# *CLUSTERING* DAN GEOVISUALISASI DATA TEKS TWITTER DENGAN ALGORITME K-MEANS UNTUK KASUS KEBAKARAN HUTAN DAN BENCANA ALAM

**HAMID DAROJAT**

**DEPARTEMEN ILMU KOMPUTER**
**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**
**INSTITUT PERTANIAN BOGOR**
**BOGOR**
**2017**

# PERNYATAAN MENGENAI SKRIPSI DAN SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa skripsi berjudul *Clustering* dan Geovisualisasi Data Teks Twitter dengan Algoritme K-Means untuk Kasus Kebakaran Hutan dan Bencana Alam adalah benar karya saya dengan arahan dari komisi pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka di bagian akhir skripsi ini.

Dengan ini saya melimpahkan hak cipta dari karya tulis saya kepada Institut Pertanian Bogor.

Bogor, Februari 2017

*Hamid Darojat*
NIM G64144022

# ABSTRAK

HAMID DAROJAT. *Clustering* dan Geovisualisasi Data Teks Twitter dengan Algoritme K-Means untuk Kasus Kebakaran Hutan dan Bencana Alam. Dibimbing oleh MUHAMMAD ABRAR ISTIADI.

Twitter merupakan *microblogging* yang mampu menyebarkan informasi penting pada masyarakat pengguna internet. Twitter memiliki fitur *tweet* yang berisi data teks, *longitude, latitude,* dan lain-lain. Penelitian ini bertujuan untuk melakukan *clustering* dengan algoritme K-Means pada data teks Twitter terkait isu kebakaran hutan dan bencana alam di Indonesia. Data penelitian ini berupa data *tweet* yang dikumpulkan berdasarkan *hashtag* tentang kebakaran hutan dan bencana alam. Penelitian ini juga menyajikan hasil *clustering* data teks Twitter dengan geovisualisasi yang direpresentasikan pada peta. Algoritme K-Means berhasil diimplementasikan pada *Term Document Matrix* dari praproses data Teks Twitter dengan pemograman R. Proses geovisualisasi berhasil diimplementasikan pada data Twitter dengan *framework* R Shiny. Hasil implementasi algoritme K-Means pada data teks Twitter menggunakan nilai *k* sebesar 7 dan nilai *sum squared error* (SSE) yang dihasilkan 5693.169. Geovisualisasi hasil *clustering* data teks Twitter menyebar di seluruh Indonesia berdasarkan *longitude* dan *latitude* tiap data *tweet* tersebut. Tidak semua kategori bencana membentuk *cluster* sendiri, melainkan masuk pada *cluster* yang dominan.



Kata kunci: *clustering, tweet,* geovisualisasi



# ABSTRACT

HAMID DAROJAT. Clustering and Geovisualization of Twitter Text Data using K-Means Algorithm for Forest Fires and Natural Disasters. Supervised by MUHAMMAD ABRAR ISTIADI.

Twitter is a microblogging platform that can spread important information to internet community. Twitter has tweet feature that contain text data, longitude, latitude and the others. This study aims to cluster Twitter text data using K-Means algorithm for forest fire and natural disaster issues. The data used are tweet data collected based on hashtags related to forest fires and natural disasters. This study also presents the clustering result of Twitter text data with geovisualization that represented on the map. K-Means algorithm has been implemented on Term Document Matrix from preprocessing of Twitter text data in R programming. Geovisualization process has been implemented on the data using R Shiny framework. The implementation results with K-Means algorithm use 7 as the best *k* value with 5693.169 sum squared error (SSE). Third cluster has the most members. Geovisualization of the clustering result spread throughout Indonesia by longitude and latitude for every tweet data. Not all disaster categories become cluster of their own, but assigned to the dominant cluster.



Keywords: clustering, tweet, geovisualization

# *CLUSTERING* DAN GEOVISUALISASI DATA TEKS TWITTER DENGAN ALGORITME K-MEANS UNTUK KASUS KEBAKARAN HUTAN DAN BENCANA ALAM

**HAMID DAROJAT**

Skripsi
sebagai salah satu syarat untuk memperoleh gelar
Sarjana Komputer
pada
Departemen Ilmu Komputer

**DEPARTEMEN ILMU KOMPUTER**
**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**
**INSTITUT PERTANIAN BOGOR**
**BOGOR**
**2017**

Penguji:

1 Husnul Khotimah, SKomp MKom
2 Rina Trisminingsih, SKomp MT

Judul Skripsi : *Clustering* dan Geovisualisasi Data Teks Twitter dengan Algoritme K-Means untuk Kasus Kebakaran Hutan dan Bencana Alam
Nama : Hamid Darojat
NIM : G64144022

Disetujui oleh

Muhammad Abrar Istiadi, SKomp MKom
Pembimbing

Diketahui oleh

Dr Ir Agus Buono, MSi MKom
Ketua Departemen

Tanggal Lulus:

# PRAKATA

*Alhamdulillah wa syukurillah* penulis panjatkan kepada Allah *subhanahu wa ta'ala* atas segala karunia-Nya sehingga karya ilmiah ini berhasil diselesaikan. Tema yang dipilih dalam penelitian yang dilaksanakan sejak bulan Agustus 2016 ini ialah *data mining*, dengan judul *Clustering* dan Geovisualisasi Data Teks Twitter dengan Algoritme K-Means untuk Kasus Kebakaran Hutan dan Bencana Alam.

Penulis menyadari bahwa dalam proses penulisan skripsi ini banyak mengalami kendala dan masalah, namun berkat bantuan, bimbingan, kerjasama dari berbagai pihak dan berkah dari Allah *subhanahu wa ta'ala* sehingga kendala-kendala yang dihadapi tersebut dapat diatasi. Penulis mengucapkan terima kasih kepada Orang Tua dan keluarga yang selalu memberi dukungan moril dan materiil. Serta ucapan terima kasih dan penghargaan kepada Bapak Muhammad Abrar Istiadi, SKomp MKom selaku pembimbing yang telah dengan sabar, tekun, tulus dan ikhlas meluangkan waktu, tenaga, dan pikiran memberikan bimbingan, motivasi, arahan, dan saran-saran yang sangat berharga kepada penulis selama menyusun skripsi.

Penulis juga menyampaikan terima kasih kepada:

1 Ibu Husnul Khotimah, SKomp MKom dan Ibu Rina Trisminingsih, SKomp MT selaku penguji.
2 Bapak Dr Ir Agus Buono, MSi MKom selaku Ketua Departemen Ilmu Komputer IPB.
3 Seluruh dosen, staf tata usaha, dan staf pegawai Departemen Ilmu Komputer IPB.
4 Seluruh teman-teman Program S1 Alih Jenis Ilmu Komputer IPB Angkatan 9.

Semoga segala bantuan, bimbingan, motivasi, dan dukungan yang telah diberikan kepada penulis senantiasa dibalas oleh Allah *subhanahu wa ta'ala*. Semoga karya ilmiah ini bermanfaat bagi semua pihak yang membutuhkan.

Bogor, Februari 2017

*Hamid Darojat*

# DAFTAR ISI


| | |
|---|---|
| DAFTAR TABEL | vi |
| DAFTAR GAMBAR | vi |
| DAFTAR LAMPIRAN | vi |
| PENDAHULUAN | 1 |
| Latar Belakang | 1 |
| Perumusan Masalah | 2 |
| Tujuan Penelitian | 2 |
| Manfaat Penelitian | 2 |
| Ruang Lingkup Penelitian | 2 |
| METODE | 3 |
| Data Penelitian | 3 |
| Tahapan Penelitian | 3 |
| Akuisisi Data Twitter | 4 |
| Praproses Data | 4 |
| *Clustering* dengan K-Means | 6 |
| Geovisualisasi dengan *Framework* Shiny | 6 |
| Evaluasi Hasil Analisis | 7 |
| Lingkungan Pengembangan | 7 |
| HASIL DAN PEMBAHASAN | 8 |
| Akuisisi Data Twitter | 8 |
| Praproses Data | 10 |
| *Clustering* dengan K-Means | 11 |
| Geovisualisasi dengan *Framework* Shiny | 13 |
| Evaluasi Hasil *Clustering* | 16 |
| SIMPULAN DAN SARAN | 18 |
| Simpulan | 18 |
| Saran | 18 |
| DAFTAR PUSTAKA | 18 |
| LAMPIRAN | 21 |
| RIWAYAT HIDUP | 29 |

# DAFTAR TABEL


| | | |
|---|---|---|
| 1 | Daftar *hashtag* yang digunakan untuk pencarian *tweet* | 3 |
| 2 | Daftar atribut data *tweet* yang diperoleh dari hasil akuisisi data | 4 |
| 3 | Kategori dari *hashtag* | 7 |
| 4 | Jumlah *tweet* berdasarkan *keywords* | 10 |
| 5 | Data *tweet* sebelum dan sesudah proses *tokenizing* | 11 |
| 6 | Data *tweet* sebelum dan sesudah normalisasi kata | 12 |
| 7 | Data teks Twitter sebelum dan sesudah penghapusan *stopwords* | 12 |
| 8 | Data teks Twitter sebelum dan sesudah *stemming* | 12 |
| 9 | Hasil term dan *sparsity* | 12 |
| 10 | Nilai SSE untuk hasil *clustering* nilai $k$ = 2 hingga 7 | 13 |
| 11 | *Package* dan fungsinya | 15 |
| 12 | Hasil *clustering* | 16 |
| 13 | Hasil eksplorasi data | 17 |
| 14 | *Confusion matrix* kemunculan kategori pada tiap *cluster* | 17 |
| 15 | Pelabelan tiap *cluster* | 17 |


# DAFTAR GAMBAR


| | | |
|---|---|---|
| 1 | Tahapan penelitian | 3 |
| 2 | Tahapan praproses data | 4 |
| 3 | API *key* dan API *secret* | 9 |
| 4 | Token *access* dan Token *secret* | 9 |
| 5 | Pencarian data *tweet* | 9 |
| 6 | Fungsi browser.get pada program Python | 9 |
| 7 | Contoh pembuatan TDM | 13 |
| 8 | *Datatable* dari menu Data explorer | 14 |
| 9 | *Datatable* untuk menu K-Means *clustering* | 15 |
| 10 | *Interactive Map* | 16 |


# DAFTAR LAMPIRAN


| | | |
|---|---|---|
| 1 | Kode program Python untuk akuisisi data Twitter | 21 |
| 2 | Kode program fungsi pada server.r | 23 |
| 3 | Kode program untuk *interface* sistem pada ui.r | 26 |
| 4 | Visualisasi sistem | 27 |

# PENDAHULUAN

## Latar Belakang

Situs *microblogging* seperti Twitter mampu memainkan peran penting dalam menyebarkan informasi tentang bencana alam. Volume dan kecepatan *tweet* yang dikirim cenderung sangat tinggi sehingga masyarakat yang terkena bencana dan para tanggap bencana profesional membutuhkan suatu *tools* untuk memproses informasi dengan tepat (Imran *et al*. 2013).

Kelebihan media sosial untuk mengomunikasikan informasi penting dari masyarakat telah terbukti efektif selama beberapa tahun terakhir. Seperti penelitian yang telah dilakukan oleh Crooks *et al.* (2012) bahwa media sosial Twitter bisa menjadi sensor sistem yang dapat memberikan informasi kepada masyarakat tentang terjadinya gempa bumi. Hal tersebut terjadi karena Twitter memiliki *hashtag* tertentu, *longitude*, dan *latitude* yang bisa memberikan informasi lokasi *tweet* tersebut di-*posting*. Sakaki *et al.* (2013) telah meneliti bahwa data *tweet* ketika bencana gempa terjadi dapat dijadikan pendeteksi gempa dengan skala *richter* sebesar 3 atau lebih. Probabilitas yang dihasilkan tinggi yaitu 96% dari *Japan Meteorologi Agency* (JMA).

