial dan SAW

Data sekunder kemudian diolah menggunakan analisis spasial dengan fungsi *Overlay* dan *Buffer* pada peta tertentu yang menghasilkan data temuan baru yang selanjutnya diolah menggunakan metode SAW. Berikut merupakan Tabel 1 menjelaskan mengenai kriteria lokasi yang digunakan berdasarkan penyesuaian dengan data primer :

Tabel 1. Kriteria lokasi berdasarkan data primer.

Kriteria	Penilaian
Bencana Banjir dan Serupa	Bukan daerah rawan bencana banjir dan longsor.
Jarak Jalan	Semakin dekat dengan akses jalan maka semakin baik
Jarak Pemukiman	Semakin dekat dengan daerah permukiman maka semakin baik.
Jarak DAS	Semakin jauh dengan daerah aliran sungai maka semakin baik.
Tata Guna Lahan (Landuse)	-Merupakan lahan terbuka/lapangan.

-Meupakan bangunan milik pemerintah kota (dalam hal ini dapat berupa kantor kelurahan).

-Merupakan Gedung Sekolah Maupun Bangunan Keagamaan (Gereja, Masjid dll.)

Curah Hujan           Semakin rendah curah hujan maka semakin baik.

Data sekunder yang ditemukan kemudian dicocokkan dengan kriteria yang ada. Table 2 akan menjelaskan mengenai data sekunder yang mewakili kriteria yang ada :

Tabel 2. Relasi Kriteria dan Data Sekunder.

Kriteria	Data Sekunder
Bencana Banjir dan Serupa	Peta Bencana Banjir
Jarak Jalan	Peta Ruas Jalan
Jarak Pemukiman	Peta Permukiman
Jarak DAS	Peta Aliran Sungai
Tata Guna Lahan ( <i>Landuse</i> )	Peta <i>Landuse</i>
Curah Hujan	Peta Curah Hujan

Didalam SAW sendiri dilakukan pembobotan pada masing-masing kriteria yang digunakan untuk melakukan penjumlahan terbobot pada perhitungan SAW. Hasil pembobotan telah disesuaikan dengan data primer yaitu hasil wawancara dengan pihak BPBD Kota Surakarta serta BMKG kota Semarang. Pembobotan akan lebih di jelaskan pada Tabel 3-8 berikut :

Tabel 3. Pembobotan kriteria Bencana Banjir dan Serupa.

No	Kelas	Bobot
1	Rendah	4
2	Sedang	3
3	Tinggi	2
4	Sangat Tinggi	1

Tabel 4. Pembobotan kriteria Jarak Jalan.

No	Jarak Buffer	Bobot
1	0 - 500m	4
2	500 - 1000m	3
3	1000 - 2000m	2
4	2000 - 2500m	1

Tabel 5. Pembobotan kriteria Jarak Pemukiman.

No	Jarak Buffer	Bobot
----	--------------	-------

1	0 - 10m	4
2	10 - 250m	3
3	250 - 500m	2
4	500 - 1000m	1

Tabel 6. Pembobotan kriteria Jarak DAS.

No	Kelas	Jarak Buffer	Bobot
1	Aman	250 m - 500 m	4
2	Agak rawan	150 m - 250 m	3
3	Rawan	50 m - 100 m	2
4	sangat rawan	0 - 50 m	1

Tabel 7. Pembobotan kriteria Tata Guna Lahan (*Landuse*).

No	Kelas	Bobot
1	Lahan Terbuka (Lapangan & Tanah Terbuka)	4
2	Bangunan Pemerintah Kota (Kelurahan)	3
3	Sekolah	2
4	Tempat Ibadah	1

Tabel 8. Pembobotan kriteria Curah Hujan.

No	Kelas	Bobot
1	151 - 200 mm	4
2	201 - 300 mm	3
3	301 - 400 mm	2
4	401 - 500 mm	1

Berdasarkan pembobotan kriteria pada tabel diatas, langkah selanjutnya adalah menentukan *range* skor kelayakan lokasi evakuasi. Untuk itu diperlukan penjumlahan skor total pada hasil proses analisis spasial *overlay* yang telah dilakukan sebelumnya dan menentukan total skor terendah dan tertinggi untuk menentukan *range* skor kelayakan lokasi. Total skor akan diklasifikasikan menjadi lima himpunan kelayakan yaitu seperti pada tabel 8 berikut :

Tabel 9. *Range* skor kelayakan lokasi.

Skor Total	Kelayakan
21-25	Sangat Layak
16-20	Layak
11-15	Cukup
6-10	Kurang Layak
1-5	Tidak Layak

Berdasarkan *range* skor kelayakan diatas maka data hasil *overlay* yang berjumlah 8704 data disortir kembali. Data yang akan digunakan dalam perhitungan SAW merupakan data yang termasuk dalam *range* kelayakan dengan “Sangat Layak” yaitu dengan skor total 21-25 dengan jumlah temuan data sebanyak 83 data.

Kriteria akan diwakilkan oleh  $C_i$  dalam perhitungan SAW. Masing-masing kriteria akan diberikan bobot sesuai dengan data primer yang ada. Berikut pada tabel 10 merupakan pemaparan pembobotan tiap kriteria :

Tabel 10. Pembobotan tiap kriteria untuk pengambilan keputusan.

$C_{ij}$	Kriteria	Bobot
C1	Bencana Banjir dan Serupa	25%
C2	Jarak Terhadap Jalan Umum	15%
C3	Jarak Terhadap Pemukiman	10%
C4	Jarak Terhadap DAS	20%
C5	Curah Hujan	15%
C6	Landuse	15%

### 3.4 Penarikan Kesimpulan

Temuan akhir dari perhitungan dengan metode SAW tersebut merupakan kesimpulan dari keseluruhan penelitian yang mana akan menentukan lokasi evakuasi terbaik berdasarkan kriteria yang telah ditentukan.

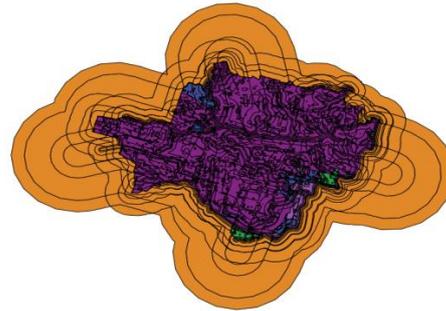
## 4. ANALISIS DAN PEMBAHASAN

Penelitian ini akan menghasilkan peta dengan informasi mengenai lokasi evakuasi terbaik yang digunakan ketika banjir terjadi di Kota Surakarta. Peta tersebut dihasilkan melalui analisis spasial dengan pemanfaatan SIG dan metode SAW dengan menggunakan data primer serta sekunder.

### 4.1 Analisis Spasial

Sebelum memasuki tahap *overlay* dan *buffer*, terlebih dahulu peta harus dikonversikan kedalam peta digital dalam bentuk *shapefile* yang kemudian disesuaikan proyeksinya agar dapat dilakukan proses *overlay* dan *buffer*. Berikut merupakan proses *overlay* dari setiap peta yang menjadi variabel pengambilan keputusan.

Peta yang diproses melalui *overlay* adalah keseluruhan data sekunder yang telah dijelaskan pada Tabel 2. Hasil *overlay* adalah berupa peta dengan atribut baru sebagai alternatif calon lokasi evakuasi. Gambar 4 akan menggambarkan hasil *overlay* setiap layer peta :



Gambar 3. Peta hasil akhir dari proses overlay Peta Bencana Banjir, Ruas Jalan, Permukiman, Aliran Sungai, Landuse dan Curah Hujan.

Peta hasil proses *overlay* tersebut memiliki data terkait informasi calon lokasi evakuasi yang akan disortir kembali untuk kemudian dilakukan operasi metode SAW untuk pembobotan. Berikut data dari peta hasil *overlay* pada Gambar 5.

Attribute table - skor\_landuse - Features total: 5049, filtered: 5049, selected: 0

	SKOR_RAIWAN	SKOR_JALAN	SKOR_PERMU	SKOR_SUNGA	SKOR_UJIAN	SKOR_SHELT	lokasi	total_skor
0	2	3	4	4	1	0	1	14
1	2	4	4	1	1	4	2	16
2	2	4	4	1	1	2	3	14
3	2	4	4	1	1	0	4	12
4	2	4	4	1	1	2	5	14
5	2	4	4	1	1	2	6	14
...								
...								
...								
8704	0	0	0	4	0	0	8705	4

Gambar 4. Data atribut hasil dari proses *overlay*.

Data hasil *overlay* pada Gambar 5 merupakan data masing-masing *layer* peta sesuai dengan masing-masing peta pada Tabel 2 yang telah di *overlay* yang menghasilkan 8704 data baru yang menjadi alternatif calon lokasi evakuasi. Dari data tersebut kemudian diambil alternatif dengan skor kelayakan “Sangat Layak” yaitu dengan total skor 21-25 sehingga ditemukan data alternatif sebanyak 83 data. Data tersebut selanjutnya akan di analisis kembali dengan menggunakan metode SAW untuk kemudian melihat alternatif yang paling sesuai sebagai lokasi evakuasi.

### 4.2 Analisis Metode Simple Additive Weighting

Data dengan jumlah 83 data yang telah ditemukan melalui analisis *overlay* tersebut kemudian dianalisis kembali dengan metode SAW guna menjumlahkan secara terbobot atribut tersebut untuk menemukan alternatif terbaik. Kriteria ( $C_i$ ) yang akan dijadikan acuan sebelumnya telah disesuaikan pada proses sebelumnya. Berikut merupakan Tabel 1 berisi 10 dari 83 data hasil *overlay* yang kemudian digunakan sebagai alternative kriteria yang telah disesuaikan :

Tabel 11. Alternatif Kriteria.

Alternatif	C1	C2	C3	C4	C5	C6
Lokasi 713	4	4	4	4	2	3
Lokasi 716	4	4	4	4	2	3
Lokasi 725	4	4	4	4	2	3
Lokasi 738	4	4	4	4	2	3
Lokasi 858	4	4	4	4	2	3
Lokasi 639	4	4	4	3	2	4
Lokasi 649	4	4	4	4	2	4
Lokasi 693	4	4	4	4	1	4
Lokasi 957	4	4	4	3	2	4
Lokasi 961	4	4	4	3	2	4
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
Lokasi 3291	4	3	4	4	2	4

Kemudian Kriteria alternatif dinormalisasi (R) berdasarkan kriteria (C<sub>i</sub>) dengan menyesuaikan jenis atribut apakah berjenis keuntungan atau biaya dan menyesuaikan dengan rumus yang belaku sesuai dengan jenis atribut. Normalisasi hanya dilakukan pada data dengan Berikut merupakan Tabel 2 hasil normalisasi 10 dari 83 data alternatif :

Tabel 12. Normalisasi matriks (R).

