IMPLEMENTASI METODE SUPPORT VECTOR MACHINE UNTUK MELAKUKAN KLASIFIKASI KEMACETAN LALU LINTAS PADA TWITTER

IMPLEMENTATION SUPPORT VECTOR MACHINE METHOD FOR TRAFFIC JAM CLASSIFICATION ON TWITTER

Elly Susilowati, Mira Kania Sabariah, ST., MT.[2], Alfian Akbar Gozali, ST., MT. [3]

[1,2,3] Jurusan Teknik Informatika, Universitas Telkom
Jl. Telekomunikasi, Dayeuh Kolot Bandung 40257 Indonesia

elly.susi237@gmail.com, mira.ijuan@gmail.com, alfian.gozali@gmail.com

ABSTRAKSI

Twitter merupakan jejaring sosial yang populer saat ini. Beragam informasi dapat diambil dari *Twitter.* Salah satunya adalah *tweet* yang mengabarkan mengenai kondisi kemacetan suatu lalu lintas. Akan tetapi, sumber yang mengabarkan kondisi lalu lintas tersebut tidak hanya satu dan tidak saling terintegrasi. Sehingga informasi yang ada menjadi kurang bermanfaat karena seseorang cenderung malas ketika harus melakukan pencarian data secara manual dari satu sumber ke sumber yang lain.

Tugas akhir ini bertujuan untuk melakukan hasil klasifikasi tweet kondisi jalan pada twitter yang telah dikumpulkan dengan melihat isi dari tweet tersebut. Data diklasifikasikan menjadi 2 kondisi, yaitu macet atau ramai lancar. Metode yang digunakan pada penelitian ini adalah Support Vector Machine (SVM). Metode ini dipilih karena mampu mengklasifikasikan data berdimensi tinggi yang dalam konteks tugas akhir ini adalah data berupa teks. Dari uji skenario yang dilakukan, hasil rata-rata akurasi berada diatas nilai 90%.

Kata kunci : kemacetan, klasifikasi, *Support Vector Machine*

ABSTRACT

Twiter is one of the popular social networks lately. We can get a lot of informations from twitter. One of them including the info about traffic jam from the tweets itself. However, the resources of the informations might not be just one and not fully integrated. So that the informations might be founded unuseful because people tend to be careless to verify the informations manually from each resources.

This research proposed to do the tweet classification by checking on the real condition from the resources on twitter. Data classification will be divided into 2 conditions, one with traffic jam and one without traffic jam. The method that will be used in this research is Support Vector Machine (SVM). This method was chosen because it is believed to be able to do the classification from the high dimention data which is formed as data text. From the testing scenario, the results showed that the percentage of system accuration are mostly above 90%.

Keyword : ***traffic jam, classification, Support Vector Machine***

1 Pendahuluan

1.1 Latar Belakang

Social media merupakan sebuah media online dimana para pengguna yang terdaftar dapat berinteraksi satu sama lain. Salah satu microblogging populer saat ini adalah *Twitter*.

Dari data yang terdapat pada Twitter, dapat dimanfaatkan untuk memperoleh informasi tertentu. User bisa mengungkapkan apapun melalui Twitter, baik itu apa yang dia rasakan, apa yang sedang dialami ataupun kondisi yang terjadi di sekitarnya. Kemacetan merupakan hal yang dihindari oleh masyarakat. Sebisa mungkin, orang akan memilih jalan yang tidak terlalu ramai ketika ingin bepergian dengan mencari informasi terkait kondisi jalan saat ini. Informasi terkait kondisi sebuah jalan sebenarnya bisa dicari melalui media *Twitter* karena terdapat beberapa akun *Twitter* yang diperuntukkan sebagai media informasi lalu lintas. Akan tetapi Karena data di *Twitter* tersebut tidak tergabung menjadi satu, maka ketika dilakukan pencarian secara manual akan menghabiskan banyak waktu dan tidak efisien. Berdasarkan hal tersebut diperlukan adanya integrasi data yang berasal dari berbagai sumber dengan memuat konten yang sama untuk selanjutnya di klasifikasikan dari data yang ada bagaimana kondisi dari sebuah jalan.

