

**KINERJA ALGORITMA *SUPPORT VECTOR MACHINE*
DALAM MENENTUKAN KEBENARAN INFORMASI BANJIR
DI TWITTER**



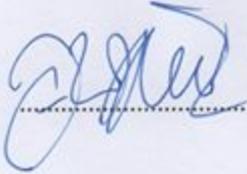
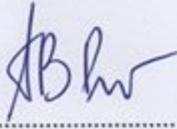
**MUHAMAD PRASETIO DWI CAHYO
5235134395**

Skripsi ini Disusun Sebagai Salah Satu Persyaratan Untuk Memperoleh Gelar
Sarjana Pendidikan

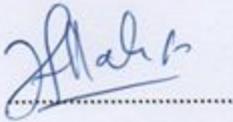
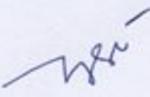
**PENDIDIKAN TEKNIK INFORMATIKA DAN KOMPUTER
FAKULTAS TEKNIK
UNIVERSITAS NEGERI JAKARTA**

2017

HALAMAN PENGESAHAN

NAMA DOSEN	TANDA TANGAN	TANGGAL
Widodo, M. Kom (Dosen Pembimbing I)		23-2-'17
Bambang P. Adhi, S.Pd, M.Kom (Dosen Pembimbing II)		23-2-17

PENGESAHAN PANITIA UJIAN SKRIPSI

NAMA DOSEN	TANDA TANGAN	TANGGAL
Lipur Sugiyanta., Ph.D (Ketua Penguji)		22-2-17
Hamidillah Ajie., MT (Sekertaris Penguji)		16-2-2017
Drs. Bachren Zaini., M.Pd (Dosen Ahli)		16-2-2017

HALAMAN PERNYATAAN

Dengan ini saya menyatakan bahwa:

1. Karya tulis skripsi saya yang berjudul “Kinerja Algoritma *Support Vector Machine* dalam Menentukan Informasi Banjir di Twitter” adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik sarjana, baik di Universitas Negeri Jakarta maupun di perguruan tinggi lain.
2. Karya tulis yang berjudul “Kinerja Algoritma *Support Vector Machine* dalam Menentukan Informasi Banjir di Twitter” adalah murni gagasan, rumusan, dan penelitian saya sendiri dengan arahan dosen pembimbing.
3. Dalam karya tulis, tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini, serta sanksi lainnya sesuai dengan norma yang berlaku di Universitas Negeri Jakarta.

Jakarta, Februari 2017

Yang membuat pernyataan



Muhamad Prasetio Dwi Cahyo

5235134395

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Allah SWT atas segala Rahmat Hidayah, dan Karunia-Nya yang diberikan kepada penulis sehingga dapat menyusun dan menyelesaikan skripsi ini. Shalawat serta salam semoga senantiasa tercurahkan kepada Nabi Muhammad SAW, serta para sahabat dan para pengikutnya.

Penulisan skripsi yang berjudul “Kinerja Algoritma *Support Vector Machine* dalam Menentukan Kebenaran Informasi Banjir di Twitter” penulis sajikan untuk memenuhi salah satu syarat untuk meraih gelar Sarjana program S1 (Strata 1) Pendidikan Teknik Informatika dan Komputer.

Dalam penyusunan skripsi ini penulis banyak mendapatkan bantuan dan dukungan moril maupun materil dari berbagai pihak. Untuk itu penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Widodo, M.Kom selaku pembimbing I dan Bapak Bambang P.Adhi, S.Pd, M.Kom selaku pembimbing II yang telah bersedia meluangkan waktunya untuk membimbing dan senantiasa memotivasi, mengarahkan penulis sehingga skripsi ini dapat terselesaikan dengan hasil terbaik.
2. Bu Dr. Yuliatri Sastrawijaya, M.Pd, selaku Kepala Program Studi Pendidikan Teknik Informatika dan Komputer Fakultas Teknik Universitas Negeri Jakarta.
3. Kedua orang tua penulis yang telah membesarkan dan mendidik, serta tiada hentinya memanjatkan doa dan memberikan dukungan kepada penulis.
4. Miftahul Fauziah Fazrin yang telah memberikan dukungan kepada penulis.
5. Teman-teman PTIK angkatan 2013 yang senantiasa memberikan dukungan dan semangat kepada penulis.
6. Serta berbagai pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu dalam pembuatan skripsi ini, semoga budi baik dan amal yang di berikan mendapat balasan yang setimpal dari Allah SWT.

Penulis menyadari bahwa dalam skripsi ini masih jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan saran dan kritik yang bersifat membangun

guna mengembangkan pengetahuan penulis dan perbaikan skripsi ini. Akhir kata semoga skripsi ini dapat berguna dan bermanfaat bagi penulis dan pembaca.

Jakarta, Februari 2017

Penulis

Muhamad Prasetio Dwi Cahyo

5235134395

**KINERJA ALGORITMA *SUPPORT VECTOR MACHINE*
DALAM MENENTUKAN KEBENARAN INFORMASI BANJIR
DI TWITTER**

Muhamad Prasetyo Dwi Cahyo

ABSTRAK

Menurut survei yang dilakukan oleh Asosiasi Penyelenggara Jaringan Internet Indonesia (APJII) mengatakan bahwa sepanjang 2016 sebanyak 132,7 juta orang telah terhubung ke internet. Dalam penggunaan internet seseorang dapat berkomunikasi melalui jejaring sosial. Jejaring sosial adalah sarana untuk bersosialisasi satu sama lain secara online didunia maya (internet). Twitter adalah salah satu dari macam-macam jejaring sosial. Menurut catatan Badan Penanggulangan Bencana Daerah (BPBD) Jakarta terdapat 700 kasus banjir selama periode Januari-Agustus 2016. Pada twitter terdapat user yang memberikan informasi tentang banjir dengan mengirimkan sebuah *tweet*. Namun penggunaan kata “banjir” tidak semua dimaksudkan untuk memberi informasi mengenai banjir. Ada yang menggunakannya hanya sebagai kata kiasan. Penelitian ini menggunakan algoritma *Support Vector Machine* untuk mengklasifikasi *tweet* yang benar memberikan informasi mengenai banjir atau tidak. Algoritma *Support Vector Machine* adalah suatu algoritma yang memiliki tingkat akurasi yang tinggi. Dalam penelitian ini untuk mengevaluasi tingkat akurasi dari algoritma *Support Vector Machine* menggunakan *Confusion Matrix*. Hasilnya adalah tingkat akurasi dari *Support Vector Machine* dalam menentukan *tweet* mengenai informasi banjir sebesar 0,96.

Kata Kunci : Twitter, Banjir, Informasi, *Support Vector Machine*, *Confusion Matrix*.

PERFORMANCE SUPPORT VECTOR MACHINE ALGORITHM IN DETERMINING THE FLOOD OF THE TRUTH INFORMATION ON TWITTER

Muhamad Prasetyo Dwi Cahyo

ABSTRACT

Based on survey from APJII (Asosiasi Penyelenggara Jaringan Internet Indonesia) said that in 2016 there are 132,7 million of peoples connected to Internet. with Internet usage, peoples can go through social networking. Social networking is a platform to communicate online with eachother by using the internet and Twitter is one of the platform. Based on BPBD (Badan Penanggulangan Bencana Daerah), we have 700 information flood during the period of Jan-Aug 2016. On Twitter, there are users gave Flood information with just a single tweet. However the usage of 'Flood' wording is not giving all of Flood information, there are many of usage for the Flood wording in the platform. This research using Support Vector Machine to identify a tweet that giving a flood information or not, inside the research, we evaluate the high accuracy from Support Vector Machine using the Confusion Matrix. As result, the accuracy from Support Vector Machine in determine the Tweet with Flood Information is 0.96.

Keyword : Twitter, Flood, Information, Support Vector Machine, Confusion Matrix.

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	ii
LEMBAR PERNYATAAN	iii
KATA PENGANTAR	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Identifikasi Masalah	4
1.3 Batasan Masalah	5
1.4 Rumusan Masalah	5
1.5 Tujuan Masalah	5
1.6 Manfaat Masalah	5
BAB II TINJAUAN PUSTAKA	6
2.1 Teori	6
2.1.1 Banjir	6
2.1.1.1 Pengertian Banjir	6
2.1.1.2 Penyebab Terjadinya Banjir	6
2.1.2 <i>Data Mining</i>	8
2.1.2.1 Pengertian <i>Data Mining</i>	8
2.1.2.2 Tahap-tahap <i>Data Mining</i>	8
2.1.3 <i>Text Mining</i>	10
2.1.4 <i>Text Preprocessing</i>	11
2.1.5 Pembobotan TF-IDF	12
2.1.6 <i>Vector Space Model</i>	13
2.1.7 Klasifikasi	14
2.1.8 <i>Support Vector Machine</i>	15
2.1.9 <i>K-fold Cross Validation</i>	17
2.1.10 Informasi	18
2.1.11 Media Sosial dan Twitter	19

2.1.12 <i>Confusion Matrix</i>	20
BAB III METODOLOGI PENELITIAN	27
3.1 Tempat dan Waktu Penelitian	27
3.2 Alat dan Bahan Penelitian	27
3.2.1 Alat	27
3.2.2 Bahan	28
3.3 Diagram Alir Penelitian	28
3.4 Teknik dan Prosedur Pengumpulan Data	29
3.4.1 Klasifikasi Secara Manual	29
3.4.2 Text Preprocessing	29
3.4.3 Pembobotan Kata	30
3.4.4 Klasifikasi menggunakan <i>Support Vector Machine (SVM)</i>	32
3.5 Teknik Analisis Data	33
BAB IV HASIL PENELITIAN	35
4.1 Deskripsi Hasil Penelitian	35
4.1.1 Pengumpulan Data	35
4.1.2 Tahap <i>Case Folding</i>	36
4.1.3 Tahap <i>Tokenizing</i>	37
4.1.4 Tahap <i>Filtering</i>	38
4.1.5 Pembobotan Kata	39
4.1.5.1 Menghitung <i>Term Frequency</i>	39
4.1.5.2 Menghitung IDF	40
4.2 Analisis Data Penelitian	40
4.3 Pembahasan	43
4.4 Aplikasi Hasil Penelitian	44
BAB V KESIMPULAN DAN SARAN	45
5.1 Kesimpulan	45
5.1 Saran	45
DAFTAR PUSTAKA	46
LAMPIRAN	48

DAFTAR TABEL

Tabel 2.1 Model <i>Confusion Matrix</i>	20
Tabel 3.1 Model <i>Confusion Matrix</i>	33
Tabel 4.1 Contoh Data Twitter yang Telah Dipeoleh	35
Tabel 4.2 Contoh Data Twitter yang Telah Melalui Tahap <i>Case Folding</i>	36
Tabel 4.3 Tabel <i>Confusion Matrix</i>	42
Tabel 4.4 Tabel Hasil <i>Precision, Recall</i>	42
Tabel 4.5 Tabel Hasil Akurasi	42

DAFTAR GAMBAR

Gambar 1.1 Contoh <i>Tweet</i> tentang Banjir yang Akurat I	2
Gambar 1.2 Contoh <i>Tweet</i> tentang Banjir yang Akurat II	2
Gambar 1.3 Contoh <i>Tweet</i> tentang Banjir yang Tidak Akurat I	3
Gambar 1.4 Contoh <i>Tweet</i> tentang Banjir yang Tidak Akurat II	3
Gambar 1.5 Contoh <i>Tweet</i> tentang Banjir yang Tidak Akurat III	3
Gambar 2.1 Ilustrasi <i>Vector Space</i>	13
Gambar 2.2 Pemisahan Dua Kelas Data dengan Margin Maksimum	16
Gambar 2.3 Prosedur Penelitian	25
Gambar 3.1 Diagram Alir Penelitian	28
Gambar 4.1 Kumpulan <i>bag-of-word</i>	38
Gambar 4.2 Kumpulan Data Bersih	38
Gambar 4.3 Hasil <i>Term Frequency</i>	39
Gambar 4.4 Hasil <i>Inverse Document Frequency</i>	40
Gambar 4.5 Hasil Klasifikasi Algoritma <i>Support Vector Machine</i>	41
Gambar 4.6 Diagram Hasil Klasifikasi dengan <i>Cross Validation</i>	43

DAFTAR LAMPIRAN

Lampiran 1 Sampel Data Penelitian	49
Lampiran 2 Stopwords_id.text	65
Lampiran 3 <i>Source Code</i> Program	67
Lampiran 4 Modul <i>Scikit Learn</i> yang Dipakai dalam <i>Source Code</i> Program	69

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Teknologi di era globalisasi saat ini sudah berkembang, terutama dalam penggunaan internet. Penggunaan internet di berbagai elemen masyarakat sudah berpengaruh besar terhadap kehidupan sehari-hari. Menurut survei yang dilakukan oleh Asosiasi Penyelenggara Jaringan Internet Indonesia (APJII) yang dimuat oleh www.tekno.kompas.com mengungkap bahwa lebih dari setengah penduduk Indonesia kini telah terhubung ke internet. Survei yang dilakukan sepanjang 2016 itu menemukan bahwa 132,7 juta orang Indonesia telah terhubung ke internet. Adapun total penduduk Indonesia sendiri sebanyak 256,2 juta orang. Dengan menggunakan internet saat ini dapat memudahkan seseorang dalam berkomunikasi satu sama lain, salah satunya melalui jejaring sosial.

Jejaring sosial adalah sarana untuk bersosialisasi satu sama lain secara online di dunia maya (internet). Twitter adalah salah satu dari macam-macam jejaring sosial. Twitter adalah layanan jejaring sosial yang memungkinkan penggunanya untuk mengirim pesan, yang dikenal dengan sebutan kicauan (*tweet*). Dengan jejaring sosial Twitter penggunanya dapat dengan mudah berkomunikasi atau memberi informasi baru terkait keadaan lingkungan sekitarnya. Informasi yang sering kita peroleh dari jejaring sosial twitter mengenai fenomena alam seperti banjir.

Seperti yang diketahui fenomena banjir sering terjadi di daerah DKI Jakarta. Menurut catatan Badan Penanggulangan Bencana Daerah (BPBD) Jakarta terdapat 700 kasus banjir selama periode Januari-Agustus 2016. Banyak pengguna twitter memberikan informasi mengenai banjir di daerah lingkungannya dengan cara membuat *tweet*. Berikut contohnya:



Gambar 1.1. Contoh *Tweet* tentang Banjir yang Akurat I



Gambar 1.2. Contoh *Tweet* tentang Banjir yang Akurat II

Gambar 1.1. dan gambar 1.2. adalah contoh *tweet* yang berisikan informasi tentang banjir yang aktual. Pada gambar 1.1. yang diposting oleh pengguna dengan *username* @mizzly_maniez memberi informasi bahwa di daerah Kampung Dukuh telah terjadi banjir yang diakibatkan oleh saluran air (got) tidak dapat menampung air lagi. Pada gambar 1.2. yang telah diposting oleh pengguna dengan *username* @Evamaria__ memberikan informasi bahwa di daerah Bidakara telah terjadi banjir.

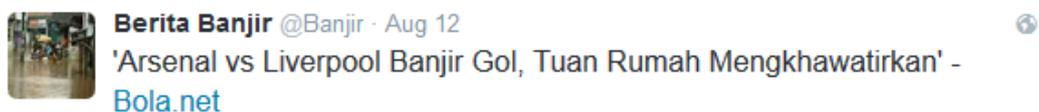
Namun tidak semua *tweet* yang berisi kata “banjir” adalah informasi mengenai lokasi yang sedang terkena banjir. Sebagian ada pengguna jejaring sosial twitter lainnya yang menggunakan kata “banjir” tetapi tidak memberikan informasi mengenai banjir namun penggunaanya hanya sebagai kiasan. Seperti pada gambar 1.3. Pengguna dengan *username* @marianasution menggunakan kata

banjir hanya sebagai kata kiasan yang dimaksudkan bahwa keringat yang berlebihan.



Gambar 1.3. Contoh Tweet tentang Banjir yang Tidak Akurat I

Namun sebenarnya sudah ada pengguna twitter dengan *username* @banjir yang dimaksudkan untuk meneruskan informasi mengenai banjir yang dikirimkan oleh pengguna lainnya dengan cara mention *username* @banjir disetiap tweet yang memberikan informasi mengenai banjir. Tetapi dalam penggunaannya ia memposting semua hal yang mengandung kata “banjir” tidak dipilah-pilih terlebih dahulu sehingga tidak membantu masyarakat dalam memberi informasi mengenai banjir yang akurat. Berikut contohnya:



Gambar 1.4. Contoh Tweet tentang Banjir yang Tidak Akurat II



Gambar 1.5. Contoh Tweet tentang Banjir yang Tidak Akurat III

Pada gambar 1.4. memang mengandung kata “banjir” tetapi kata “banjir” tersebut digunakan sebagai kata kiasan yang bermaksud sebagai banyaknya jumlah gol dalam pertandingan Arsenal vs Liverpool. Dan pada gambar 1.5. kata “banjir” jua sebagai kata kiasan yang dimaksud kata “banjir” tersebut adalah banyaknya diskon yang ada pada acara GIIAS 2016.

Atas dasar tersebut, maka peneliti merasa perlu untuk mengklasifikasi *tweet* yang berisi informasi tentang banjir yang akurat, agar memudahkan masyarakat untuk dapat informasi yang mereka inginkan. Dalam mengklasifikasi text terdapat beberapa metode algoritma, seperti *Support Vector Machine*, *Naive Bayes*, *decision Tree*, *etc.* Untuk mengklasifikasi *tweet* yang berisikan informasi mengenai banjir, peneliti menggunakan algoritma *Support Vector Machine*. *Support Vector Machine* memiliki kelebihan yaitu mampu mengidentifikasi *hyperplane* terpisah yang memaksimalkan margin antara dua kelas yang berbeda (Chou et al., 2014). Pertimbangan peneliti menggunakan algoritma *Support Vector Machine* karena algoritma tersebut memiliki tingkat akurasi yang tinggi seperti dalam penelitian yang dilakukan oleh Achamd Nurhadi (2015) yang menggunakan algoritma tersebut dalam mengklasifikasi konten berita digital bahasa Indonesia, menghasilkan nilai akurasi sebesar 95.42%. Jadi penelitian ini difokuskan untuk implementasi algoritma *Support Vector Machine* dalam pengklasifikasi *tweet* mengenai banjir.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah dijabarkan, maka dapat diidentifikasi beberapa masalah sebagai berikut:

- a. Banyaknya data *tweet* yang berisi satu kata “banjir” namun tidak dipastikan *tweet* tersebut mengandung informasi tentang daerah yang terjadi banjir atau hanya menggunakan kata “banjir” sebagai kata kiasan.
- b. Seberapa baik algoritma *Support Vector Machine* dalam menentukan *tweet* yang berisi kata “banjir” yang merupakan informasi tentang daerah yang terjadi banjir atau bukan.

1.3 Batasan Masalah

Untuk mencegah penelitian keluar dari topik yang dibahas maka penelitian ini dibatasi oleh beberapa point sebagai berikut:

- a. Penelitian dilakukan hanya membahas tentang kicauan (*tweet*) mengenai banjir.
- b. Data yang digunakan dalam penelitian hanya *tweet* yang berisi kata banjir, tidak termasuk kata sinonim dari banjir seperti terendam, tergenang, genangan, luapan, meluap.
- c. Penelitian ini menggunakan data tentang banjir yang terjadi di daerah DKI Jakarta.

1.4 Rumusan Masalah

Berdasarkan latar belakang masalah diatas, maka dirumuskan sebuah rumusan masalah yang akan dibahas dalam skripsi ini adalah “Bagaimana kinerja *Support Vector Machine* dalam menentukan kebenaran informasi banjir di twitter?”

1.5 Tujuan Penelitian

Berdasarkan perumusan masalah yang telah dirumuskan sebelumnya maka tujuan dari penelitian ini adalah mengklasifikasi *tweet* mengenai banjir dengan menggunakan algoritma *Support Vector Machine*.

1.6 Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk mengetahui klasifikasi *tweet* yang berisi tentang kebenaran informasi banjir dari twitter.

BAB II

TINJAUAN PUSTAKA

2.1 Teori

2.1.1 Banjir

2.1.1.1 Pengertian Banjir

Banjir adalah fenomena alam yang terjadi di kawasan yang banyak dialiri oleh aliran sungai. Sedangkan secara sederhana, banjir didefinisikan sebagai hadirnya air suatu kawasan luas sehingga menutupi permukaan bumi kawasan tersebut. Berdasarkan SK SNI M-18-1989-F (1989) dalam Suparta 2004, bahwa banjir adalah aliran air yang relatif tinggi, dan tidak tertampung oleh alur sungai atau saluran.

Terdapat beberapa istilah yang memiliki arti kata yang sama dengan banjir, seperti air besar, air pasang, bah, genangan, luapan, meluap dan melimpah.

2.1.1.2 Penyebab terjadinya banjir

Penyebab Terjadinya Banjir - Banjir dapat terjadi karena beberapa faktor antara lain sebagai berikut:

1. Penyumbatan aliran sungai ataupun sekolah sering membuang sampah di sungai.

Masyarakat beranggapan dengan tingkah lakunya karena jika sampah dibakar, maka akan menyebabkan polusi udara dan bau tidak sedap. Sehingga masyarakat mengambil jalan pintas tanpa memikirkan sebab dan akibatnya. Penyumbatan yang terjadi karena sedimentasi atau

pengendapan area hilir sungai yang dapat mengurangi kemampuan sungai dalam menampung air.

2. Curah hujan yang tinggi

Curah hujan yang relatif tinggi dapat menyebabkan sungai tidak dapat menampung *volume* air yang dapat melampau kapasitas.

3. Pendirian rumah di sepanjang sungai

Masyarakat yang mendirikan rumah di pinggir sungai biasanya mengurangi lebar sungai. Dengan berkurangnya lebar sungai dapat menyebabkan sirkulasi air tidak optimal.

4. Penggundulan hutan

Sikap manusia yang berfikir singkat tanpa berfikir kedepannya sebelum bertindak, menyebabkan manusia bertindak sewenang-wenang terhadap lingkungan. Tindakan tersebut berupa penebangan hutan yang tidak menggunakan sistem tebang pilih. Akibatnya tidak ada pohon untuk menyerap air sehingga air mengalir tanpa terkendali.

5. Sedikitnya daerah serap

Di zaman modern, daerah resapan cenderung ditemukan. Khususnya di daerah perkotaan yang pada dasarnya sangat rentan terhadap banjir, mengingat kondisi kota yang berada di dataran rendah. Daerah serap justru banyak tertutup dengan aspal ataupun pembetonan sehingga air tidak dapat meresap ke dalam lapisan tanah.