Alternatif	C1	C2	C3	C4	C5	C6
Lokasi 713	1	0.75	0.75	1	0.50	0.75
Lokasi 716	1	0.75	0.75	1	0.50	0.75
Lokasi 725	1	0.75	0.75	1	0.50	0.75
Lokasi 738	1	0.75	0.75	1	0.50	0.75
Lokasi 858	1	0.75	0.75	1	0.50	0.75
Lokasi 639	1	0.75	0.75	0.75	0.50	1
Lokasi 649	1	0.75	0.75	1	0.50	1
Lokasi 693	1	0.75	0.75	1	1	1
Lokasi 957	1	1	1	1	1	1
Lokasi 961	1	1	1	1	1	1
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
Lokasi 3291	1	1	0.75	1	0.5	1

Hasil normalisasi matriks (R) diatas kemudian digunakan untuk melakukan perankingan bobot yang telah ditentukan sesuai dengan keputusan yang dijelaskan pada pada Tabel 10. Hasil dari perkalian tersebut akan menghasilkan nilai akhir dari alternatif (V<sub>i</sub>). Nilai terbesar yang akan dipilih sebagai alternatif terbaik (A<sub>i</sub>) yang dianggap sebagai lokasi ideal untuk dijadikan lokasi evakuasi bencana banjir

di kota Surakarta. Data dengan alternatif terbaik yang akan dipaparkan pada Tabel 4 yang kemudian kata Lokasi akan di singkat menjadi L dan Alternatif menjadi A.

Tabel 13. Alternatif tebaik (A<sub>i</sub>).

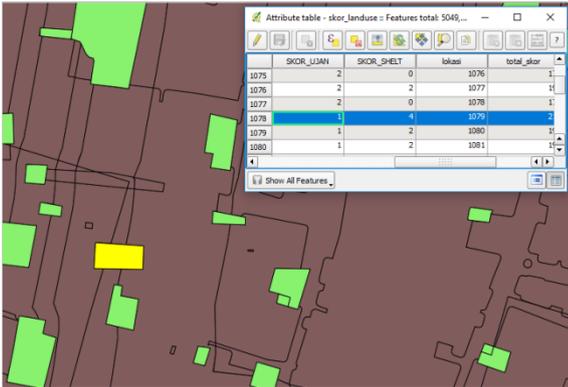
A	C1	C2	C3	C4	C5	C6	V
L1079	1	0.75	0.75	1	1	1	0.94
L1084	1	0.75	0.75	1	1	1	0.94
L1085	1	0.75	0.75	1	1	1	0.94
L1165	1	0.75	0.75	1	1	1	0.94
L1169	1	0.75	0.75	1	1	1	0.94
L1585	1	0.75	0.75	1	1	1	0.94

Alternatif yang didapat memiliki 6 kriteria yang sama di masing-masing 6 data yaitu diantaranya dijelaskan pada Tabel 15:

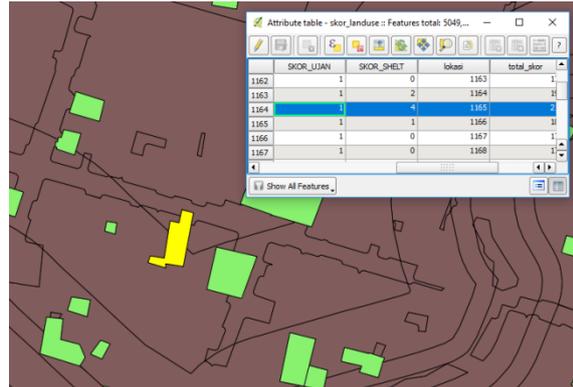
Tabel 14. Alternatif 1

Kriteria	Sub Kriteria
Bencana Banjir dan Serupa	Rendah
Jarak Terhadap Jalan Umum	0 - 500m
Jarak Terhadap Pemukiman	0 - 10m
Jarak Terhadap DAS	Aman (250 - 500m)
Curah Hujan	401 - 500 mm
Landuse	Lahan Terbuka

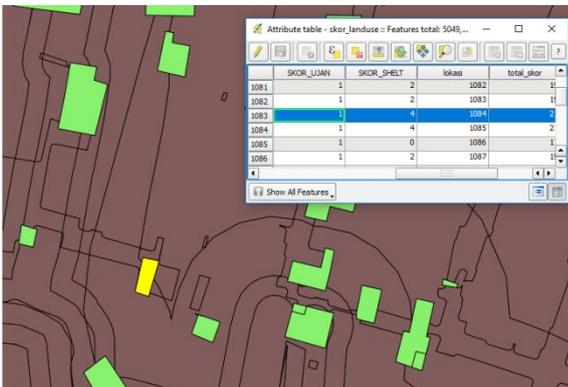
Lokasi yang telah dijelaskan seluruhnya telah disesuaikan dengan setiap lokasi hasil *overlay* peta yang merupakan data sekunder dari penelitian ini. Setiap lokasi tersebut telah diberi bobot yang telah disesuaikan dengan pembobotan SAW yang telah diberikan di awal. Melalui perhitungan SAW tersebut maka telah ditemukan alternatif terbaik yang memiliki nilai preferensi sebesar 0.94 dengan jumlah alternatif lokasi evakuasi sebanyak 6 lokasi yang tersebar di peta kota Surakarta. Titik lokasi tersebut akan ditandai dengan polygon berwarna kuning yang ada pada Gambar 6-11. Tabel yang berada pada gambar akan menjelaskan mengenai atribut terpilih sebagai lokasi yang ditandai dengan polygon kuning.



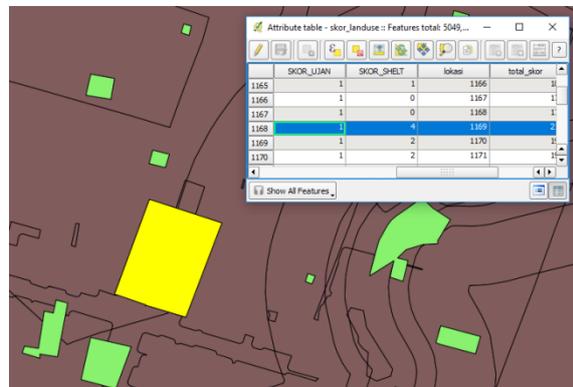
Gambar 5. Lokasi 1 berada di Kelurahan Panularan, Kecamatan Laweyan.



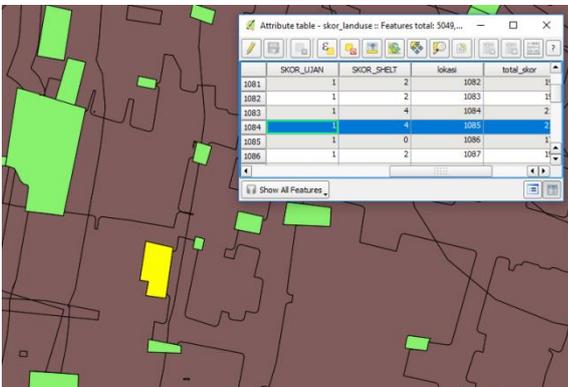
Gambar 8. Lokasi 4 berada di Kelurahan Danukusuman, Kecamatan Serengan.



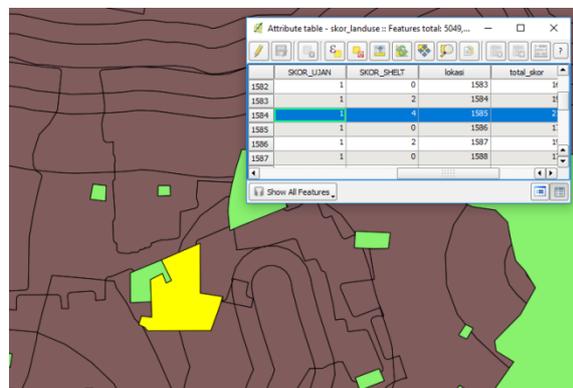
Gambar 6. Lokasi 2 berada di Kelurahan Panularan, Kecamatan Laweyan.



Gambar 9. Lokasi 5 berada di Kelurahan Buluwarti, Kecamatan Pasar Kliwon.



Gambar 7. Lokasi 3 berada di Kelurahan Sriwedari, Kecamatan Laweyan.



Gambar 10. Lokasi 6 berada di Kelurahan Jebres, Kecamatan Jebres.

## 5. SIMPULAN

Berdasarkan keseluruhan pembahasan maka melalui analisis menggunakan SIG dan kemudian dilanjutkan dengan penjumlahan pembobotan dengan menggunakan metode SAW untuk penentuan lokasi evakuasi guna menjadi tempat singgah sementara ketika banjir melanda di Kota Surakarta telah didapatkan temuan yaitu terdapat 6 lokasi terbaik. Keenam lokasi tersebut memiliki kriteria beserta sub kriteria yang sama yaitu yang telah dijelaskan pada Tabel 15.

Berdasarkan keseluruhan penelitian yang telah dilaksanakan, dapat disimpulkan bahwa masih terdapat beberapa saran untuk penelitian ini kedepannya diantaranya adalah perlunya analisis terkait perbandingan jumlah pengungsi dengan lokasi evakuasi yang dituju apakah sudah mencukupi dan memadahi atau belum. Serta perlu dilakukannya penelitian terkait temuan dari penelitian ini apakah bisa digunakan pada setiap tingkatan bencana banjir yang terjadi di kota Surakarta atau tidak.