Data hasil rekapitulasi dari KLKH (2016) luas kebakaran hutan dan lahan (Ha) di Indonesia dari tahun 2011 sampai dengan tahun 2015 mengalami peningkatan. Luas total kebakaran paling parah terjadi pada tahun 2015 yaitu 261.060,44 Ha. Berdasarkan data tingkat kewaspadaan terhadap kebakaran hutan terhitung dari 1 Januari 2013 sampai dengan 15 September 2015 yang dikumpulkan oleh *World Resource Institute* (WRI) menunjukkan bahwa tingkat kewaspadaan mencapai puncaknya di angka 1189 pada 8 September 2015. Berdasarkan rekapitulasi BNPB (2015) bencana alam yang sering terjadi di Indonesia yaitu banjir, gemba bumi, puting beliung, tanah longsor, dan letusan gunung api. Jumlah total bencana yang terjadi pada periode 1 Januari sampai dengan 30 November 2015 sebanyak 1482 kejadian.

Beberapa penelitian menggunakan data dari media sosial Twitter telah dilakukan sebelumnya. Denatari (2015) melakukan penelitian tentang *clustering* data teks Twitter untuk kasus pertanian Indonesia. Data Twitter yang diuji sejumlah 51 data *tweet* dan data konten *uniform resource locator* yang diuji sejumlah 51 data. Kedua data tersebut dibandingkan dan dikelompokkan dengan algoritme *hierarchial clustering* untuk mendapatkan c*luster* terbaiknya.

Penelitian yang dilakukan oleh Susanto *et al*. (2014) menggunakan *tweet* berbahasa Indonesia dengan menggunakan teknik *clustering* untuk menganalisis *sentiment tweet* dengan topik pemilu dan membuat visualisasinya. Penelitian tersebut menggunakan algoritme *clustering* K-Means, *cascade* K-Means, dan *self organizing map* (SOM) *Kohonen*. Tujuan dari penelitan tersebut yaitu membandingkan ketiga algoritme tersebut.

Nadilah (2016) meneliti asosiasi dan geovisualisasi terkait kasus kebakaran hutan dengan data cuaca di Provinsi Riau dan Kepulauan Riau. Penelitian dilakukan untuk menganalisis asosiasi dengan algoritme Apriori terhadap *tweet* terkait isu kebakaran hutan dan cuaca. Informasi disajikan dengan geovisualisasi yang direpresentasikan dengan peta.

Pada penelitian ini dilakukan *clustering* data teks Twitter terkait kasus kebakaran hutan dan bencana alam. Algoritme yang digunakan pada penelitian ini yaitu K-Means. Sebelum proses *clustering*, dilakukan tahap praproses seperti *tokenizing*, normalisasi kata, penghapusan *stopwords*, dan *stemming* sehingga menghasilkan data teks Twitter menjadi *Term Document Matrix* (TDM). Data hasil *clustering* divisualisasikan dengan menggunakan *framework shiny* untuk menerapkan geovisualisasi data teks Twitter terkait kebakaran hutan dan bencana alam. Proses geovisualisasi dilakukan dengan tujuan menampilkan data *tweet* hasil *clustering*-nya dan lokasi kebakaran hutan dan bencana alam. Pengujian dilakukan dengan mengevaluasi hasil analisis *clustering* dengan mencari nilai SSE-nya.

## Perumusan Masalah

Rumusan permasalahan pada penelitian ini adalah sebagai berikut:

1 Bagaimana penerapan algoritme *clustering* K-Means untuk mengolah data teks Twitter?
2 Bagaimana menerapkan geovisualisasi hasil *clustering* data teks Twitter terkait kebakaran hutan dan bencana alam?

## Tujuan Penelitian

Tujuan penelitian ini adalah:

1 Mengimplementasikan algoritme *clustering* K-Means untuk mengolah data teks Twitter.
2 Geovisualisasi hasil *clustering* data teks Twitter terkait kebakaran hutan dan bencana alam dengan *framework* R Shiny.

## Manfaat Penelitian

Hasil *clustering* dan geovisualisasi data *tweet* terkait isu kebakaran hutan dan bencana alam di Indonesia diharapkan dapat bermanfaat untuk memudahkan proses penarikan informasi tentang lokasi *tweet* kebakaran hutan dan bencana alam di Indonesia.

## Ruang Lingkup Penelitian

Ruang lingkup penelitian ini ialah:

1 Penelitian ini menggunakan data *tweet* dengan *hashtag* tentang kebakaran hutan dan bencana alam di Indonesia tahun 2015.
2 Data *tweet* yang digunakan yaitu *tweet* berbahasa Indonesia.
3 Proses geovisualisasi hanya untuk *clustering* data teks Twitter dengan algoritme K-Means.

# METODE

## Data Penelitian

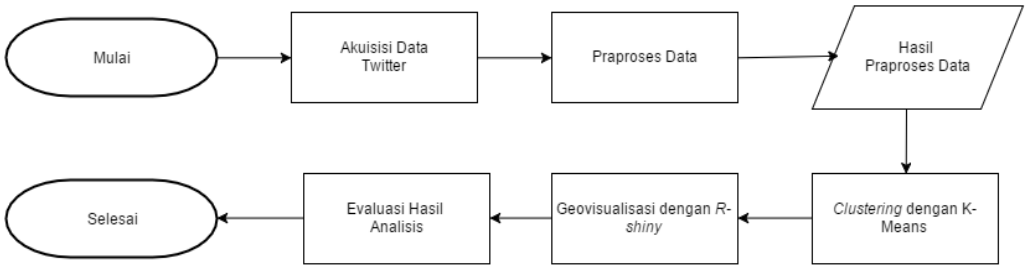
Data yang digunakan pada penelitian ini adalah data teks Twitter terkait kebakaran hutan dan bencana alam di Indonesia tahun 2015. Penelitian ini menggunakan *hashtag* terkait isu kebakaran hutan dan bencana alam di Indonesia sebagai data sampel *tweet*. Pencarian *hashtag* kebakaran hutan merujuk pada *hashtag* paling populer selama tahun 2015 yang telah dirangkum oleh Simangungsong (2015) dan pada penelitian Nadilah (2016). Pencarian *hashtag* bencana alam merujuk pada rekapitulasi bencana oleh BNPB (2015). Daftar *hashtag* dapat dilihat pada Tabel 1 sedangkan nama dan atribut dari data *tweet* yang telah diakusisi dapat dilihat pada Tabel 2.

Tabel 1 Daftar *hashtag* yang digunakan untuk pencarian *tweet*

| No | *Hashtag* |
|---|---|
| 1 | #kebakaranhutan |
| 2 | #melawanasap |
| 3 | #daruratasap |
| 4 | #melawanapi |
| 5 | #maribersikap |
| 6 | #kabutasap |
| 7 | #banjir |
| 8 | #tanahlongsor |
| 9 | #gempabumi |
| 10 | #putingbeliung |
| 11 | #gunungmeletus |
| 12 | #erupsi |

## Tahapan Penelitian

Tahapan pada penelitian ini berupa akuisisi data Twitter, praproses data, *clustering* dengan K-Means, geovisualisasi dengan *framework* Shiny, dan evaluasi hasil analisis. Masing-masing tahapan tersebut dapat dilihat pada Gambar 1.




Gambar 1 Tahapan penelitian

## Akuisisi Data Twitter

Tahap akuisisi ini dilakukan untuk mendapatkan data *tweet* terkait isu kebakaran hutan dan bencana alam tahun 2015. Akuisisi data Twitter menggunakan proses *web crawling* yang ada pada program Python dan Twitter API. *Web crawling* dan *scraping* dengan *library* Selenium pada Python merupakan metode yang digunakan untuk mendapatkan data *tweet*. Program Python yang digunakan merujuk dari *source code* pada penelitian Nadilah (2016).

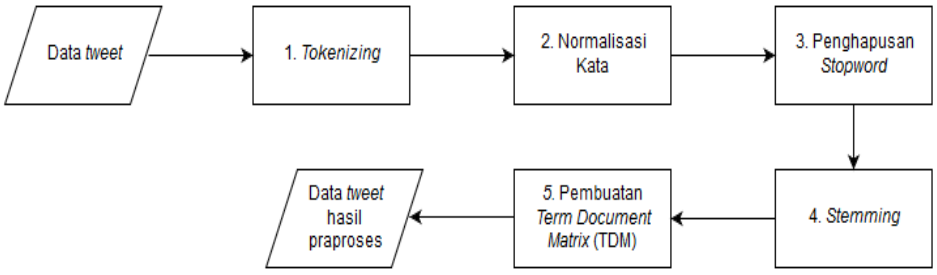
Proses akuisisi menggunakan *browser* Google Chrome yang dikombinasikan dengan aplikasi Chromedriver. Kemudian akuisisi dilakukan dengan memasukkan *hashtag* yang telah ditentukan dan memasukkan tanggal 1 Januari 2015 sampai dengan 31 Desember 2015 pada halaman pencarian Twitter. Tahap akuisisi ini juga menghasilkan *output* dalam format CSV. Metode *web crawling* dan *scraping* ini menghasilkan *username*, *date*, *time*, *text*, dan *place id*. Twitter API digunakan untuk mendapatkan data lokasi, pusat kota, *longitude* dan *latitude*. Program Python dimodifikasi pada penelitian Nadilah (2016) untuk mendapatkan data *geolocation* secara otomatis.

Tabel 2 Daftar atribut data *tweet* yang diperoleh dari hasil akuisisi data

| No | Nama Atribut Data *Tweet* | Sumber Data |
|---|---|---|
| 1 | *tweet id* | Twitter |
| 2 | *Username* | Twitter |
| 3 | *Date* | Twitter |
| 4 | *Time* | Twitter |
| 5 | *text* | Twitter |
| 6 | *place id* | Twitter |
| 7 | lokasi | Twitter API |
| 8 | pusat kota | Twitter API |
| 9 | *longitude* | Twitter API |
| 10 | *latitude* | Twitter API |

## Praproses Data

Praproses data dilakukan untuk mendapatkan data yang dibutuhkan pada proses *clustering*. Pada tahap ini data *tweet* memasuki tahap praproses data. Implementasi kode program *text mining* menggunakan R dari penelitian Wahyuningtyas (2016). Tahap praproses data terbagi lagi menjadi 5 tahap seperti pada Gambar 2.




Gambar 2 Tahapan praproses data

***Tokenizing***

*Tokenizing* merupakan pengambilan kata-kata (term) dari kumpulan dokumen menjadi kumpulan term. *Tokenizing* dilakukan dengan melakukan instalasi *package* koleksi dokumen teks "tm" pada R. Data *tweet* diubah menjadi *dataset* dengan mengubah menjadi format *comma separated values* (CSV). Dataset *tweet* kemudian diubah menjadi bentuk *corpus*. *Corpus* merupakan entitas secara konseptual mirip dengan *database* dalam menyimpan dan mengolah dokumen teks. Dengan kata lain *corpus* adalah entitas sebuah kumpulan dokumen teks (Meyer *et al.* 2008). Semua huruf pada *corpus* diubah menjadi huruf kecil dengan menggunakan fungsi *tolower*. Fungsi *gsub* pada R digunakan untuk menghapus *mention*, URL, tanda baca, dan angka.