2.1.2 Data Mining

2.1.2.1 Pengertian Data Mining

Data mining adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual (Pramudiono,2006).

Menurut (Arif Djunaidy dan Rully Soelaiman. 2001) *data mining* dapat didefinisikan sebagai suatu metode pencarian informasi yang tersembunyi dan merupakan bagian yang penting dalam suatu basis data berukuran besar yang sulit untuk diperoleh dengan hanya menggunakan *query* basis data biasa atau analisis statistik biasa.

Menurut Garnet Group, *data mining* adalah suatu proses menemukan hubungan yang berarti, pola dan kecendrungan dengan memeriksa dengan sekumpulan besar data yang tersimpan dalam penyimpanan dengan menggunakan teknik pengenalan pola seperti teknik statistik dan matematika (Larose, 2005).

Jadi, dapat di simpulkan bahwa *data mining* adalah suatu metode untuk mencari suatu informasi dari suatu kumpulan data yang sulit diperoleh hanya dengan menggunakan *query* basis data biasa.

2.1.2.2 Tahap-tahap Data Mining

Istilah *data mining* dan *knowledge discovery in database* (KDD) sering kali digunakan secara bergantian untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar. Sebenarnya kedua istilah tersebut memiliki konsep yang berbeda, tetapi berkaitan dengan satu sama lain. Dan salah

satu tahapan dalam keseluruhan proses KDD adalah *data mining*. Proses KDD secara garis besar dapat dijelaskan sebagai berikut (Fayyad, 1996):

1. *Data selection*

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang akan digunakan untuk proses data mining, disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. *Pre-processing / Cleaning*

Sebelum proses data mining dapat dilaksanakan, perlu dilakukan proses cleaning pada data yang menjadi fokus KDD. Proses cleaning mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (tipografi). Juga dilakukan proses enrichment, yaitu proses “memperkaya” data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.

3. *Transformation*

Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses data mining. Proses *coding* dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. *Data mining*

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode atau

algoritma yang tepat sangat tergantung pada tujuan dan proses KDD secara keseluruhan.

5. *Interpretation (Evaluation)*

Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut *interpretation*. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya.

2.1.3 *Text Mining*

Menurut Han & Kamber (2006) yang diacu dalam Kestrilia Rega Prilianti & Hendra Wijaya (2014: 1) *text mining* adalah satu langkah dari analisis teks yang dilakukan secara otomatis oleh komputer untuk menggali informasi yang berkualitas dari suatu rangkaian teks yang terangkum dalam sebuah dokumen.

Menurut Fedman, R & Sanger, J (2007) yang diacu dalam Bambang Kurniawan, dkk (2012: 1) *text mining* adalah salah satu teknik yang dapat digunakan untuk melakukan klasifikasi dimana, *text mining* merupakan variasi dari *data mining* yang berusaha menemukan pola yang menarik dari sekumpulan data tekstual yang berjumlah besar. Selain klasifikasi, *text mining* juga digunakan untuk menangani masalah *clustering*, *information extraction*, dan *informationretrival*.

Jadi, dapat di simpulkan bahwa text mining adalah suatu teknik analisis teks yang digunakan untuk menggali informasi dari suatu rangkaian teks dengan menemukan pola menarik dari sekumpulan teks tersebut.

2.1.4 *Text Preprocessing*

Menurut Muchammad Mustaqfiri, dkk (2012) *text preprocessing* adalah tahapan untuk mempersiapkan teks menjadi data yang akan diolah di tahapan berikutnya. Inputan awal pada proses ini adalah berupa dokumen. *Text preprocessing* pada penelitian ini terdiri dari beberapa tahapan, yaitu: proses *case folding*, proses *tokenizing* kata, proses *filtering*, dan proses *stemming*.

- *Case Folding*

Case folding adalah tahapan proses mengubah semua huruf dalam teks dokumen menjadi huruf kecil, serta menghilangkan karakter selain a-z.

- *Tokenizing*

Tokenizing adalah proses pemotongan string input berdasarkan tiap kata yang menyusunnya. Pemecahan kalimat menjadi kata-kata tunggal dilakukan dengan menscan kalimat dengan pemisah (delimiter) *white space* (spasi, *tab*, dan *newline*).

- *Filtering*

Filtering merupakan proses penghilangan *stopword*. *Stopword* adalah kata-kata yang seringkali muncul dalam dokumen namun artinya tidak deskriptif dan tidak memiliki keterkaitan dengan tema tertentu. Didalam bahasa Indonesia *stopword* dapat disebut sebagai kata tidak penting, misalnya “di”, ”oleh”, “pada”, ”sebuah”, ”karena” dan lain sebagainya.

- *Stemming*

Stemming merupakan proses mencari akar (root) kata dari tiap token kata yaitu dengan pengembalian suatu kata berimbuhan ke bentuk dasarnya (stem) (Tala,2003).

2.1.5 Pembobotan TF-IDF

Term Frequency (TF) dan *Inverse Document Frequency* (IDF) merupakan pembobotan yang sering digunakan dalam penelusuran informasi dan *text mining* (K.S. Jones:2004).

Menurut Robertson (2005) yang diacu dalam Rolly Intan dan Andrew Defeng (2006:3) Metode Tf-Idf merupakan suatu cara untuk memberikan bobot hubungan suatu kata (term) terhadap dokumen. Pembobotan dapat diperoleh berdasarkan jumlah kemunculan suatu term dalam sebuah dokumen *term frequency* (tf) dan jumlah kemunculan term dalam koleksi dokumen *inverse document frequency* (idf). Bobot suatu istilah semakin besar jika istilah tersebut sering muncul dalam suatu dokumen dan semakin kecil jika istilah tersebut muncul dalam banyak dokumen (Grossman, 1998). Nilai idf sebuah *term* (kata) dapat dihitung menggunakan persamaan sebagai berikut:

$$IDF = \log\left(\frac{D}{dfi}\right)$$

Keterangan:

D = jumlah dokumen yang berisi *term* (t)

dfi = jumlah kemunculan (frekuensi) *term* terhadap D.

Adapun algoritma yang digunakan untuk menghitung bobot (W) masing-masing dokumen terhadap kata kunci (*query*), yaitu:

$$W_{d,t} = tf_{d,t} * IDF_t$$

Keterangan :

d = dokumen ke-d

t = *term* ke- t dari kata kunci

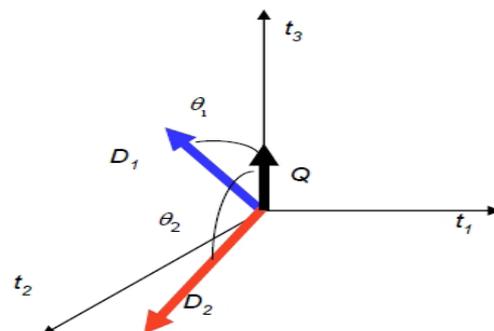
tf = *term* frekuensi/frekuensi kata

W = bobot dokumen ke- d terhadap *term* ke- t

Setelah bobot (W) masing-masing dokumen diketahui, maka dilakukan proses pengurutan (*sorting*) dimana semakin besar nilai W , semakin besar tingkat kesamaan (*similarity*) dokumen tersebut terhadap kata yang dicari, demikian pula sebaliknya.

2.1.6 Vector Space Model

Vector Space Model (VSM) adalah metode melihat tingkat kedekatan atau kesamaan (*similarity*) *term* dengan cara pembobotan *term*. Dokumen dipandang sebagai sebuah vektor yang memiliki *magnitude* (jarak) dan *direction* (arah). Pada *Vector Space Model*, sebuah istilah direpresentasikan dengan sebuah dimensi dari ruang vektor. Relevansi sebuah dokumen ke sebuah *query* didasarkan pada similaritas diantara vektor dokumen dan vektor *query* (Baeza, 1999).



Gambar 2.1 Ilustrasi *Vector Space*

Keterangan:

t_i = Kata di database

D_i = Dokumen

Q = Kata kunci (query)

Cara kerja dari *vector space model* adalah dengan menghitung nilai *cosines* sudut dari dua vektor, yaitu vektor kata kunci terhadap vektor tiap dokumen. Perhitungan vektor space model menggunakan persamaan (1),(2) dan (3)

$$\text{cosine } \theta_{D_i} = \text{Sim} (Q_i D_i) \quad (1)$$

Keterangan:

Q = Kata kunci (query)

D_i = Dokumen ke-i

$$\text{Sim} (Q_i D_i) = \frac{\sum_j w_{i,j} w_{q,j}}{\sqrt{\sum_j w_{i,j}^2} \sqrt{\sum_j w_{q,j}^2}} \quad (2)$$

Keterangan:

$w_{q,j}$ = bobot term j pada query Q

$w_{i,j}$ = bobot term j pada query i

$$\text{cosine } \theta_{D_i} = \frac{Q \cdot D_i}{|Q| |D_i|} \quad (3)$$

Keterangan:

Q = Kata kunci (query)

D_i = Dokumen ke-i

$|Q|$ = Vektor Q

$|D_i|$ = Vektor D_i

2.1.7 Klasifikasi

Menurut (Han,2006) yang diacu dalam Henny Leidiyana (2013:2) Klasifikasi adalah proses penemuan model (atau fungsi) yang menggambarkan dan

membedakan kelas data atau konsep yang bertujuan agar bisa digunakan untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui.

Klasifikasi adalah proses menemukan sekumpulan model atau fungsi yang menjelaskan dan membedakan data kedalam kelas-kelas tertentu, dengan tujuan menggunakan model tersebut dalam menentukan kelas dari suatu objek yang belum diketahui kelasnya (Warih, 2009).

Jadi, dapat disimpulkan klasifikasi adalah proses menemukan model yang membedakan data kedalam kelas-kelas tertentu.

2.1.8 Support Vector Machine

Menurut (Nello Christianini dan John S. Taylor, 2000) yang diacu dalam Dwi Astuti (2007:2) *Support Vector Machine* (SVM) adalah sistem pembelajaran yang pengklasifikasiannya menggunakan ruang hipotesis berupa fungsi-fungsi linear dalam sebuah ruang fitur (*feature space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan learning bias yang berasal dari teori pembelajaran statistik.

Menurut (Lin et al., 2008) yang diacu dalam Acmad Nurhadi (2015:4) *Support Vector Machines* (SVM) adalah suatu metode klasifikasi pola populer dengan banyak aplikasi yang beragam. Pengaturan parameter kernel dalam prosedur pelatihan SVM, bersama dengan pilihan fitur, secara signifikan mempengaruhi akurasi klasifikasi.

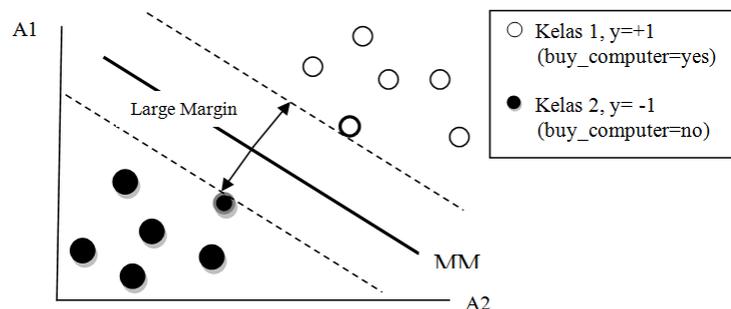
Langkah awal suatu algoritma SVM adalah pendefinisian persamaan suatu hyperplane pemisah yang dituliskan dengan:

$$W \cdot X + b = 0 \quad (1)$$

W adalah suatu bobot vektor, yaitu $W = \{W_1, W_2, \dots, W_n\}$; n adalah jumlah atribut dan b merupakan suatu skalar yang disebut dengan bias. Jika berdasarkan pada atribut A_1, A_2 dengan permisalan tupel pelatihan $X = (x_1, x_2)$, x_1 dan x_2 merupakan nilai dari atribut A_1 dan A_2 , dan jika b dianggap sebagai suatu bobot tambahan w_0 , maka persamaan suatu hyperplane pemisah dapat ditulis ulang sebagai berikut:

$$w_0 + w_1w_1 + w_2w_2 = 0 \quad (2)$$

Setelah persamaan dapat didefinisikan, nilai x_1 dan x_2 dapat dimasukkan ke dalam persamaan untuk mencari bobot w_1, w_2 , dan w_0 atau b .



Gambar 2.2 Pemisahan Dua Kelas Data dengan Margin Maksimum

Pada Gambar 2.1, SVM menemukan hyperplane pemisah maksimum, yaitu hyperplane yang mempunyai jarak maksimum antara tupel pelatihan terdekat. *Support vector* ditunjukkan dengan batasan tebal pada titik tupel. Dengan demikian, setiap titik yang terletak di atas hyperplane pemisah memenuhi rumus:

$$w_0 + w_1w_1 + w_2w_2 > 0 \quad (3)$$

Sedangkan, titik yang terletak di bawah hyperplane pemisah memenuhi rumus:

$$w_0 + w_1w_1 + w_2w_2 < 0 \quad (4)$$

Melihat dua kondisi di atas, maka didapatkan dua persamaan hyperplane yaitu:

$$H_1: w_0 + w_1w_1 + w_2w_2 \geq 1 \text{ untuk } y_i = +1 \quad (5)$$

$$H_2: w_0 + w_1w_1 + w_2w_2 \leq -1 \text{ untuk } y_i = -1 \quad (6)$$

Perumusan model SVM menggunakan trik matematika yaitu formula Lagrangian. Berdasarkan Lagrangian formulation, *Maksimum Margin Hyperplane* (MMH) dapat ditulis ulang sebagai suatu batas keputusan (*decision boundary*) yaitu:

$$d(X^T) = \sum_{i=1}^l y_i a_i X_i X^T + b_0 \quad (7)$$

y_i adalah label kelas dari support vector X_i . X^T merupakan suatu tupel test. a_i dan b_0 adalah parameter numerik yang ditentukan secara otomatis oleh optimalisasi algoritma SVM dan l adalah jumlah vector support.

2.1.9 *K-Fold Cross Validation*

Menurut Fu (1994) yang diacu didalam Bangun Asanurjaya (2012:2) *Cross validation* merupakan metode membagi data menjadi k-subset yang ukurannya hampir sama satu sama lain. Himpunan subset yang dihasilkan yaitu S_1, S_2, \dots, S_k yang digunakan sebagai data latih dan data uji. Dalam metode ini dilakukan perulangan sebanyak k kali dimana salah satu subset dijadikan data uji dan k-1 subset lainnya dijadikan data latih.

Menurut Nursalim et al. (2014) *Cross-Validation* adalah metode statistik untuk mengevaluasi dan membandingkan algoritma pembelajaran dengan membagi data menjadi dua segmen: satu digunakan untuk belajar atau melatih model dan yang lainnya digunakan untuk memvalidasi model. Dalam *cross-validation*,

pelatihan dan validasi set harus menyeberang dalam putaran berturut-turut sehingga setiap titik data memiliki kesempatan yang divalidasi. Bentuk dasar *cross-validation* adalah *k-fold cross-validation*. Bentuk lain dari *cross-validation* adalah kasus khusus *dari k-fold cross-validation* atau melibatkan putaran berulang *k-fold cross-validation*.

2.1.10 Informasi

Definisi kata ‘informasi’ sendiri secara internasional telah disepakati sebagai ‘hasil dari pengolahan data’ yang secara prinsip memiliki nilai atau value yang lebih dibandingkan dengan data mentah.

Menurut (Tata Sutabri 2005:23) diacu dalam Slamet Pebrianto (2010:2) Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.

Sumber informasi adalah data. Data merupakan bentuk jamak dan bentuk tunggal atau data data item. Data adalah kenyataan yang menggambarkan suatu.

Ciri-ciri informasi menurut Davis (1999)

1. Benar atau salah. Dalam hal ini, informasi berhubungan dengan kebenaran terhadap kenyataan. Jika penerima informasi yang salah mempercayainya, efeknya seperti kalau informasi itu benar.
2. Baru. Informasi benar-benar baru bagi si penerima.
3. Tambahan. Informasi dapat memperbaharui atau memberikan perubahan terhadap informasi yang telah ada.
4. Korektif. Informasi dapat digunakan untuk melakukan koreksi terhadap informasi sebelumnya yang salah atau kurang benar.

5. Penegas. Informasi dapat mempertegas informasi yang telah ada sehingga keyakinan terhadap informasi semakin meningkat.

2.1.11 Media Sosial dan Twitter

Media sosial adalah media bersosialisai antar pengguna internet yang dilakukan secara *online* dimana pengguna dapat berinteraksi satu sama lain secara virtual tanpa terbatas ruang dan waktu (Rustian, 2013).

Twitter merupakan suatu media jejaring sosial dimana para penggunanya dapat saling berinteraksi satu dengan yang lainnya. Twitter dapat diakses penggunanya melalui jaringan internet.

Menurut Domikus Juju & MataMaya Studio (2009: 2-3) di acu dalam Dika Putri Utama, dkk (2015: 5) definisi dari twitter adalah sebuah web dan layanan mikroblog yang bisa digunakan untuk melakukan pembaharuan (*update*) berupa sebuah teks panjang maksimum sebanyak 140 karakter, pembaharuan (*update*) di twitter dikenal sebagai *tweets*.

Adapun beberapa fungsi twitter adalah seperti yang diungkapkan oleh Dominikus Juju dan Matamaya Studio (2009: 3) di acu dalam Dika Putri Utama, dkk (2015: 5), yaitu:

- 1) Untuk menuliskan berbagai hal yang ingin ditulis tidak mewajibkan untuk menulis apa yang sedang dilakukan, dapat menuliskan apa saja mengenai perasaan anda.
- 2) Bisa digunakan untuk berbagi (*share*) foto, lagu, dan video kepada teman-teman.
- 3) Mencari teman-teman baru di dunia maya.

- 4) Mempopulerkan web atau blog yang dimiliki dengan cara berbagi tautan (link) artikel yang dibuat pada twitter.
- 5) Bisa digunakan sebagai media informasi secara realtime, seperti memberikan informasi mengenai suatu bencana atau kejadian tertentu.
- 6) Bisa digunakan untuk berbisnis.

2.1.12 *Confusion matrix*

Metode ini menggunakan tabel matriks seperti pada Tabel 1 jika data set hanya terdiri dari dua kelas, kelas yang satu dianggap sebagai positif dan yang lainnya negatif (Bramer, 2007). Confusion Matrix adalah tools yang digunakan untuk evaluasi model klasifikasi untuk memperkirakan objek yang benar atau salah.

Tabel 2.1. Model *Confusion Matrix* (Bramer, 2007)

Klasifikasi yang benar	Diklasifikasikan sebagai	
	+	-
+	TP <i>true positives</i>	FN <i>false negatives</i>
-	FP <i>false positives</i>	TN <i>true negatives</i>

True positives adalah jumlah *record* positif yang diklasifikasikan sebagai positif, *false positives* adalah jumlah *record* negatif yang diklasifikasikan sebagai positif, *false negatives* adalah jumlah *record* positif yang diklasifikasikan sebagai negatif, *true negatives* adalah jumlah *record* negatif yang diklasifikasikan sebagai negative. Evaluasi dan validasi hasil dihitung menggunakan rumus akurasi, *precision*, *recall* dan *f-measure* berikut ini:

1. Akurasi

Perhitungan akurasi dilakukan dengan cara membagi jumlah data yang diklasifikasi secara benar dengan total sample data testing yang diuji.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Precision

Menghitung nilai *precision* dengan cara membagi jumlah data benar yang bernilai positif (*True Positive*) dibagi dengan jumlah data benar yang bernilai positif (*True Positive*) dan data salah yang bernilai positif (*False Negative*).

$$Precision = \frac{TP}{TP + FP}$$

3. Recall

Sedangkan *recall* dihitung dengan cara membagi data benar yang bernilai positif (*True Positive*) dengan hasil penjumlahan dari data benar yang bernilai positif (*True Positive*) dan data salah yang bernilai negatif (*False Negative*).

$$Recall = \frac{TP}{TP + FN}$$

4. F-Measure

Nilai *F-Measure* didapat dari perhitungan pembagian hasil dari perkalian *precision* dan *recall* dengan hasil penjumlahan *precision* dan *recall*, kemudian dikalikan dua.

$$F - Measure = 2 * \frac{precision * recall}{precision + recall}$$

2.2 Hasil Penelitian yang Relevan

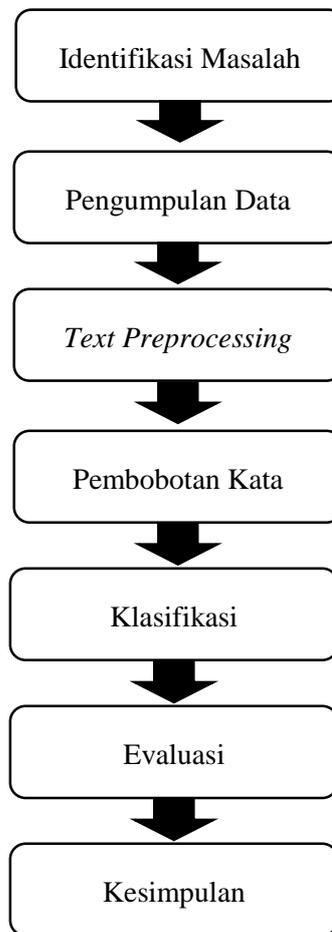
Penelitian mengenai kinerja *support vector machine* dalam menentukan informasi banjir yang akurat di Twitter relevan dengan penelitian yang dilakukan sebelumnya, yaitu:

1. Klasifikasi *tweet* dan pengenalan entitas bernama pada *tweet* bencana dengan *support vector machine*
 - a. Penelitian ini dilakukan oleh Rizky Dermawan di Universitas Gadjah Mada
 - b. Tujuan dari penelitian ini adalah terbentuknya sistem klasifikasi *tweet* tentang bencana dan sistem pengenalan entitas bernama.
 - c. Hasil penelitian tersebut adalah telah dibuat sistem yang dapat mengklasifikasi apakah sebuah *tweet* terkait bencana atau tidak. Sistem pengklasifikasian *tweet* menggunakan metode *tf idf* untuk pembobotan dan *support vector machine* untuk pembuatan model. Rata-rata akurasi dari sistem pengklasifikasian *tweet* adalah 78.37% dan Telah dibuat sistem pengenalan entitas bernama pada *tweet* bencana. Pemisahan entitas pada *tweet* bencana menggunakan metode BIO. Sistem pengenalan entitas bernama menggunakan *support vector machine* untuk membuat model dengan F1score 86%, *recall* 80.75%, dan *precision* 91.98%.
2. Aplikasi pengenalan wajah dengan menggunakan metode klasifikasi *support vector machine* (SVM)

- a. Penelitian ini dilakukan oleh Sri Andriarti Asri di Politeknik Negeri Bali dan telah dimuat dalam Jurnal Logic Vol. 11 No.1 edisi Maret 2011
 - b. Tujuan dari penelitian ini adalah membuat suatu aplikasi pengenalan wajah dengan menggunakan metode klasifikasi. Pengklasifikasi yang digunakan adalah *Support Vector Machine*.
 - c. Hasil penelitian tersebut adalah dari hasil uji coba aplikasi pengenalan wajah dengan metode klasifikasi SVM dengan basis data Yale dan ORL dapat disimpulkan bahwa untuk basis data Yale tingkat keberhasilan pengenalan maksimal adalah 100 %, untuk basis data ORL tingkat keberhasilan pengenalan maksimal adalah 97,5 %, semakin banyak jumlah sampel pelatihan yang digunakan semakin tinggi tingkat keberhasilan pengenalan dan metode klasifikasi SVM terbukti mampu melakukan proses klasifikasi secara maksimal untuk setiap sampel citra yang diuji.
3. Kinerja Algoritma *Naive Bayes* untuk mendeteksi akurasi berita gempa bumi melalui *twitter* berbahasa indonesia
- a. Penelitian ini dilakukan oleh Ivan Bagaskara Arisputranto di Universitas Negeri Jakarta
 - b. Tujuan dari penelitian ini adalah dapat mengklasifikasi tweet berbahasa Indonesia mengenai gempa bumi dengan menggunakan algoritma *Naive Bayes* dan mengetahui keoptimalan algoritma *Naive Bayes* dalam mengklasifikasi teks pendek

- c. Hasil dari penelitian ini dapat diambil kesimpulan bahwa dengan menggunakan algoritma *Naive Bayes* dalam mendeteksi tweet/dokumen pendek tentang gempa bumi memberikan hasil yang memuaskan yaitu dengan rata-rata akurasi sebesar 80,59%.