## 6. DAFTAR PUSTAKA

- ArcGIS for Desktop. *Understanding Overlay Analysis*. Diperoleh 16 Maret 2017, dari <http://desktop.arcgis.com/en/arcmap/10.3/tools/spatial-analyst-toolbox/understanding-overlay-analysis.htm>.
- BNPB. *Definisi Dan Jenis Bencana*. Diperoleh 14 Maret 2017, dari <http://www.bnpb.go.id/home/definisi>.
- CHERNOVITA, H.P. 2013. Pemetaan Wilayah Risiko Bencana Banjir Kabupaten Kudus Berdasarkan Aspek Ancaman, Kerentanan, dan Kapasitas Berbasis Sistem Informasi Geografis. *Skripsi*. Program Studi Sistem Informasi Pariwisata Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Salatiga.
- ESRI. *GIS Dictionary Look Up Terms Related to GIS Operations, Cartography, and Esri Technology*. Diperoleh 16 Maret 2017, dari <http://support.esri.com/other-resources/gis-dictionary/term/overlay>.
- ILMU GEOGRAFI. (2016, 17 September). *Upaya Penanggulangan Banjir*. Diperoleh 16 Maret 2017, dari <http://ilmugeografi.com/bencana-alam/upaya-penanggulangan-banjir>.
- NUGRAHANI, LUSIANA RISKY. 2016. Penerapan Metode Simple Additive Weighting Untuk Penerimaan Karyawan. *Repository Universitas Dian Nuswantoro*.
- PRIYONO, JUMADI. 2010. Pemodelan Spatial Menggunakan Sistem Informasi Geografis (SIG) Berbasis Web Untuk Pengelolaan Wilayah Yang Terintegrasi Dengan Kebijakan Pemerintah. *Seminar Nasional Penginderaan Jauh Dan Sistem Informasi Geografis*.
- PURBANI, DINI. 2015. Penentuan Jalur Evakuasi, Tempat Evakuasi Sementara (TES) Beserta Kapasitasnya Di Kota Pariaman Dengan Sistem Informasi Geografis (SIG). *Jurnal Segara*, vol. 11, no. 1.
- REPUBLIK INDONESIA. 2007. *Undang-Undang No. 24 Tahun 2007 tentang Penanggulangan Bencana*.
- ROSYIDIE, ARIEF. 2013. Banjir: Fakta dan Dampaknya, Serta Pengaruh dari Perubahan Guna Lahan. *Jurnal Perencanaan Wilayah dan Kota*, vol. 24, no. 3, pp.241–249.
- RUDYANTO, R. 2015. Identifikasi Zona Genangan Banjir Kota Makassar Berbasis SIG. *Repository Universitas Hasanudin*.
- SOLOPOS. (2016, 20 Juni). *BPBD Akui Penanganan Banjir Kurang Diantisipasi*. Diperoleh 5 Februari 2017, dari <http://m.solopos.com/2016/06/20/banjir-solo-bpbd-akui-penanganan-banjir-kurang-diantisipasi-730606>.
- SURAKARTA. *BPBD Kota Surakarta Waspada 21 Kelurahan Rawan Banjir*. Diperoleh 5 Februari 2017, dari <http://www.surakarta.go.id/konten/bpbd-kota-surakarta-waspada-21-kelurahan-rawan-banjir>.
- TANAAMAH, A. R. 2013. Perancangan Sistem Informasi Geografis Sebaran Tanaman Pangan Berbasis Spasial Dan Multi Tier Menggunakan Map Server. *Teknologi Komputer dan Informatika*, vol.9, no. 2.

## ANALISIS DAN PERANCANGAN SISTEM INFORMASI PENJUALAN BERDASARKAN STOK GUDANG BERBASIS CLIENT SERVER (STUDI KASUS TOKO GROSIR “RESTU ANDA”)

Anthony Anthony<sup>1</sup>, Andeka Rocky Tanaamah<sup>2</sup>, Agustinus Fritz Wijaya<sup>3</sup>

<sup>1,2,3</sup>Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana  
Email: <sup>1</sup>682013035@student.uksw.edu, <sup>2</sup>atanaamah@staff.uksw.edu, <sup>3</sup>agustinus.wijaya@staff.uksw.edu

(Naskah masuk: 21 April 2017, diterima untuk diterbitkan: 7 Mei 2017)

### Abstrak

Restu Anda merupakan toko grosir dengan sistem penjualan yang belum terkomputerisasi atau masih manual. Dimana sistem penjualan yang berjalan adalah dengan menjual barang kepada *downline* berdasarkan pada berapa banyak barang yang masih tersedia di gudang. Karena proses pencatatan barang masih manual sehingga ketika pemilik toko memeriksa barang-barang di gudang, kadang-kadang jumlah item yang tercatat tidak sesuai dengan yang dilaporkan Akibatnya saat menerima pesanan pembelian barang dari *downline*, pemilik toko yang melihat pencatatan stok barang dan barang yang diinginkan *downline* tercatat habis atau kosong maka terpaksa pemilik toko menolak dan menawarkan barang lain yang pastinya tidak semua *downline* menerima tawarang tersebut. Maka dari itu untuk menyelesaikan masalah tersebut adalah dengan merancang sistem informasi penjualan berbasis *client-server*. Aplikasi ini dibangun dengan menggunakan teknologi RMI (*Remote Method Invocation*) yang memudahkan pengguna dalam mengelola data dari dua atau lebih komputer yang berbeda. Teknologi ini memudahkan pemilik toko untuk memantau jumlah stok barang yang ada digudang tanpa membutuhkan waktu yang lama, sehingga penjualan barang di toko Restu Anda dapat dijalankan dengan benar. Dan dengan aplikasi ini pemilik toko dapat memprediksikan berapa banyak kebutuhan barang yang harus disediakan untuk dijual kepada *downline* dengan melihat data penjualan yang sudah tersinkronisasi dengan data member.

**Kata kunci:** *Sistem informasi penjualan, Client-server, RMI, Toko Grosir*

### Abstract

*Restu Anda is a grocery store with a sales system that has not been computerized or still manual. Where the sales system that running is to sell goods to downline based on how many goods are still available in the warehouse. Because the recording process of items is still manual so when the store owners checking the items in the warehouse, sometimes the number of items that recorded do not correspond to the reported. As a result, when receiving the purchasing order from downline, the store owner must see the list of goods and when the items that downline wants already out of stock or empty then the store owner will tell the downline that the items is already out of stock, and offer another item that not all downline receive it. Therefore to resolve the problem is design a sales information system based on client-server. This application is built using the RMI technology (Remote Method Invocation) that allows users to manage data from two or more different computers. This technology facilitates the store owner to directly monitoring the number of items in the warehouse without taking much time, so it can make the selling of item on Restu Anda store running properly. And with this application, the store owner can predicted how much item that required for sale to the downline with looking from selling data that already synchronized with the member data.*

**Keywords:** *sales information system, Client-server, RMI, Grocery Store*

## 1. PENDAHULUAN

Teknologi yang berkembang sekarang ini sangat berguna untuk membantu setiap orang dalam menyelesaikan masalah, hanya saja bagaimana caranya kita bisa menggunakan dan memanfaatkannya dengan baik sesuai kebutuhan kita. Karena dengan penggunaannya yang benar, tidak menutup kemungkinan teknologi akan membawa dampak positif. Salah satu dampak yang dapat dirasakan dari perkembangan teknologi adalah di bidang bisnis. Bisnis sekarang ini tidak lepas dari

yang namanya teknologi, para pengusaha sekarang tanpa teknologi tidak bisa memantau produktifitas dari perusahaannya dengan baik, selain itu dengan berkembangnya teknologi dalam bisnis, dapat dipastikan pekerjaan yang masih manual dapat dengan cepat dan tepat terselesaikan. Tapi tidak semua perusahaan mau menggunakan teknologi, karena mereka merasa perlu waktu cukup lama untuk belajar lagi dalam menggunakannya padahal tidak.

Toko Restu Anda merupakan toko grosir tingkat menengah yang terletak di Jalan Raya Barat No 80, Kecamatan Randublatung, Kabupaten Blora, yang didirikan sejak tahun 1960 oleh bapak Joyo Semito

yang merupakan orang tua dari bapak Budi Santoso yang sekarang meneruskan usaha ini. Toko ini merupakan toko grosir yang menjual kebutuhan pokok atau sembako dan rokok secara langsung kepada para pelanggan. Selain menjual barang secara langsung, toko Restu Anda juga melakukan penjualan barang secara tidak langsung, yaitu dengan melakukan pengiriman barang kepada *downline* atau pelanggan tetap yang berada di luar kecamatan Randublatung. Total pelanggan seminggu dari toko Restu Anda sampai sekarang kurang lebih ada 250 orang, dengan rata-rata 60 orang perhari, sudah termasuk *downline*.

Sejak berdiri hingga sekarang sistem penjualan yang dijalankan toko Restu Anda masih manual termasuk sistem penjualan kepada *downline*. Dalam melakukan penjualan terhadap *downline* toko Restu Anda melihat dari berapa banyak stok barang yang masih tersedia di gudang. Karena proses pencatatan stok barang yang masih manual dimana saat dicek oleh pemilik toko secara langsung ditemukan stok barang yang tidak sesuai dengan kondisi real yang dilaporkan oleh pegawai. Dampaknya, saat pemilik toko menerima pesanan dari *downline* barang yang diinginkan oleh *downline* sebenarnya masih ada tapi karena di pencatatan stok barang, barang yang diinginkan tercatat habis atau kosong maka terpaksa pemilik toko menolak dan menawarkan barang lain, pastinya tidak semua *downline* menerima tawaran tersebut. Alhasil jika *downline* menolak tawaran tersebut pemilik toko menjalankan strategi sementara yaitu dengan menghubungi langsung pembeli tersebut dan memberitahukan jumlah barang yang dikirim serta kapan pastinya barang yang diminta tersebut datang, lalu akan mengirimkannya setelah tiba. Hal ini membuat *downline* harus menunggu untuk mendapatkan barang yang diinginkan padahal mungkin saja barang yang diinginkan tersebut dibutuhkan untuk dijual saat itu juga.

Dari permasalahan diatas, maka toko Restu Anda perlu mengembangkan sistem informasi penjualan berbasis *client-server* yang mampu memajemen penjualan dan stok gudang yang dimiliki dengan baik, maka tujuan dari penelitian ini adalah untuk merancang sistem informasi penjualan berdasarkan stok gudang berbasis *client-server* pada toko grosir Restu Anda yang berguna dan membantu toko Restu Anda dalam memajemen penjualan barang-barang pada *downline* secara lebih rapi dan membantu menghindari terjadinya kesalahan pencatatan stok gudang yang ada.

## 2. TINJAUAN PUSTAKA

### 2.1 Penelitian Terdahulu

Penelitian terdahulu digunakan sebagai bahan pertimbangan yang memiliki kaitan dengan penelitian yang dilakukan saat ini. Terdapat beberapa penelitian terdahulu yang menjadi referensi, antara lain penelitian berjudul “Analisa, Perancangan dan Implementasi Sistem Informasi Penjualan Pelumas

Studi Kasus : Perusahaan “PT. Pro Roll International”” oleh Radiant Victor Imbar dan Eric Tirta. Penelitian ini dilakukan bertujuan untuk merancang sistem informasi penjualan pada “PT. Pro Roll International” yang masih manual menjadi sistem informasi penjualan yang terkomputerisasi. Penelitian ini diharapkan setelah sistem informasi penjualan di komputerisasi, maka pengumpulan, penyimpanan, dan pengolahan data transaksi dapat dilakukan secara akurat dan tepat. Perangkat lunak yang digunakan untuk membangun sistem antara lain adalah menggunakan PHP, MySQL dan *Asynchronous Javascript And XML (AJAX)* (Radiant Victor Imbar, Eric Tirta. 2007).