**Normalisasi Kata**

Banyak *tweet* yang mengandung kata yang tidak baku. Kata yang tidak baku dapat berupa perulangan karakter sehingga sulit dikenali, tanggal, jumlah mata uang, dan akronim. Aziz (2013) melakukan normalisasi teks dengan penghilangan karakter berulang dan penggantian kata tidak baku menjadi baku. Pada penelitian ini dilakukan penggantian kata tidak baku menjadi baku berdasarkan kamus kata pada penelitian Aziz (2013).

**Penghapusan *Stopwords***

*Stopwords* adalah sebuah kata-kata dalam bahasa tertentu yang sangat umum dan kebanyakan tidak memiliki nilai informasi (Meyer *et al.* 2008). *Stopwords* mempunyai definisi term yang tidak berhubungan dengan sesuatu data teks meskipun term tersebut sering muncul pada data teks. Tahap penghapusan *stopwords* dilakukan untuk membuang kata-kata yang termasuk dalam daftar *stopwords.* Beberapa contoh kata yang termasuk dalam daftar *stopwords* adalah "yang", "di", "ke", "dari", "adalah", "dan", "atau", dan "semua". Acuan daftar *stopwords* didapatkan dari penelitian yang telah dilakukan oleh Tala (2003) sebanyak 759 kata dalam bahasa Indonesia.

***Stemming***

*Stemming* merupakan proses konversi term ke bentuk kata dasarnya. Proses *stemming* dilakukan untuk menghapus awalan (*prefix*) dan akhiran (*suffix*). Tujuan dari *stemming* ini adalah untuk mendapatkan kata dasar yang sesuai. Proses *stemming* ini menggunakan kode program yang terdapat pada penelitian Wahyuningtyas (2016) untuk menghapus berbagai variasi awalan dan akhiran.

**Pembuatan *Term Document Matrix* (TDM)**

Proses pembuatan *Term Document Matrix* menggunakan fungsi *TermDocumentMatrix* pada R. Menurut Nadilah (2016) tahap pembuatan *Term Document Matrix* (TDM) dilakukan untuk membuat matriks jumlah kemunculan suatu term pada dokumen. Satu *tweet* menandakan satu dokumen. Pada proses ini dilakukan reduksi term untuk memperkecil dimensi matriks dengan fungsi *removeSparseTerm* pada R. *Sparse* memiliki rentang nilai term 0 dan 1. Kolom matriks pada TDM menunjukkan kata yang ada pada data *tweet,* sedangkan baris matriks menunjukkan indeks dari dokumen pada kumpulan *corpus*. Hasil praproses data dalam bentuk TDM digunakan pada tahap *clustering*.

## *Clustering* dengan K-Means

Menurut Han *et al*. (2012) algoritme K-Means merupakan teknik berbasis *centroid* dan termasuk salah satu metode *partitioning* atau mempartisi data yang ada ke dalam bentuk dua atau lebih kelompok. Tujuan pengelompokan data ini adalah meminimalkan fungsi objektif yang diatur dalam proses pengelompokan. Pada umumnya proses pengelompokan menimalkan variasi dalam suatu kelompok dan memaksimalkan variasi antar kelompok.

Secara garis besar algoritme K-means sebagai berikut:

1. Memilih *k* objek sebagai *centroid* (titik pusat awal dari *cluster*).
2. Masukkan objek-objek ke dalam *cluster* yang objeknya paling mirip berdasarkan nilai rata-rata dari objek-objek yang berada dalam sebuah *cluster*.
3. Memperbaharui *centroid* dengan menghitung nilai rata-rata dari objek-objek tiap *cluster*.
4. Melakukan iterasi sampai objek-objek yang berada pada *cluster* tidak ada yang berbeda.

Pada tahapan ini dilakukan *clustering* dengan menggunakan algoritme K-Means untuk data Twitter yang telah dipraproses menjadi data *Term Document Matrix*. Proses *clustering* menggunakan nilai *k* sebesar 2 sampai dengan 7. *Clustering* digunakan untuk mendapatkan *cluster* setiap dokumen berdasarkan term terkait kebakaran hutan dan bencana alam untuk proses geovisualisasi. Penentuan kelas dengan maksimal 7 berdasarkan hasil pembagian kategori dari *hashtag* yang digunakan. Kategori untuk setiap *hashtag* dapat dilihat pada Tabel 3.

Kualitas sebuah *cluster* dihitung dengan menggunakan *Sum Squared Error* (SSE). *Error* merupakan jarak tiap titik diukur ke *cluster* yang terdekat, *p* adalah titik yang merepresentasikan objek data, objek pada penelitian ini merupakan term, *k* merupakan jumlah *cluster* yang dibentuk, $C_i$ merupakan *centroid* dari kelas *i*, dan *dist* merupakan fungsi jarak. Kelas yang memiliki SSE terkecil merupakan *k* terbaik untuk *clustering*. Semakin kecil nilai SSE menunjukkan *instance* lebih seragam pada *cluster* yang dikelompokkan. Nilai SSE dapat dirumuskan pada Persamaan 1.

Zwitch (2013) telah menganalisis term yang digunakan *visitor* untuk mengunjungi suatu laman (*search keywords*) dan term yang digunakan pada saat berada di laman tersebut (*on-site search*) dengan algoritme K-Means. Tujuan dari penelitiannya yaitu untuk melakukan pelabelan pada setiap *cluster*. Penentuan nilai *k* dilakukan untuk memperkirakan topik atau kategori setiap *cluster* sebagai asumsi awal pencarian term yang dominan sebelum melakukan pelabelan.

## Geovisualisasi dengan *Framework* Shiny

Menurut Yasobant *et al.* (2015) geovisualisasi merupakan proses analisis data geospasial dengan visualisasi yang diaktifkan melalui alat konvergensi informasi, kartografi dan metode geografi. Proses geovisualisasi membutuhkan proses *geocoding* (*longitude* dan *latitude*). Visualisasi dapat dilakukan dengan membuat *point patterns, line patterns,* dan *area patterns*. Penelitian ini menggunakan *point patterns* sebagai dasar visualisasi untuk mendistribusikan data

hasil *clustering* berdasarkan *geolocation* setiap data *tweet.* Menurut Nadilah (2016) geovisualisasi digunakan dalam proses eksplorasi geospasial yang direpresentasikan melalui peta. Informasi yang diperlukan dalam geovisualisasi dapat berupa *longitude* atau *latitude*. Implementasi geovisualisasi dengan pemilihan fitur warna, pola, ukuran dan bentuk dapat memudahkan orang untuk memahami informasi yang didapat dari geovisualisasi tersebut.

Tabel 3 Kategori dari *hashtag*

| No | *Hashtag* | Kategori |
|---|---|---|
| 1 | #kebakaranhutan, #maribersikap, #melawanapi | Kebakaran Hutan |
| 2 | #kabutasap, #melawanasap, #daruratasap | Asap |
| 3 | #putingbeliung | Puting beliung |
| 4 | #banjir | Banjir |
| 5 | #tanahlongsor | Tanah longsor |
| 6 | #gempabumi | Gempa bumi |
| 7 | #erupsi, #gunungmeletus | Erupsi |

$$\text{SSE}= k \sum_{i=1} \sum_{n \epsilon \text{Ci}} \text{dist}(\text{p}, Ci)^2 \qquad (1)$$

Shiny merupakan *package* dari bahasa pemograman R untuk membuat aplikasi berbasis web yang interaktif dan mudah. Shiny memiliki fungsi reaktif yang otomatis yaitu ui.r dan server.r. Kedua fungsi tersebut menjadi dasar *input* dan *output* suatu aplikasi dan bisa digunakan untuk membangun aplikasi yang *user friendly*, *responsif* dan dapat diintegerasikan dengan bermacam-macam *widget* pendukung.

Pada tahapan ini dilakukan visualisasi data *tweet* hasil *clustering* menggunakan *library* Leaflet pada R. Proses visualisasi juga menggunakan *framework* Shiny dengan membuat dua *file* yaitu *file* ui.r dan *file* server.r. *File* ui.r merupakan kumpulan baris program yang merepresentasikan *interface* atau antar muka yang akan ditampilkan. *File* server.r merupakan kumpulan baris program berisi fungsi-fungsi yang digunakan pada aplikasi yang dibangun.

## Evaluasi Hasil Analisis

Tahapan ini menganalisis hasil *cluster* dari *clustering* dengan K-Means. Analisis dilakukan dengan melihat kategori yang sering muncul dalam satu *cluster*. Analisis juga dilakukan dengan melihat term yang dominan dan relevan pada masing-masing *cluster* sebagai dasar pemberian label pada data geovisualisasi.

## Lingkungan Pengembangan

Spesifikasi perangkat keras dan perangkat lunak yang digunakan untuk penelitian ini adalah sebagai berikut:

1 Perangkat keras yang digunakan berupa komputer personal dengan spesifikasi:

- Intel® Core™ i5 CPU @2.50 GHz.

- RAM 4 GB.
- *Harddisk Internal* 320 GB.

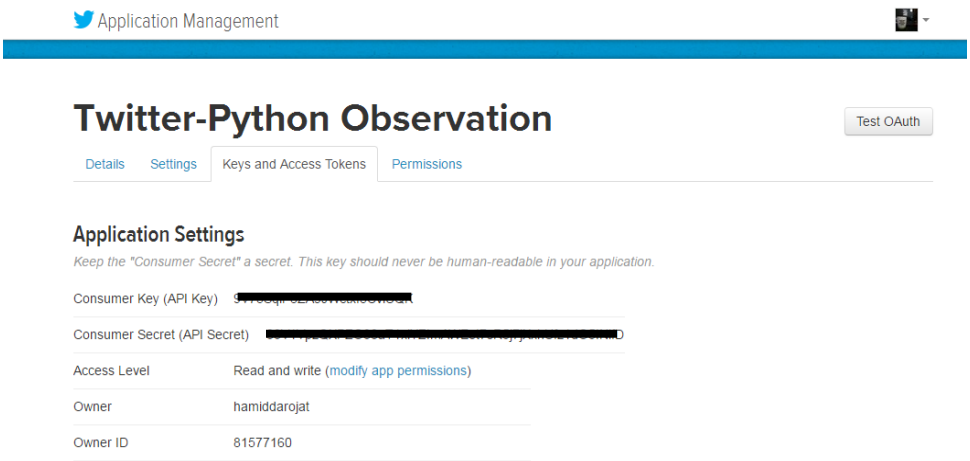
2 Perangkat lunak yang digunakan:
- Sistem Operasi Windows 7 32-bit.
- Bahasa pemrograman R versi 3.3.0 untuk menjalankan R Studio.
- RStudio versi  dengan *framework* Shiny untuk membangun aplikasi dan *package* Leaflet untuk membuat *interactive map*.
- Microsoft Excel  mengolah data Twitter.
- MySQL Xampp versi 3.2.2 sebagai *server* basis data
- Python 3. 5. 2.
- Selenium 2.53.
- Chromedriver 2.27.