2.3 Prosedur



Gambar 2.3. Prosedur Penelitian

Berdasarkan identifikasi masalah yang telah dijelaskan pada bab 1, langkah pertama adalah mengumpulkan data *tweet* dari twitter yang akan digunakan sebagai *data train* dan *data test*.

Setelah data terkumpul, proses selanjutnya adalah *text preprocessing* pada setiap *tweet* meliputi proses *case fold*, *tokenizing*, *filtering*. Selanjutnya *tweet* dibuat menjadi sebuah *Vector Space Model* dengan pendekatan *bag of word* dan setiap kata pada *bag of word* akan di beri bobot dengan metode TF-IDF.

Setelah melewati tahap pembobotan kata, kemudian dilakukan teknik klasifikasi dengan menggunakan *Support Vector Machine*. Hasil dari klasifikasi tersebut lalu dievaluasi terlebih dahulu. Tahap terakhir penelitian penulis akan menarik kesimpulan berdasarkan evaluasi hasil dari klasifikasi pada tahap sebelumnya.

BAB III

METODOLOGI PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Program Studi Pendidikan Teknik Informatika dan Komputer Fakultas Teknik Universitas Negeri Jakarta. Penelitian ini dilaksanakan sejak bulan Oktober 2016 hingga Januari 2017.

3.2 Alat dan Bahan Penelitian

Alat dan bahan yang akan digunakan dalam penelitian ini antara lain:

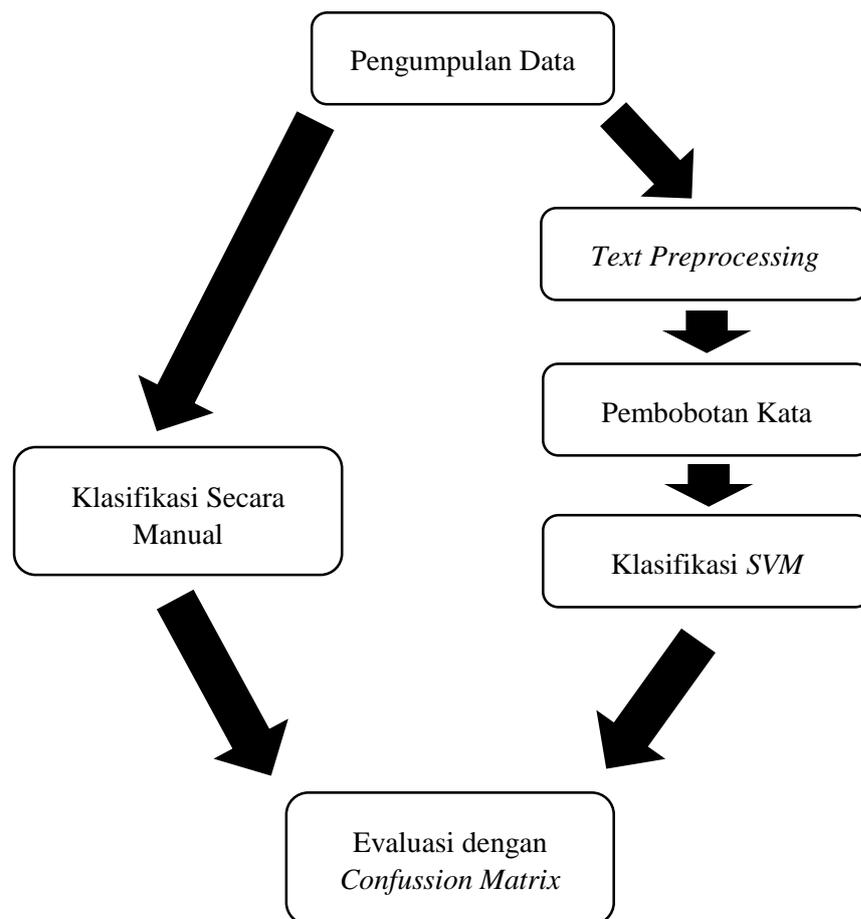
3.2.1 Alat

1. Hardware
 - a. *Processor* AMD E2-1800, 1.7GHz
 - b. *Memory* RAM 4 GB DDR 3
 - c. *Harddisk* 500 GB
2. Software
 - a. Sistem Operasi Windows 7 Ultimate 64-bit *Operating System*
 - b. Microsoft Excel 2016 64-bit
 - c. Notepad
 - d. Anaconda 3 64-bit
 - e. PyCharm Community Edition 2016.2.3
 - f. Menggunakan modul dan fungsi yang ada pada www.scikit-learn.org.

3.2.2 Bahan

Bahan yang akan dipakai pada penelitian ini adalah sekumpulan *tweet* mengenai banjir yang telah dikumpulkan secara manual dalam waktu 2 bulan (November 2016 – Desember 2016).

3.3 Diagram Alir Penelitian



Gambar 3.1 Diagram Alir Penelitian

Proses pertama pada penelitian ini adalah pengumpulan *tweet*. *Tweet* yang terkumpul untuk penelitian ini sebanyak 400 *tweet*. Setelah *tweet* terkumpul *tweet* tersebut akan melewati 2 proses yang berbeda. Proses 1, *tweet* akan diklasifikasikan secara manual oleh peneliti. Proses 2, *tweet* akan masuk tahap *text processing*,

setelah melalui tahap *text processing*, data di buat menjadi *vector space model* dengan pendekatan *bag of word* lalu setiap kata dibobotkan dengan metode TF_IDF. Selanjutnya data akan di klasifikasikan dengan algoritma *Support Vector Machine*. Lalu hasil dari klasifikasi secara manual akan dibandingkan dengan hasil klasifikasi dengan *Support Vector Machine* dan di evaluasi menggunakan metode *Confusion Matrix*.

3.4 Teknik dan Prosedur Pengumpulan Data

Data yang dibutuhkan adalah *tweet-tweet* yang diambil dari *website* Twitter berbahasa Indonesia. Data yang diambil berupa *tweet* sebanyak 400 *tweet*. Proses pengumpulan data dilakukan secara manual.

3.4.1 Klasifikasi Secara Manual

Pada proses 1 dari data yang telah terkumpul akan dilakukan pengklasifikasian untuk menentukan *tweet* mana yang akurat dan tidak akurat. Maksud dari akurat disini adalah *tweet* yang mengandung kata “banjir” yang benar memberikan informasi mengenai banjir, bukan *tweet* yang mengandung kata “banjir” tetapi tidak ada hubungannya dengan peristiwa banjir (penggunaan kata “banjir” sebagai kata khiasan).

3.4.2 Text Preprocessing

Pada proses 2 dari data yang telah terkumpul maka langkah selanjutnya adalah text preprocessing. Terdapat beberapa tahapan, yaitu: proses *case folding*, proses *tokenizing*, dan proses *filtering*.

- *Case Folding*

Pada tahap ini semua huruf dalam teks dokumen menjadi huruf kecil, serta menghilangkan karakter selain huruf ‘a’ sampai dengan ‘z’.

- *Tokenizing*

Selanjutnya data yang sudah melalui tahap *case folding* akan di cacah menjadi perkata. Pemecahan kalimat menjadi kata-kata tunggal dilakukan dengan menscan kalimat dengan pemisah (delimiter) *white space* (spasi, *tab*, dan *newline*).

- *Filtering*

Selanjutnya proses penghilangan *stopword*. *Stopword* adalah kata-kata yang seringkali muncul dalam dokumen namun artinya tidak deskriptif dan tidak memiliki keterkaitan dengan tema tertentu. Didalam bahasa Indonesia *stopword* dapat disebut sebagai kata tidak penting, misalnya “di”, ”oleh”, “pada”, ”sebuah”, ”karena” dan lain sebagainya.

3.4.3 Pembobotan Kata

Dalam pembobotan kata peneliti menggunakan metode TF-IDF.

Tahapannya sebagai berikut:

1. Menghitung Term Frequency

Dalam tahap ini memberikan nilai pada setiap kata yang ada pada dokumen dengan cara menghitung jumlah masing-masing kata pada tiap dokumen.

2. Menghitung DF dan IDF

Selanjutnya menghitung *document frequency* (DF), DF adalah jumlah masing-masing kata pada keseluruhan dokumen, sedangkan Inverse Document Frequency (IDF) untuk mengurangi bobot suatu term jika

kemunculannya banyak tersebar diseluruh koleksi dokumen. Nilai idf sebuah *term* (kata) dapat dihitung menggunakan persamaan sebagai berikut:

$$IDF = \log\left(\frac{D}{dfi}\right) \quad (1)$$

Keterangan:

D = jumlah dokumen yang berisi *term* (t)

dfi = jumlah kemunculan (frekuensi) *term* terhadap D.

3. Menghitung bobot

Adapun algoritma yang digunakan untuk menghitung bobot (W) masing-masing dokumen terhadap kata kunci (*query*), yaitu:

$$W_{d,t} = tf_{d,t} * IDF_t \quad (2)$$

Keterangan :

d = dokumen ke-d

t = *term* ke-t dari kata kunci

tf = *term* frekuensi/frekuensi kata

W = bobot dokumen ke-d terhadap *term* ke-t

Setelah bobot (W) masing-masing dokumen diketahui, maka dilakukan proses pengurutan (*sorting*) dimana semakin besar nilai W, semakin besar tingkat kesamaan (*similarity*) dokumen tersebut terhadap kata yang dicari, demikian pula sebaliknya.

3.4.4 Klasifikasi menggunakan *Support Vector Machine* (SVM)

Sebelum mengklasifikasi menggunakan SVM data harus diubah menjadi bentuk linear sehingga kernel SVM yang digunakan adalah kernel linear dengan menggunakan pendekatan bag-of-words atau Vector Space Model.

Langkah awal suatu algoritma SVM adalah pendefinisian persamaan suatu hyperplane pemisah yang dituliskan dengan:

$$W \cdot X + b = 0 \quad (1)$$

Keterangan:

W = suatu bobot vektor, yaitu $W = \{W_1, W_2, \dots, W_n\}$

b = merupakan suatu skalar yang disebut dengan bias.

X = tupel pelatihan X

Jika b dianggap sebagai suatu bobot tambahan w_0 , maka persamaan suatu hyperplane pemisah dapat ditulis ulang sebagai berikut:

$$w_0 + w_1 w_1 + w_2 w_2 = 0 \quad (2)$$

Setelah persamaan dapat didefinisikan, nilai x_1 dan x_2 dapat dimasukkan ke dalam persamaan untuk mencari bobot w_1 , w_2 , dan w_0 atau b .

3.5 Teknik Analisis Data

Untuk menganalisis performa algoritma dari *Support Vector Machine* peneliti menggunakan metode Confusion Matrix.

Tabel 3.1 Model Confusion Matrix (Bramer, 2007)

Klasifikasi yang benar	Diklasifikasikan sebagai	
	+	-
+	TP <i>true positives</i>	FN <i>false negatives</i>
-	FP <i>false positives</i>	TN <i>true negatives</i>

Evaluasi dan validasi hasil dihitung menggunakan rumus akurasi, *precision*, dan *recall* berikut ini:

1. Akurasi

Akurasi adalah proporsi jumlah prediksi yang benar. Perhitungan akurasi dilakukan dengan cara membagi jumlah data yang diklasifikasi secara benar dengan total sample data testing yang diuji.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Precision

Precision proporsi kasus dengan hasil positif diidentifikasi secara benar. Menghitung nilai *precision* dengan cara membagi jumlah data benar yang bernilai positif (*True Positive*) dibagi dengan jumlah data benar yang bernilai positif (*True Positive*) dan data salah yang bernilai positif (*False Negative*).

$$Precision = \frac{TP}{TP + FN}$$

3. *Recall*

Recall adalah proporsi kasus positif yang diidentifikasi dengan benar. *Recall* dihitung dengan cara membagi data benar yang bernilai positif (*True Positive*) dengan hasil penjumlahan dari data benar yang bernilai positif (*True Positive*) dan data salah yang bernilai negatif (*False Negative*).

$$Recall = \frac{TP}{TP + FN}$$

BAB IV

HASIL PENELITIAN

4.1 Deskripsi Hasil Penelitian

Penelitian ini terdiri dari pengumpulan data, melalui tahap *case folding*, tahap *tokenizing*, tahap *filtering*, pembobotan kata lalu mengklasifikasi data dengan algoritma *Support Vector Machine*. Data sebanyak 400 *tweet* akan diuji di dalam program implementasi algoritma *Support Vector Machine* untuk menentukan apakah *tweet* tersebut termasuk “false” atau “true”. Yang dimaksud dengan “false” adalah apabila *tweet* tersebut menggunakan kata “banjir” tetapi tidak berkaitan dengan bencana alam berupa banjir. Sedangkan yang dimaksud dengan “true” adalah apabila *tweet* tersebut menggunakan kata “banjir” yang berisikan informasi mengenai suatu daerah yang sedang terjadi bencana alam berupa banjir.

4.1.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini berasal dari *Twitter* yang berisikan kata “banjir”. Data yang terkumpul sebanyak 400 *tweet*. Dengan rincian 200 *tweet* mengandung kata “banjir” yang terjadi di DKI Jakarta dan 200 *tweet* mengandung kata “banjir” yang penggunaannya hanya sebagai kata kiasan atau yang sama sekali tidak berhubungan dengan bencana alam banjir.

Tabel 4.1 Contoh Data Twitter yang Telah Diperoleh

No	Username	Tweet	Label
1	@JuveArena	Banjir Kritik, Juventus Butuh Waktu Perbaiki Performa http://bit.ly/2cOckQx via @GOAL_ID	0 (false)
2	@ridwan_ritonga	pulang KERJA kehujanan sepanjang jalan menuju rumah BANJIR	0 (false)
3	@10_paklevih	Mobil ditinggal grgr kejebak banjir tuh rasanya.....??	0 (false)

4	@Arif12P	Dejan Antonic Banjir Kritik, Ini Kata Manajemen Persib http://tinyurl.com/jdo6x23 #bolaindo #news	0 (false)
5	@Panji_Nonton_Tv	Juan Mata: Banjir Kritik Takkan Menghancurkan Man United! http://bit.ly/2e6PzJs olahraga #Bola	0 (false)
6	@kel_ancol	Sabtu (02/04/2016) Genangan akibat rembasan kali Jl. Gunung Sahari Raya #banjir @petajkt @BPBDJakarta	1 (true)
7	@TMCPoldaMetro	04.57 #Banjir 20 Cm didepan Citraland Grogol Jakbar. Hati-hati bila melintas dilokasi"	1 (true)
8	@NopieGodOff	Banjir setinggi 40-50cm orang dewasa di RW. 06 kelurahan Tegal Alur. #Banjir @petajkt @BPBDJakarta @RadioElshinta	1 (true)
9	@cepyamstronk	Ibukota ku tercinta terendam banjir setinggi tumit orang dewasa. @ Jl.Tanah Tinggi Senen Jakarta... https://www.instagram.com/p/BMolh7kg6ns/	1 (true)
10	@DevinEnverfrey	@BPBDJakarta 06.20: #banjir di Komplek KFT Cengkareng barat sudah 3 hr blom surut surut, ketinggian air ud mencapai 1 meter. Mhn bantuannya	1 (true)

Sumber: www.twitter.com

Sesudah mendapatkan data, selanjutnya memberikan label sesuai dengan penggunaan kata “banjir” dalam *tweet* tersebut.

4.1.2 Tahap Case Folding

Dalam tahap ini peneliti menggunakan cara manual dengan bantuan MS. Excel 2016. Tahap ini mengubah setiap huruf dalam data menjadi huruf kecil dan menghilangkan karakter selain huruf ‘a’ sampai dengan ‘z’.

Tabel 4.2 Contoh Data Twitter yang Telah Melalui Tahap Case Folding

No	Username	Tweet	Label
1	@JuveArena	banjir kritik juventus butuh waktu perbaiki performa	0 (false)
2	@ridwan_ritonga	pulang kerja kehujanan sepanjang jalan menuju rumah banjir	0 (false)
3	@10_paklevih	mobil ditinggal gara gara kejemak banjir tuh rasanya	0 (false)
4	@Arif12P	dejan antonic banjir kritik ini kata manajemen persib	0 (false)
5	@Panji_Nonton_Tv	juan mata banjir kritik takkan menghancurkan man united	0 (false)

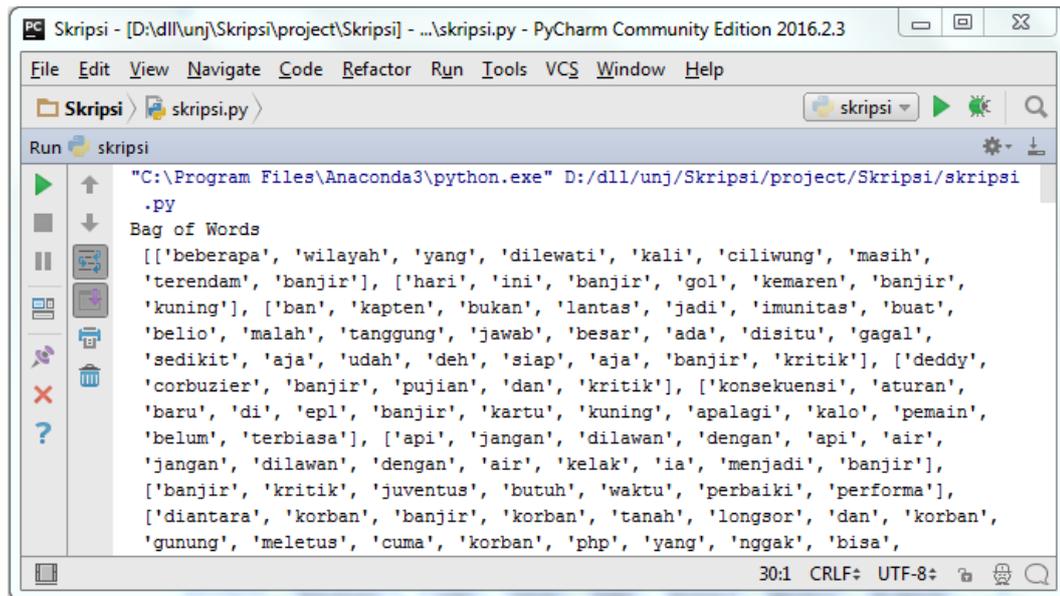
6	@kel_ancol	sabtu genangan akibat rembasan kali jalan gunung sahari raya banjir petajkt bpbdjakarta	1 (true)
7	@TMCPoldaMetro	banjir cm didepan citraland grogol jakbar hati hati bila melintas dilokasi	1 (true)
8	@NopieGodOff	banjir setinggi cm orang dewasa di rw kelurahan tegal alur banjir petajkt bpbdjakarta radioelshinta	1 (true)
9	@cepyamstronk	ibukota ku tercinta terendam banjir setinggi tumit orang dewasa jalan tanah tinggi senen jakarta	1 (true)
10	@DevinEnverfrey	bpbdjakarta banjir di komplek kft cengkareng barat sudah hari belum surut surut ketinggian air udah mencapai meter mohon bantuannya	1 (true)

Di dalam proses ini juga peneliti membenarkan sebagian kata di dalam datanya dikarenakan sifat data yang berasal dari twitter tersebut tidak memiliki standarisasi dan cenderung seenaknya user dalam memposting kata-kata dalam tweetnya tersebut. Contohnya kata “banjirrr” diubah menjadi “banjir”. Dan menghilangkan atribut *tweet* yang tidak diperlukan, contohnya menghilangkan URL. Lalu data tweet tersebut harus dipindahkan ke dalam format .txt satu-persatu, karena program bantu yang dibuat hanya menerima file .txt.

4.1.3 Tahap *Tokenizing*

Di tahap ini peneliti sudah mulai menggunakan program *PyCharm*. Inti dari tahap ini adalah menjadikan *tweet* yang awalnya berupa kalimat dipisah menjadi perkata dan masukan ke dalam *bag-of-word*.