Penelitian lain dilakukan oleh Suprayitno dan Uli Indah Wardati dengan judul “Pembangunan Sistem Stok Barang dan Penjualan pada Toko Sero Elektronik”. Penelitian ini bertujuan untuk membuat sistem informasi untuk mengelola stok barang dan penjualan. Pembuatan sistem ini hanya meliputi sistem stok barang, pencarian barang, laporan per bulan, dan penjualan di toko Sero Elektronik menggunakan *Visual Basic 6.0* dan *Microsoft Access 2003*. Dari sistem yang dirancang diharapkan membantu proses pembuatan laporan penjualan bulanan serta memudahkan penyampaian informasi pada pegawai lain dengan fitur-fitur yang ada didalamnya. (Suprayitno, Uli Indah Wardati. 2012).

Penelitian selanjutnya adalah penelitian dari Meliana Christianti, Eric Kurniawan dengan judul “Sistem Informasi Toko Atom Komputer untuk Mengelola Proses Penjualan dan Pembelian Barang Menggunakan PHP dan Openwave”. Penelitian ini membahas mengenai perubahan dari cara-cara pengelolaan data manual dari toko Atom Komputer di Bandung menjadi sistem yang terkomputerisasi guna mendukung sistem kerja dari toko tersebut. Dan juga aplikasi ini diharapkan dapat menjadi alat yang mempermudah dalam memantau berbagai macam laporan di mana saja dan kapan saja guna menunjang mobilitas kerja yang tinggi. (Meliana Christianti, Eric Kurniawan. 2009).

Perbedaan dari penelitian ini dengan penelitian yang sudah pernah ada sebelumnya ialah perancangan sistem informasi penjualan yang akan dilakukan adalah berbasis *client-server*. Dimana yang diharapkan disini adalah data-data yang berada di tiap bagiannya dapat di satukan atau dikumpulkan dalam satu sistem yang terintegrasi sehingga nantinya akan mempermudah dalam menerima laporan di akhir. Sistem informasi penjualan ini dibuat dengan menggunakan bahasa pemrograman *Java* serta berbasis desktop. Selain itu teknologi yang digunakan disini adalah teknologi RMI (*Remote Method Invocation*). Untuk *database* yang digunakan adalah MySQL. Selain itu batasan masalah yang diteliti adalah bagian penjualan yang berdasarkan dengan stok gudang dari pemilik toko, melalui sistem informasi penjualan ini maka pemilik toko akan lebih cepat mendapatkan informasi mengenai jumlah stok

gudang sehingga saat melakukan penjualan barang kepada *downline* pemilik toko dapat memberikan informasi mengenai jumlah stok barang yang tersisa secara cepat. Serta pemilik toko dengan mudah mendapatkan informasi berapa banyak barang yang harus dibeli dan disediakan setiap minggunya, agar saat *downline* melakukan pembelian, barang yang harus dijual masih ada dan penjualan tidak terganggu. Selain itu, dari sisi pemilik toko sistem informasi ini akan membantu dalam pemesanan barang yang

## 2.2 Konsep Sistem Informasi

Sistem informasi adalah kumpulan elemen yang saling berhubungan dan berfungsi untuk memproses, mengumpulkan, mendistribusikan serta menyimpan informasi yang berguna sebagai pendukung dalam pembuatan keputusan juga pengawasan dalam organisasi (Kadir, Abdul. 2003). Maka dengan adanya sistem informasi, pemilik toko akan lebih mudah dalam memberikan informasi sehingga penjualan yang dilakukan tidak terganggu.

## 2.3 Konsep Penjualan

Yang dimaksud dengan penjualan ialah menjual barang dagang yang menjadi usaha pokok suatu perusahaan dengan dilakukan secara terus menerus dan tepat (Marom, Chairul. 2002). penjualan juga adalah salah satu langkah pemasaran dari suatu perusahaan, sehingga perusahaan dapat memperoleh keuntungan yang akibatnya adalah kegiatan operasional perusahaan dapat tetap dijalankan. Berikut pengertian penjualan menurut beberapa ahli: Basu Swasta(2001:1) diambil dari bukunya yang merupakan buku manajemen penjualan edisi ketiga, yaitu "Menjual adalah ilmu dan seni mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang dan jasa yang ditawarkan" (Basu Swastha, 2001). Menurut Winardi(2005:26), "Penjualan adalah berkumpulnya seorang pembeli dan penjual dengan tujuan melaksanakan tukar menukar barang dan jasa berdasarkan pertimbangan yang berharga misalnya pertimbangan uang". Menurut Thamrin Abdullah dan Francis Tantri,(2016:3) "Penjualan adalah bagian dari promosi dan promosi adalah salah satu bagian dari keseluruhan sistem pemasaran".

Dalam pengertiannya suatu perusahaan mempunyai tiga tujuan dalam penjualan yaitu:

1. Mencapai batas tertentu dari penjualan
2. Mendapat keuntungan
3. Mendukung perkembangan perusahaan (Basu Swastha, 2001).

Penjualan bila diidentifikasi berdasarkan perusahaannya maka dapat dikategorikan menjadi beberapa jenis, yaitu :

1. **Penjualan langsung** dimana penjualan ini adalah dengan cara mengambil barang dari *supplier* kemudian secara langsung dikirim ke customer
2. **Penjualan stok gudang** dimana penjualan ini adalah dengan cara menjual barang dari stok yang ada di gudang.

3. **Penjualan kombinasi** adalah penjualan dimana dengan mengambil sebagian barang dari *supplier* serta sebagian dari stok yang ada di gudang (Martin S, Colleran G, 2006).

Dilihat dari pengertian tersebut maka suatu proses penjualan haruslah memberikan keuntungan bagi suatu usaha yang sedang berlangsung, karena itu diperlukan sebuah sistem informasi penjualan sehingga proses penjualan yang dilakukan dapat berjalan dengan efektif serta efisien.

Jika melihat jenis dan bentuk penjualan tanpa diidentifikasi dari perusahaannya, maka dapat dibedakan jenis penjualan adalah sebagai berikut:

1. **Trade selling** yaitu penjualan yang dapat terjadi jika antara produsen dan pedagang mengizinkan pengecer untuk memperbaiki distribusi produk mereka.
2. **Missionary Selling**, yaitu proses untuk meningkatkan penjualan dengan cara mendorong para pembeli agar membeli barang-barang dari penyalur.
3. **Technical Selling** yaitu proses untuk meningkatkan penjualan dengan cara memberikan saran serta nasihat kepada para pembeli barang dan jasa.
4. **New Business Selling** adalah usaha untuk membuka transaksi baru dengan calon pembeli seperti yang dilakukan perusahaan asuransi.
5. **Responsive Selling** adalah dimana setiap tenaga penjual dapat memberikan reaksi terhadap permintaan serta pembeli melalui *route driving and retailing*. (Basu Swastha, 2001).

Sedangkan untuk bentuk bentuk dari penjualan antara lain:

- 1) **Penjualan Tunai/Cash** adalah penjualan bersifat *cash and carry* yang mana penjualan dilakukan setelah terdapat kesepakatan harga antara penjual dengan pembeli, lalu pembeli dapat membayar secara langsung dan barang dapat langsung dimiliki.
- 2) **Penjualan Kredit/Non-Cash** yaitu penjualan *non-cash*, dengan memberikan tenggat waktu tertentu, biasanya diatas satu bulan.
- 3) **Penjualan Tender** adalah penjualan yang dilaksanakan melalui prosedur tender yang digunakan untuk memenuhi permintaan pihak pembeli yang membuka tender.
- 4) **Penjualan Ekspor** yaitu penjualan yang dilaksanakan dengan pihak pembeli dimana mengimpor barang dari luar negeri yang biasanya dengan menggunakan *letter of credit*.
- 5) **Penjualan Konsinyasi** yaitu penjualan barang dengan cara menitipkan kepada pembeli yang juga berperan sebagai penjual dimana jika barang

yang dititipkan tersebut tidak terjual maka barang akan dikembalikan lagi kepada penjual.

- 6) **Penjualan Grosir** yaitu penjualan yang dilakukan tidak langsung kepada pembeli, tetapi melalui perantara pedagang. (Basu Swastha, 2001).

#### 2.4 Konsep Sistem Informasi Penjualan

Yang dimaksud dengan sistem informasi penjualan ialah suatu pembuatan pernyataan penjualan atau kegiatan yang dijelaskan melalui prosedur-prosedur penjualan (Niswonger, C. Rolling; E.Fess, Philip ; S.Warre, Carl. 1999.)

#### 2.5 Konsep Persediaan

Persediaan diartikan sebagai aset atau harta yang ada untuk dijual dalam kegiatan usaha biasa, dalam proses produksi penjualan atau dalam bentuk bahan atau perlengkapan untuk digunakan dalam proses produksi atau pemberian jasa (Ikatan Akutansi Indonesia, 2008). Maka dari itu mengapa suatu penjualan yang berdasarkan stok gudang atau persediaan barang sangat sensitif.

Persediaan (*inventory*) adalah semua barang yang ada pada perusahaan dalam kondisi tertentu dengan tujuan nantinya untuk dijual atau dikonsumsi dalam perusahaan. Aktiva atau hal hal lain yang dimiliki perusahaan akan tapi tidak dijual atau dikonsumsi tidak termasuk dalam persediaan. Persediaan (*inventory*) untuk setiap tempat akan berbeda, tergantung kepada jenis perusahaan yang bersangkutan:

Pada perusahaan dagang, berupa persediaan barang dagangan (*merchandise inventory*). Sedangkan untuk perusahaan pabrik (manufaktur) persediaan terdiri atas:

- Persediaan bahan baku (*direct materials inventory*)
- Persediaan barang dalam proses (*work in process inventory*)
- Persediaan barang jadi (*finished good inventory*) (Ahmed Riahi dan Belkaoui, 2000).

