

## 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>brillianarc@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 *Laustrialian* 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:

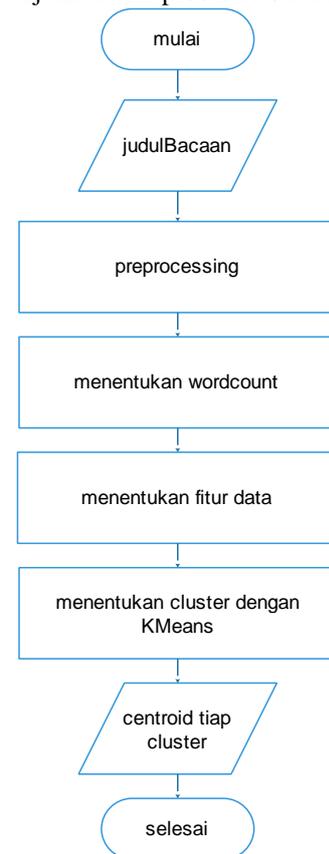
$$v_j = \frac{\sum_{i=1}^n m(c_j|x_i)x_i}{\sum_{i=1}^n m(c_j|x_i)} \text{ for } j = 1, \dots, k \quad (1)$$

- 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;

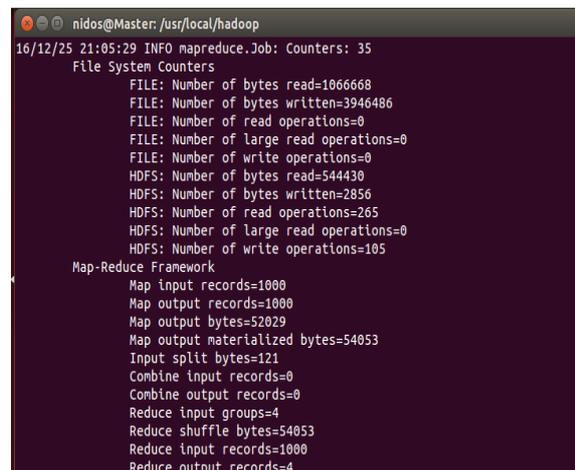
System.out.println("Should continue: "+
shouldContinue);

}
}
}

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
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
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.