Gambar 4.1 adalah hasil dari proses *tokenizing*. Data tersebut sudah menjadi kata perkata. Yang selanjutnya akan di proses dalam tahap *filtering*.



```

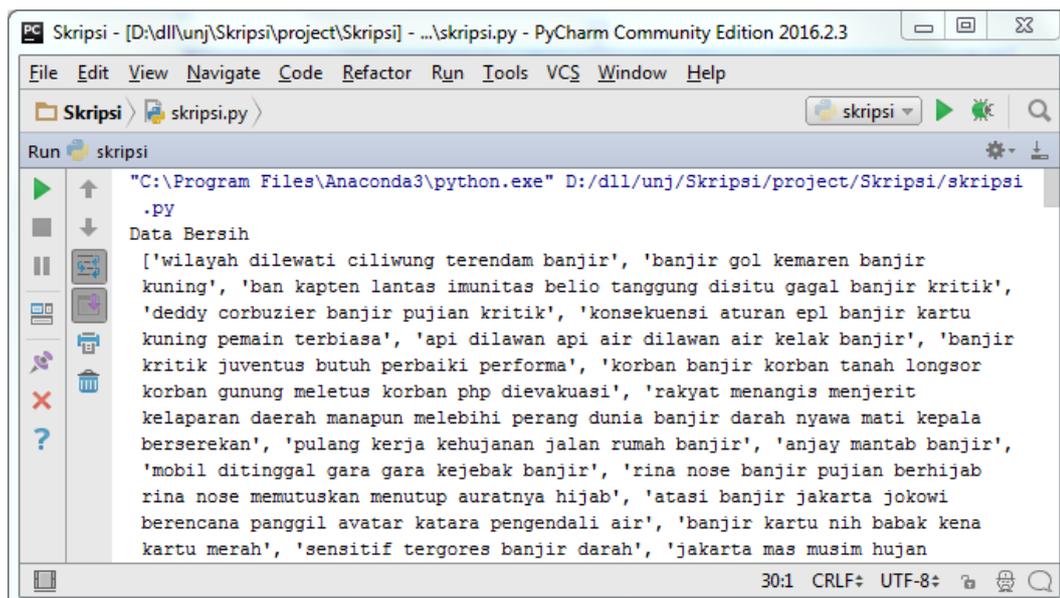
PC Skripsi - [D:\dll\unj\Skripsi\project\Skripsi] - ...\skripsi.py - PyCharm Community Edition 2016.2.3
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Skripsi > skripsi.py >
Run skripsi
"C:\Program Files\Anaconda3\python.exe" D:/dll/unj/Skripsi/project/Skripsi/skripsi
.py
Bag of Words
[['beberapa', 'wilayah', 'yang', 'dilewati', 'kali', 'ciliwung', 'masih',
'terendam', 'banjir'], ['hari', 'ini', 'banjir', 'gol', 'kemaren', 'banjir',
'kuning'], ['ban', 'kapten', 'bukan', 'lantas', 'jadi', 'imunitas', 'buat',
'belio', 'malah', 'tanggung', 'jawab', 'besar', 'ada', 'disitu', 'gagal',
'sedikit', 'aja', 'udah', 'deh', 'siap', 'aja', 'banjir', 'kritik'], ['deddy',
'corbuzier', 'banjir', 'pujian', 'dan', 'kritik'], ['konsekuensi', 'aturan',
'baru', 'di', 'epl', 'banjir', 'kartu', 'kuning', 'apalagi', 'kalo', 'pemain',
'belum', 'terbiasa'], ['api', 'jangan', 'dilawan', 'dengan', 'api', 'air',
'jangan', 'dilawan', 'dengan', 'air', 'kelak', 'ia', 'menjadi', 'banjir'],
['banjir', 'kritik', 'juventus', 'butuh', 'waktu', 'perbaiki', 'performa'],
['diantara', 'korban', 'banjir', 'korban', 'tanah', 'longsor', 'dan', 'korban',
'gunung', 'meletus', 'cuma', 'korban', 'php', 'yang', 'nggak', 'bisa',

```

Gambar 4.1 Kumpulan *bag-of-word*

4.1.4 Tahap *Filtering*

Pada tahap *filtering* akan dihilangkan kata-kata yang tidak bermakna dari kumpulan *bag-of-word*. Kata-kata yang tidak bermakna tersebut sudah dikumpulkan dalam *stopword list* dan dapat dilihat pada bagian lampiran 2. Berikut hasil dari proses tahap *filtering*.



```

PC Skripsi - [D:\dll\unj\Skripsi\project\Skripsi] - ...\skripsi.py - PyCharm Community Edition 2016.2.3
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Skripsi > skripsi.py >
Run skripsi
"C:\Program Files\Anaconda3\python.exe" D:/dll/unj/Skripsi/project/Skripsi/skripsi
.py
Data Bersih
['wilayah dilewati ciliwung terendam banjir', 'banjir gol kemaren banjir
kuning', 'ban kapten lantas imunitas belio tanggung disitu gagal banjir kritik',
'deddy corbuzier banjir pujian kritik', 'konsekuensi aturan epl banjir kartu
kuning pemain terbiasa', 'api dilawan api air dilawan air kelak banjir', 'banjir
kritik juventus butuh perbaiki performa', 'korban banjir korban tanah longsor
korban gunung meletus korban php dievakuasi', 'rakyat menangis menjerit
kelaparan daerah manapun melebihi perang dunia banjir darah nyawa mati kepala
berserekan', 'pulang kerja kehujanan jalan rumah banjir', 'anjay mantab banjir',
'mobil ditinggal gara gara kejabak banjir', 'rina nose banjir pujian berhijab
rina nose memutuskan menutup auratnya hijab', 'atasi banjir jakarta jokowi
berencana panggil avatar katara pengendali air', 'banjir kartu nih babak kena
kartu merah', 'sensitif tergores banjir darah', 'jakarta mas musim hujan

```

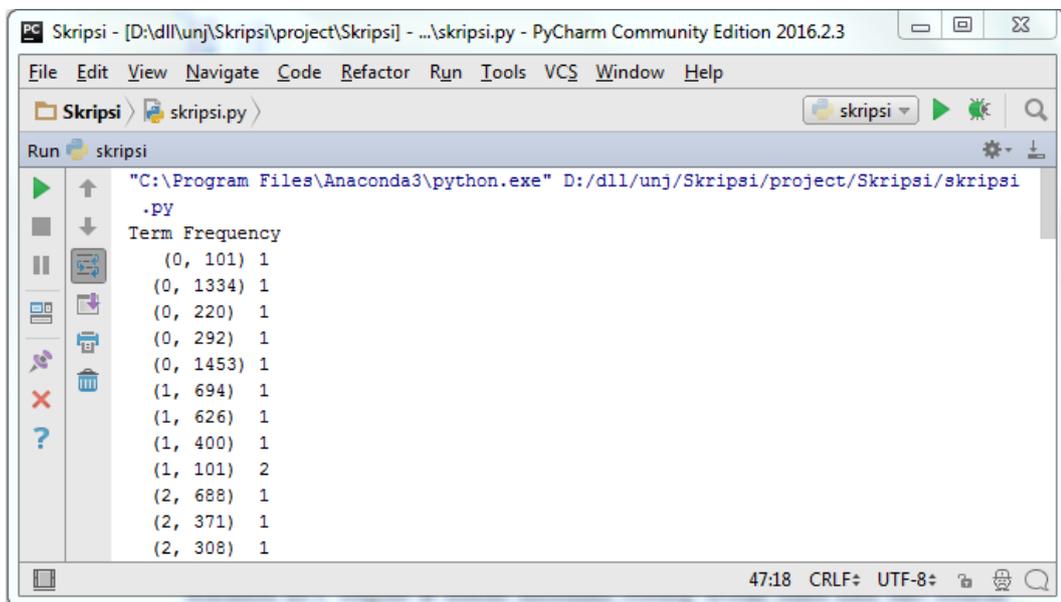
Gambar 4.2 Kumpulan Data Bersih

Gambar 4.2. adalah hasil dari proses *filtering*. Dapat disebut data bersih karena dalam data tersebut sudah tidak mengandung kata-kata yang tidak bermakna lagi.

4.1.5 Pembobotan Kata

4.1.5.1 Menghitung *Term Frequency*

Dalam tahap ini akan menghitung TF (*Term Frequency*). TF adalah jumlah suatu term (kata) dalam suatu dokumen.



```

Skripsi - [D:\dll\unj\Skripsi\project\Skripsi] - ...\skripsi.py - PyCharm Community Edition 2016.2.3
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Skripsi skripsi.py
Run skripsi
"C:\Program Files\Anaconda3\python.exe" D:/dll/unj/Skripsi/project/Skripsi/skripsi
.py
Term Frequency
(0, 101) 1
(0, 1334) 1
(0, 220) 1
(0, 292) 1
(0, 1453) 1
(1, 694) 1
(1, 626) 1
(1, 400) 1
(1, 101) 2
(2, 688) 1
(2, 371) 1
(2, 308) 1
47:18 CRLF+ UTF-8+

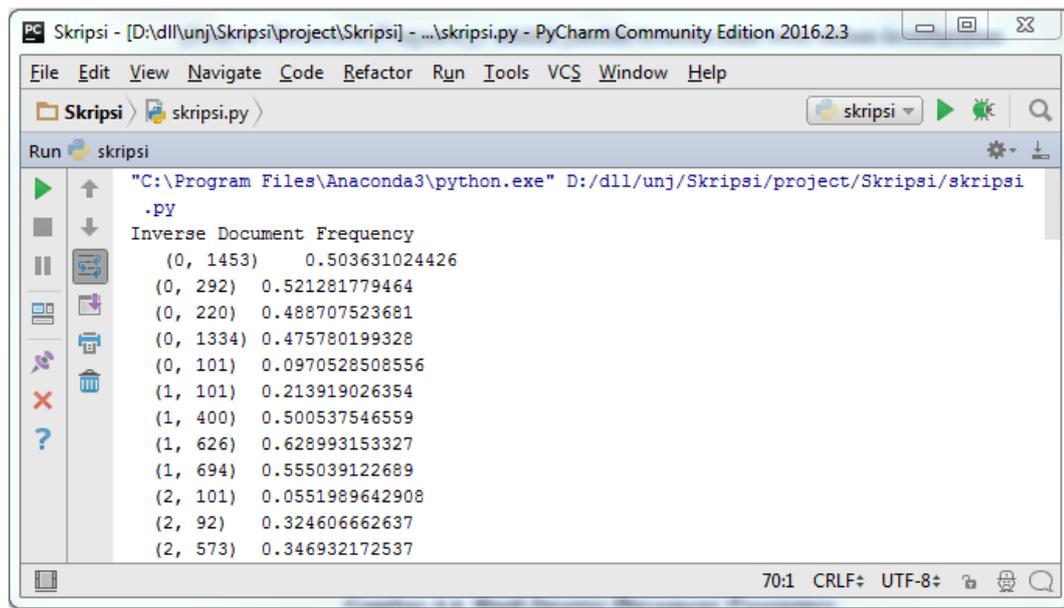
```

Gambar 4.3 Hasil *Term Frequency*

Gambar 4.3 adalah hasil dari penghitungan TF dalam bentuk *Vector Space Model*. Pada baris kedua terdapat angka (0, 101) 1. Peneliti inisialisasikan menjadi (A, B) C. Pada bagian A adalah informasi suatu kata yang terdapat di urutan dokumen ke-i. Bagian B adalah informasi tentang urutan suatu kata dari seluruh kata yang terdapat di seluruh dokumen yang sebelumnya sudah di sorting menurut abjad. Dan bagian C adalah informasi tentang banyaknya kata dalam dokumen ke-i. Dalam menampilkan *output Term Frequency*, urutan kata dimulai dari bagian akhir ke awal.

4.1.5.2 Menghitung IDF

Dalam tahap ini akan menghitung IDF dari *Vector Space Model*. IDF (*Inverse Document Frequency*) adalah jumlah kemunculan term dalam keseluruhan dokumen.



```

"C:\Program Files\Anaconda3\python.exe" D:/dll/unj/Skripsi/project/Skripsi/skripsi
.PY
Inverse Document Frequency
(0, 1453) 0.503631024426
(0, 292) 0.521281779464
(0, 220) 0.488707523681
(0, 1334) 0.475780199328
(0, 101) 0.0970528508556
(1, 101) 0.213919026354
(1, 400) 0.500537546559
(1, 626) 0.628993153327
(1, 694) 0.555039122689
(2, 101) 0.0551989642908
(2, 92) 0.324606662637
(2, 573) 0.346932172537

```

Gambar 4.4 Hasil *Inverse Document Frequency*

Gambar 4.4 adalah hasil dari penghitungan TF. Pada baris kedua terdapat angka (0, 1453) 0.503631024426. Sama seperti pada *output Term Frequency* pada *output Inverse Document Frequency* terdapat 3 bagian (A, B) C. Pada bagian A adalah informasi suatu kata yang terdapat di urutan dokumen ke-i. Bagian B adalah informasi tentang urutan suatu kata dari seluruh kata yang terdapat di seluruh dokumen yang sebelumnya sudah di sorting menurut abjad. Dan bagian C adalah nilai IDF suatu kata. Dalam menampilkan *output Inverse Document Frequency*, urutan kata dimulai dari bagian awal ke akhir.

4.2 Analisis Data Penelitian

Setelah menghitung nilai IDF dan diberi label, selanjutnya masuk ke tahap klasifikasi data. Kumpulan data *tweet* tersebut akan di klasifikasi menggunakan

algoritma *Support Vector Machine*. Pada pengujian algoritma *Support Vector Machine* di penelitian ini menggunakan metode *K-fold Cross Validation*. Sehingga pengklasifikasian akan dilakukan sebanyak 10 kali (dengan presentase data test sebanyak 10% dari data keseluruhan). Dengan menggunakan data test yang berbeda-beda diharapkan hasil dari klasifikasi akan semakin valid.

Instrumen yang digunakan oleh peneliti untuk mengaplikasikan algoritma *Support Vector Machine* dalam bahasa pemrograman *python* adalah *PyCharm Community Edition 2016.2.3*. Dalam pemrosesan klasifikasi, data yang digunakan adalah hasil dari *VSM* yang menghitung nilai *IDF* lalu akan diklasifikasi menggunakan algoritma *Support Vector Machine*.

```

Skripsi - [D:\dll\unj\Skripsi\project\Skripsi] - ...\skripsi.py - PyCharm Community Edition 2016.2.3
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Skripsi skripsi.py
Run skripsi
"C:\Program Files\Anaconda3\python.exe" D:/dll/unj/Skripsi/project/Skripsi/skripsi.py
Confusion Matrix
[[198  2]
 [ 12 188]]
           precision    recall  f1-score   support

   False      0.94      0.99      0.97      200
   True       0.99      0.94      0.96      200

 avg / total      0.97      0.96      0.96      400

Score [ 0.925  0.975  0.875  1.    1.    1.    0.975  0.975  0.975  0.95 ]
Akurasi: 0.96

Process finished with exit code 0
16:1 CRLF UTF-8

```

Gambar 4.5 Hasil Klasifikasi Algoritma *Support Vector Machine*

Pada gambar 4.5 adalah hasil akhir dari klasifikasi menggunakan algoritma *Support Vector Machine*. Berikut adalah detail hasil klasifikasi dengan menggunakan *Confusion Matrix*:

Tabel 4.3 Tabel *Confusion Matrix*

		Diklasifikasikan sebagai	
		False	True
Klasifikasi yang benar	False	198	2
	True	12	188

Pada kelas false algoritma *Support Vector Machine* berhasil mengklasifikasikan sebanyak 210 *tweet*. Namun yang sesuai dengan kelas false hanya sebanyak 198 *tweet* dan 12 *tweet* lainnya algoritma salah dalam mengklasifikasinya. Dan 12 *tweet* seharusnya kelas true dianggap kelas false oleh algoritma *Support Vector Machine*. Pada kelas true algoritma *Support Vector Machine* mengklasifikasi sebanyak 190 *tweet*. Dengan kelas yang sesuai sebanyak 188 *tweet* dan 2 lainnya salah dalam pengklasifikasiannya. Berikut adalah hasil *Precision*, *Recall* dan Akurasi tabel *Confusion Matrix*:

Tabel 4.4 Tabel Hasil *Precision*, *Recall*

Kelas	<i>Precision</i>	<i>Recall</i>
False	0.94	0.99
True	0.99	0.94
avg	0.97	0.96

Berdasarkan tabel 4.4 pada kelas false memiliki nilai *precision* sebesar 0.94 dan nilai *recall* 0.99. dan pada kelas true memiliki nilai *precision* sebesar 0.99 dan nilai *recall* 0.94.

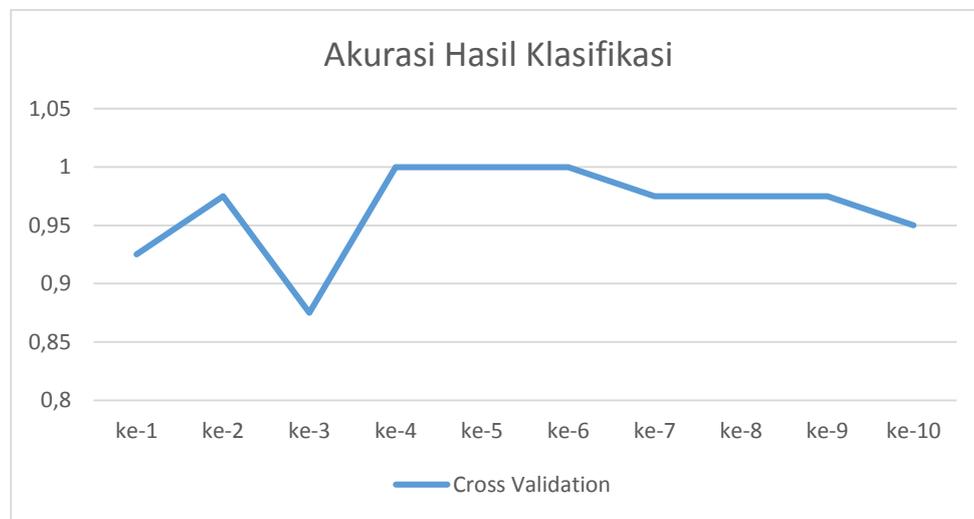
Tabel 4.5 Tabel Hasil Akurasi

<i>Cross Validation</i>	Akurasi
1	0.925
2	0.975
3	0.875
4	1
5	1
6	1
7	0.975
8	0.975
9	0.975
10	0.95
Akurasi Akhir	0.96

Berdasarkan hasil akurasi dari seluruh *Cross Validation* mendapatkan nilai rata-rata sebesar 0.96.

4.3 Pembahasan

Berdasarkan hasil penelitian diatas, peneliti akan membahas mengenai hasil dari klasifikasi algoritma *Support Vector Machine* dengan pengujian algoritma menggunakan metode *K-fold Cross Validation* dengan *K* sebanyak 10 kali.



Gambar 4.6 Diagram Hasil Klasifikasi dengan *Cross Validation*

Berdasarkan gambar 4.6 pada proses awal pengklasifikasian mendapatkan hasil akurasi yang fluktuatif dari CV ke-1 sampai dengan CV ke-4. Tetapi dari CV ke-4 sampai CV-10 sudah mengalami kestabilan dalam mendapatkan hasil akurasi. Terdapat penurunan akurasi yang besar dibandingkan dengan perubahan nilai akurasi dari seluruh hasil akurasi yang berada di antara CV ke-2 dan CV ke-3 dengan perbedaan akurasi sebesar 0,1. Dan terdapat juga kenaikan nilai akurasi yang terbesar yang berada di antara CV ke-3 dan CV ke-4 dengan perbedaan akurasi sebesar 0,125. Dari setiap *Cross Validation* terdapat 3 CV dengan nilai tertinggi, yaitu pada *Cross Validation* ke-4,5 dan 6 dengan akurasi 1. Dan yang terendah dengan nilai akurasi 0,875 dari *Cross Validation* ke-3. Dengan rata-rata akurasi

sebesar 0.96 dari 10 klasifikasi yang berbeda-beda maka dapat disimpulkan bahwa algoritma *Support Vector Machine* memiliki kinerja yang bagus dalam menentukan *tweet* yang berisi informasi mengenai suatu daerah yang sedang terkena bencana banjir.

4.4 Aplikasi Hasil Penelitian

Penelitian ini dapat diterapkan di dunia teknologi informasi khususnya pada aplikasi *website* seperti media sosial, atau aplikasi *website* lainnya untuk membantu dalam mengklasifikasi *tweet* dengan skala data yang besar. Dengan tujuan bila dikembangkan lebih lanjut dapat digunakan sebagai mendeteksi *tweet*/dokumen teks pendek tentang banjir secara *realtime*, maka apabila sudah diketahui suatu daerah sedang terkena bencana banjir selanjutnya lembaga-lembaga yang terkait dapat dengan cepat mengatasinya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian ini dengan judul “Kinerja Algoritma *Support Vector Machine* dalam Menentukan Informasi Banjir di Twitter” dapat disimpulkan bahwa hasil klasifikasi dari kinerja algoritma *Support Vector Machine* dalam menentukan tweet mengenai banjir dianggap bagus karena memiliki nilai *precision* sebesar 0.97, *recall* sebesar 0.96 dan akurasi 0.96.

5.2 Saran

Peneliti memiliki saran untuk penelitian lainnya yang akan dilakukan terkait dengan penggunaan algoritma *Support Vector Machine* dalam mengklasifikasi teks, yaitu :

- Menggunakan metode *K-Fold Cross Validation* sehingga hasil klasifikasi menjadi semakin valid.
- Memperbanyak jumlah data, sehingga algoritma dapat mempelajari berbagai macam karakteristik jenis data. Dan semakin akurat juga hasilnya.
- Menambahkan satu atau beberapa algoritma sebagai pembandingan hasil akurasi, sehingga menemukan algoritma yang terbaik.