#### 2.6 Konsep Pencatatan Persediaan Barang Dagang

Selain itu, karena penjualan berdasarkan stok gudang atau persediaan barang sangat sensitif maka diperlukanlah pencatatan persediaan barang yang akan membantu pemilik toko untuk memantau berapa banyak persediaan barangnya secara langsung serta jika persediaan yang dimiliki masih banyak pemilik dapat melakukan pembatasan pembelian barang sehingga persediaan tidak melebihi kapasitas dari gudang yang dimiliki.

Pengelolaan transaksi pembelian atau penjualan persediaan barang didalam akutansi terdapat dua cara dalam mencatat persediaan hal ini membuat sebuah perusahaan dapat memilih salah satu metode tersebut, metode itu antara lain metode pencatatan persediaan

perpetual dan metode pencatatan persediaan fisik/periodik.

Pencatatan Perpetual yaitu pencatatan atas transaksi persediaan yang dilaksanakan setiap waktu, baik terhadap pemasukan maupun terhadap pengeluaran persediaan, dalam penggunaan metode ini pencatatan atas transaksi dilakukan secara terus-menerus untuk setiap jenis persediaan dan untuk menjamin keakuratan jumlah persediaan perhitungan fisik persediaan biasanya dilakukan setahun sekali. Pencatatan persediaan dengan menggunakan metode ini ditujukan terutama untuk barang yang bernilai tinggi dan untuk barang yang mudah dicatat pemasukan dan pengeluarannya digudang (A. Dunia, Firdaus, 2005), sistem perpetual (*perpetual inventory system*), adalah pencatatan terhadap mutasi persediaan secara konsisten, dengan mencatat semua transaksi yang menyebabkan pengurangan atau penambahan persediaan. Untuk lebih jelasnya, sistem fisik memiliki ciri-ciri:

- Transaksi pembelian barang dagang, dicatat debit pada akun “persediaan”, kredit akun “kas” atau “utang dagang”.
- Transaksi penjualan barang dagang dicatat sebagai berikut:
  - o Harga (hasil) penjualan dicatat debit pada akun “kas” atau akun “piutang dagang”, dan kredit “akun penjualan”.
  - o Harga pokok barang dagang yang dijual, dicatat debit pada akun “Harga Pokok Penjualan” dan kredit pada akun “persediaan”.
- Saldo akun “persediaan” menunjukkan harga pokok (nilai) persediaan barang yang tersedia di gudang (James M. Reeve, 2008).

Sedangkan pencatatan persediaan fisik/periodik dikutip dari Kieso, Weygant & Warfield “*The quantity of inventory in the hands of determined, as implied by its name, periodically. All purchase of inventory during the by debiting the account purchase accounting period are recorded*” adalah kuantitas persediaan ditangan ditentukan, seperti yang tersirat oleh namanya, secara periodik. Semua pembelian persediaan selama periode akuntansi dicatat dengan mendebet akun pembelian. (E. Kieso, Donald, Jerry J, Weygant and Teery D. Warfield, 2007 ). Metode fisik / periodik (*periodical inventory system*) berdasarkan sistem persediaan ditentukan dengan menghitung fisik terhadap persediaan. Penghitungan dilakukan secara periodik. Dalam sistem ini pencatatan tidak selalu diikuti dengan mutasi persediaan. Maka dari itu prosedur ini harus dilakukan pada akhir periode (*mandatory procedure*) guna menentukan persediaan yang akan dilaporkan pada laporan keuangan. Hasil perhitungan ini digunakan untuk landasan penentu nilai persediaan.

Untuk lebih jelasnya, metode ini memiliki ciri-ciri sebagai berikut:

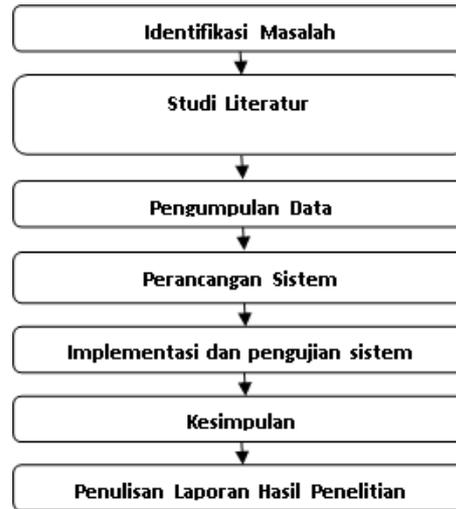
- a. Pembelian serta penjualan barang dagangan, tidak dicatat sehingga akun persediaan hanya menunjukkan informasi nilai persediaan barang dagang awal periode.
- b. Transaksi pembelian barang dagang dicatat debit pada akun “pembelian” dan transaksi penjualan barang dagang dicatat sebesar harga jualnya.
- c. Nilai persediaan pada akhir periode diketahui setelah dilakukan pemeriksaan dan penghitungan barang secara fisik (James M. Reeve, 2008).

**2.7 Konsep Client-Server**

*Client-Server* merupakan suatu model komunikasi antara dua atau lebih komputer guna melakukan pembagian tugas (W. Komputer, 2010). Dalam perkembangan teknologi *Client-Server* terdapat beberapa arsitektur yang telah dikembangkan, yaitu *1-Tier (standalone)*, *2-Tier* dan *3-Tier/Multi-Tier* (W. Komputer, 2010). Pembangunan aplikasi yang memanfaatkan konsep komputasi tersebar telah digantikan oleh teknologi *Web Service* namun pada area yang masih memprioritaskan kecepatan adalah hal yang utama, maka konsep teknologi tersebar seperti Java RMI bisa lebih baik dan dapat diandalkan (A. Nugroho dan A. Ashari, 2011). Teknologi Java RMI tidak hanya dapat dibangun dalam satu komputer melainkan ke banyak komputer. Dari hasil implementasi teknologi Java RMI terdapat keuntungan pengaksesan data yang cepat karena adanya pembagian fungsi antara RMI server dan RMI client (S. Lestari, M. Somantri, R. R. Isnanto, 2011). *Remote Method Invocation (RMI)* dapat didefinisikan sebagai sebuah fasilitas standar Java yang berguna melakukan pemanggilan (*invocation*) suatu *methode* dari jarak jauh (*remote*) didalam jaringan (B. Susanto, 2003).

**3. METODOLOGI PENELITIAN**

Penelitian yang dilakukan meliputi beberapa tahapan yaitu: 1) Identifikasi masalah, 2) Studi Literatur 3) Pengumpulan Data, 4) Perancangan Sistem, 5) Implementasi dan Pengujian Sistem, 6) Kesimpulan, 7) Laporan Hasil Penelitian. Tahapan yang dilakukan dapat dilihat pada Gambar 1:



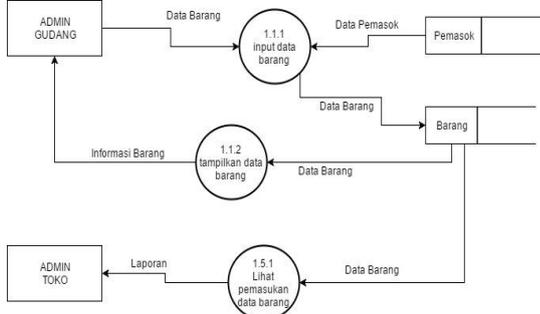
**Gambar 1.** Tahapan Penelitian

Tahapan pertama dalam merancang sebuah sistem yaitu pengidentifikasian masalah dimana hasil dari identifikasi masalah digunakan untuk merancang *prototype* Sistem Informasi Penjualan Berdasarkan Stok Gudang. Pada tahap ini digunakan metode wawancara untuk memperoleh informasi dari permasalahan yang ada. Wawancara dilakukan dengan narasumber yang bersangkutan yaitu bapak Bambang Budi Santoso selaku pemilik toko Restu Anda untuk memperoleh penjelasan lengkap guna memperkuat data, wawancara dilakukan selama bulan Desember 2016 sampai dengan Januari 2017, dari hasil pengidentifikasian masalah yang dilakukan didapatkan informasi mengenai masalah sistem penjualan berbasis stok gudang yang dijalankan toko restu anda masih manual, dimana penjualan didasarkan pada permintaan *downline* dan pemilik langsung menerima saja tanpa melihat terlebih dahulu jumlah stok sebenarnya di gudang. Salah satu akibatnya adalah *downline* kecewa karena barang tidak ada, selain itu karena pencatatan gudang yang dilakukan masih bersifat seadanya dalam arti berubah ubah, bisa sehari sekali, bisa dua hari sekali dilakukan pencatatan karena itu pemilik toko merasa kesulitan dalam menganalisa permintaan barang kepada *supplier* padahal *supplier* melakukan pengiriman barang seminggu sekali dan permintaan barang harus dilakukan seminggu sebelum pengiriman. Jika pemilik toko sembarangan dalam melakukan permintaan barang maka akibatnya adalah *overstock* dan juga nota pembelian barang yang diterima akan menumpuk, pemilik toko Restu Anda berharap dengan memanfaatkan sistem yang sudah terkomputerisasi maka pencatatan persediaan barang yang biasanya dirasa kurang efektif dapat lebih efektif, *overstock* dapat dihindari, penjualan berdasarkan persediaan barang digudang dapat dijalankan.

Pada tahap selanjutnya dilakukan studi literatur tentang beberapa penelitian terdahulu terkait dengan perancangan sistem informasi penjualan untuk membangun konsep awal penelitian serta

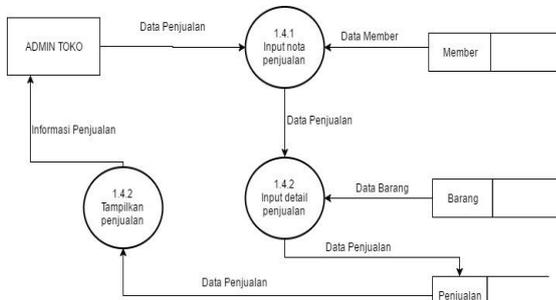


konteks pada Gambar 3, saat data barang dimasukkan dalam sistem oleh admin gudang, sistem akan memproses di dalam olah data barang dengan mengambil data *supplier* untuk melengkapi data barang, kemudian hasil pemrosesan yang berupa informasi barang akan dikirimkan ke admin gudang, setelah itu data barang akan disimpan kedalam data *store* barang. Untuk admin toko, bisa melakukan olah data member, olah data *supplier*, dan olah data penjualan. Selain itu, admin toko juga mendapatkan laporan pemasukan barang yang berasal dari olah data pemasukan barang yang mana sistem dalam olah data pemasukan barang berasal dari data *store* barang. Sistem penjualan yang dijalankan di olah data penjualan adalah memanggil data dari data *store* member serta data *store* barang yang berguna sebagai acuan dasar siapa yang membeli barang serta barang apa saja yang dibeli.