# HASIL DAN PEMBAHASAN

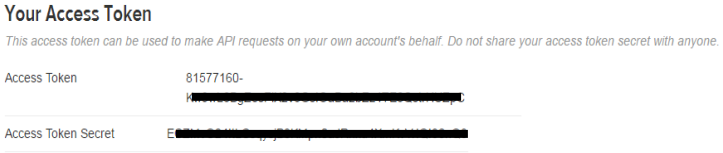
## Akuisisi Data Twitter

Hasil akuisisi data *tweet* didapatkan dengan metode *web crawling* dan *scraping* dengan *script* Python. *Script* Python dikombinasikan dengan Selenium dan Chromedriver untuk mendapatkan data *tweet.* Modifikasi *source code* program Python dilakukan untuk mendapatkan data *geolocation* secara otomatis. Data *tweet* disimpan dalam format *file* CSV. Kode program yang digunakan untuk mendapatkan data *tweet* dapat dilihat pada Lampiran 1. Langkah-langkah mengambil data *tweet* adalah sebagai berikut:

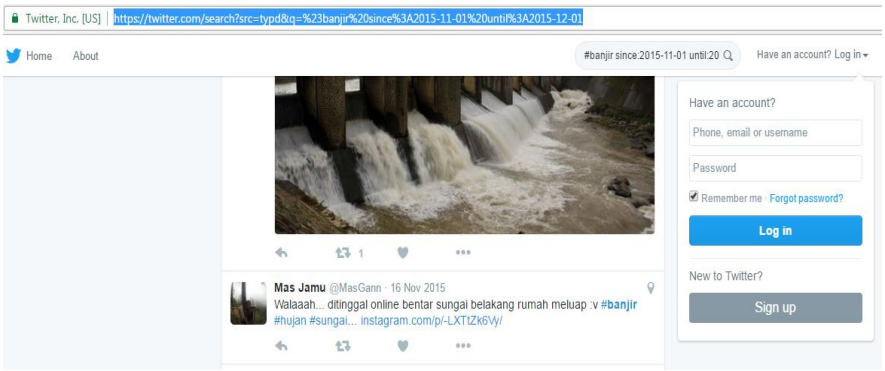
1 Membuat koneksi ke Twitter API
  Membuat manajemen aplikasi terlebih dahulu pada https://apps.twitter.com/ untuk mendapatkan *consumer key* (API *key*), *consumer secret* (API *secret*), *token,* dan *token secret.* Contoh hasil pembuatan manajemen aplikasi dapat dilihat pada Gambar 3 dan Gambar 4. *Key* dan *token* digunakan pada proses *authorization* untuk mendapatkan data *geolocation* langsung dari *script* Python. Data *geolocation* yaitu berupa atribut pusat kota, lokasi, *longitude,* dan *latitude*. Proses *authorization* dibuat dengan menggunakan fungsi API seperti pada kode program pada Lampiran 1.
2 Pencarian menggunakan fungsi *searchtweet* pada *platform* Twitter dengan parameter (#*keyword* since: yyyy-mm-dd until: yyyy-mm-dd) seperti pada Gambar 5. Contoh pemanggilan fungsi dapat dilihat pada Gambar 6 untuk contoh pencarian: #banjir *since*:2015-11-01 *until* 2015-09-26. *Query* ini berfungsi untuk mengambil *hashtag*  banjir dari tanggal 26 September 2015 sampai dengan 1 November 2015.
3 Mengatur jumlah *scroll* untuk halaman pencarian *tweet* dengan parameter (n). *Scrolling* halaman pencarian *tweet* dilakukan sebanyak n tersebut. Penentuan *scroll* diperkirakan banyaknya *tweet* dan rentang tanggal yang ditentukan.
4 Hasil *running* program tersebut disimpan dalam bentuk CSV dengan *header* kolom *username, date, time, text, tweetID, placeID,* pusat kota, lokasi, *longitude,* dan *latitude*.




Gambar 3 API *key* dan API *secret*




Gambar 4 Token *access* dan Token *secret*




Gambar 5 Pencarian data *tweet*

```
browser.get("https://twitter.com/search?src=typd&q=%23banjir
%20since%3A2015-11-01%20until%3A2015-12-01")
```

Gambar 6 Fungsi browser.get pada program Python

Pada tahap ini diperoleh sebanyak 33689 data *tweet* dari 12 *keywords* yang dapat dilihat pada Tabel 4. Jumlah *tweet* yang memiliki data *geolocation* sebanyak 2160 *tweet.* Pada 2160 data *tweet* tersebut dilakukan penghapusan data *duplicate* dan penghapusan data *tweet* dengan *geolocation* selain dari Indonesia menjadi 1681 *tweet.* Penghapusan data *tweet* yang *duplicate* dengan cara diamati satu persatu dan dilakukan penghapusan secara manual pada *Microsoft excel.*

## Praproses Data

***Tokenizing***

Pada tahap ini dilakukan perubahan semua huruf kapital pada data teks Twitter menjadi huruf kecil. Kemudian menghapus *mention,* URL, tanda baca dan angka menggunakan fungsi *gsub*. Selain itu, pada tahapan ini dilakukan beberapa perintah pemanggilan fungsi. *Source code tokenizing* dapat dilihat pada Lampiran 2. Berikut langkah-langkah yang dilakukan pada *tokenizing*.

1 Mengaktikan *package "tm"*.
2 Membaca *dataset tweet* dalam bentuk CSV dan mengubahnya menjadi suatu korpus.
3 Mengubah semua karakter huruf pada data teks Twitter menjadi huruf kecil.
4 Menghapus *mention,* URL, tanda baca, *hashtag,* dan angka.

Tabel 4 Jumlah *tweet* berdasarkan *keywords*

| Keywords | *Tweet* | *Tweet* dengan *geolocation* | *Tweet* dengan *geolocation* di Indonesia | *Tweet* yang tidak *duplicate* |
|---|---|---|---|---|
| #kebakaranhutan | 1014 | 53 | 53 | 32 |
| #melawanasap | 12840 | 758 | 750 | 650 |
| #daruratasap | 4095 | 149 | 145 | 84 |
| #melawanapi | 210 | 19 | 19 | 10 |
| #maribersikap | 115 | 3 | 3 | 3 |
| #kabutasap | 5154 | 508 | 496 | 446 |
| #banjir | 7612 | 533 | 458 | 383 |
| #tanahlongsor | 46 | 4 | 4 | 4 |
| #gempabumi | 1902 | 87 | 50 | 40 |
| #putingbeliung | 449 | 25 | 19 | 9 |
| #erupsi | 195 | 20 | 20 | 19 |
| #gunungmeletus | 57 | 1 | 1 | 1 |
| Total | 33689 | 2160 | 2018 | 1681 |

Data teks Twitter sebelum dan sesudah proses *tokenizing* dapat dilihat pada Tabel 5. Semua tanda baca dan angka pada dokumen sudah hilang. Semua karakter huruf pada setiap dokumen menjadi karakter huruf kecil.

**Normalisasi Kata**

Normalisasi kata dilakukan dengan mengganti kata tidak baku menjadi kata baku berdasarkan kamus kata Aziz (2013). Contoh data *tweet* sebelum dan sesudah dilakukan proses normalisasi kata dapat dilihat Tabel 6. Ada singkatan kata seperti "hrus", "dll", "tdk" dan "yg" dinormalisasi menjadi "harus", "dan lain-lain", "tidak" dan "yang".

**Penghapusan *Stopwords***

Tahap ini dilakukan penghapusan kata-kata yang sering muncul sesuai dengan penelitian Tala (2003). Kata yang dihapus adalah kata yang sering muncul dalam kehidupan sehari-hari seperti kata sambung, kata depan, kata seru

dan kata keterangan setelah dilakukan normalisasi. Kata “dan” , “lain”, “tidak” terhapus pada proses ini. Hasil penghapusan *stopwords* dapat dilihat pada Tabel 7.

***Stemming***

*Stemming* merupakan proses mengubah term ke bentuk term dasarnya. Proses *stemming* ini menggunakan kode program dari penelitian Wahyunintyas (2016) untuk melakukan *stemming* berbahasa Indonesia. Kamus kata dasar untuk proses *stemming* disimpan ke dalam *database* MySQL. Dalam *stemming* ini awalan seperti “me-” atau akhiran seperti “-kan” hilang. Hasil dari *stemming* ini mengubah kata menjadi bentuk dasarnya dapat dilihat pada Tabel 8.

**Pembuatan *Term Document Matrix* (TDM)**

Proses pembuatan *Term Document Matrix* (TDM) bertujuan untuk mendapatkan matriks jumlah kemunculan suatu term pada dokumen. Pada tahap penelitian ini menghasilkan 1681 dokumen dengan 4037 term. Banyaknya term yang dihasilkan membuat dimensi matriks menjadi besar, untuk memperkecil dimensi matriks dapat dilakukan dengan cara mereduksi term yang memiliki tingkat kemunculan rendah. Fungsi yang digunakan yaitu *removeSparseTerm* pada *console* R. Hasil *sparsity* menggambarkan kemunculan tiap term dalam setiap dokumen. Pada penelitian ini dilakukan percobaan dengan 3 parameter *sparse* sebesar 98.5%, 99%, dan 99.5%. Alasan pemilihan 3 parameter tersebut karena jika *sparse* di bawah 98.5% term yang dihasilkan menjadi terlalu sedikit dan jika *sparse* di atas 99.5% term yang dihasilkan terlalu banyak. Untuk mengetahui hasil reduksi term dapat dilihat pada Tabel 9.

Berdasarkan term yang dihasilkan maka dipilih nilai *sparse* 99% untuk mereduksi term menjadi 80 term. Berikut adalah contoh hasil dari pembuatan TDM dapat dilihat pada Gambar 7. Angka yang muncul pada TDM merupakan jumlah kemunculan term atau frekuensi term pada setiap dokumen.

Tabel 5 Data *tweet* sebelum dan sesudah proses *tokenizing*

| *Docs* | Sebelum *tokenizing* | Sesudah *tokenizing* |
|---|---|---|
| 1 | '#banjir : Hindari semua makanan yg tercemar air banjir makanan yg basi dll.'" | banjir hindari semua makanan yg tercemar air banjir makanan yg basi dll |
| 2 | #banjir : Pastikan sumber air yg digunakan tdk tercemar air banjir. Sblum dikonsumsi air hrus dimasak sampai mendidih.' | banjir pastikan sumber air yg digunakan tdk tercemar air banjir sblum dikonsumsi air hrus dimasak sampai mendidih |

***Clustering* dengan K-Means**

Pada tahap ini *Term Document Matrix* hasil reduksi dimensi dan penghapusan term dengan fungsi *removesparseTerm* dilakukan *clustering* dengan fungsi K-Means. Sebelum dilakukan *clustering dengan* K-Means, TDM diubah dulu menjadi bentuk matriks dan di-*transpose* pada R. Fungsi K-Means pada R digunakan untuk *clustering* pada TDM setelah proses pengecilan dimensi matriks dengan jumlah term sebanyak 80. Tujuan dari *clustering* ini adalah untuk mencari

kemiripan term antar dokumen sehingga dapat ditentukan suatu dokumen masuk *cluster* tertentu. Pada *clustering* dengan algoritme K-Means ini ditentukan terlebih dahulu dokumen sebagai *centroid* secara acak.