DAFTAR PUSTAKA

- Amin, F. (2011). Implementasi Search Engine (Mesin Pencari) Menggunakan Metode Vector Space Model. *Jurnal Ilmiah Dinamika Teknik*, 5(1).
- Amin, F. (2012). Sistem Temu Kembali Informasi dengan Metode Vector Space Model. *Jurnal Sistem Informasi Bisnis*, 2(2), 078-083.
- Asanurjaya, B. (2012). Identifikasi tanaman jati menggunakan Probabilistic Neural Network dengan ekstraksi fitur ciri morfologi daun.
- Dika Putri Utama, A., & Permana, P. Penggunaan Twitter sebagai Media Pembelajaran untuk Meningkatkan Kemampuan Menulis Kalimat Sederhana dalam Pembelajaran Bahasa Jerman.
- Dina, M. (2015). *Skripsi Penerapan Data Mining untuk Rekomendasi Beasiswa Pada SMA Muhammadiyah Gubug Menggunakan Algoritma C4. 5.*, Jurusan Teknik Informatika, Fakultas Ilmu Komputer, UNIDUS, Semarang.
- Intan, R., & Defeng, A. (2006). Hard: Subject-Based Search Engine Menggunakan Tf-Idf Dan Jaccard S Coefficient. *Jurnal Teknik Industri*, 8(1), pp-61.
- Kadir, A. (2014). *Pengenalan Sistem Informasi Edisi Revisi*. Yogyakarta: ANDI
- Kurniawan, B., Effendi, S., & Sitompul, O. S. (2012). Klasifikasi Konten Berita Dengan Metode Text Mining. *Dunia Teknologi Informasi-Jurnal Online*, 1(1), 14-19.
- Kurniawan, D., & Supriyanto, C. (2013) Optimasi Algoritma Support Vector Machine (SVM) Menggunakan Adaboost untuk Penilaian Resiko Kredit. *Jurnal Teknologi Informasi*, 9(1). 38-49.
- Kusrini dan Emha Taufik Luthfi. (2009) *Algoritma Data Mining*. Yogyakarta: ANDI

- Leidiyana, H. (2013). Penerapan Algoritma K-Nearest Neighbor untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor. *PIKSEL (Penelitian Ilmu Komputer Sistem Embedded dan Logic)*, 1(1). 65-76.
- Mustaqhfiri, M., Abidin, Z., & Kusumawati, R. (2012). Peringkasan Teks Otomatis Berita Berbahasa Indonesia Menggunakan Metode Maximum Marginal Relevance. *MATICS*, 4(4). 134-147.
- Nofriansyah, D., Kom, S., & Kom, M. (2015). *Konsep Data Mining Vs Sistem Pendukung Keputusan*. Deepublish.
- Nursalim, S., & Himawan, H. (2014). Klasifikasi Bidang Kerja Lulusan Menggunakan Algoritma K-Nearest Neighbor. *Jurnal Teknologi Informasi*, 10(1), 31-43.
- Pandu, R. (2016). *Skripsi Analisis Metode Extending Document Term dengan algoritma K-Means untuk Meningkatkan Hasil Akurasi algoritma Support Vector Machine Pada Klasifikasi*. Program Studi Pendidikan Teknik Informatika dan Komputer, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Negeri Jakarta, Jakarta.
- Pebrianto, S. (2011). Pembangunan Sistem Informasi Perpustakaan Pada Perpustakaan Umum Kabupaten Pacitan. *Speed-Sentra Penelitian Engineering dan Edukasi*, 2(2), 43-47.
- Pontianak, A. B., & Nurhadi, A. (2015). Klasifikasi Konten Berita Digital Bahasa Indonesia Menggunakan Support Vector Machines (SVM) Berbasis Particle Swarm Optimization (PSO). *Bianglala Informatika*, 3(2). 1-9.
- Prilianti, K. R., & Wijaya, H. (2014). Aplikasi Text Mining untuk Automasi Penentuan Tren Topik Skripsi dengan Metode K-Means Clustering. *Jurnal Cybermatika*, 2(1).
- Ridwan, M., Suyono, H., & Sarosa, M. (2013). Penerapan Data Mining Untuk Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naive Bayes Classifier. *Jurnal EECCIS*, 7(1), 59-64.

Widiastuti, D. (2007). Analisa Perbandingan Algoritma SVM, Naive Bayes, dan Decision Tree dalam Mengklasifikasikan Serangan (Attacks) pada Sistem Pendeteksi Intrusi. *Jur. Sist. Inf. Univ. Gunadarma*, 1-8.

Yogi, K. 2012. *Skripsi Implementasi Antar Muka Aplikasi Data Mining Algorithm Collection dan Modul Preprocessing Data*, Program Studi Ilmu Komputer, Fakultas Ilmu Komputer, Universitas Indonesia, Depok.

Zakapedia, 2015. Pengertian Banjir, Penyebab, Dampak, Cara Menanggulangi. <http://www.artikelsiana.com/2015/08/pengertian-banjir-penyebab-dampak-cara.html> (diakses pada tanggal 4/1/17)

Lampiran 1. Data Penelitian 400 Tweet

No	Username	Twit
1	@LelyLupita	Hari ini banjir gol, kemaren banjir kartu kuning
2	@IndoLiverpoolfc	Ban kapten bukan lantas jadi imunitas buat belio. Malah tanggung jawab besar ada disitu. Gagal sedikit aje, udah deh, siap2 aja banjir kritik
3	@brilionet	Deddy Corbuzier banjir pujian dan kritik.
4	@hari19u	Konsekuensi aturan2 baru di EPL: banjir kartu kuning. Apalagi kl pemain belum terbiasa.
5	@EiyaDania	Api jangan dilawan dengan api, air jangan dilawan dengan air, kelak ia menjadi banjir.
6	@JuveArena	Banjir Kritik, Juventus Butuh Waktu Perbaiki Performa http://bit.ly/2cOckQx via @GOAL_ID
7	@Sandyhp17	Diantara korban banjir, korban tanah longsor dan korban gunung meletus, cuma korban PHP yang nggak bisa dievakuasi :D
8	@pujisyukur361	ingat rakyat sudah menangis menjerit kelaparan seluruh daerah manapun melebihi perang dunia banjir darah nyawa mati kepala berserekan semua
9	@ridwan_ritonga	pulang KERJA kehujanan sepanjang jalan menuju rumah BANJIR
10	@JF_SungJae95	Anjay mentab banjir lagi :v
11	@10_paklevih	Mobil ditinggal grgr kejebak banjir tuh rasanya.....??
12	@HiJakarta	Rina Nose Banjir Pujian Setelah Berhijab: Rina Nose akhirnya memutuskan untuk menutup auratnya dengan hijab.
13	@MH_MuhammadHady	Untuk atasi banjir Jakarta, Jokowi berencana panggil Avatar dan katara Pengendali Air.
14	@ebaidarus_	klo gini terus banjir kartu nih babak kedua , bisa2 ada yg kena kartu merah malah
15	@gadingmaulana9	terlalu sensitif, hanya tergores sedikit udah banjir darah !!!
16	@Gilang41508279	Jakarta mas pada udh musim ujan ...yg rmhnya sering banjir waspada..
17	@angelicancho	pasukan oren, pebangunan mrt, penggusuran rumah liar, hilangnya titik banjir di era pemerintahan siapa sih emang nya sampai di bikin jd anti?
18	@PosPohonTumbang	Sebagian wilayah DKI Jakarta mulai diguyur hujan. Tetap hati2, waspadai jarak pandang terbatas, banjir, pohon & papan reklame tumbang.
19	@pujisyukur361	lihat buktikan sistem yahudi jokowi cina komunis pkc atheis pki diseluruh daerah manapun banjir darah daerah pusat dalam luar negeri dunia
20	@Arif12P	Dejan Antonic Banjir Kritik, Ini Kata Manajemen Persib http://tinyurl.com/jdo6x23 #bolaindo #news
21	@D_Shiryu13	FT Espanyol 3 vs Eibar 3. Perico bangkit setelah ketinggalan 3 gol. Tiga pekan terakhir La Liga banjir gol. Seru.
22	@Bagus_dah	Banjir pikiran deras dibawa darah lihat aku pucat menunggu silaunya, bangkitkan aku sebelum fajar tau dan senja menghentikan ku
23	@Panji_Nonton_Tv	Juan Mata: Banjir Kritik Takkan Menghancurkan Man United! http://bit.ly/2e6PzJs olahraga #Bola
24	@kompascom	Review Bermunculan, Google Pixel Banjir Pujian dan Kritik.

25	@inasetyawati	jangan kecil hati yg gk punya kartu sehat yg pnting ada korek api pd waktu banjir dan lampu mati :D :))
26	@ryan_kevin90	WAFATNYA RASULULLAH SAW, BANJIR AIR MATA PARA SAHABAT, BANJIR AIR MATA KITA http://bit.ly/1UeMVMo
27	@pujisyukur361	mainkan kartu seluruh negara manapun didunia sekalipun banjir darah nyawa ribuan jutaan manusia diseluruh dunia negara manapun apapun
28	@GreenRanger1926	Tekel Brutal ke Pemain Spanyol, Vardy Banjir Hujatan http://ift.tt/2fTYKLe
29	@MTVNsports	Lerby Eliandry Antara Hujatan dan Banjir Pujian http://metrotvn.ws/dN6dEArk
30	@nfitrianingrum	bahkan akdang ketika ndengerin orang lain curhat, yang curhat masih tegar biasa aja, sayanya sudah banjir air mata
31	@pujisyukur361	Bakal Terjadi Keributan Besar di Jakarta banjir darah jutaan nyawa manusia mati dari bayi perempuan ,semuanya sudah murka akibat ahok cina
32	@uju_deder	Aku kasihan pada hujan. Selalu disalahkan. #banjir #longsor
33	@SelancarBola	Mkhitaryan Banjir Pujian setelah Laga Kontra Feyenoord: Henrikh Mkhitaryan banjir pujian... #selancarbola #okezn
34	@OryzaTitisNasti	hujan itu air,, bukan kenangan ☺ jd kalo liat hujan jangan baper berlebihan, ntar banjir, banjir kenangan ☺ keblebek, keblebek deh lu..
35	@UusRsd	10:45 biasanya disini sering banjir air (depan kahatex) kini banjir manusia #CiamisJakarta
36	@KATADATANews	Pegawainya Terlibat Kasus, Menkeu Dapat Banjir Kritik dari DPR
37	@dausmaliki7	Galau itu adalah saat mata anda kriyep kriyep gak tertahan lagi untuk tidur tapi,Man.United mainnya nya lagi banjir gol haha ●●
38	@Healty04	Ibukotanya sllu diserang banjir*gk beda jauh sama mantan yg sllu membanjiri pikiranku dengan kenangan manis*huhu galon☺ #KitaIndonesia
39	@Topi_Jerami	Timnas Argentina Banjir Kritik, Lionel Messi Geram http://dlvr.it/MnN69V
40	@Muzzamilah	udah banjir air mata ini, tp tetep aja gk diizinin.. yg paling bikin sedih.... ☹☹
41	@hernita_samsir	Ya Robb banjir air mata menyaksikan lautan ummat islam di jkt..Panggilan iman serta kecintaan terhadap Alloh&Al qur'an lah yg menggerakannya
42	@arif260989	Selain program booking last minute, @tiket juga banjir diskon dan promo yg nggak kalah menarik. #NgopiBarengTiket
43	@sasabela1	Untung ngga banjir kenangan, tapi banjir kasih sayang. Muehehehe
44	@didi_seprianto	@CNNIndonesia Aksi 212 banjir Simpati n Pujian Aksi 412 banjir Sampah n Taman Rusak Jelas sudah...
45	@fanhad	Banjir apresiasi atas laku baik yg sederhana belakangan ini menandakan sdh terlalu lama kita hidup dlm ide-ide normatif muluk dan tak nyata.
46	@rendiooo	Stadion Pakansari banjir pujian. Ok mulai hari ini siap2 di ig bakal byk remaja yg popotoan. ☺
47	@iwanokawan	@CNNIndonesia Aksi 212 banjir pujian Aksi 412 banjir keluhan Kalo lihat gambar saya jadi bisa bedakan Mana yg beradab dan mana yg biadab

48	@kenangan378	Air mata ibarat hujan , tak akan sia-sia selama berguna. Tapi jangan terus-terusan ya takut banjir :D
49	@TheGalauman	Banjir di Jakarta itu salah satu penyebabnya adalah jomblo, yang selalu berdoa minta hujan
50	@kompasmuda	Hati-hati jg thd dampak yg dapat ditimbulkan, spt banjir, tanah longsor, banjir bandang, genangan, pohon tumbang, dan jalan licin
51	@_Hrst	Kalo main bola, sebisa mungkin di sekitar gawang kiper sepak bola jangan banjir. Kalo sampe lawan cetak satu gol aja, bisa banjir gol nanti.
52	@RCTISports_	Laga banjir Gol ada di Bournemouth kontra Liverpool. Dan pesta Goal Spurs membawa Swansea terpuruk di dasar klasemen sementara. #PLRCTI
53	@TamaraCentil	Aksi Damai 212 Banjir Pujian, Aksi 412 Banjir Kritik
54	@gunturjacks	Giliran dikandang sendiri beuh ramee.. banjir pujian. Pas dikandang lawan mental-nye ciut kayak kerupuk di basahin. #TimnasGagalFinal 😊
55	@mynomyco	Indo vs Viet banjir kartu kuning dan merah ☐
56	@kabarindoinfo	Jelang Lawan Indonesia, Pelatih Vietnam Banjir Hujatan
57	@detikhot	Masuk Nominasi Grammy Lagi, Joey Alexander Banjir Pujian di Lini Masa
58	@NdaNiNda_	Pengen punya Rumah diatas bukit, biar bisa terhindar dari banjir air mata jomblo *eh #DukungSemen
59	@realkartika91	Ya Allah banjir ucapan dan doa dr adek2 shawol hiks makasih kaliaan... Gk berasa uda dr 2009 ak ada d twitter sbg shawol 🙏 berasa pling tua
60	@Evannisa11	Yap bener bgt.. Tapi kalo kyk gini, suka kasian bgt sama pemain kita, udh mah banjir pujian sana sini..
61	@jonhtyd	Banjir pujian kadang bisa membuat seseorang lupa daratan, itulah sebabnya kadang ada orang yg sengaja menghinakan diri agar tak lupa diri.
62	@okezonenews	FOTO: Samakan Habib Rizieq dengan KH. Hasyim Asyari, Ahmad Dhani Banjir Kritik http://okz.me/CXkc7
63	@PLAYRPE	Saya ga tanggung jawab kalo tiba tiba kamarnya jadi banjir air mata yaa :) -T.Haneul
64	@SOSHIBASE	Sontak saja, pelantun "I Just Wanna Dance" itu langsung banjir pujian netter. Selain itu, fans juga jadi teringat masa-masa Tiffany -c-
65	@shyfa99	Lagu Pamit dari Tulus sukses bikin malam mingguku banjir air mata
66	@sayazakyy	Banjir air mata mengaminkan doa Ustdz. @bachtiarnasir #SubuhBerjamaah1212
67	@nengsunshine	udah lama bantal gak banjir air mata, jadi aku rasa kali ini gak apa-apa aku membiarkannya mengalir. bermuara menuju lega.
68	@bngpy	Ahok nangis aja heboh, air mata banjir di mata kaum muslim di Suriah, Rohingnya, Palestina dan tempat lain pada biasa aja.... #template
69	@rendybenz	@kemalpalevi mal setiap lu update apa2 skr banjir hujatan hahahaha. Pelajaran aja. Public figur kayak lu harus jaga kata di medos. Semangat
70	@VilfanVenci	Banjir...banjirrr...!! Yang abis diputusin banjir air mata..!
71	@NaylaNafisa2	Semoga jakarta bisa terbebas dari kebanjiran mau banjir karena air hujan atau banjir air mata karena putus pacaran :v #DjarotDKII

72	@Utora25	Untung di #jakarta itu cuma #Banjir kalau #Gempa ...!!! Pasti pada males jadi #Gubernur
73	@KWSGOfficial	Semoga dunia dibukakan mata hatinya oleh Allah SWT dan segera mengambil sikap u/ menghentikan banjir darah #SaveAllepo Suriah ☺
74	@HaritsAbuUlya	Ada yg asyik dgn permainan, di sebrang ada saudara seakidah yg didera nestapa;cemas, takut, tangisan,banjir darah&kematian#Save Aleppo
75	@dikadallyn	Artikel bagus di tengah banjir ekspektasi media2 Indonesia. #OptimisBolehEkspektasiJangan
76	@NoraAmirah	Banjir sana sini. Allah sedang menegur kita hamba-Nya yang lemah. Tanda Allah uji Bermakna Allah masih peduli. Sabarlah menghadapi dugaan.
77	@tribunjateng	Jelang Final, Unggahan Foto Piala Stefano Lilipaly Banjir Doa dan Dukungan Netizen!
78	@NoteLucu	jeng, udah siap siap buat banjir kiriman? iya udah siap jeng.. denger denger dikirimnya pake jasa JNE jeng.. - - - -
79	@NaylaNafisa2	Kamu butuh barang yang susah di temukan ? cuma disini kamu bisa menemukan selain itu dsni banjir diskon nya Kunjungi 121 Booth Senci
80	@ibadfr	Klo yg cowo lagi banjir error, boleh gak dipeluk dulu ma si cewe biar tenang dulu
81	@dwiiaprillia	KURNIA MEIGA !!!! Langsung banjir pujian
82	@sindojabar	Manahati Lestusen Banjir Ucapan Ultah' http://soccer.sindonews.com/read/1163628/59/manahati-lestusen-banjir-ucapan-ultah-1481942403 ... via @SINDOnews #TimnasDay #TimnasJuara
83	@piscok99	Masuk final heboh di lebay lebay in yakin menang.. Pas kalah heboh juga banjir keritikan.. Ya! MEDIA!!!
84	@NETSPORT_ID	Sempat dicaci, Hansamu kini banjir pujian #AYOGARUDA, hayo siapa yg dulu ikut bully? ;)
85	@mediocrefans	@dnizzz88 @HarukaN_JKT48 saya jg nonton dan nyaris saja menitikan air mata. Tapi kalo last stage theater mungkin bakal banjir air mata.
86	@kimtaebeon	@KAIXROS hujan tinggi bikin banjir sekaligus bikin jemuran pada basah, kemarau bikin tenggorokan sariawan bibir pecah2 kata cjr
87	@sonyaram_	Aku pengen tuh kalo banjir kamu hanyut, tapi ke hati aku aja hanyutnya.
88	@ju2rliet	banjir disini gara gara nurhadi nangis.teu
89	@mama_costaart	banjir emang bnyk jenisnya kang mamam....yg paling mnyenangkan adlah....Banjir diskon di akhir tahun...yeyeyeee....
90	@Kusumaayupaloo	Banjir hadiah besar-besaran di p0kermulia.c0m gak bakal bosan, pastinya selalu happy dan mantap selalu ^^ BB : D3F98F26 #22
91	@uidrus6	Banjir WNA China Ilegal, Indonesia dalam Ancaman Kejahatan Transnasional, Kemana Polisi Yang Gencar Bacot Teroris ? http://fb.me/ArM4OjBH
92	@epjjw94	baru login notif dm udah banjir, ingin berkata kasar :v
93	@WaspadaOnline	Bila Juara, Madrid Bakal Banjir Rekor http://waspada.co.id/sports/internasional/bila-juara-madrid-bakal-banjir-rekor/ ...
94	@sloppypoppy	diem aja kok bisa gitu banget ya pas ngomong makin banjir gantengnya

95	@putskih	Dikatain nggak simpati sama Suriah karena gak posting di medsos. Pas gue posting soal banjir pengungsi Suriah tahun 2014 lo ke mane?
96	@qyuthpie	Dora sama seblak bikin banjir
97	@EndahNurrr	Endingnya bkin banjir air mata :(dan akhirnya bukan Arlan yg tiada, tapi Arkan :(http://my.w.tt/UiNb/10vech0Lcz
98	@Logika_Tuhan	Gempa, tsunami, banjir, gunung meletus, kapal karam, Pesawat jatuh, korupsi, telah melengkapi sebab Indonesia akan jadi bangsa besar. #sdIT
99	@scorpeach	Gue tunggu banjir diskon natalan dulu lah baru nyari memorycard. daripada ntar nyeselwkwkwk
100	@Interisti_Ahmad	Hedew banjir kartu kuning...
101	@Bolonet	https://goo.gl/4MCs7Z - Banjir Tawaran, Andik Vermansah Tetap Setia di Selangor FA
102	@pledizboss	Ini mention box saya banjir udah kaya rekening saya :)
103	@arief8d	Kalau ada lowongan jadi komentator bola, pasti banjir lamaran dari seluruh posok Indonesia.
104	@kataa2_motivasi	Kita harus terus membangun tanggul keberanian untuk menahan banjir rasa takut
105	@2494_aris	Ktk Allah melindungi nabi Ibrahim dr api, nabi Nuh dr banjir bandang & angin kencang...perlindungan Tuhan mana lg yg pantas didustakan
106	@anandaanugerah1	Lebih baik Negera ini di rusak bangsa sendiri dari pada di rusak negara lain #Natkoba Banjir dari China
107	@rendra_rei	Lgsg banjir mentionn hahaha ☺☺☺
108	@kweesiongwey1	Mimpi lainnya Jakarta punya transportasi air,yang saya sampaikan di DISHUB DKI.INI penting untuk,mengatasi banjir, lp pekerjaan,
109	@azisrighthere	Ngeluh macet, tapi setiap tahun beli mobil baru. Ngeluh banjir, tapi setiap hari buang sampah di bantaran kali. Sakit jiwa lu pada.
110	@Irmaerfiana27	orang mungkin gak suka banjir tapi tidak dengan aku sebaliknya aku suka banjir karna kebanjiran cinta kamu @jscmila haha :p
111	@Kurotashiro48	Orang Yunani kuno percaya bahwa mendatangkan bencana alam, seperti banjir dan badai, adalah cara dewa Zeus menghukum orang-orang jahat.
112	@ozradiobandung	Pertandingan Indonesia VS Thailand kemarin banjir ucapan semangat dari beberapa artis salah satunya Raisa #TANGTOP #OZITSWEEKEND
113	@PeluangU	Pohon ditebang tak ditanam ulang; sampah dibuang sembarangan. Panas seminggu kekeringan; hujan seharian banjir yang datang.
114	@monitorindo	Polri Telusuri Berita `Banjir` Pekerja China di Indonesia
115	@WSuratman	Prediksi Jakarta #Banjir apabila #hujan terus menerus selama 4 jam ditambah kenaikan Pintu Air di Katulampa, Depok & Mangarai smpai Siaga 1
116	@nisaadiva	REklamsi tidak menyebabkan banjir itu cuman menurut oknum yg tidak ingin adanya perubahan #MasalahTelukBenoa @DPR_RI @Jokowi @KementerianLHK
117	@AniesaZarai	Sebelum banjir, Kaum Nuh tinggal di sebelah selatan Iraq, yang sekarang terletak di kota Kufah.
118	@N19AB	Sekalinya ngomongin VLado Banjir mention Hahahaha asikin kang #KritikanGakAdaGuna
119	@MCFC_Supporters	Seperti yang sudah diketahui Josep Guardiola Sendiri memang tengah mendapat banjir kritikan belakangan ini.