**Gambar 5.** Data Flow Diagram Lvl 2 proses 1.1 dari sistem informasi penjualan toko Restu Anda

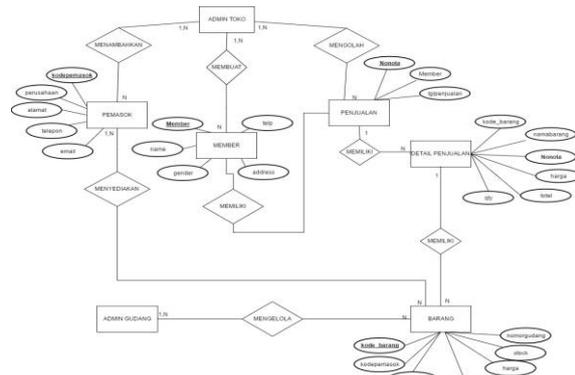
Pada Gambar 5, diperlihatkan bila dalam proses olah data penjualan terdapat proses yang lebih rinci, yaitu Admin gudang akan melakukan penginputan data barang akan mengambil data pemasok terlebih dulu. Data pemasok digunakan sebagai acuan admin gudang untuk memberikan informasi terhadap barang yang ada digudang, jadi ketika admin toko melakukan proses lihat pemasukan data barang, maka admin toko dapat memperoleh informasi lengkap dari barang tersebut sampai pada siapa pemasok barang tersebut.



**Gambar 6.** Data Flow Diagram Lvl 2 proses 1.4 dari sistem informasi penjualan toko Restu Anda

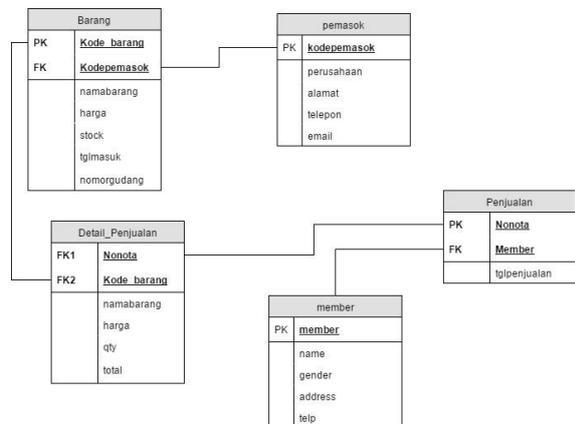
Pada Gambar 6, diperlihatkan juga bila proses penjualan yang terdapat dalam sistem berjalan lebih kompleks, dimana dalam proses olah data penjualan secara rinciannya terdapat proses input nota penjualan, dalam proses ini, admin harus

menginputkan no nota penjualan yang sedang berlangsung, tanggal nota tersebut dibuat, serta no member pembeli barang. Selanjutnya setelah input nota penjualan berhasil, data penjualan diteruskna ke bagian input detail penjualan, dalam proses tersebut, pihak admin sudah dapat melakukan input pembelian barang yang akan dibeli oleh member. Setelah selesai, data penjualan tersebut disimpan kedalam data *store* penjualan, setelah itu admin toko bisa mengakses data penjualan di proses tampilkan penjualan, dalam proses tersebut sistem akan memanggil data penjualan di data *store* penjualan dan ditampilkan kepada admin toko langsung.



**Gambar 7.** Entity Relationship Diagram dari dari sistem informasi penjualan toko Restu Anda

Pada Gambar 7 adalah gambaran Entity Relationship Diagram (ERD) dari sistem informasi penjualan toko Restu Anda yang dirancang. ERD tersebut menggambarkan hubungan hubungan yang terbentuk antar entitas yang ada dalam sistem informasi penjualan toko restu anda. Hasil rancangan ERD sistem informasi penjualan toko restu anda ini akan diimplementasikan kedalam suatu *database* yang didalamnya terdapat beberapa tabel yang berelasi satu sama lain.



**Gambar 8.** Relasi Antar Tabel dari sistem informasi penjualan toko Restu Anda

Pada Gambar 8 merupakan tampilan dari tabel tabel yang akan dibuat dan dimana relasinya. Bisa dilihat, setiap tabel yang dirancang saling berelasi satu sama lain dan dari rancangan tersebut terdapat satu tabel

yang tidak memiliki *primary key* dan hanya memiliki *foreign key* saja yaitu pada tabel *detail\_penjualan* yang berguna untuk memberikan informasi mendetail mengenai penjualan yang ada, jadi semisal ada perulangan data pada tabel *detail\_penjualan* tersebut maka tidak akan terdapat *error* dikarenakan yang digunakan adalah *foreign key* maka data yang diinputkan boleh berulang atau sama. Rincian dari Tabel yang dibuat pada Gambar 8 antara lain adalah Tabel Pemasok, Tabel Barang, Tabel *Detail\_Penjualan*, Tabel *Penjualan* dan Tabel *Member*

#### 4. HASIL DAN PEMBAHASAN

Hasil dari tahap awal penelitian yang dilakukan yaitu pengidentifikasian masalah, didapatkan informasi mengenai permasalahan sistem penjualan berbasis stok gudang yang dijalankan toko restu anda masih manual, dimana penjualan didasarkan pada permintaan *downline* dan pemilik langsung menerima saja tanpa melihat terlebih dahulu jumlah stok sebenarnya di gudang. Salah satu akibatnya adalah *downline* kecewa karena barang tidak ada, selain itu karena pencatatan gudang yang dilakukan masih bersifat seadanya dalam arti berubah ubah, bisa sehari sekali, bisa dua hari sekali dilakukan pencatatan karena itu pemilik toko merasa kesulitan dalam menganalisa permintaan barang kepada *supplier* padahal *supplier* melakukan pengiriman barang seminggu sekali dan permintaan barang harus dilakukan seminggu sebelum pengiriman, jika pemilik toko sembarangan dalam melakukan permintaan barang maka akibatnya adalah *overstock* dan juga nota pembelian barang yang diterima akan menumpuk, pemilik toko Restu Anda berharap dengan memanfaatkan sistem yang sudah terkomputerisasi maka pencatatan persediaan barang yang biasanya dirasa kurang efektif dapat lebih efektif, *overstock* dapat dihindari, penjualan berdasarkan persediaan barang digudang dapat dijalankan Berdasarkan permasalahan tersebut dirancangkan aplikasi yang membantu pemilik toko kedepannya dalam mengelola data barang di gudang serta data penjualan berbasis stok gudang.

Dengan perancangan sistem berbasis *prototype* yang sudah ditetapkan diawal oleh peneliti maka selama proses perancangan peneliti melakukan komunikasi secara aktif antara peneliti yang akan merancang sistem serta pengguna sistem tersebut yaitu pemilik toko. Selama perancangannya dirancangkan beberapa macam *prototype* yang berbeda, rancangan ini dibuat sampai sesuai dengan keinginan pemilik toko.

Dalam perancangan sistem informasi penjualan ini, ditentukan terlebih dahulu bahwa *user* atau pengguna yang akan menggunakan aplikasi ini terdiri dari 2 admin yang berbeda, yaitu admin gudang yang akan mengelola semua data barang barang yang masuk setiap harinya serta admin toko yang akan mengelola semua data penjualan, data member, data pemasok

barang serta melihat laporan dari pemasukan barang yang diinputkan oleh admin gudang tiap harinya.

Gambar 9 merupakan menu login yang nantinya akan membedakan mana hak akses Admin Toko dan mana hak akses Admin Gudang. Dimana untuk Admin Gudang hanya bertugas mengelola barang dan Admin Toko yang mengelola penjualan, member, dan pemasok



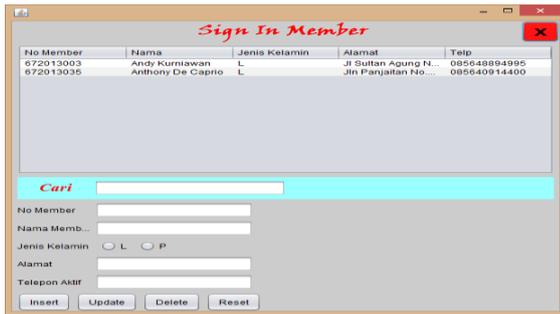
**Gambar 9.** Design menu awal Login dari system informasi penjualan toko Restu Anda

Gambar 10 merupakan *design* awal aplikasi yang akan digunakan untuk admin toko. Dimana pada *design* awal, admin toko dapat melihat hal apa saja yang dapat di lakukan mulai dari olah barang, olah member, olah pemasok dan laporan barang.



**Gambar 10.** Design awal menu olah member admin toko dari sistem informasi penjualan toko Restu Anda

Menu selanjutnya pada Gambar 11 adalah merupakan *design* menu olah member yang digunakan admin, pada menu ini admin toko dapat melakukan pencarian data member dengan mengetikkan kata apa saja, dimana kata tersebut nantinya akan otomatis menyaring data member yang diinginkan. Data yang diinputkan oleh admin toko sendiri antara lain hanya data no member, nama member, jenis kelamin, alamat member, serta telepon aktif. Menu ini membantu admin toko untuk mengetahui siapa saja pembeli tetap yang ada pada toko restu anda, memudahkan pemilik toko menghubungi jika ada keperluan. Serta dengan adanya member, penjualan yang dilakukan nantinya dapat dipantau dengan baik oleh admin toko.



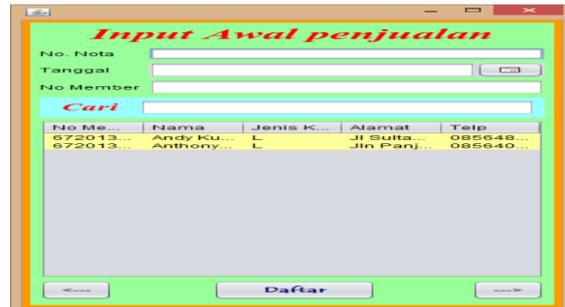
**Gambar 11.** Design menu olah member admin toko dari sistem informasi penjualan toko Restu Anda

Pada Gambar 12 adalah menu olah pemasok, menu ini dirancang untuk memudahkan pemilik toko untuk menghubungkan pemasok yang mengelola dan menyediakan barang yang dibutuhkan oleh pemilik toko, pada menu ini, admin toko dapat menambahkan data data pemasok barang yang belum ada serta mengedit data pemasok bila sewaktu waktu terdapat perubahan *salesman*.