Tabel 6 Data *tweet* sebelum dan sesudah normalisasi kata

| *Docs* | Sebelum normalisasi kata | Sesudah normalisasi kata |
|---|---|---|
| 1 | banjir hindari semua makanan yg tercemar air banjir makanan yg basi dll | banjir hindari semua makanan yang tercemar air banjir makanan yang basi dan lain-lain |
| 2 | banjir pastikan sumber air yg digunakan tdk tercemar air banjir sblum dikonsumsi air hrus dimasak sampai mendidih | banjir pastikan sumber air yang digunakan tidak tercemar air banjir sebelum dikonsumsi air harus dimasak sampai mendidih |

Tabel 7 Data teks Twitter sebelum dan sesudah penghapusan *stopwords*

| *Docs* | Sebelum penghapusan *stopwords* | Sesudah penghapusan *stopwords* |
|---|---|---|
| 1 | banjir hindari semua makanan yang tercemar air banjir makanan yang basi dan lain-lain | banjir hindari makanan tercemar air banjir makanan basi - |
| 2 | banjir pastikan sumber air yang digunakan tidak tercemar air banjir sebelum dikonsumsi air harus dimasak sampai mendidih | banjir pastikan sumber air tercemar air banjir dikonsumsi air dimasak mendidih |

Tabel 8 Data teks Twitter sebelum dan sesudah *stemming*

| *Docs* | Sebelum *stemming* | Sesudah *stemming* |
|---|---|---|
| 1 | banjir hindari makanan tercemar air banjir makanan basi - | banjir hindar makan cemar air banjir makan basi |
| 2 | banjir pastikan sumber air tercemar air banjir dikonsumsi air dimasak mendidih | banjir pasti sumber air cemar air banjir konsumsi air masak didih |

Tabel 9 Hasil term dan *sparsity*

| *Sparse* | Jumlah term yang dihasilkan |
|---|---|
| 98.5% | 44 |
| 99.0% | 80 |
| 99.5% | 217 |

Setiap dokumen dicari kemiripan termnya dengan menggunakan fungsi *Euclidean distance* seperti pada Persamaan 2. Fungsi *Euclidean distance* yaitu fungsi pencarian jarak. Variabel $x_i$ merupakan nilai term pada dokumen *centroid* pusat sedangkan $y_i$ merupakan nilai term pada dokumen yang ingin dilakukan

*clustering*. Semakin kecil hasil *Euclidean distance* menunjukkan kemiripan suatu dokumen pada *cluster* tertentu.

| | and | asap | bakar | bal | banjir | bantu | bencana | bikin | biru |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| 14 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 15 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 16 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 17 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 18 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| 19 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 20 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |

Gambar 7 Hasil pembuatan TDM

Pemilihan nilai *k* terbaik dengan menentukan nilai SSE terkecil. Pencarian nilai SSE dilakukan dengan menjalankan kode program yang terdapat pada Lampiran 2. Nilai *k* yang menghasilkan nilai SSE terkecil yaitu pada nilai *k* sebesar 7. Maka *clustering* yang akan dilakukan pada nilai *k* sebesar 7. Hasil nilai SSE dapat dilihat pada Tabel 10.

$$distance\ (x, y) = \sqrt{\sum_i (xi - yi)^2} \qquad (2)$$

Tabel 10 Nilai SSE untuk hasil *clustering* nilai *k* = 2 hingga 7

| Nilai *k* | SSE |
|---|---|
| 2 | 7623.983 |
| 3 | 6818.443 |
| 4 | 6076.148 |
| 5 | 5822.506 |
| 6 | 5782.923 |
| 7 | 5693.169 |

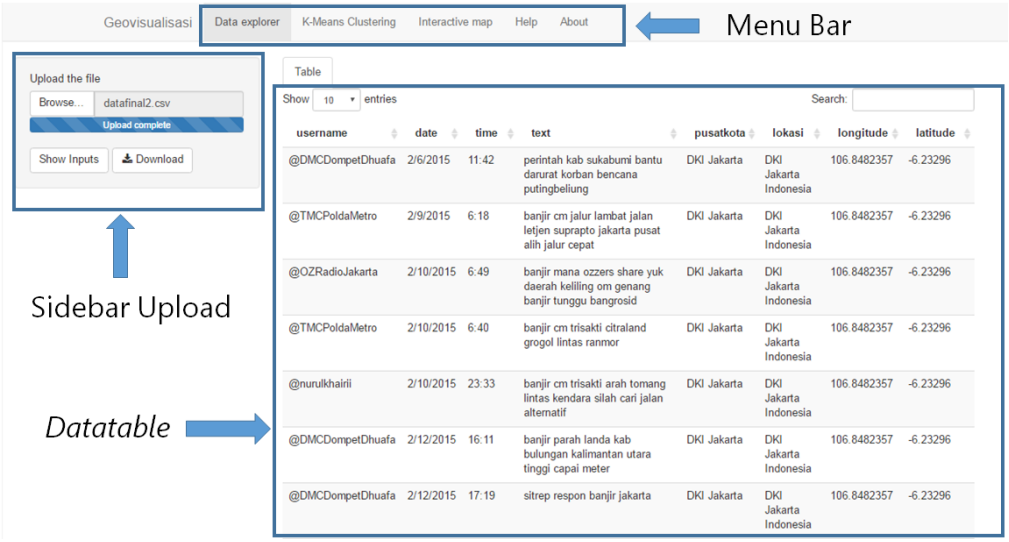
**Geovisualisasi dengan *Framework* Shiny**

Data *tweet* yang telah dipraproses kemudian digabungkan dengan data hasil *cluster* yang diperoleh dari proses *clustering* dengan algoritme K-Means. Data *tweet* divisualisasikan berdasarkan *longitude*, *latitude* dan hasil *cluster* tiap dokumen. Proses geovisualisasi data hasil *clustering* dengan K-Means ini meliputi pembuatan *data explorer*, K-Means *clustering* dan *interactive map* ke dalam suatu sistem. *Interface* pada geovisualisasi dibuat pada *file* ui.r pada *framework* Shiny. Kode program di ui.r dapat dilihat pada Lampiran 3. Fungsi

pada geovisualisasi dibuat pada *file* server.r pada *framework* Shiny. Kode program di server.r dapat dilihat pada Lampiran 2.

**Pembuatan *Data Explorer***

Eksplorasi data dibuat untuk praproses data. Data di*upload* kemudian dilakukan praproses data. Data hasil praproses divisualisasikan dalam bentuk tabel. Fungsi yang digunakan untuk membuat *datatable* yaitu *renderDatatable*. Pada fungsi *datatable* sudah dilengkapi *fitur* seperti pencarian, filter berdasarkan kolom, dan data yang ditampilkan. Hasil *datatable Data Explorer* dapat dilihat pada Gambar 8.




Gambar 8 *Datatable* dari menu Data explorer

**Pembuatan K-Means *Clustering***

K-Means *clustering* dibuat untuk *clustering* data hasil praproses. Proses *clustering* dengan K-Means dilakukan dengan memberikan pilihan parameter *sparsity* dan nilai *k*. Pada geovisualisasi ini *default* parameter disesuaikan dengan proses *clustering* dengan K-Means pada *console* R. Hasil *clustering* digabungkan dengan hasil praproses dan ditampilkan dalam bentuk *datatable* dapat dilihat pada Gambar 9.

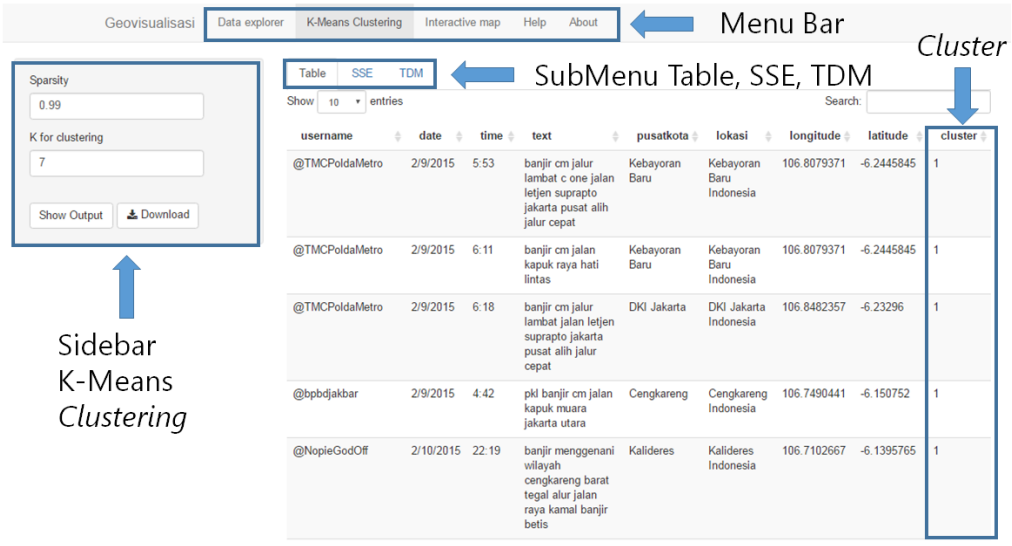
**Pembuatan *Interactive Map***

Pada proses ini dibutuhkan *package* Leaflet untuk membuat visualisasi peta yang interaktif. Untuk mendukung proses visualisasi ini dibutuhkan beberapa *package* yang lain yaitu: Scales, Lattice dan Dplyr. Berikut adalah fungsi dari masing-masing *package* seperti pada Tabel 11.

Eksplorasi data *tweet* hasil *clustering* divisualisasikan dalam *marker* lingkaran yang diberi warna pada masing-masing *cluster*. *Interactive map* sudah dilengkapi fitur *zoom in* dan *zoom out* sehingga dapat mencari area tertentu dengan *cluster* tertentu. Pada visualisasi dibuat *cluster explorer* yang memiliki fitur seperti *checkbox* legenda dan histogram frekuensi setiap *cluster*. Hasil *interactive map* dapat dilihat pada Gambar 10. Pada proses pembuatan *interactive map* ini ada beberapa fungsi penting yang digunakan yaitu:

1 Peta
Fungsi yang dibuat untuk memvisualisasikan peta yaitu *renderLeaflet*. Di dalam fungsi tersebut terdapat beberapa *tools* untuk membuat kelengkapan pada peta interaktif seperti penambahan *marker* , *popup*, dan legenda.

2 *Popup*
*Popup* jika diklik akan memberikan informasi langsung dari setiap *marker*. Informasi tersebut berupa *username*, tanggal, waktu, data teks, dan lokasi dari setiap dokumen.



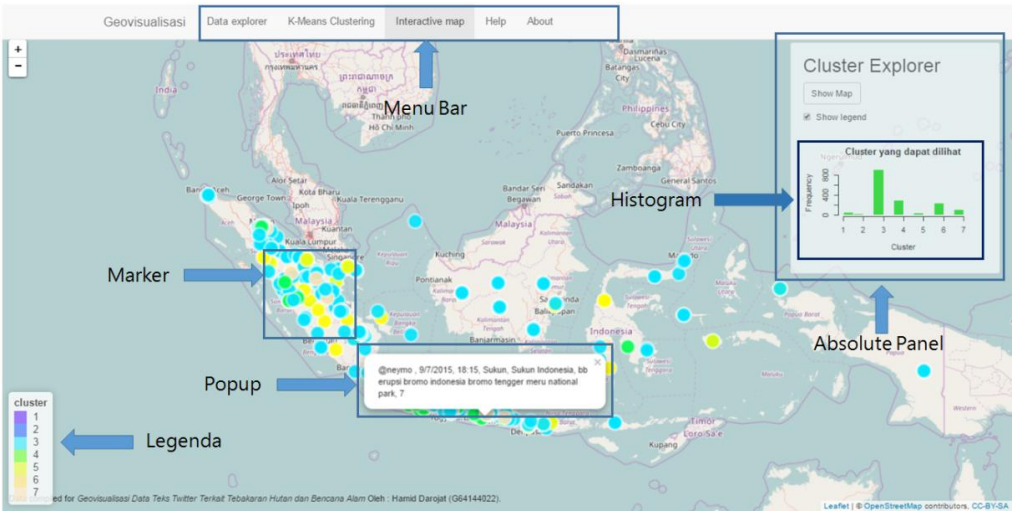

Gambar 9 *Datatable* untuk menu K-Means *clustering*

Tabel 11 *Package* dan fungsinya

| *Package* | Fungsi |
|---|---|
| Leaflet | Memvisualisasikan peta interaktif dengan JavaScript leaflet |
| Scales | Memberi fungsi skala pada visualisasi |
| Lattice | *Library* untuk pembutan grafik |
| Dplyr | Memvisualisasi data, manipulasi data, dan menghubungkan ke *database* |

3 Histogram
Histogram pada peta interaktif ini berfungsi untuk menunjukkan frekuensi setiap *cluster*. *Zoom in*, *zoom out*, dan menggeser posisi peta dapat merubah frekuensi *cluster*. Perubahan frekuensi tergantung area dan perbesaran peta yang ditampilkan (hanya menghitung pada area tersebut).