120	@beylaspriana	Sudah berapa pesawat yg jatuh, kapal yg tenggelam, daerah yg banjir & guncangan gempa. Adakah semua ini tanpa arti bagi manusia yg BERAKAL?
121	@_k2ost	Tampilkan Panggung Pertama di 'Inkigayo', Big Bang Banjir Pujian Netizen https://indokpop.com/tampilkan-panggung-pertama-di-inkigayo-big-bang-banjir-pujian-netizen.html?utm_source=dlvr.it&utm_medium=twitter...-http://indokpop.com
122	@marco_harlian	Trus ntar klo banjir, pemerintah yg disalahin. Brain, where are you? http://fb.me/4Ct8xMX6Z
123	@itsBTStwt	@Renvelvt enak dong cilf kalau banjir bs berenang gratis gr" jim
124	@SuporterFC	Banjir gol di Napoli. Genoa takluk dikandang. Inter Milan menang tipis...
125	@Indomaret	Banjir Hadiah bertransaksi dengan Rekening Ponsel CIMB Niaga di Indomaret selama 1 Desember 2016 - 28 Februari 2017.
126	@monot0ne	Banjir tenaga kerja Cina bukan isapan jempol. #InvasiCina membonceng rezim
127	@_ammargeddon	Banjir tl dgn tweet fans Arsenal yg tgh sakit hati
128	@apry23	Ga semua banjir itu nyebelin, ada juga yg asik, misalnya banjir hadiah
129	@rememorari	gak bisa teriak abis baca penpiik bagus banget mau nangis banjir air mata jadinya capslock di timeline aja
130	@pujisyukur361	Gak peduli sekalipun kerusakan geger pecinan banjir darah nyawa ratusan juta kepala mati di jalan raya solo medan aceh jakarta
131	@IniHeka	kamu itu ibarat comberan yg mampet yang selalu buat hati aku meluap hingga banjir ;)
132	@SipitNahh	Kamu tau gak sih, kamu itu kaya Jakarta kalo lagi musim hujan. Bikin otakku ini macet total gara-gara banjir pikiran tentang kamu.
133	@gua_oon	katanya motor bebek, giliran banjir kok malah ga bisa berenang? ;(Goreng aja ya goreng :v
134	@aqilan_aqilun	Mungkin Arsenal menempati lahan pemerintah atau bantaran sungai hingga menyebabkan banjir. Makanya digusur.
135	@pujisyukur361	Pasukan api air angin halilintar badai puting beliung banjir topan gempa bumi meletus gunung berapi
136	@panggitowe	Perlu kesiapan untk mengatasi banjir dan hujan di hari natal dan taun baru ini brow...
137	@PictureTawa	Waspada Mblo, Banyak Putri Duyung Saat Banjir. Wkwkwk
138	@Finsa07	Ibrahimovic Banjir Pujian: Kepalanya Yahud, Kakinya Hebat http://dlvr.it/MwhQ5G
139	@armenpch	Pasang Banner Ini di Outletnya, Pengusaha Asal Payakumbuh Banjir Doa dan Pujian
140	@DianaLaras88	Keluarga Anang Hermansyah - Tampil Natural Tanpa Make Up Berlebih, Aurel Justru Banjir Pujian http://style.tribunnews.com/2016/12/24/keluarga-anang-hermansyah-tampil-natural-tanpa-make-up-berlebih-aurel-justru-banjir-pujian?utm_source=dlvr.it&utm_medium=twitter...
141	@MedioClub	45+2' Babak pertama di Turf Moor selesai. Banjir kartu kuning namun hasilimbang tanpa gol untuk #BFC & #Boro. #EPL

142	@thanniyya	Berbagi makanan & obat saat Natal, aksi Stefan William banjir pujian http://ouo.io/GlZmiP
143	@_AhmadHifni	Alhamdulillah atas nikmat sehat, dan yg paling penting, masih bisa baca buku agar pikiran tetap waras di tengah2 banjir informasi hoax
144	@hendra494	Apa hebatnya ahok, jakarta masih maceeettttt, banjir..... #BebaskanAhok
145	@viviantgunawan	Banjir Tenaga Kerja China Ilegal, Jokowi: Jangan Percaya Fitnah
146	@taufikalam110	Berlimpah Gelar pada 2016, Real Madrid Banjir Bonus.
147	@delermotorjogja	FILM TERBARU : Ditonton 800.000 Orang dalam 5 Hari, Hangout Raditya Dika Banjir Pujian http://dlvr.it/MzF93h
148	@Mimin_concetha	Instagram Korban Pembunuhan Sadis di Pulomas Banjir Ucapan Belasungkawa, Inilah Sosoknya http://dlvr.it/Mz9IGq
149	@deliknewscom	Polres Jakarta Utara Banjir Apresiasi Karena Natal Kondusif
150	@ErnaNoviani7	Sumpah banjir air mata lihat Arka nangis, dan nangis itu kyak k hati bgt....bner2 bapeeeerrrr @ACRCTI
151	@EnggarJKT48	Takut Banjir Pekerja Cina, Buruh Tuntut Bebas Visa Dicabut
152	@manjalarap	#cristianoronaldo #realmadrid Banjir Trofi di Tahun 2016, Ronaldo Jawab Semua Keraguan.
153	@popsiepops	Abis ngeludahin fans nya,sekarang temen sendiri diludahin sampe banjir Inget2 aja jangan sampe besok2 ngeludahin diri sendiri..
154	@Robert_Pei	Ahok Janji Selesaikan Masalah Banjir di Kalisari
155	@Diana_aryYulian	Alhamdulillah, desmber banjir! :D Alhamdulillah 12 Dus besar habis semua laris manis dlm wktu 2 minggu krywn lpngn mghsilkn 3 jt :D
156	@scripkidding	Antisipasi Banjir, RT Setempat Memasang Behel Di Pinggiran Sungai Agar Tak Meluap. #scripkidding
157	@olstore_unik	Baguslah kalau kanal banjir timur sudah mulai ada buaya muara. Biar yang pada mesum di untal buaya.
158	@erliyani	Klo yang sebelah sono, biar gak banjir, kotanya kudu dinaekin. Yakaliiii...
159	@kwarnas	Pramuka banjir kenangan manis ☐ #PramukaKeren
160	@ArthalitaDiah	Tak Setuju Penggusuran, lalu untuk mengatasi banjir dengan cara apa? Membuat kota yang bisa pencet tombol terbang gitu?
161	@R_ER_EF3	Teman2..realita nya adlh jakarta itu aj udh pusing ngatasi banjir klo ada rumah apung,mw ngapung dmn?dilaut apa iya?
162	@Sijo_su	th 2017, kata dukun di infoteimen: jakarta bakal banjir, bakal ada artis top cerai, ada politisi ketangkap KPK--ini semua mah bukan ramalan
163	@p36ol	Uwow... Kelas FB Personal Banjir Orderan udah keisi 150-an seat ajah. Kamu beneran nih gak mau daftar? Bayarnya suka-suka looohh...
164	@stellamanyari	#BeritaIndonesia Pembunuhan di Pulomas Terungkap, Kapolda Metro Jaya Banjir Apresiasi
165	@favoreast	@baesically kirain bahagia di luar doang dalemnya mah banjir air mata
166	@susiamalia037	Anak ayam di pinggir kali Singgah sebentar sekedar mampir Sekolahku bagus sekali Kalau hujan langsung banjir
167	@Bangcendil	Bagi seekor semut, beberapa tetes air hujan adalah Banjir.

168	@awandanu1	Duet Ibrahimovic dan Pogba Banjir Pujian
169	@IsmawatiDasuki	Kenapa banyak sekali baju-baju lucu di saat akhir tahun banjir diskon begini? D'uh
170	@Red_Squad20	Liverpool vs stoke 4-1 banjir gol :)
171	@Adot1998	rumah apung ndasmu banjir zaman si zalim Ahok gusur alasan biar ngak banjir ngak taunya buat pengembang cina
172	@SPORTKU	Tottenham Hotspur Banjir Gol, Dele Alli Tampil Impresif (Video) --> https://goo.gl/WNSA8P #Bola #EPL #LigaPrimer
173	@KicauPulitik	Unggah Foto Ini di Instagram, Indra Sjafri Banjir Sanjungan http://ift.tt/2isCvxZ #KicauPulitik
174	@BJZweir	WNA CINA, LAGI, LAGI. Kali ni seenaknya bekerja di Bogor. Petugas ga cukup tuk tangani banjir #WNACina ke negeri ni.
175	@jugqooqy	tugas deras sekali. arus banjir aja kalah[]
176	@selvyrama	Giliran ngetweet foto lembaran buku+ penulisnya, gilaaaa banyak skali yg retweet dan love2ki ahaha Banjir notification. Sering2 ah wakakaa
177	@opiekbali	Badminton dulu... Banjir keringat
178	@fannyardianti_	Tugas udah kaya banjir ga surut-surut
179	@Maya25_	Mulai banjir dimana mana. Tugas liputan ngaa usah susah" nih pak dosen 😊😊
180	@mllnx_	Mau uas malah banjir tugas+ulangannya 😊 susah idup 😊😊
181	@ilah_hermawan	Hari minggu : semangat ngerjain tugas. Semoga bs selesain 2 tugas sekaligus di hari ini. Lagi banjir tugas
182	@siayip	latihan yg bikin banjir keringat *ngos2an*
183	@irmaTandi	Minggu ini banjir dengan tugas 😊😊👊👊👊
184	@nrsvvr	Kantor gue tiga hari ini banjir kunjungan. Dua hari kemaren Decil...
185	@SelaAgraini	Serek" banjir 😊😊😊😊
186	@Aiy_Moel	Ini tuh dua2.nya mie goreng tapi yang satu kena banjir 😊
187	@pradanaprilia	Desember banjir film, banjir tugas, banjir-banjir-banjir
188	@Rouliberation	Salut bgt tukang gw kerja semangat pdhl panasnya poll bgt euy, gw aja smpe banjir keringat cuma ngeliatin. Kerja keras itu ga ad yg sia2 yes
189	@benzbara_	ngakunya tahan pedes, tapi makan ayam geprek pake cabe satu aja udah banjir keringat. #bara
190	@Wennyyyy_	Pusing gue politik tahun ini gua mah coblos siap bae lah inntinya bisa bikin jakarta bebas banjir, terus siapin tempat tinggal buat orang
191	@naffaadewi	Desember banjir lemburan
192	@FerryStanislaus	Banjir manusia
193	@MozzaFP	banjir keringat
194	@bangsari	Barusan lewat rombongan paspampres banjir keringat. "Nyari es dulu. Panas bgt, nih", kata seseorang pada lainnya. :))
195	@Larasatiii93	Karna akhir tahun banjir diskon 😊😊😊
196	@nagarukun	Kipasan masuk angin, ra kipasan banjir keringat
197	@kamandaka	Itu namany anti kritik. Banjir TKA ilegal Cina cuma dibantah; jutaan pengangguran di dlm negeri tdk dcarikn solusi
198	@idafitrih	JAFRA banjir hadiah di January ini , untuk kamu yang sudah lama gak aktif di Jafra 4 bulan lalu,... https://www.instagram.com/p/BOq9wqdBmVs/
199	@teesIndonesia	Sambil menikmati pesta tahun baru, bisa kali mampir dulu ke http://buff.ly/2hBXikb , karena ada banjir diskon!

200	@alfaisal_95	Gak biasa kepanasan, kalo kepanasan langsung banjir keringat..😓😓
201	@hendinPR	@TMCPoldaMetro 03.10 #Banjir sekitar 20cm kp.bulak seri,kalideres,jakarta barat
202	@TMCPoldaMetro	04.57 #Banjir 20 Cm didepan Citraland Grogol Jakbar. Hati-hati bila melintas dilokasi"
203	@TMCPoldaMetro	06:44 #Banjir di Bukit Gading Raya & Boulevard Barat Kelapa Gading Jakarta Utara, lalin tersendat. @sute88
204	@TMCPoldaMetro	07:39 #Banjir menggenangi Pemukiman Warga di RT. 10 & 12 RW. 13 Cengkareng Timur #Jakarta Barat. @rohim_saber
205	@TMCPoldaMetro	07:56 #Banjir di Jl. Hybrida s/d Sport Mall Kelapa Gading #Jakarta Utara, berimbas lalin tersendat. https://twitter.com/newpendi64/status/703020071858237440/photo/1pic.twitter.com/wjsla4c4lx @newpendi64
206	@NopieGodOff	Banjir setinggi 40-50cm orang dewasa di RW. 06 kelurahan Tegal Alur. #Banjir @petajkt @BPBDJakarta @RadioElshinta
207	@aimancollege	Genangan di Kelurahan Cipinang Melayu RT.004/004 5-30 cm #banjir @petajkt @BPBDJakarta
208	@DevinEnverfrey	@BPBDJakarta 06.20: #banjir di Komplek KFT Cengkareng barat sudah 3 hr blom surut surut, ketinggian air ud mencapai 1 meter. Mhn bantuannya
209	@yosepten	@petajkt #banjir kavling polri jelambar 10-30 cm
210	@TMCPoldaMetro	10:30 #Banjir 40 cm di Jl. Tanjung Pura 5 Kp. Koang Kalideres #Jakarta Barat. @ChaYoung4nch4
211	@TMCPoldaMetro	10:32 #Banjir di Jl. Basoka Raya Joglo Kembangan #Jakarta Barat. @Ranggara1005
212	@mbakje	#banjir RT 06 RW 07 Cililitan @ Jl.dewi sartika Cililitan kecil
213	@danihamdani99	@petajkt #Banjir depan SD 07 Kalibata Jaksel dekat kantor kelurahan. Banjir rutin datang tiap hujan sudah menahun
214	@Deden_32	#Banjir jl pengangsaan 2 klapa gading tergenang air @RadioElshinta @TMCPoldaMetro @RadioDakta @RadioANDIKA
215	@BPBDJakarta	Genangan di Jl.Pancoran Barat IV,Jkt Sel ; 10-20 cm #banjir @petajkt
216	@BPBDJakarta	17.20 Info #Banjir Jl. MH. Thamrin sepan Gedung Thamrin, Genangan 10 cm
217	@kel_ancol	Sabtu (02/04/2016) Genangan akibat rembasan kali Jl. Gunung Sahari Raya #banjir @petajkt @BPBDJakarta
218	@TMCPoldaMetro	00:43 #Banjir di Jl. Kemandoran Kel. Grogol Utara sudah surut & siap dilintasi kendaraan. @Jumadi171
219	@SonoraFM92	#Lalin92FM 23:50 Imbas #banjir, Traffic Light Cilandak arah ke Kampung Rambutan lalin padat. @may_warko"
220	@BPBDJakarta	#banjir di RW.03/Rt.10,11,Kel.pondok Labu,jkt Sel ; 20-100 cm @petajkt @BeritaJakarta
221	@BPBDJakarta	RT; 07.00 Genangan air dekat sekolah strada pejabat pasar minggu. lalin tersendat @sigit_dbritto #banjir @petajkt
222	@ShipitMala	#banjir daerah kp.makasar rw.07 kel.makasar jak-tim banjir sedada orang dewasa,@petajkt @BPBDJakarta
223	@Damsaddam	#banjir di pademangan barat @petajkt @Ahok @Ahok_BasukiTP @AhokDjarot
224	@PapanyaAbin	#banjir makin tinggi di komp. Bumi Harapan Permai @RadioElshinta @TMCPoldaMetro @petajkt @basuki_btp @BPBDJakarta

225	@aimancollege	@BPBDJakarta @petajkt #banjir luapan kali cipinang 30-70 cm
226	@firestealth	@BPBDJakarta @petajkt @basuki_btp Banjir di Komp Pondok Karya Mampang Prapatan Jaksel. Tgl: 21/4/2016 pkl: 12:11
227	@tetsuansil	@petajkt #banjir 30cm jl buncitpersada
228	@irsoed	@petajkt #banjir Banjir di Jl. Nusa Indah 1 dst kel. Jagakarsa, jam 09.00 WIB, ketinggian 40cm. Skrg berangsur surut
229	@kirismaul	@petajkt #banjir di kampung rambutan kecamatan ciracas setinggi paha orang dewasa
230	@ikurniawaaan	@petajkt #banjir di sepanjang Jalan Bangau, Gunung Sahari nih. Sekitar 30 Cm.
231	@Saputronikodwi	@petajkt @petajkt Banjir sekitar setengah meter di Jalan Tanah Merdeka kampung rambutan. Motor tidak bisa melintas.
232	@oktavnugroho	@petajkt Banjir komp pondok karya,pela mampang,JakSel
233	@fazafarha	@petajkt banjir rt 12 rw 05 kel kalibata, mohon bantuannya @BPBDJakarta #banjir
234	@malindamayanti	@petajkt kondisi banjir di Hek, Kr Jati 07.30. Motor tdk bisa lewat. Hanya bbrp mobil berukuran sedang yg melintas
235	@onlinesuzuki	04:45 #Banjir di Komplek Paspampres Kramat Jati #Jakarta Timur. @djumadri
236	@IrwanNopiyanto	15.20 WIB Air masih meluap d kwsn Jln. P. Jayakarta Dalam @petajkt @BPBDJakarta @SonoraFM92 @RadioElshinta #Banjir
237	@BPBDJakarta	16.00: RT @bambang_irawan: Kondisi terkini banjir di Mampang Prapatan XVI, Jaksel sekitar 50cm @petajkt
238	@TMCPoldaMetro	22:39 Imbas #banjir di KM 3 Tol Taman Mini, Tol Slipi arah ke Cawang-Cibubur lalin padat.
239	@glienmourinsie	Akses jalan dari jl raya bogor menuju TMII dan dari TMII menuju Kramatjati terputus #banjir @TMCPoldaMetro
240	@n_vievie	Banjir di kelurahan pinang ranti jakarta timur @petajkt #banjir
241	@IrwanNopiyanto	Jam 9 Pagi ini banjir di jln Pangeran Jayakarta Jakarta Pusat @MaybankID @KompasNews @BPBDJakarta @petajkt #Banjir
242	@kel_ancol	Kamis (21/04/2016) Jl. Pelabuhan Ratu samping Aston terdampak 30 cm #banjir @petajkt @BPBDJakarta
243	@Damsaddam	Kawasan mangga dua,wtc, banjir..silakan cari jalur lain @petajkt @AhokDjarot @basuki_btp
244	@gunturwijaya27	#Banjir .. Banjir.. Banjir (at Jl kp pulo, rt 08 rw 04, pinang ranti)
245	@TMCPoldaMetro	15:32 #Banjir di wilayah Taman Ratu Jakarta Barat. @NNidyanti
246	@TMCPoldaMetro	15:34 #Banjir di KM 4 Tol Kedoya (arah ke Tomang) & berimbas lalin tersendat. @SusantoFery
247	@TMCPoldaMetro	15:41 #Banjir 30-40 di Jl. Kedoya Raya Jakarta Barat & lalin tersendat. @Guna_Artha
248	@TMCPoldaMetro	01:44 #Banjir 40-80 cm di Patra Tomang 1 Duri Kepa Jakbar. @ronaldalfons
249	@uehara_tengaar	@petajkt #banjir #BanjirRob jalan danau sunter barat depan pengadilan jakarta utara, tinggi sekitar 40cm dua arah parah!!
250	@basukiDKI	Test #banjir @petajkt banjir di abdul muis 50 cm

251	@SilmiKaaffah15	@petajkt banjir di kelurahan pinang Ranti, air kali Cipinang luber
252	@edhos12	@petajkt Jl. H. Amsir Kel. Cipinang Melayu Kec. Makasar #Banjir
253	@TMCPoldaMetro	19:59 #Banjir setelah Layang Roxy (arah ke Harmoni), agar hati2 bila sdg melintas. @cikhal83
254	@Rahindradi84	Banjir kali cipinang meluap Jkt timur kluahan dukuh kcamatan kramat jati perum bumi harapan permai #banjir @petajkt
255	@wihantoro_y	@BPBDJakarta @petajkt #Banjir jl lapangan tembak cilandak timur
256	@naldt	@petajkt hindari jalan belakang citos, banjir ga bisa dilewati mobil dan motor #banjir
257	@muliakur	@BPBDJakarta : banjir 120cm di pemukiman jl.kebagusan 2 rt.03/06 bagian bawah,kebagusan pasar minggu jakarta selatan
258	@wheeqo	@petajkt #banjir di komplek Perdatam, Ulujami
259	@NdoRie	@petajkt banjir kemang selatan 8 kel bangka sepinggang / 1m... #banjir #Jakarta #banjirjakarta
260	@NdoRie	@petajkt kondisi #banjir #jakarta jln kemang selatan 8 kel bangka siang ini...
261	@topikes	Banjir hujan semalam di depan smp n 257 kp rambutan hingga komplek nuansa biru @ Kp.Rambutan... https://www.instagram.com/p/BJT0zUeAhqk/
262	@rinohere	#Banjir #kalicipinang Jl.penggilingan baru 1 kp.dukuh
263	@Fferentya	#banjir @petajkt banjir sudah melanda Jl. Abuserin Cilandak,Jakarta Selatan dan sekitarnya .
264	@FFerentya	@BPBDJakarta #banjir Banjir sudah melanda Gandaria Selatan ,Jaksel dan saat ini sudah mulai surut
265	@TMCPoldaMetro	21:25 #Banjir dpn Tamani Jl. Kemang Raya Jaksel, sementara tidak bisa dilintasi semua jenis ranmor. @ArnaldoPellini
266	@antonwilliam	Genangan Kali Mampang sudah sejak sore @petajkt #banjir
267	@ndoblely	Help pak polisi,@TMCPoldaMetro, @DKIJakarta @RadioElshinta @basuki_btp kejebak banjir di darmawangsa ga bisa kmn mna
268	@kittyjelita	Jalan raya antasari banjir bagi pengendara harap berhari" @NTMCLantasPolri @NEToficial @NTMCLantasPolri
269	@willy_betawie	#Banjir belum ada tanda2 surut, masih pada level 45 - 100 cm kp Makassar Jakarta Timur cuaca masih mendung
270	@willy_betawie	#banjir kp makassar RT 07 RW 07 Jakarta Timur
271	@TMCPoldaMetro	13:01 #Banjir di Jl. Pattimura Jaksel, agar hati2 bila sdg melintas. @zie_aja
272	@bangzul_1988	Jln Fatmawati Cipete Jaksel, pas gg sawo antara ITC lumpuh krna #banjir @petajkta
273	@mizzly_maniez	Masjid albashor kampung dukuh #banjir pdhl dataran tinggi krn got tdk menampung tolong perbaiki sdh msk dlm rmh
274	@tifanytifa	#banjir di sebagian besar daerah kelurahan cipinang melayu (dkt univ. borobudur) sudah 1.5 meter di luar rumah. @petajkt
275	@TMCPoldaMetro	15:44 #Banjir 30 cm di Jl. Taman Mini Jaktim & berimbas lalin tersendat
276	@TMCPoldaMetro	16:07 #Banjir 40 cm di wilayah RW. 01 Cililitan Besar Jakarta Timur.