**Gambar 12.** Menu olah pemasok pada admin toko dari sistem informasi penjualan toko Restu Anda

Gambar 13 adalah menu olah penjualan yang dapat dilakukan oleh admin toko, menu ini berisi beberapa tahap, yaitu pada menu ini, admin toko dapat mengelola transaksi penjualan dan mengambil data member untuk memberikan informasi barang apa saja yang dibeli member tersebut pada tanggal yang saat itu sedang berlangsung. Pada menu ini disertakan juga keterangan bila pembeli atau *downline* yang melakukan pembelian barang pada toko restu anda bukan merupakan member maka atau belum terdaftar sebagai member maka mereka harus mendaftar terlebih dahulu barulah mereka bisa melakukan transaksi, hal ini untuk memudahkan pemilik toko, mengetahui jumlah barang yang dapat disediakan setiap minggunya oleh pemilik toko untuk masing masing member.



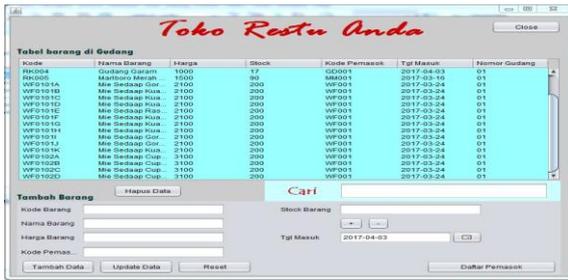
**Gambar 13.** Menu olah penjualan pada admin toko dari sistem informasi penjualan toko Restu Anda

Gambar 14 merupakan tampilan dari menu input detail penjualan yang nantinya dapat diakses setelah melalui menu olah penjualan di Gambar 13. Pada menu ini admin toko akan menginputkan pembelian yang dilakukan oleh member atau *downline* toko Restu Anda. Dimana pembelian tersebut secara otomatis akan mengurangi jumlah stok barang yang ada di gudang, jadi admin gudang tidak perlu mengupdate secara manual karena sudah otomatis bisa terupdate pada proses ini.



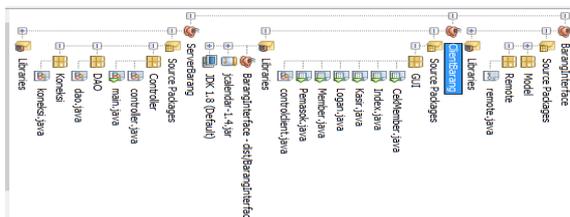
**Gambar 14.** Menu input detail penjualan dari sistem informasi penjualan toko Restu Anda

Pada Gambar 15 merupakan tampilan menu olah barang yang dilakukan oleh admin gudang, pada menu ini admin gudang dapat menginputkan data data barang yang masuk setiap harinya, kemudian bila ingin mencari data barang yang masih ada atau tidak, admin dapat melakukan pencarian dengan memasukkan kata yang berhubungan dengan keterangan barang yang dicari, selain itu bila admin gudang lupa dengan kode pemasok barang yang akan diinputkan atau bila barang yang akan diinputkan merupakan barang yang berbeda namun pemasoknya sama, maka admin gudang bisa melihat daftar pemasok terlebih dahulu kemudian mengkonfirmasi.



**Gambar 15.** Menu olah barang yang dilakukan oleh admin gudang dari sistem informasi penjualan toko Restu Anda.

Berikut alur RMI pada sistem informasi penjualan toko Restu Anda, terdiri dari tiga alur yaitu *Barang Interface*, *Client Barang*, *Server Barang*, seperti pada Gambar 16.



**Gambar 16.** Printscren proyek yang digunakan pada aplikasi Sistem Informasi Penjualan

Berikut penjelasan potongan kode yang ada dalam penggabungan aplikasi sistem informasi penjualan yang sudah dirancangan.

Kode Program 1. merupakan *Bind* nama *remote-remote* yang digunakan oleh *client* untuk me-request sebuah *method* ke *server*. *Method-method* yang disediakan oleh *server* dijelaskan dari baris 2 hingga 7 dimana berisi *requestgetInfo*, *insert*, *getaAll*, *getcari*, *delete*, dan *update*.

```

1. public interface remote extends
Remote{
2. public Boolean getInfo(String id,
String password) throws
RemoteException,SQLException;
3. public Boolean insertData
(modelbarang b) throws
RemoteException,SQLException;
4. public List<modelbarang> getall()
throws
RemoteException,SQLException;
5. public List<modelbarang>
getcari(String cari) throws
RemoteException,SQLException;
6. public Boolean deletedata(String
kode) throws
RemoteException,SQLException;
7. public Boolean
updatedata(modelbarang b) throws
RemoteException,SQLException;}
    
```

**Kode Program 1** Bind RMI pada sistem informasi penjualan toko Restu Anda

Kode Program 2. adalah *Controller Server* RMI berperan sebagai penerima pemancar dari *remote client*, seperti pada baris 3 `dongo = new dao(new koneksi().getConnection()).getInfo(id, password);` menjelaskan

variabel `dongo` dengan membawa data (`id, password`) untuk diteruskan ke DAO/tempat *method server* berada. Begitupula hal tersebut berfungsi sama seperti *method insertData*.

```

1. @Override
2. public Boolean getInfo(String id,
String password) throws
RemoteException, SQLException{
3. boolean dongo = false; dongo = new
dao(new
koneksi().getConnection()).getInfo(id,
password);
4. return dongo; }
5. @Override
6. public Boolean
insertData(modelbarang b) throws
RemoteException,
SQLException {
7. boolean cek = false; cek = new
dao(new
koneksi().getConnection()).insertdata(b)
; return cek; } }
    
```

**Kode Program 2.** Controller Server RMI system informasi penjualan toko Restu Anda

Kode Program 3. Adalah *DAO Server* RMI yang merupakan *class* yang berisi *method-method* yang dibutuhkan aplikasi dalam sepanjang pengoperasian. Dijelaskan pada baris 3 merupakan *method* boolean “insertdata”, *method* tersebut berfungsi untuk menginputkan data baru dari *request client* yang akan diteruskan ke *database* melalui kode baris ke-4. Pada baris 5 ada *method* “getData” dari *database* ke *user interface client*.

```

1. public class dao {
2. private Connection conn;
3. public Boolean insertdata(modelbarang
b) throws SQLException {
4. PreparedStatement pstm =
conn.prepareStatement("INSERT INTO
barang
5. VALUES (?, ?, ?, ?, ?, ?)"); }
6. public List<modelbarang> getData() {
7. List<modelbarang> data = new
ArrayList<>(); try{
8. PreparedStatement ps =
conn.prepareStatement("SELECT * FROM
barang
9. order by kode"); } } }
    
```

**Kode Program 3.** DAO Server RMI system informasi penjualan toko Restu Anda

Kode Program 4 adalah *Controller Client* RMI yang merupakan sarana *remote* untuk me-request sebuah *method controller server* kemudian diteruskan ke *DAO server* RMI. Pada class “controlclient” baris 3 merupakan *registration* ke alamat *server* dan sekaligus *port* dari *server* tersebut dan juga *remote server* atau nama *server* yang dituju yaitu “server”. Pada baris 5 hingga 14 dijelaskan *method-method* yang berfungsi menyalurkan apa yang diminta *client* untuk di *remote* ke *controller server* sesuai alamat *bind* RMI yang telah dibuat sebelumnya.

```

1. public class controlclient {
2. public controlclient() throws
RemoteException, NotBoundException {
3. try{
4. regis =
LocateRegistry.getRegistry("localhost",
1234); rem =
5. (remote) regis.lookup("server");
6. } catch (Exception e) {
System.out.println("Kesalahan pada
Server/
7. Server belum menyala"); }}
8. public boolean getInfo(String id,
String password) throws
9. RemoteException, SQLException
{return rem.getInfo(id, password); }
10. public List<modelbarang> getData()
throws RemoteException,
11. SQLException {
12. List<modelbarang> data =
rem.getAll(); return data; }
13. public boolean
insertData(modelbarang b) throws
RemoteException,
14. SQLException { return
rem.insertData(b); }}
    
```

**Kode Program 4** *Controller Client RMI Bind* pada sistem informasi penjualan toko Restu Anda

Sistem ini secara perancangannya sudah sesuai dengan yang diharapkan oleh pemilik toko, namun dalam perancangan sistem ini terdapat beberapa kendala dimana sumber daya pegawai yang dimiliki pemilik toko perlu diberikan pengarah terlebih dahulu dalam menggunakan aplikasi ini. Selain itu data barang yang diinputkan didalam sistem ini barulah separuh dari jumlah data barang yang sebenarnya karena banyaknya jenis dan jumlah barang yang harus dicek ulang sebelum diinputkan. Untuk pemasok barang yang diinputkan juga harus melihat jenis barang yang dijual terlebih dahulu karena banyak barang yang terkadang berbeda jenis namun memiliki pemasok yang sama. Untuk menginputkan data member pihak toko tidak mengalami kendala dikarena toko Restu Anda sudah memiliki catatan sendiri untuk masing masing member jadi mudah untuk segera diinputkan.

Pengujian aplikasi dilakukan dengan menggunakan metode *blackbox* yaitu dengan menguji masing masing fungsi yang telah dibuat dalam aplikasi ini gunanya adalah untuk menemukan kesalahan atau *bug* pada sistem yang sudah dirancang sehingga pada saat sistem ini digunakan dapat berjalan sesuai dengan apa yang diinginkan dan diharapkan oleh pengguna serta dapat memenuhi kebutuhan pengguna. Pengujian *blackbox* sendiri adalah pengujian langsung pada fungsi fungsi aplikasi tanpa memperhatikan alur eksekusi program. Pengujian ini dilakukan dengan memperhatikan apakah fungsi yang dibuat sudah sesuai atau tidak dengan yang diharapkan. Pada Tabel 1 merupakan hasil pengujian aplikasi.