4 Legenda
Legenda ditampilkan dengan pilihan *checkbox* pada *absolute panel*. Legenda menampilkan perbedaan warna tiap *cluster*. Pemanggilan warna menggunakan fungsi *colorfactor*. *Colorfactor* merupakan fungsi warna yang terkategori. Pemberian warna berdasarkan nomor *cluster*.




Gambar 10 *Interactive Map*

## Evaluasi Hasil *Clustering*

Hasil *clustering* dengan K-Means pada *k* sebesar 2 sampai 7 pada *Term Document Matrix* memperoleh nilai SSE terkecil pada *cluster* 7 yaitu 5593.169. Sehingga K-Means dilakukan dengan nilai *k* sebesar 7. Hasil *clustering* dengan nilai *k* sebesar 7 dapat dilihat pada Tabel 12. Berdasarkan hasil eksplorasi data pada *interactive map* dan *datatable* terdapat kategori yang sering muncul pada tiap *cluster*. Hasil eksplorasi data terdapat pada Tabel 13.

Semua kategori muncul pada *cluster* 3. Kategori banjir sering muncul pada *cluster* 1, 2, dan 4. Kemunculan kategori banjir pada *cluster* 3 dan 6 karena data teks Twitter mengandung kategori "asap". Pada *cluster* 5, 6, dan 7 dominan kategori tentang "asap". Kemunculan kategori pada tiap *cluster* dapat dilihat pada Tabel 13.

Geovisualisasi dari hasil *clustering* tentang kebakaran hutan dan bencana alam menyebar hampir di seluruh Indonesia. *Cluster* 3 memiliki jumlah dokumen terbanyak. Pulau Sumatera, Jawa, Kalimantan, Sulawesi, Papua, Bali, dan sekitarnya didominasi oleh *cluster* 3. Contoh persebaran *cluster* dapat dilihat pada Lampiran 4. Tidak semua *tweet* tentang kebakaran hutan dan bencana alam berada langsung di lokasi kejadian. Sebagai contoh kebakaran hutan di Riau namun *geolocation tweet* berada di Jakarta.

Tabel 12 Hasil *clustering*

| *Cluster* | Jumlah dokumen | Presentase |
|---|---|---|
| 1 | 57 | 3.39% |
| 2 | 26 | 1.55% |
| 3 | 914 | 54.37% |
| 4 | 294 | 17.49% |
| 5 | 36 | 4.32% |
| 6 | 248 | 14.75% |
| 7 | 106 | 6.31% |

Tabel 13 Hasil eksplorasi data

| *Hashtag* | Kategori | *Cluster* |
|---|---|---|
| #kebakaranhutan, #maribersikap, #melawanapi | Kebakaran hutan | 3, 6, 7 |
| #kabutasap, #melawanaasap, #daruratasap | Asap | 1, 3, 5, 6, 7 |
| #putingbeliung | Puting beliung | 3 |
| #banjir | Banjir | 1,2,3,4,6 |
| #tanahlongsor | Tanah longsor | 3 |
| #gempabumi | Gempa bumi | 3,7 |
| #erupsi, #gunungmeletus | Erupsi | 3,7 |

Tabel 14 *Confusion matrix* kemunculan kategori pada tiap *cluster*

| *Cluster* | Kategori | | | | | | |
|---|---|---|---|---|---|---|---|
| | Kebakaran Hutan | Asap | Puting Beliung | Banjir | Tanah Longsor | Gempa Bumi | Erupsi |
| 1 | 0 | 1 | 0 | 56 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 26 | 0 | 0 | 0 |
| 3 | 25 | 827 | 9 | 1 | 4 | 39 | 17 |
| 4 | 0 | 0 | 0 | 294 | 0 | 0 | 0 |
| 5 | 1 | 37 | 0 | 0 | 0 | 0 | 0 |
| 6 | 14 | 243 | 0 | 1 | 0 | 0 | 0 |
| 7 | 3 | 118 | 0 | 0 | 0 | 1 | 3 |

Tabel 15 Pelabelan tiap *cluster*

| *Cluster* | Label | Warna | Frekuensi term terbanyak |
|---|---|---|---|
| 1 | Banjir | Ungu | banjir |
| 2 | Banjir | Biru | banjir |
| 3 | Melawan asap | Biru langit | melawanasap |
| 4 | Banjir | Hijau | banjir |
| 5 | Melawan asap | Kuning | melawanasap |
| 6 | Asap | Oranye | asap |
| 7 | Darurat Asap | Coklat muda | daruratasap |

Hasil dari *confusion matrix* pada Tabel 14 menunjukkan bahwa kategori tentang bencana alam seperti "puting beliung", "tanah longsor", "gempa bumi", dan "erupsi" masuk pada *cluster* 3. Hal tersebut terjadi karena keempat kategori tersebut tidak membentuk *cluster* sendiri. Jumlah data *tweet* masing-masing kategori tersebut sedikit dan kemunculan term pada *cluster* juga sedikit sesuai pada Tabel 4.

Berdasarkan kemunculan term pada setiap *cluster*, term tentang "asap" dan "banjir" menyebar hampir di setiap *cluster* karena *tweet* dengan *hashtag* #melawanasap, #kabutasap, dan #banjir memiliki jumlah data *tweet* yang dominan dibandingkan *tweet* dengan *hashtag* yang lainnya sesuai dengan Tabel 4. Term "banjir" meruapakan term terbanyak pada *Cluster* 1, 2, dan 4. Term "melawanasap" merupakan term terbanyak pada *Cluster* 3 dan 5. "Asap"

merupakan term terbanyak pada *Cluster* 6, sedangkan *cluster 7* memiliki term terbanyak yaitu "daruratasap". Term tentang bencana alam seperti "erupsi", "gempabumi", "gunungmeletus", "tanahlongsor" dan "putingbeliung" masuk pada *cluster* 3 karena jumlah *tweet* tentang bencana alam sedikit seperti pada Tabel 4. Pemberian label pada masing-masing *cluster* seperti pada Tabel 15.

# SIMPULAN DAN SARAN

## Simpulan

Hasil implementasi *clustering* dengan algoritme K-Means pada data teks Twitter terkait kebakaran hutan dan bencana alam menggunakan *sparsity* 99% pada *Term Document Matrix* dan menggunakan nilai *k* sebesar 2 sampai dengan 7 menghasilkan SSE terkecil pada nilai *k* sebesar 7. Nilai *Sum Squared Error* (SSE) yang dihasilkan adalah 5693.169.

Sistem Geovisualisasi berhasil diimplementasikan pada data teks Twitter hasil *clustering* dengan algoritme K-Means. Geovisualisasi direpresentasikan dengan peta Indonesia menggunakan *library* Leaflet. Proses geovisualisasi dimulai dari praproses data, *clustering* dengan algoritme K-Means, dan geoisualisasi hasil *clustering* pada peta. Geovisualisasi hasil *clustering* data teks Twitter menyebar di seluruh Indonesia berdasarkan *longitude* dan *latitude* tiap data *tweet* tersebut. *Geolocation* merepresentasikan posisi *username* melakukan *tweet* dan bukan merepresentasikan posisi kejadian kebakaran hutan atau bencana alam. *Cluster* 3 memiliki jumlah dokumen terbanyak. Pulau Sumatera, Jawa, Kalimantan, Sulawesi, Papua, Bali dan sekitarnya didominasi oleh *cluster* 3. Tidak semua kategori mendapat *cluster* sendiri. Kategori tentang bencana alam masuk pada *cluster* 3 (melawanasap).

## Saran

Terdapat beberapa hal yang dapat ditambahkan atau diperbaiki untuk penelitian selanjutnya yaitu:

1 Menambahkan jumlah dokumen data *tweet* yang digunakan sebagai korpus dapat diperbanyak lagi. Menambahkan jumlah *hashtag* untuk tiap kategori bencana diperbanyak lagi.
2 Menambahkan fungsi *searching map* area pada *interactive map*.

# DAFTAR PUSTAKA

Aziz ATA. 2013. Sistem Pengklasifikasian Entitas pada Pesan Twitter Menggunakan Ekspresi Reguler dan Naïve Bayes [skripsi]. Bogor (ID): Institut Pertanian Bogor.

[BNPB] Badan Nasional Penanggulangan Bencana. 2015. Informasi Kebencanaan Bulalan Teraktual. Laporan. [Internet]. [Diunduh tanggal

10/08/2016].Tersedia: http://www.bnpb.go.id/uploads/publication/1125/2015-12-8_info_bencana_november.pdf.

Crooks A, Croitoru A, Stefanidis A, Radzikowski J. 2012. *#Earthquake: Twitter as a Distributed Sensor System*.Virginia(US): Blackwell Publishing Ltd.

Longueville B, Smith RS, Luraschi G. 2009. Omg, from here, i can see the flames!: a use case of mining location based social networks to acquire spatio-temporal data on forest fires. Di dalam: Zhou X, Xie X, editor. *In Proceedings of the 2009 International Workshop on Location Based Social Networks*; 2009 Nov 04-06; Seattle, Washington , USA. New York (US):ACM. hlm 73-80.

Denatari, RB. 2015. Clustering Data Teks Twitter untuk Kasus Pertanian di Indonesia.[skripsi]. Bogor (ID): Institut Pertanian Bogor.

Han J, Kamber M, Pei J. 2012. Data Mining: Concepts and Techniques, 3rd ed. New York (US): Morgan Kaufmann.

Imran M, Elbassuoni SM, Castillo C, Diaz F, Meier P. 2013. Extracting information nuggets from disaster-related messages in social media. Di dalam: Comes T, Fiedrich F, Fortier S, Geldermann J, Muller T, editor. *10th International Conference on Information Systems for Crisis Response and Management*; 2013 Mei 12-15; Baden-Baden, Germany. Baden-Baden (DE): Karlsruhe Institute of Technology.

[KLKH] Direktorat PKHL Kementrian Lingkukan Hidup dan Kehutanan RI. 2015. KLKH: Rekapitulasi Luas Kebakaran Hutan dan Lahan (Ha) Per Provinsi Di Indonesia Tahun 2011-2016 [Internet]. [diunduh 2016 Agustus 2]. Tersedia pada: http://sipongi.menlhk.go.id/hotspot/luas_kebakaran.

Meyer D, Hornik K, Feinerer I. 2008. Text mining infrastructure in R. Di dalam: Grun B, Hothom T, Pebesma E, Zeileis A, editor. *Journal of Statistical Software*.Volume 25. Vienna (AT): American Statistical Association. hlm 1-54.

Nadilah F. 2016. Asosiasi dan Geovisualisasi Antara Data Tweet Terkait Kebakaran Hutan dengan Data Cuaca di Provinsi Riau dan Kepulauan Riau [skripsi]. Bogor (ID): Institut Pertanian Bogor.

[Rstudio]. 2014. Package ‘shiny’ [internet]. [diunduh 2016 Jul 27]. Tersedia pada: http://cran.r-project.org/web/packages/shiny/shiny.pdf.

Sakaki T, Okazaki M, Matsuo Y. 2010. Earthquake shakes Twitter users: real-time event detection by social sensors. Di dalam: Rappa M, Jones P, Freire J, Chakrabarti S, editor. *In Proceedings of the 19th international conference on World Wide Web*; 2010 Apr 26-30; North Carolina, USA. North Carolina (US):ACM. hlm 851-860.