277	@TMCPoldaMetro	17.25 #Banjir di taman kemang raya Jaksel 40Cm, silahkan cari jalur alternatif
278	@TMCPoldaMetro	19.57 #Banjir 40 cm di Jl. Swadarma Cipulir
279	@BPBDJakarta	21.00 Info #Banjir di RT 06 RW 13 Kel. Cipinang Muara terdampak 187 KK, 288 Jiwa, JakTim, Akibat dari hujan lebat dan luapan Kali Sunter
280	@BPBDJakarta	21.10 Info #Banjir di Kel. Petogogan RW.01, RW.02, RW.03, ketinggian air 10 s.d 60 cm, Akibat dari Hujan dan luapan Kali Krukut, JakSel
281	@BPBDJakarta	21.43: Banjir akibat saluran air tersendat di Rt08/02 kel. Kedaung Kaliangke, Cengkareng, Jakbar
282	@M_Vebby027	Banjir 20-30 cm di area Halim PK. Jaktim @BPBDJakarta
283	@87dadodut	Banjir 50-70 cm di Kel. Cipinang Melayu RW 04 pukul 20.30 tadi
284	@maiya_ranoem	Banjir daerah bambu apus jakarta timur
285	@jumadi_40932	Halaman kantor Kelurahan Kamal Muara, Kec Penjaringan, Jakut terendam banjir 20cm.
286	@theponirin	Jatijati waringin arah gamprit banjir, harap berhati-hati...!!!
287	@TulusYanti	Jln. Raya Kemang, Jaksel banjir. Ketinggian air sampe setengah meter.
288	@shindysaras	Kalau nggak urgent males banget ke daerah sini #banjir (at http://Jl.Bank Raya Kemang)
289	@erwinwinardi55	Kalibata timur samping Apartemen Nifarro, Jaksel banjir
290	@shaadl	Pak @basuki_btp di cililitan besar RW 1 banjir hingga betis orang dewasa, pdhl kami 30 th tdk pernah mgalami hal ini. #banjir #banjirjakarta
291	@AnwarSa9914	@petajkt jln.cakung-cacing #banjir.Ketinggian air kurang lebih 50cm.
292	@luck_7379	#Banjir 30cm di Jl Pangeran Jayakarta #Jkt
293	@si_PutriRani	#Banjir sktr 30cm dari arah lampu merah Kalimalang hingga Cawang UKI, sebrang HIVE, Jaktim.
294	@BPBDJakarta	#Banjir ; Kondisi pompa air di Petogogan, dpn SMA Tarakanita Pulo Raya, Jaksel.
295	@achietmokoginta	#Banjir di #TolJORR TB Simatupang KM 25, arah keluar Pasar Minggu agak tersendat.
296	@sulistiyoagus15	#Banjir di depan Wisma Mulia, Jl. Jend. Gatot Subroto, #Jkt.
297	@mfcf_semy	#Banjir di dpn Psr Buncit, Jl. Bangka Utara IX tdk bisa di lewati kendaraan. via
298	@Kiki95046365	#Banjir di Jl Pulo Raya, belakang Kantor Walikota Jakarta Selatan.
299	@TMCPoldaMetro	#Banjir di Jl. TB Simatupang dpn Gd Nestle arah Ragunan tidak bisa dilintasi kendaraan.
300	@reporterjail	#Banjir di Rt12/03, Kemang Timur 11, Bangka, Jaksel.
301	@faisalanzavi	#Banjir sebatas lutut orang dewasa di Kel. Kuningan Barat, Jaksel.
302	@MilassugA	#Banjir setinggi 20 cm di Jl. H Ali, Kel. Tengah Kramat Jati, Jaktim. @BPBDJakarta
303	@MasAcuy	#Banjir Tutup Jalan TB Simatupang Arah Ragunan
304	@ahmadbudiarto	@petajkt banjir di daerah Pete VI Radio Dalam. Masih sebatas mata kaki belum sampai ke mata hati
305	@RadioElshinta	14.32: #Banjir setinggi 70 Cm di Jl. Falatehan, Mabes Polri, Blok M, Jaksel, ada beberapa mobil yg terendam banjir.
306	@RadioElshinta	17.09: #Banjir menggenangi Komplek Pejabatn indah 1, Jl. H. Samali Ujung sejak hujan deras.
307	@nahrudin13	Banjir dikalibata Pulo...

308	@fraswita	Genangan air 40-50cm depan Ps impres cipete..mobil sedan sebaiknya tdk melintas @TMCPoldaMetro #banjir
309	@WaAnjas	Hujan Deras, Jalan Kemang Raya Terendam #Banjir
310	@InfodariAnda_ID	Kawasan Pejaten, Jaksel direndam banjir
311	@WaAnjas	Kondisi #banjir di Jl. Kemang Raya depan Kem Chicks, #Jkt.
312	@PREMIEREMAGZ	Kondisi saat Ini di daerah Kampung Dukuh, belakang Gandaria City, Jaksel #banjir.
313	@missnurula	#Banjir di Jl. Pondok Bambu Raya Jaktim, lalin ramai lancar.
314	@PKS_Jaktim	#Banjir di Bidaracina , Jaktim
315	@SonoraFM92	Pada Pkl 2.30 Dini hari tadi #Banjir 10 s.d 90 cm di Kel. Pejaten Timur, akibat kenaikan Bendung Katulampa.
316	@Hotel_Singapura	20:00 Imbas #Banjir 40 cm di ruas jalan Raya KJalimalang, lalu lintas di exit tol Jatiwaringin padat merayap.
317	@TMCPoldaMetro	#Banjir 30 cm di RT 08/06 Pekayon, Pasar Rebo, Jakarta Timur.
318	@PasangMata	#Banjir Banjir di Jl Pengangsaan Dua Arah Kelapa Gading
319	@PasangMata	#Banjir Usai Hujan, Jalan Depan RS Mata Aini Setiabudi Tergenang
320	@soulmate_gown	15:14 #Banjir 30-40 cm di lajur kanan KM 3 Tol Kebon Jeruk (arah ke Tomang) & berimbas lalin tersendat.
321	@RusliIshak1	#Banjir di depan Roxy Mas #Jkt
322	@maison110571	#Banjir di Jl D.I Panjaitan, Cawang, Jaktim. @BPBDJakarta
323	@supendifendi51	#Banjir di Jl. Letjen Panjaitan, Jaktim, yang arah Pedati Prumpung tdk bisa dilewati.
324	@steveJFC	#Banjir selepas turunan FO Roxy Mas, mengakibatkan macet panjang, mhn ada petugas yang mengatur.
325	@Evamaria__	Banjir!! Belakang bidakara atah pancoran #banjir #hujan @ Jln. Rasamala Raya Menteng Dalam
326	@SonoraFM92	11.42 #Banjir dikarenakan air laut yang pasang/Rob disekitar Kamal Kapuk Jakarta RT @soulmate_gown
327	@PKS_Jaktim	#Banjir di Kampung Arus Cawang , Kramat Jati Jaktim
328	@BPBDJakarta	Pkl 10.00 #Banjir akibat luapan sungai ciliwung di Kel. Kebon baru, Jakarta Selatan. Kondisi saat ini sudah surut.
329	@BPBDJakarta	Pkl 10.10 #Banjir akibat luapan sungai ciliwung di Kel. Bidara Cina, Jakarta Timur sudah mulai surut. Saat ini ketinggian 10 s.d 20 cm.
330	@BPBDJakarta	Pkl 11.30 #Banjir akibat luapan sungai ciliwung di Kel. Pejaten Timur, Jakarta Selatan sudah surut. Saat ini petugas melakukan pembersihan.
331	@BPBDJakarta	Pkl 11.40 #Banjir akibat luapan sungai ciliwung di Kel. Kp Melayu, Jakarta Timur sudah mulai surut. Saat ini ketinggian 10 s.d 50 cm.
332	@BPBDJakarta	Pkl 11.55 #Banjir akibat luapan sungai ciliwung di Rw. 07 Kel. Kp Melayu, Jakarta timur. Kondisi saat ini sudah surut.
333	@cepyamstronk	Ibukota ku tercinta terendam banjir setinggi tumit orang dewasa. @ Jl.Tanah Tinggi Senen Jakarta... https://www.instagram.com/p/BMolh7kg6ns/
334	@galgitzizu	#Banjir 35 Cm di sekitar Taman Dharmawangsa, pengendara diimbau mencari alternatif lain
335	@adn_on	#Banjir di kawasan Pondok Karya dan Bangka.
336	@TMCPoldaMetro	15.41 #banjir 40Cm Jl sultan iskandar muda dpn Gandaria city, lalin trpantau padat

337	@TMCPoldaMetro	15.46 #Banjir 30Cm di Jalan Hang Lekiu 1, lalin smntr dialihkan
338	@TMCPoldaMetro	15.55 #banjir 40Cm di jl kalibata timur IV Jaksel dpn kantor mjlh Gatra
339	@TMCPoldaMetro	15.56 #banjir 40Cm depan Gandaria city, lalin padat, hati2 saat melintas
340	@TMCPoldaMetro	15:00 #Banjir 60 cm di sekitar traffic light Garuda, TMII, Jakarta Timur. Pengguna jalan diimbau mencari alternatif lain.
341	@TMCPoldaMetro	15:30 #Banjir di ruas jalan Hanglekir, Kebayoran Baru, Jakarta Selatan.
342	@TMCPoldaMetro	15:35 #Banjir 60cm di sktr TL Garuda TMII Jaktim, bg kendaraan sedan agar tdk melintas.
343	@amirgomir	15:46 #Banjir di depan Apartemen The Mansion Jl. Kemang Raya Jaksel & berimbas lalin tersendat.
344	@dhenybrutality	15:55 #Banjir di Gandaria City Arteri Pondok Indah Jaksel & berimbas lalin tersendat
345	@TMCPoldaMetro	16.06 #banjir 50Cm dpn gandaria city, lalin arteri pondok indah trpantau padat.
346	@TMCPoldaMetro	16:04 #Banjir di depan Wika Jl. DI. Panjaitan Jaktim & berimbas lalin tersendat.
347	@mariskova	16:29 Imbas #banjir di Jl. DI Panjaitan, MT. Haryono arah ke Cawang lalin padat.
348	@lusi_prihadi	16:31 Imbas #banjir di Gandaria City, Pondok Indah arah ke Permata Hijau & sebaliknya lalin padat.
349	@thyan666	16:33 Imbas #banjir, Tendea arah ke Pancoran lalin padat
350	@adamWH68	17:43 #Banjir di Gandaria City Jl. Sultan Iskandar Muda Jaksel, bagi kendaraan sejenis sedan agar tidak melintas.
351	@nala_jati	Genangan #banjir setinggi 60 cm di RT01/RW03, Petogogan, KebBar, JakSel. Mohon asistensinya segera @TataAirJakarta @KotaJaksel @Humas_DKI
352	@infonitascom	Hujan Deras Kemang Raya Banjir 80 cm #jakarta #banjir
353	@arespendil	Imbas #banjir di Apartemen The Mansion, Jl. Wijaya 1 arah Prapanca-Antasari lalin padat.
354	@RestiArdyati	Jl Hang Lekir Keb. Baru Jaksel #Banjir kira2 60cc, tdk bisa dilewati kendaraan.
355	@BPBDJakarta	Pkl 16.55 Penanganan #banjir di Jl. Raya Teber (Depan Stasiun Tebet), Jakarta Selatan.
356	@BPBDJakarta	Pkl 17.05 Penanganan #banjir Jl. Tanjung Duren (Depan Pasar Kopro), Jakarta Barat.
357	@BPBDJakarta	Pkl 17.10 Penanganan #banjir di Jl. Perjuangan Raya, Kebon Jeruk, Jakarta Barat.
358	@BPBDJakarta	Pkl 17.19 #banjir di sekitar TMII sudah surut, Jalan sudah dapat dilalui kendaraan.
359	@ridwan_ai	#banjir di Jalan Ciledug Raya setelah Pasar Cipulir arah Ciledug. Pertigaan Jl. Swadarma. Membuat motor mogok masuk air.
360	@nurchamidch	@petajkt #banjir rawa buaya, tinggi 1 meter,
361	@Haryoristamaji	07:02 #Banjir di dpn Yadika Joglo sdh surut & bisa dilintasi semua jenis kendaraan.
362	@BPBDJakarta	Info #Banjir dampak dari Kali Pesanggrahan di Rw.05, Rt.10, Kel. Ulujami, Kec. Pesanggraha, Jak-Sel. air sudah mulai surut.
363	@BPBDJakarta	Info #Banjir di RW.01/07 Kel. Kembangan Utara dampak dari kenaikan Kali Angke Hulu.
364	@jumadi_40932	Ruas jalan Panjang, Green Garden, Jakbar terendam banjir setinggi 30cm. #NewsAndTalk

365	@TMCPoldaMetro	#Banjir 20cm sepanjang 40m di dpn Kampus Triksakti, Jakarta, hati2 melintas.
366	@NitaOftianti	#Banjir di RT 012/02 Kedoya,Pesing.
367	@NengHusni	#Banjir lagi setinggi 50cm di Green Garden RT04/RW08 Kedoya Utara, Jakbar.
368	@kokgabobo	@lewatmana arah pik dr pluit sekolah permai banjir hampir sedengkul hati2
369	@samnauli	@petajkt @BPBDJakarta Green Garden banjir 30cm di dlm perumahan #banjir
370	@ramlanirwanto	@RadioElshinta , banjir di PLTU Muara Karang pertigaan masjid Raudhatul Jannah. Kedua arah macet
371	@MartdeElsanico	@RadioElshinta jl pejuangan depan rs siloam kebon jeruk kedua arah macet total imbas banjir green garden
372	@mifoodjournal	@TMCPoldaMetro jalan panjang dekat pintu tol kebon jeruk terendam air 1/2 ban #banjir #jakarta
373	@NTMCLantasPolri	15.19 wib Banjir di Taman Ratu Kemuning setinggi 30-40 cm, cari jalur alternatif lainnya.
374	@NTMCLantasPolri	15.55 wib Banjir depan Bakmi Siantar Muara Karang, Jakut, Pengendara harap berhati2 melintas di jalan ini.
375	@NTMCLantasPolri	16.22 wib Banjir 20-30 cm di dpn RS. Siloam Kebon Jeruk Jakbar, hati2 bila sedang melintas kawasan ini.
376	@TMCPoldaMetro	16:05 Banjir sepanjang 40 meter di Jl.S.Parman dpn kampus Trisakti dan Kampus Untar arah ke Tomang setinggi 20 Cm, masih bisa dilintasi R2
377	@TMCPoldaMetro	16:08 Banjir menggenangi wilayah Pejagalan, Kapuk Jakarta Barat, diperkirakan ketinggian mencapai 30-40 cm @syaifulrizaldii
378	@TMCPoldaMetro	17:01 #Banjir 20-30 cm di dpn Greenville Jl. Panjang Jakbar & berimbas lalin tersendat. @FaySonora
379	@vaniyeay	20.30 #Banjir 30 cm di Taman Ratu Kedoya Jakbar, hati2 bila melintas
380	@Arif_Rizqi78	Banjir di jalan patra raya duri kepa 40-60 cm,masih turun hujan sedang.
381	@hendtanoto	Banjir di Kb. Jeruk Siloam dan Gerbang Tol Kb. Jeruk.
382	@ramlanirwanto	Banjir di PLTU muara karang, pertigaan masjid Raudhatul Jannah..
383	@Vdock7	Banjir di Taman Ratu Kemuning setinggi 30-40cm. Hati-hati melintas bnyk anak2 bermain air
384	@ellys_khouw	Jalan mangga 6, banjir ketinggian 20-30cm. Hati" ya. @Banjir @TMCPoldaMetro @LMJakarta
385	@HusenRR	Jl Kopo 2arah (dkt Panjunan) banjir...
386	@willy_hgg	Kapuk gg sinar dalam banjir disebabkan banyak deker yg berlapis2 menghambat jln air ke kali angke tlg phk terkait tl @BeritaJakarta
387	@infonitascom	Hujan Deras Sepanjang Malam Pasar Tanah Merah Banjir
388	@infonitascom	Sistem Drainase Tak Optimal, Jalan Muara Karang PLTU Pluit Banjir
389	@BPBDJakarta	#Banjir di jln Kamboja Gn Balong Rt.08/08 Lbk BIs, 25 KK mengungsi di Mushola Al Mabrul.
390	@BPBDJakarta	Info #Banjir di Jl. H. Ipin RW. 01, RT 011 dan RW. 010, Kel. Pondok Labu, Kec. Cilandak, Jak-Sel, "SAAT INI BANJIR SUDAH SURUT"
391	@BPBDJakarta	Info #Banjir Jl. H. Kamang bawah, Pondok Labu, ketinggian air 50 cm, Jakarta Selatan.
392	@BPBDJakarta	Kondisi pengungsian Korban #Banjir di Mushola Al - Mabrul RT.08, RW.08 Lebak bulus Jaksel, Warga akan kembali kerumah setelah surut

393	@BPBDJakarta	Update gambar #Banjir di Jl. H. Ipin depan Madrasah Tarbiyah Bunaiya, RW. 01, Kel. Pondok Labu, Jak-Sel
394	@RadioElshinta	17.11: Banjir di jl Pulo Mawar 1, Kemandoran Kel Grogol Utara, Kec.Kebayoran Lama,Jaksel. (P. Atuman) #Jkt @PolresJaksel
395	@benalirastafari	Habis hujan terbit pelangi 🌈☺️ #allhamdulillah #banjir (at Kampung Pegangsaan 1) [pic] —
396	@SunnyGreise	@petajkt #banjir di Jalan perintis kemerdekaan, kayu mas, kayu putih sudah mencapai 30 cm. Kali sudah meluap.
397	@TMCPoldaMetro	19:09 #Banjir di dpn Isuzu Jl. Perintis Kemerdekaan Pulo Gadung & berimbas lalin tersendat. @tri3su
398	@TMCPoldaMetro	18.35 #Banjir didepan PTC Pulo Gadung, lalin padat. Hati2 bila melintas (foto : @asnoex)
399	@TMCPoldaMetro	18.30 #Banjir didalam kawasan industri Pulo Gadung (foto : @asnoex)
400	@RadioElshinta	17.12: RT @jaminsocool : Banjir di terowongan TolJorr W2 Meruya, Jln Meruya Selatan, dr Univ Mercu ke Kembang Kerep

Lampiran 2. Stopwords_id.text

ada adalah adanya adapun agak agaknya agar ah aja akan akankah akhir akhiri akhirnya aku akulah amat amatlah anda andalah antar antara antaranya apa apaan apabila apakah apalagi apatah artinya asal asalkan atas atau ataukah ataupun awal awalnya bagai bagaikan bagaimana bagaimanakah bagaimanapun bagi bagian bahkan bahwa bahwasanya baik bakal bakalan balik banyak bapak baru bawah beberapa begini beginian beginikah beginilah begitu begitukah begitulah begitupun bekerja belakang belakangan belum belumlah benar benarkah benarlah berada berakhir berakhirlah berakhirnya berapa berapakah berapalah berapapun berarti berawal berbagai berdatangan beri berikan berikut berikutnya berjumlah berkali kali berkata berkehendak berkeinginan berkenaan berlainan berlalu berlangsung berlebihan bermacam bermacam macam bermaksud bermula bersama bersama sama bersiap bersiap siap bertanya bertanya tanya berturut berturut turut berturut berujar berupa besar betul betulkah biasa biasanya bila bilakah bisa bisakah boleh bolehkah bolehlah buat bukan bukannya bukanlah bukannya bulan bung cara caranya cukup cukupkah cukuplah cuma dah dahulu dalam dan dapat dari daripada datang deh dekat demi demikian demikianlah dengan depan di dia diakhiri diakhirinya dialah diantara diantaranya diberi diberikan diberikannya dibuat dibuatnya didapat didatangkan digunakan diibaratkan diibaratkannya diingat diingatkan diinginkan dijawab dijelaskan dijelaskannya dikarenakan dikatakan dikatakannya dikerjakan diketahui diketahuinya dikira dilakukan dilalui dilihat dimaksud dimaksudkan dimaksudkannya dimaksudnya diminta dimintai dimisalkan dimulai dimulailah dimulainya dimungkinkan dini dipastikan diperbuat diperbuatnya dipergunakan diperkirakan diperlihatkan diperlukan diperlukannya dipersoalkan dipertanyakan dipunyai diri dirinya disampaikan disebut disebutkan disebutkannya disini disinilah ditambahkan ditandaskan ditanya ditanyai ditanyakan ditegaskan ditunjukan ditunjuk ditunjuki ditunjukkan ditunjukkannya ditunjuknya dituturkan dituturkannya diucapkan diucapkannya diungkapkan dong dua dulu eh empat enggak enggaknya entah entahlah gini guna gunakan hal hampir hanya hanyalah hari harus haruslah harusnya hendak hendaklah hendaknya hingga ia ialah ibarat ibaratkan ibaratnya ibu ikut ingat ingat ingin inginkah inginkan ini inikah inilah itu itukah itulah iya jadi jadilah jadinya jangan jangankan janganlah jauh jawab jawaban jawabnya jelas jelaskan jelaslah jelasnya jika jikalau juga jumlah jumlahnya justru kala kalau kalaulah kalaupun kalian kalo kami kamilah kamu kamulah kan kapan kapankah kapanpun karena karenanya kasus kata katakan katakanlah katanya kayak ke keadaan kebetulan kecil kedua keduanya keinginan kelamaan kelihatan kelihatannya kelima keluar kembali kemudian kemungkinan kemungkinannya kenapa kepada kepadanya kesampaian keseluruhan keseluruhannya keterlaluhan ketika khususnya kini kinilah kira kira kiranya kita kitalah kok ku kurang lagi lagian lah lain lainnya lalu lama lamanya lanjut lanjutnya lebih lewat lima luar macam mah maka makanya makin malah malahan mampu mampukah mana manakala manalagi masa masalah masalahnya masih masihkah masing masing masing mau maupun melainkan melakukan melalui melihat melihatnya memang memastikan memberi memberikan membuat memerlukan

memihak meminta memintakan memisalkan memperbuat mempergunakan
memperkirakan memperlihatkan mempersiapkan mempersoalkan mempertanyakan
mempunyai memulai memungkinkan menaiki menambahkan menandakan
menanti menanti nanti menantikan menanya menanyai menanyakan mendapat
mendapatkan mendatang mendatangi mendatangkan menegaskan mengakhiri
mengapa mengatakan mengatakannya mengenai mengerjakan mengetahui
menggunakan menghendaki mengibaratkan mengibaratkannya mengingat
mengingatn menginginkan mengira mengucapkan mengucapkannya
mengungkapkan menjadi menjawab menjelaskan menuju menunjuk menunjuki
menunjukkan menunjuknya menurut menuturkan menyampaikan menyangkut
menyatakan menyebutkan menyeluruh menyiapkan merasa mereka merekalah
merupakan meski meskipun meyakini meyakinkan minta mirip misal misalkan
misalnya mula mulai mulailah mulanya mungkin mungkinkah nah naik namun
nanti nantinya nggak ntar nyaris nyatanya oleh olehnya pada padahal padanya pak
paling panjang pantas para pasti pastilah penting pentingnya per percuma perlu
perlukah perlunya pernah persoalan pertama pertama tama pertanyaan pertanyakan
pihak pihaknya pukul pula pun punya rasa rasanya rata rupanya saat saatnya saja
sajalah saling sama sama sama sambil sampai sampai sampai sampaikan sana
sangat sangatlah satu saya sayalah se sebab sebabnya sebagai sebagaimana
sebagainya sebagian sebaik sebaik baiknya sebaiknya sebaliknya sebanyak sebegini
sebegitu sebelum sebelumnya sebenarnya seberapa sebesar sebetulnya sebisanya
sebuah sebut sebutlah sebutnya secara secukupnya sedang sedangkan sedemikian
sedikit sedikitnya seenaknya segala segalanya segera seharusnya sehingga seingat
sejak sejauh sejenak sejumlah sekadar sekadarnya sekali sekali kali sekalian
sekaligus sekalipun sekarang sekarang sekecil seketika sekiranya sekitar sekitarnya
sekurang kurangnya sekurangnya sela selain selaku selalu selama selama lamanya
selamanya selanjutnya seluruh seluruhnya semacam semakin semampu
semampunya semasa semasih semata semata mata semaunya sementara semisal
semisalnya sempat semua semuanya semula sendiri sendirian sendirinya seolah
seolah olah seorang sepanjang sepantasnya sepantasnyalah seperlunya seperti
sepertinya sepihak sering seringnya serta serupa sesaat sesama sesampai sesegera
sesekali seseorang sesuatu sesuatunya sesudah sesudahnya setelah setempat
setengah seterusnya setiap setiba setibanya setidak tidaknya setidaknya setinggi
seusai sewaktu si siap siapa siapakah siapapun sih sini sinilah soal soalnya suatu
sudah sudahkah sudahlah supaya tadi tadinya tahu tahun tak tambah tambah
tampak tampaknya tandas tandasnya tanpa tanya tanyakan tanyanya tapi tegas
tegasnya telah tempat tengah tentang tentu tentulah tentunya tepat terakhir terasa
terbanyak terdahulu terdapat terdiri terhadap terhadapnya teringat teringat ingat
terjadi terjadilah terjadinya terkira terlalu terlebih terlihat termasuk ternyata
tersampaikan tersebut tersebutlah tertentu tertuju terus terutama tetap tetapi tiap tiba
tiba tiba tidak tidakkah tidaklah tiga tinggi toh tuh tunjuk turut tutur tuturnya ucap
ucapnya udah ujar ujarnya umum umumnya ungkap ungkapnya untuk usah usai
waduh wah wahai waktu waktunya walau walaupun wong ya yaitu yakin yakni
yang yg