**Tabel 1** Hasil Pengujian

No	Fungsi yang diuji	Status Pengujian
1	<i>Login</i>	<i>Valid</i>
2	Menambahkan data member, data pemasok, data penjualan, dan data barang.	<i>Valid</i>
3	Mengubah data member, data pemasok dan data barang.	<i>Valid</i>
4	Menghapus data member, data pemasok, data penjualan, dan data barang.	<i>Valid</i>
5	Menampilkan data barang, data member, data pemasok, dan data detail penjualan	<i>Valid</i>
6	Menambahkan detail penjualan di menu input penjualan untuk Admin toko.	<i>Valid</i>
7	Menampilkan data detail penjualan	<i>Valid</i>
8	Mencari data member, data pemasok, dan data barang	<i>Valid</i>

Pengujian yang dilakukan terhadap setiap fungsi pada aplikasi ini adalah dengan menggunakan tiap fungsi dan memberikan skenario penggunaan atau kondisi pada penggunaan aplikasi ini, untuk lebih rincinya dapat di lihat pada Tabel 1, berdasarkan skenario tersebut diberikan juga output yang diharapkan dari aplikasi ini. Bila output dari aplikasi ini sesuai dengan output yang diharapkan maka pengujian dapat dikatakan valid. Berdasarkan pengujian yang sudah dilakukan pada aplikasi ini dapat dilihat status pengujian dari setiap fungsi valid karena output dari aplikasi ini hasilnya sesuai dengan output yang diharapkan, maka aplikasi ini sudah berjalan dengan baik dan sesuai yang diharapkan.

**5. SIMPULAN**

Pada penelitian yang telah dilakukan, ditemukan bahwa sistem penjualan barang yang dijalankan toko Restu Anda belum terkomputerisasi sehingga mengakibatkan penjualan barang yang dilakukan tidak berjalan dengan benar. Maka dari itu untuk menyelesaikan masalah yang terjadi, dibangunlah sistem informasi penjualan berdasarkan stok gudang berbasis *client-server*.

Berdasarkan hasil pengujian yang telah dilakukan, diharapkan dengan menggunakan aplikasi ini dapat membantu pemilik toko dalam menjalankan proses penjualan barang kepada *downline* dengan benar. Selain itu aplikasi ini juga membantu dalam mengelola data member, data pemasok, dan data penjualan barang yang dilakukan oleh pemilik toko, sehingga secara keseluruhan data data tersebut dapat disimpan dengan rapi serta memudahkan pemilik toko mengakses data tersebut dengan cepat jika

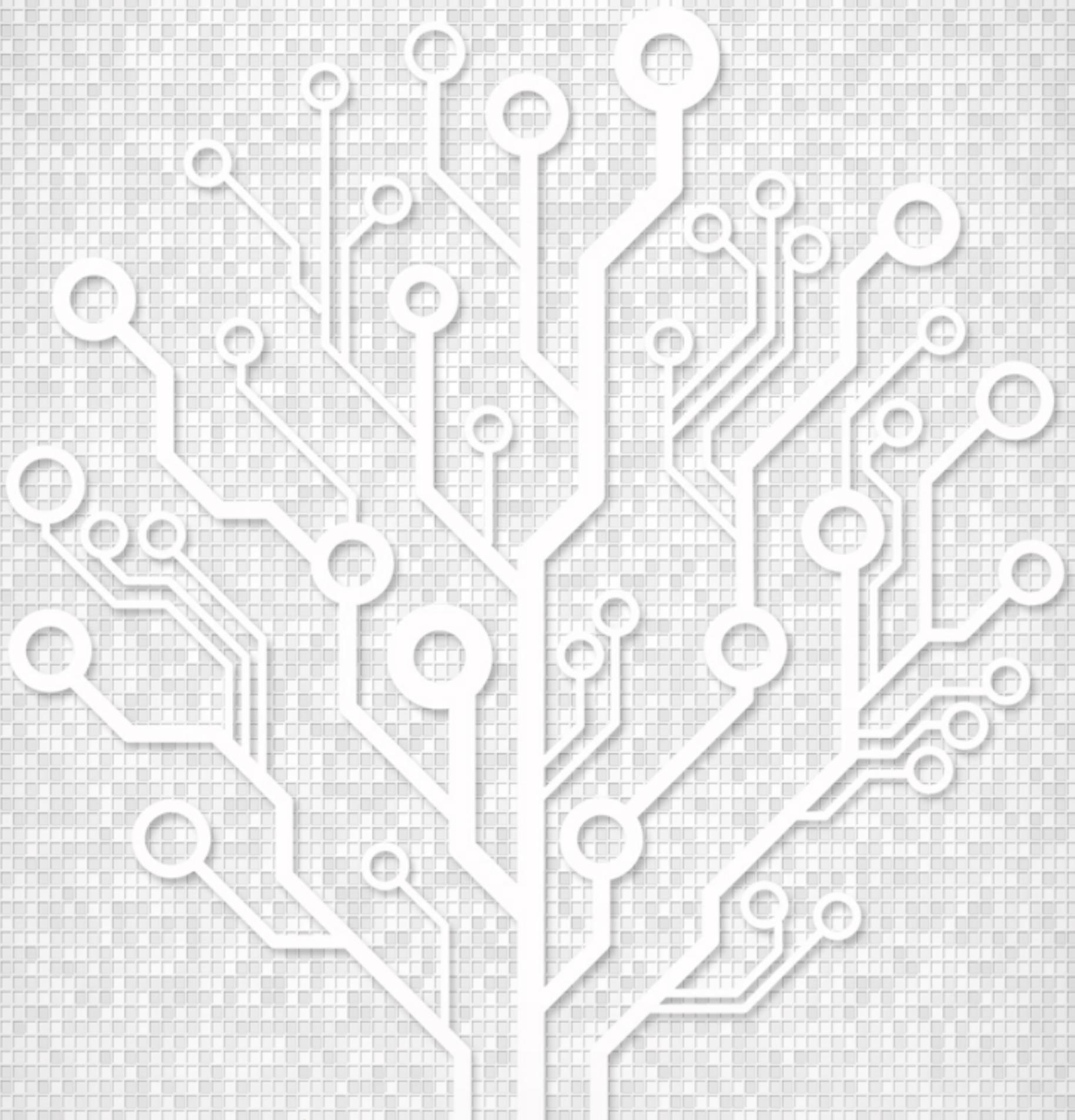
sewaktu waktu diperlukan. Dan juga pemilik toko dengan adanya data penjualan yang tersinkronisasi dengan data member dapat memprediksikan berapa banyak barang yang harus disediakan untuk dijual. Hal ini dapat memberikan kepuasan kepada *downline* sehingga *downline* tidak perlu menunggu untuk mendapatkan barang yang diinginkan.

Selain itu karena aplikasi ini dibangun berbasis *client-server* dengan menggunakan teknologi RMI, sangatlah membantu dalam menyatukan kegiatan yang dilakukan antar Admin toko dan Admin Gudang yang berbeda komputer. Pemilik toko tidak lagi membutuhkan banyak waktu untuk memantau jumlah stok barang yang ada di gudang, dan admin gudang tidak perlu merekap sisa stok barang yang ada di gudang karena pemilik toko dapat memantau secara langsung dari *database* barang yang dimiliki Admin gudang.

Pengembangan pada aplikasi ini dapat dilakukan dengan memberikan ROP(*Reorder Point*) yaitu batas/titik jumlah pemesanan kembali yang memudahkan pemilik toko mengetahui kapan harus melakukan pemesanan barang.

#### DAFTAR PUSTAKA

- RADIANT VICTOR IMBAR, ERIC TIRTA. 2007. *Analisa, Perancangan dan Implementasi Sistem Informasi Penjualan Pelumas Studi Kasus : Perusahaan "PT. Pro Roll International"*, Jurusan S1 Sistem Informasi Fakultas Teknologi Informasi Universitas Kristen Maranatha Bandung. Jurnal Informatika, Vol. 3. No 1, Juni 2007:119-149
- SUPRAYITNO, ULI INDAH WARDATI. 2012. *Pembangunan Sistem Stok Barang dan Penjualan pada Toko Sero Elektronik*. Fakultas Teknologi Informatika Universitas Surakarta. Indonesian Jurnal on Computer Science Speed-FTI UNSA Vol 9 No 3 – Desember 2012 – ISSN:1979-9330(Print) – 2088-0154(Online) – 2088-0162(CDROM)
- MELIANA CHRISTIANI, ERIC KURNIAWAN. 2009. *Sistem Informasi Toko Atom Komputer untuk Mengelola Proses Penjualan dan Pembelian Barang Menggunakan PHP dan Openwave*. Jurusan Teknik Informatika, Fakultas Teknologi Informasi Universitas Kristen Maranatha Bandung. Jurnal Sistem Informasi, Vol. 4, No.2, September 2009:111-119
- KADIR, ABDUL. 2003. *Pengenalan Sistem Informasi*. Yogyakarta: Andi
- MAROM, CHAIRUL. 2002. *Sistem Akuntansi Perusahaan Dagang*. Grasindo: Jakarta.
- BASU SWASTHA, 2001. *Manajemen Pemasaran Modern*; Yogyakarta: BPFE
- MARTIN S, COLLERAN G, 2006, *Sold! Bagaimana Memudahkan Konsumen Membeli dari Anda*, Erlangga, Jakarta.
- NISWONGER, C. ROLLING; E.FESS, PHILIP; S.WARRE, CARL. 1999. *Prinsip-prinsip Akuntansi*. Jilid-1. Edisi ke-19. Erlangga, Jakarta
- IKATAN AKUTANSI INDONESIA, 2008, *Pernyataan Standar Akuntansi Keuangan*, Jakarta : Salemba Empat
- AHMED RIAHI DAN BELKAOUI, 2000, *Teori Akuntansi*, Jakarta: Salemba Empat.
- A. DUNIA, FIRDAUS, 2005, *Pengantar Akutansi 2*, Edisi revisi, Fakultas Ekonomi – Universitas Indonesia, Jakarta.
- JAMES M. REEVE, 2008, *Pengantar Akuntansi*, Jakarta: Salemba Empat.
- E. KIESO, DONALD, JERRY J, WEYGANDT AND TEERY D. WARFIELD, 2007, *Intermediate Accounting*, Edisi 12: by Erlangga.
- W. KOMPUTER, “ Panduan Aplikatif dan Solusi Membuat Aplikasi Client Server dengan Visual Basic 2008”, Yogyakarta : Penerbit Andi, 2010.
- A. NUGROHO DAN A. ASHARI, “ DCOM, CORBA, JAVA RMI: Konsep dan Teknik Dasar Pemrograman”, Jurnal Sistem Informasi, Vol.7, No.2, Oktober 2011.
- S. LESTARI, M. SOMANTRI, R. R. ISNANTO, “Implementasi Java RMI pada Rancang Bangun Tes Toefl Online Berbasis Web”, TRANSMISI, 13 (3), pp. 103-107, 2011.
- B. SUSANTO, “Pemrograman Client/Server dengan Java2”, Jakarta : Elex Media Komputindo, 2003.
- PRESSMAN, R.S. 2001. *Software Engineering : A Practitioner's Approach*. Amerika Serikat : R.S. Pressman and Associates.



JURNAL TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA



9 772355 769000