Simangungsong R. 2015. YearOnTwitter 2015: Twitter sebagai Platform Pemersatu Masyarakat Indonesia. [Internet]. Tersedia pada: https://blog.twitter.com/id/2015/yearontwitter-2015-twittersebagai-platform-pemersatu-masyarakatindonesia.

Susanto H, Sumpeno S, Racmadi RF. 2014.Visualisasi Data Teks Twitter Berbasis Bahasa Indonesia Menggunakan Teknik Pengklasteran [Thesis] Surabaya(ID): Institut Teknologi Sepuluh November.

Tala FZ. 2003. A study of stemming effects on information retrieval in Bahasa Indonesia [tesis]. Amsterdam (NL): Universiteit van Amsterdam.

Wahyuningtyas A. 2016. Deteksi Spam pada Twitter Menggunakan Algoritme Naïve Bayes [Skripsi]. Bogor (ID): Institut Pertanian Bogor.

[WRI] World Resource Institute. 2015.WRI: Land and Forest Fires in Indonesia Reach Crisis Levels [Internet]. Tersedia pada: http://www.wri.org/blog/2015/09/land-and-forest-fires-indonesia-reach-crisis-levels

Yasobant S, Vora KS, Hughes C, Upadhyay A, Mavalankar DV. 2015. Geovisualization: A Newer GIS Technology for Implementation Research in Health. *Journal of Geographic Information System*. Wuhan (CN): Scientific Research Publishing. hlm 20-28.

Zwitch R. 2013. Clustering Search Keywords Using K-Means Clustering.[Internet]. Tersedia pada: https://www.r-bloggers.com/clustering-search-keywords-using-k-means-clustering/

Lampiran 1 Kode program Python untuk akuisisi data Twitter

```
import time,datetime
import codecs
import json
from Tw
itterAPI import TwitterAPI
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from TwitterAPI import TwitterAPI
#TwitterAPI Settings
api = TwitterAPI
('9[redacted]',
'8[redacted]O5INlfD',
'8[redacted]trHUZpC',
'E[redacted]Q3')

#Open session to write
outputFile = codecs.open("Output Dalam CSV.csv", "w+", "utf-
8")
outputFile.write
('username;date;time;text;tweetID;
placeID;pusatkota;lokasi;longitude;latitude\n')

#Open webdriver Chrome
#Use location to your chromedriver.exe
browser = webdriver.Chrome
("D:/kuliah/akuisisi/banjir/chromedriver.exe")

#search url
browser.get
("https://twitter.com/search?src=typd&q=
%23banjir%20since%3A2015-11-01%20until%3A2015-12-01")
time.sleep(1)

elem = browser.find_element_by_tag_name("body")

#Scroll value
scroll = 1000

while scroll:
       elem.send_keys(Keys.PAGE_DOWN)
       time.sleep(0.2)
       scroll-=1

#tweet_contents
tweet_contents =
browser.find_elements_by_class_name("js-stream-tweet")

for tweet in tweet_contents:
       #Tweet text
       text = tweet.find_element_by_css_selector
       ("div.js-tweet-text-container > p")
       .text.replace('\n',' ')
       .encode('ascii', 'ignore')

       #Username
       username = tweet.find_element_by_css_selector
       ("div.stream-item-header > a
       > span.username.js-action-profile-name > b")
       .text.encode('ascii', 'ignore')
```

Lampiran 1 lanjutan

```
      #Date of tweet
       dateInt = int(tweet.find_element_by_css_selector
       ("div.stream-item-header > small > a > span")
       .get_attribute('data-time'))

       dateFormat =    datetime.datetime.
                      fromtimestamp(dateInt)
#Place ID
       if len
       (tweet.find_elements_by_class_name
      ("Tweet-geo")) > 0:
              placeID = tweet.find_element_by_css_selector
              ("div.stream-item-header > span > a")
              .get_attribute('data-place-id')
              r = api.request('geo/id/:%s' % placeID)
              json_var = r.json()

              #name
              pusatkota = json_var['name']
              #fullname
              lokasi = json_var['full_name']
              #longitude
              longitude = json_var['centroid'][0]
              #latitude
              latitude = json_var['centroid'][1]

       else:
              placeID = ''
              lokasi = ''
              pusatkota = ''
              longitude = ''
              latitude = ''

       #Tweet ID
       tweetID = tweet.get_attribute("data-tweet-id")

       outputFile.write
       ('%s;%s;%s;%s;%s;%s;%s;%s;%s;%s\n' %
       (username, dateFormat.strftime("%Y-%m-%d")
       ,dateFormat.strftime("%H:%M")
       , text, tweetID, placeID
       ,pusatkota,lokasi, longitude,latitude))
```

Lampiran 2 Kode program fungsi pada server.r

**Fungsi *tokenizing***

```
hapusURL <- function(x) gsub("(http)[[:graph:]]*", "", x)
korpus <- tm_map(korpus, content_transformer(hapusURL))
enter <- function(x) gsub("\n", " ", x)
korpus <- tm_map(korpus, content_transformer(enter))
karakter1 <- function(x) gsub("\\S*(\\S)\\1\\1\\S*\\s?", "", x)
korpus <- tm_map(korpus, content_transformer(karakter1))
karakter2 <- function(x) gsub("(@)[[:graph:]]+", "", x)
korpus <- tm_map(korpus, content_transformer(karakter2))
karakter4 <- function(x) gsub("&lt;-"," ", x)
korpus <- tm_map(korpus, content_transformer(karakter4))
karakter5 <- function(x) gsub("[[:punct:]]", " ", x)
korpus <- tm_map(korpus, content_transformer(karakter5))
karakter6 <- function(x) gsub("[[:digit:]]", " ", x)
korpus <- tm_map(korpus, content_transformer(karakter6))
```

**Fungsi pemanggilan proses normalisasi kata**

```
#normalisasi dengan mengubah singkatan
 nbaris <- length(splitgue)
 for (i in 1:nbaris) {
   nkolom <- length(splitgue[i][[1]])
   if(nkolom != 0){
     nkolom <- length(splitgue[i][[1]])
     for (j in 1:nkolom) {
       term <- splitgue[i][[1]][[j]]
       norm <- cari.singkatan(term)
       splitgue[i][[1]][[j]] <- norm
     }}}
 Length <- sapply(splitgue, length)
 max.length <-max(sapply(splitgue,length))
```

**Fungsi menghapus *stopwords***

```
#menghapus stopwords
dataframe1 <-data.frame(text=unlist(sapply(korpus, "content")))
newdat <- data.frame(text=str_trim(do.call(paste, hasilnormal)),
stringsAsFactors=FALSE)
baru<-data.frame(newdat)
names(baru)[names(baru)=="hasilnormalbaru"] <- "text"
korpus1 <- Corpus(VectorSource(baru$text))
korpus1 <- tm_map(korpus1, stripWhitespace)
stopword<- read.csv("stopword.csv")
stopword <- tolower(stopword[, 1])
stopword <- c(stopword, "", "a", "-", "rt", "...", "&", "|",
"ada")
korpus1=tm_map(korpus1,removeWords,stopword)
```

Lampiran 2 lanjutan

**Fungsi pemanggilan proses *stemming***

```
#Stemming
nbaris1 <- length(splitgue1)
for (m in 1:nbaris1) {
  nkolom1 <- length(splitgue1[m][[1]])
    if(nkolom1 != 0){
      for (n in 1:nkolom1) {
        #print(n)
        term_stem <- splitgue1[m][[1]][[n]
        #print(term_stem)
        norm_stem <- stemming(term_stem)
        #print(norm_stem)
        splitgue1[m][[1]][[n]] <- norm_ste
      }}}
Length <- sapply(splitgue1, length)
max.length <-max(sapply(splitgue1,length))
```

**Fungsi *Clustering* dengan K-Means**

```
#TDM
baru1
v <- Corpus(VectorSource(baru1$text))
tdm <- TermDocumentMatrix(v)
#tdm to matrix
m <- as.matrix(tdm)
#mereduksi terms dengan remove sparse term
s <- input$numberofsparse
tdms <- removeSparseTerms(tdm, s)
inspect(tdms)
m1 <- as.matrix(tdms)
#binding tdm before and after
tdm_before <- dim(m)
df_m<-as.data.frame(tdm_before)
tdm_after <- dim(m1)
df_m1<-as.data.frame(tdm_after)
df_tdm <- cbind(df_m, df_m1)
tdm_total <<- df_tdm
#Tranpose Matrix m1 ke m2
m2 <- t(m1)
# set a fixed random seed
set.seed(122)

#K-Means dengan memasukkan input kelas
k <- input$numberofk
kmeansResult <- kmeans(m2, k)
#count SSE
s <- 2
finish <- input$numberofk
for(start in s : finish){
set.seed(1000)
hasil <- kmeans(m2, start)
centers <- hasil$centers[hasil$cluster,,drop=FALSE]
jarak <- sqrt((m2 - centers)^2)
total <- sum(jarak)
```

Lampiran 2 lanjutan

```
persen <- 100*(hasil$betweenss/hasil$totss)
gab<- data.frame(start, total, hasil$tot.withinss, persen)
if(start == 2) {
  sse <- gab
  sse <- as.matrix(sse)}
else{sse <- rbind(sse, gab)
}}
names(sse) <- c("kelas", "sse", "tot.within", "persen")

#write result of SSE
analisis <<- sse
```

**Fungsi *interactive map* dengan Leaflet**

```
output$map <- renderLeaflet({
factpal <- colorFactor(topo.colors(7), cleantable$cluster)
leaflet(cleantable) %>% addTiles() %>%

#Add Marker
addCircleMarkers
(~long, ~lat, fillColor = factpal(cleantable$cluster),
fillOpacity = 0.9,
stroke = TRUE, color="#fff", weight=3,
popup = ~htmlEscape(paste
( username, tanggal, waktu, kota, lokasi,
  teks, cluster, sep=", ")))%>%

#View Map Indonesia
setView(lng = 113.9213, lat = 0.7893, zoom = 5)
 })
```

**Fungsi legenda**

```
#Show Legend
observe({
proxy <- leafletProxy("map", data = cleantable)
proxy %>% clearControls()
if (input$legend) {
  #pal <- colorpal()
  factpal <- colorFactor(topo.colors(100),
             cleantable$cluster)
  proxy %>% addLegend(position = "bottomleft",
  pal = factpal, values = ~cluster
  )}
})
```

**Fungsi histogram**

```
# Make Histogram to show frequency of cluster
clusterBreaks <- hist(plot = FALSE,
cleantable$cluster, breaks = 10)$breaks
output$histCluster <- renderPlot({
 if (nrow(zipsInBounds()) == 0)
 return(NULL)
  hist(zipsInBounds()$cluster,
   breaks = clusterBreaks,
   main = "Cluster yang dapat dilihat",
   xlab = "Cluster",
   xlim = range(cleantable$cluster),
   col = '#00DD00',
   border = 'white')})
```

Lampiran 3 Kode program untuk *interface* sistem pada ui.r

**Visualisasi *Data Exlporer***

```
tabPanel("Data explorer",
   sidebarLayout(fluid = TRUE,
    sidebarPanel(
     fileInput('datafile', 'Upload the file'),
       actionButton(inputId = "input_action", label =
       "Show Inputs"),
       downloadButton('downloadData', 'Download'),
    width=3),
    mainPanel(fluid = TRUE,
     tabsetPanel(
       tabPanel("Table", dataTableOutput("td"))
    ))),
```