Lampiran 3. Source Code Program

```

1  __author__ = 'prasetic'
2
3  from sklearn import svm, metrics
4  from sklearn.model_selection import cross_val_score,
   cross_val_predict
5  from sklearn.feature_extraction.text import CountVectorizer,
   TfidfTransformer
6  import re
7
8  data = []
9  bagOfWords = []
10 stopWords = []
11 stopWordList = []
12 Y = []
13
14 # load data
15 for i in range(1, 401):
16     d = open("Data/%d.txt" % i, "r+")
17     string = d.read()
18     data.append(string)
19     d.close()
20 print("Data Awal \n",data)
21
22 # tokenizing
23 bagOfWords = data
24 for j in range(0, 400):
25     kalimat = data[j]
26     bagOfWords[j] = kalimat.split()
27 print("Bag of Words\n",bagOfWords)
28
29 # load stopword
30 d = open("Data/stopwords_id.txt", "r+")
31 kata = d.read()
32 kata = re.sub("[^a-z]", " ", kata)
33 stopWords.append(kata)
34 d.close()
35 print("stopword \n",stopWords)
36
37 stopWordList = stopWords
38 for k in range(0, 1):
39     kalimat = stopWords[k]
40     stopWordList[k] = kalimat.split()
41 print("Stop Word List\n",stopWordList)
42
43 # filtering
44 for p in range(0, 400):
45     for q in range(0, len(bagOfWords[p])):
46         for r in range(0, 799):
47             if (bagOfWords[p][q] == stopWordList[0][r]):
48                 bagOfWords[p][q] = ''

```

```
49 for l in range(0, len(bagOfWords)):  
50     bagOfWords[l] = filter(None, bagOfWords[l])  
51     data[l] = ' '.join(bagOfWords[l])  
52 print("Data Bersih\n",data)  
53  
54 # tfidf dan VSM  
55 count_vect = CountVectorizer()  
56 X_count = count_vect.fit_transform()  
57 tfidf = TfidfTransformer()  
58 X_tfidf = tfidf.fit_transform(X_count)  
59  
60 print("Term Frequency\n",X_count)  
61 print("TFIDF\n",X_tfidf)  
62  
63 # pemberian label  
64 for z in range(len(data)):  
65     if z < 200:  
66         Y.append(0)  
67     elif z < 400:  
68         Y.append(1)  
69  
70 # klasifikasi  
71 clf = svm.LinearSVC().fit(X_tfidf, Y)  
72 cv = 10  
73  
74 # Predict  
75 predicted = cross_val_predict(clf, X_tfidf, Y, cv=cv)  
76 print("Hasil prediksi \n", predicted)  
77  
78 # Score  
79 print("Confusion Matrix \n", metrics.confusion_matrix(Y,  
    predicted))  
80 label = ['False', 'True']  
81 print(metrics.classification_report(Y, predicted,  
    target_names=label))  
82  
83 scores = cross_val_score(clf, X_tfidf, Y, cv=cv)  
84 print("Score", scores)  
85 print("Akurasi : %0.2f " % scores.mean())
```

Lampiran 4. Modul *Scikit Learn* yang Dipakai dalam *Source Code* Program

```

1 class CountVectorizer(BaseEstimator, VectorizerMixin):
2     def __init__(self, input='content', encoding='utf-8',
3                 decode_error='strict', strip_accents=None,
4                 lowercase=True, preprocessor=None, tokenizer
5                 =None,
6                 stop_words=None, token_pattern=r"(?u)\b\w\w
7                 +\b",
8                 ngram_range=(1, 1), analyzer='word',
9                 max_df=1.0, min_df=1, max_features=None,
10                vocabulary=None, binary=False, dtype=np.
11                int64):
12        self.input = input
13        self.encoding = encoding
14        self.decode_error = decode_error
15        self.strip_accents = strip_accents
16        self.preprocessor = preprocessor
17        self.tokenizer = tokenizer
18        self.analyzer = analyzer
19        self.lowercase = lowercase
20        self.token_pattern = token_pattern
21        self.stop_words = stop_words
22        self.max_df = max_df
23        self.min_df = min_df
24        if max_df < 0 or min_df < 0:
25            raise ValueError("negative value for max_df or
26            min_df")
27        self.max_features = max_features
28        if max_features is not None:
29            if (not isinstance(max_features, numbers.Integral
30            ) or
31                max_features <= 0):
32                raise ValueError(
33                    "max_features=%r, neither a positive
34                    integer nor None"
35                    % max_features)
36        self.ngram_range = ngram_range
37        self.vocabulary = vocabulary
38        self.binary = binary
39        self.dtype = dtype
40
41    def _sort_features(self, X, vocabulary):
42        sorted_features = sorted(six.iteritems(vocabulary))
43        map_index = np.empty(len(sorted_features), dtype=np.
44        int32)
45        for new_val, (term, old_val) in enumerate(
46        sorted_features):
47            vocabulary[term] = new_val
48            map_index[old_val] = new_val
49
50        X.indices = map_index.take(X.indices, mode='clip')

```

```

43     return X
44
45     def _limit_features(self, X, vocabulary, high=None, low=
None, limit=None):
46         if high is None and low is None and limit is None:
47             return X, set()
48         # Calculate a mask based on document frequencies
49         dfs = _document_frequency(X)
50         tfs = np.asarray(X.sum(axis=0)).ravel()
51         mask = np.ones(len(dfs), dtype=bool)
52         if high is not None:
53             mask &= dfs <= high
54         if low is not None:
55             mask &= dfs >= low
56         if limit is not None and mask.sum() > limit:
57             mask_inds = (-tfs[mask]).argsort()[:limit]
58             new_mask = np.zeros(len(dfs), dtype=bool)
59             new_mask[np.where(mask)[0][mask_inds]] = True
60             mask = new_mask
61
62         new_indices = np.cumsum(mask) - 1 # maps old indices
to new
63         removed_terms = set()
64         for term, old_index in list(six.iteritems(vocabulary)
):
65             if mask[old_index]:
66                 vocabulary[term] = new_indices[old_index]
67             else:
68                 del vocabulary[term]
69                 removed_terms.add(term)
70         kept_indices = np.where(mask)[0]
71         if len(kept_indices) == 0:
72             raise ValueError("After pruning, no terms remain
. Try a lower min_df or a higher max_df.")
73         return X[:, kept_indices], removed_terms
74
75     def _count_vocab(self, raw_documents, fixed_vocab):
76         if fixed_vocab:
77             vocabulary = self.vocabulary_
78         else:
79             vocabulary = defaultdict()
80             vocabulary.default_factory = vocabulary.__len__
81
82         analyze = self.build_analyzer()
83         j_indices = []
84         indptr = _make_int_array()
85         values = _make_int_array()
86         indptr.append(0)
87         for doc in raw_documents:
88             feature_counter = {}

```

```

89         for feature in analyze(doc):
90             try:
91                 feature_idx = vocabulary[feature]
92                 if feature_idx not in feature_counter:
93                     feature_counter[feature_idx] = 1
94                 else:
95                     feature_counter[feature_idx] += 1
96             except KeyError:
97                 # Ignore out-of-vocabulary items for
fixed_vocab=True
98                 continue
99
100             j_indices.extend(feature_counter.keys())
101             values.extend(feature_counter.values())
102             indptr.append(len(j_indices))
103
104         if not fixed_vocab:
105             # disable defaultdict behaviour
106             vocabulary = dict(vocabulary)
107             if not vocabulary:
108                 raise ValueError("empty vocabulary; perhaps
the documents only"
109                                 " contain stop words")
110
111             j_indices = np.asarray(j_indices, dtype=np.intc)
112             indptr = np.frombuffer(indptr, dtype=np.intc)
113             values = frombuffer_empty(values, dtype=np.intc)
114
115             X = sp.csr_matrix((values, j_indices, indptr),
116                             shape=(len(indptr) - 1, len(
vocabulary)),
117                                 dtype=self.dtype)
118             X.sort_indices()
119             return vocabulary, X
120
121     def fit(self, raw_documents, y=None):
122         self.fit_transform(raw_documents)
123         return self
124
125     def fit_transform(self, raw_documents, y=None):
126         if isinstance(raw_documents, six.string_types):
127             raise ValueError(
128                 "Iterable over raw text documents expected
129                 , "
130                 "string object received.")
131
132         self._validate_vocabulary()
133         max_df = self.max_df
134         min_df = self.min_df
max_features = self.max_features

```

```

135
136     vocabulary, X = self._count_vocab(raw_documents,
self.fixed_vocabulary_)
137
138     if self.binary:
139         X.data.fill(1)
140
141     if not self.fixed_vocabulary_:
142         X = self._sort_features(X, vocabulary)
143
144         n_doc = X.shape[0]
145         max_doc_count = (max_df
146             if isinstance(max_df, numbers.
Integral)
147                 else max_df * n_doc)
148         min_doc_count = (min_df
149             if isinstance(min_df, numbers.
Integral)
150                 else min_df * n_doc)
151         if max_doc_count < min_doc_count:
152             raise ValueError(
153                 "max_df corresponds to < documents than
min_df")
154         X, self.stop_words_ = self._limit_features(X,
vocabulary, max_doc_count, min_doc_count, max_features)
155
156         self.vocabulary_ = vocabulary
157
158     return X
159
160     def transform(self, raw_documents):
161         if isinstance(raw_documents, six.string_types):
162             raise ValueError(
163                 "Iterable over raw text documents expected
, "
164                 "string object received.")
165
166         if not hasattr(self, 'vocabulary_'):
167             self._validate_vocabulary()
168
169         self._check_vocabulary()
170
171         # use the same matrix-building strategy as
fit_transform
172         _, X = self._count_vocab(raw_documents, fixed_vocab=
True)
173
174         if self.binary:
175             X.data.fill(1)
176         return X

```

```

177     def inverse_transform(self, X):
178         self._check_vocabulary()
179
180         if sp.issparse(X):
181             X = X.tocsr()
182         else:
183             X = np.asmatrix(X)
184         n_samples = X.shape[0]
185
186         terms = np.array(list(self.vocabulary_.keys()))
187         indices = np.array(list(self.vocabulary_.values()))
188         inverse_vocabulary = terms[np.argsort(indices)]
189
190         return [inverse_vocabulary[X[i, :].nonzero()[1]].
191                 ravel()
192                 for i in range(n_samples)]
193
194     def get_feature_names(self):
195         self._check_vocabulary()
196
197         return [t for t, i in sorted(six.iteritems(self.
198             vocabulary_), key=itemgetter(1))]
199
200     class TfidfTransformer(BaseEstimator, TransformerMixin):
201         def __init__(self, norm='l2', use_idf=True, smooth_idf=
202             True,
203                 sublinear_tf=False):
204             self.norm = norm
205             self.use_idf = use_idf
206             self.smooth_idf = smooth_idf
207             self.sublinear_tf = sublinear_tf
208
209         def fit(self, X, y=None):
210             if not sp.issparse(X):
211                 X = sp.csc_matrix(X)
212             if self.use_idf:
213                 n_samples, n_features = X.shape
214                 df = _document_frequency(X)
215
216                 df += int(self.smooth_idf)
217                 n_samples += int(self.smooth_idf)
218
219                 idf = np.log(float(n_samples) / df) + 1.0
220                 self._idf_diag = sp.spdiags(idf, diags=0, m=
221                 n_features,
222                                     n=n_features, format
223                 ='csr')
224
225             return self

```

```

222     def transform(self, X, copy=True):
223         if hasattr(X, 'dtype') and np.issubdtype(X.dtype, np
.float):
224             X = sp.csr_matrix(X, copy=copy)
225         else:
226             X = sp.csr_matrix(X, dtype=np.float64, copy=copy
)
227
228         n_samples, n_features = X.shape
229
230         if self.sublinear_tf:
231             np.log(X.data, X.data)
232             X.data += 1
233
234         if self.use_idf:
235             check_is_fitted(self, '_idf_diag', 'idf vector
is not fitted')
236
237             expected_n_features = self._idf_diag.shape[0]
238             if n_features != expected_n_features:
239                 raise ValueError("Input has n_features=%d
while the model"
240                                 " has been trained with
n_features=%d" % (
241                                     n_features,
expected_n_features))
242                 # *= doesn't work
243                 X = X * self._idf_diag
244
245             if self.norm:
246                 X = normalize(X, norm=self.norm, copy=False)
247
248             return X
249
250     @property
251     def idf_(self):
252         if hasattr(self, "_idf_diag"):
253             return np.ravel(self._idf_diag.sum(axis=0))
254         else:
255             return None
256
257     class LinearSVC(BaseEstimator, LinearClassifierMixin,
258                   _LearntSelectorMixin, SparseCoefMixin):
259
260         def __init__(self, penalty='l2', loss='squared_hinge',
dual=True, tol=1e-4,
261                     C=1.0, multi_class='ovr', fit_intercept=
True,
262                     intercept_scaling=1, class_weight=None,
verbose=0,

```



```

303 pre_dispatch='2*n_jobs',
304         method='predict'):
305
306     X, y, groups = indexable(X, y, groups)
307
308     cv = check_cv(cv, y, classifier=is_classifier(estimator)
309 )
310     cv_iter = list(cv.split(X, y, groups))
311
312     if not callable(getattr(estimator, method)):
313         raise AttributeError('{} not implemented in
314 estimator'
315                             .format(method))
316
317     parallel = Parallel(n_jobs=n_jobs, verbose=verbose,
318 pre_dispatch=pre_dispatch)
319     prediction_blocks = parallel(delayed(_fit_and_predict)(
320 clone(estimator), X, y, train, test, verbose,
321 fit_params, method)
322 for train, test in cv_iter)
323
324     # Concatenate the predictions
325     predictions = [pred_block_i for pred_block_i, _ in
326 prediction_blocks]
327     test_indices = np.concatenate([indices_i
328 for _, indices_i in
329 prediction_blocks])
330
331     if not _check_is_permutation(test_indices, _num_samples(
332 X)):
333         raise ValueError('cross_val_predict only works for
334 partitions')
335
336     inv_test_indices = np.empty(len(test_indices), dtype=int
337 )
338     inv_test_indices[test_indices] = np.arange(len(
339 test_indices))
340
341     if sp.issparse(predictions[0]):
342         predictions = sp.vstack(predictions, format=
343 predictions[0].format)
344     else:
345         predictions = np.concatenate(predictions)
346     return predictions[inv_test_indices]
347
348 def confusion_matrix(y_true, y_pred, labels=None, samp
349 le_weight=None):
350
351     y_type, y_true, y_pred = _check_targets(y_true, y_pred)
352     if y_type not in ("binary", "multiclass"):

```

```

342     raise ValueError("%s is not supported" % y_type)
343
344     if labels is None:
345         labels = unique_labels(y_true, y_pred)
346     else:
347         labels = np.asarray(labels)
348         if np.all([l not in y_true for l in labels]):
349             raise ValueError("At least one label specified
350 must be in y_true")
351
352     if sample_weight is None:
353         sample_weight = np.ones(y_true.shape[0], dtype=np.
354 int)
355     else:
356         sample_weight = np.asarray(sample_weight)
357
358     check_consistent_length(sample_weight, y_true, y_pred)
359
360     n_labels = labels.size
361     label_to_ind = dict((y, x) for x, y in enumerate(labels)
362 )
363     # convert yt, yp into index
364     y_pred = np.array([label_to_ind.get(x, n_labels + 1) for
365 x in y_pred])
366     y_true = np.array([label_to_ind.get(x, n_labels + 1) for
367 x in y_true])
368
369     # intersect y_pred, y_true with labels, eliminate items
370 not in labels
371     ind = np.logical_and(y_pred < n_labels, y_true < n_label
372 s)
373     y_pred = y_pred[ind]
374     y_true = y_true[ind]
375     # also eliminate weights of eliminated items
376     sample_weight = sample_weight[ind]
377
378     CM = coo_matrix((sample_weight, (y_true, y_pred)),
379 shape=(n_labels, n_labels)
380 ).toarray()
381
382     return CM
383
384 def classification_report(y_true, y_pred, labels=None, ta
385 rget_names=None,
386 sample_weight=None, digits=2):
387     if labels is None:
388         labels = unique_labels(y_true, y_pred)
389     else:
390         labels = np.asarray(labels)
391

```

```

384     last_line_heading = 'avg / total'
385
386     if target_names is None:
387         target_names = ['%s' % l for l in labels]
388     name_width = max(len(cn) for cn in target_names)
389     width = max(name_width, len(last_line_heading), digits)
390
391     headers = ["precision", "recall", "f1-score", "support"]
392     fmt = '%% %ds' % width # first column: class name
393     fmt += ' '
394     fmt += ' '.join(['% 9s' for _ in headers])
395     fmt += '\n'
396
397     headers = [""] + headers
398     report = fmt % tuple(headers)
399     report += '\n'
400
401     p, r, f1, s = precision_recall_fscore_support(y_true,
402                                                  y_pred,
403                                                  labels=labels,
404                                                  average=average,
405                                                  sample_weight=sample_weight)
406
407     for i, label in enumerate(labels):
408         values = [target_names[i]]
409         for v in (p[i], r[i], f1[i]):
410             values += ["{0:0.{1}f}".format(v, digits)]
411         values += ["{0}".format(s[i])]
412         report += fmt % tuple(values)
413
414     report += '\n'
415
416     # compute averages
417     values = [last_line_heading]
418     for v in (np.average(p, weights=s),
419             np.average(r, weights=s),
420             np.average(f1, weights=s)):
421         values += ["{0:0.{1}f}".format(v, digits)]
422     values += ["{0}".format(np.sum(s))]
423     report += fmt % tuple(values)
424     return report
425
426 def cross_val_score(estimator, X, y=None, groups=None, scoring=None, cv=None,
427                    n_jobs=1, verbose=0, fit_params=None,
428                    pre_dispatch='2*n_jobs'):

```

```
429     X, y, groups = indexable(X, y, groups)
430
431     cv = check_cv(cv, y, classifier=is_classifier(estimator
432     ))
433     cv_iter = list(cv.split(X, y, groups))
434     scorer = check_scoring(estimator, scoring=scoring)
435     parallel = Parallel(n_jobs=n_jobs, verbose=verbose,
436     pre_dispatch=pre_dispatch)
437     scores = parallel(delayed(_fit_and_score)(clone(estima
438     tor), X, y, scorer,
439     train, test,
440     verbose, None,
441     fit_params)
442     for train, test in cv_iter)
443     return np.array(scores)[: , 0]
```

RIWAYAT HIDUP



Muhamad Prasetio Dwi Cahyo dilahirkan di kota Pandeglang, pada tanggal 7 April 1995 dari pasangan Bapak Agus Dwiyanto dan Ibu Sri Nurahayu yang merupakan anak kedua dari dua bersaudara. Beralamat di Paninggilan Utara RT 01 RW 07 No 24, Ciledug Tangerang, Banten.

Penulis memulai pendidikan di taman kanak-kanan Tunas Harapan pada tahun 2000 dan lulus pada tahun 2001. Kemudian melanjutkan pendidikan ke SD Sukabumi Selatan 01 Pagi dan lulus pada tahun 2007. Lalu penulis melanjutkan ke jenjang SMP di SMPN 206 Jakarta Barat dan lulus pada tahun 2010. Pada tahun 2010 penulis melanjutkan pendidikan di SMK Bina Informatika yang terletak di Bintaro Tangerang Selatan dan lulus tahun 2013. Setelah itu, pada tahun 2013 penulis melanjutkan pendidikan S1 di Universitas Negeri Jakarta pada Program Studi Pendidikan Teknik Informatika dan Komputer dengan peminatan Rekayasa Perangkat Lunak. Penulis melaksanakan Praktik Kerja Lapangan (PKL) di Televisi Republik Indonesia pada tahun 2016. Di bulan Agustus 2016, penulis melakukan Praktik Keterampilan Mengajar (PKM) di SMK Laboratorium Jakarta. Pada akhir perkuliahan, penulis melakukan penelitian skripsi dengan judul “Kinerja Algoritma *Support Vector Machine* dalam Menentukan Informasi Banjir di twitter”. Selama perkuliahan di tahun 2015, penulis bekerja di Lembaga Penelitian Universitas Negeri Jakarta selama setahun.