**Visualisasi K-Means *Clustering***

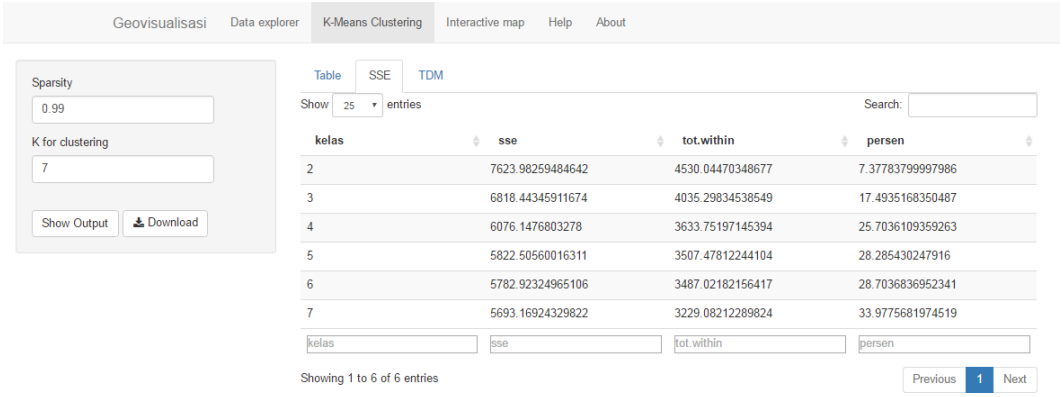
```
tabPanel("K-Means Clustering",
    sidebarLayout(fluid = TRUE,
     sidebarPanel(
       numericInput("numberofsparse",
        "Sparsity", 0.99, min = 0.985, max = 0.995),
       numericInput("numberofk",
        "K for clustering", 7, min = 2, max = 7),br(),
       actionButton(inputId = "input_clustering",
        label = "Show Output"),
       ownloadButton('downloadData1', 'Download'),
     width=3),
     mainPanel(fluid = TRUE,
       tabsetPanel(
        tabPanel("Table", dataTableOutput("td1")),
        tabPanel("SSE", dataTableOutput("sse")),
        tabPanel("TDM", tableOutput("tdm"))
     ))),
```

**Visualisasi *Interactive Map***

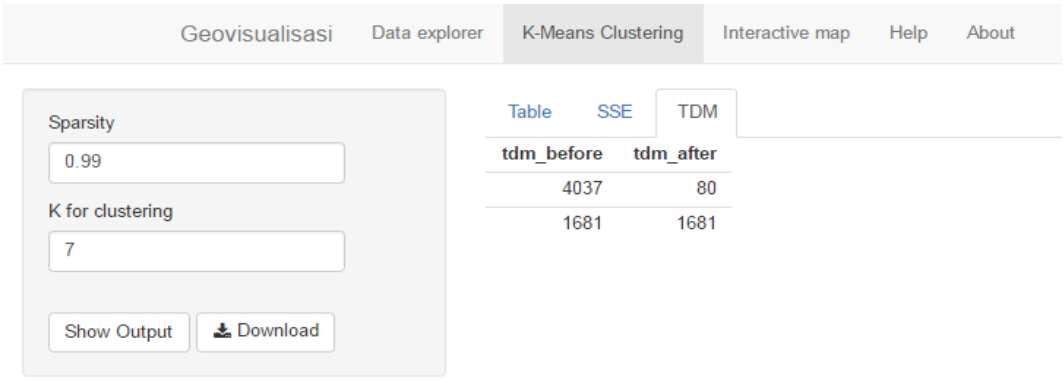
```
  tabPanel("Interactive map",
    div(class="outer",
     tags$head(
     # Include CSS
     includeCSS("styles.css"),
     includeScript("gomap.js")),
      leafletOutput("map", width="100%", height="100%"),
       absolutePanel(id = "controls",
       class = "panel panel-default", fixed = TRUE,
       draggable = TRUE, top = 60,
       left = "auto", right = 20, bottom = "auto",
       width = 330, height = "auto",
       h2("Cluster Explorer"),
       actionButton(inputId = "show_map", label = "Show Map"),
       #show legend
       checkboxInput("legend", "Show legend"),
       plotOutput("histCluster", height = 200)),
       tags$div(id="cite",'Data compiled for ',   tags$em(
         'Hamid Darojat (G64144022).'))),
```

Lampiran 4 Visualisasi sistem

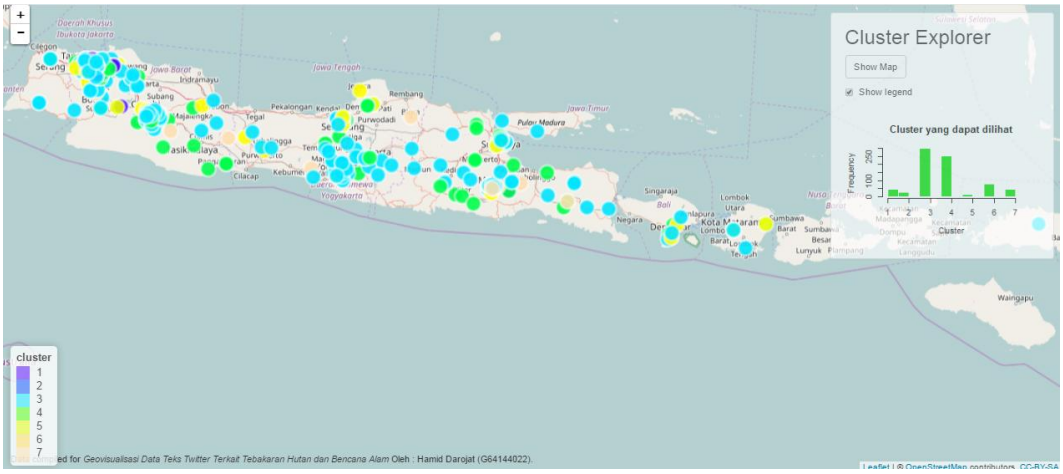
## Visualisasi analisis SSE




## Visualisasi TDM sebelum dan sesudah *sparsity*

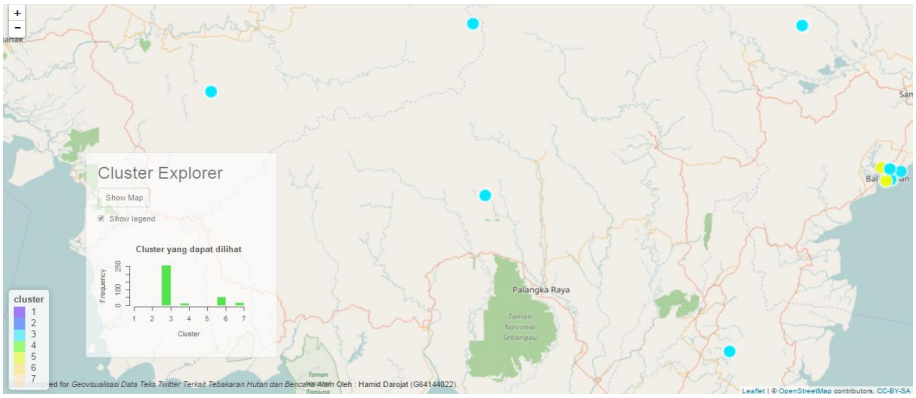



## Geovisualisasi sebaran *cluster* di Jawa, Bali dan sekitarnya

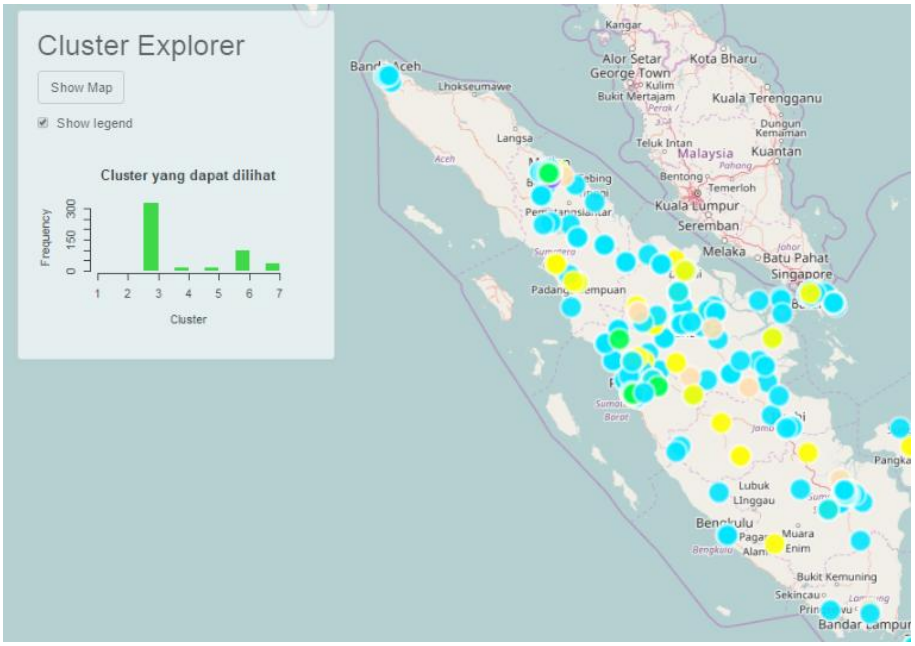


Lampiran 4 lanjutan

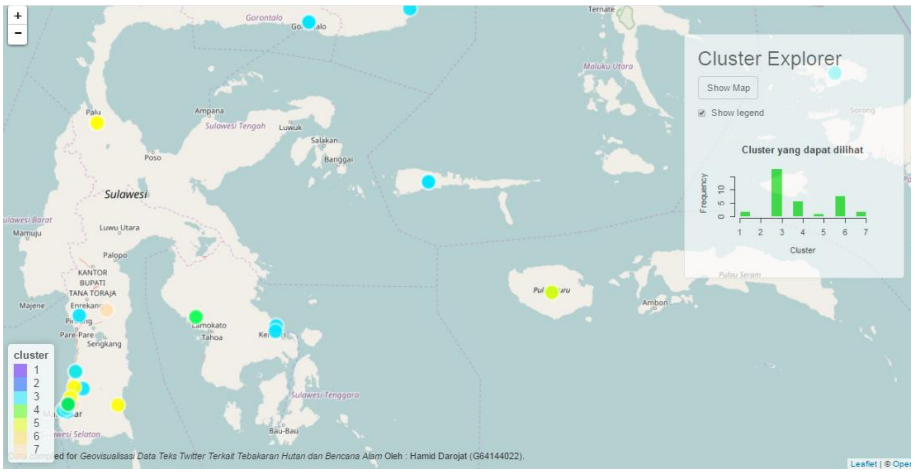
**Geovisualisasi sebaran *cluster* di Kalimantan**




**Geovisualisasi sebaran *cluster* di Sumatera dan sekitarnya**




**Geovisualisasi sebaran *cluster* di Sulawesi dan sekitarnya**



# RIWAYAT HIDUP

Penulis dilahirkan di Purworejo, Jawa Tengah pada tanggal 27 Juli 1992. Penulis adalah anak keempat dari empat bersaudara, anak dari pasangan Alm. Wahono dan Paryati. Penulis menempuh pendidikan Sekolah Menengah Atas di SMA 1 Purworejo pada tahun 2007 hingga 2010. Kemudian penulis melanjutkan pendidikan Program Diploma 3 (D3) di Universitas Gadjah Mada Sekolah Vokasi Jurusan Komputer dan Sistem Informasi pada tahun 2010 hingga 2013. Kemudian penulis melanjutkan pendidikan Program Sarjana Alih Jenis (S1) di Institut Pertanian Bogor Fakultas Matematika dan Ilmu Pengetahuan Alam Departemen Ilmu Komputer.