Dalam penelitian ini digunakan metode Suport Vector Machine dalam melakukan proses klasifikasi. SVM biasa diterapkan pada kasus *Pattern Recognition* berupa gambar maupun video. Pada penelitian kali ini SVM diujikan untuk mengetahui performansi dalam melakukan klasifikasi data yang berupa teks yang selanjutnya data hasil klasifikasi di *transformasi*-kan ke dalam bentuk visualisasi *Google Map API*.

1.2 Perumusan Masalah

Berdasarkan pada latar belakang di atas, permasalahan yang akan diuraikan dan diteliti adalah:

1. Bagaimana cara menerapkan algoritma *Support Vector Machine* untuk mengklasifikasi data kemacetan dari *Twitter*?
2. Bagaimana performansi algoritma *Support Vector Machine* untuk akurasi dan kecepatan terhadap parameter akurasi

1.3 Tujuan

Mengacu pada masalah-masalah diatas, tujuan Tugas Akhir ini adalah :

1. Mampu mengimplementasikan data kemacetan menggunakan *Support Vector Machine* ke dalam bentuk klasifikasi
2. Melakukan evaluasi hasil performansi dari pengklasifikasian yang dihasilkan menggunakan *Support Vector Machine*

1.4 Batasan Masalah

Adapun batasan-batasan masalah pada Tugas Akhir ini antara lain :

1. Data tweets yang digunakan adalah tweet berbahasa Indonesia
2. Data tweet yang diproses hanya data yang menyebutkan lokasi dan waktu
3. Klasifikasi topik hanya berupa macet atau ramai lancar
4. Data yang diambil hanya berupa teks, tidak termasuk suara, image maupun video.
5. Sistem melakukan pembelajaran secara offline

2 Landasan Teori

2.1 *Data Mining*

Data Mining adalah proses mengekstraksi pola-pola yang menarik (implisit, tidak diketahui sebelumnya, dan berpotensi untuk dapat dimanfaatkan) dari data yang berukuran besar [**3**].

Dalam implementasinya, teknik data mining dibagi menjadi 2 kategori, yaitu [**3**] :

a. Prediktif
Tujuannya adalah untuk memprediksi nilai dari atribut tertentu berdasarkan nilai dari atribut lainnya.

b. Deskriptif
Tujuannya adalah untuk mengenali pola (korelasi,tren,cluster, trajector dan anomali) yang merupakan summary dari relasi-relasi dalam data.

2.2 *Klasifikasi*

Merupakan proses pembangunan suatu model yang mengklasifikasikan suatu objek berdasarkan atribut-atributnya. Kelas label sudah tersedia dari data sebelumnya sehingga terfokus untuk bagaimana mempelajari data yang ada agar klasifikator bias mengklasifikasikan secara otomatis. [**3**]

2.3 *Text Mining*

Text mining merupakan penggalian data yang berupa teks yang didapatkan dari dokumen atau kumpulan kalimat yang memiliki tujuan mencari inti dari konten dan selanjutnya dianalisa untuk didapatkan sebuah informasi.Text mining merupakan area yang menarik dari penelitian ilmu computer yang mengatasi krisis informasi yang berlebihan dengan menggabungkan teknik data mining, *machine learning, natural language processing, information retrieval* dan *knowledge management* [**4**].

2.4 Data Preprocessing

2.4.1 Data Cleaning

Dataset yang telah dikumpulkan dari beberapa artikel website disatukan dalam sebuah file Excel untuk dilakukan perbaikan tata bahasa agar sesuai dengan inputan yang dapat diterima oleh sistem, kemudian diubah menjadi file text yang beberapa atribut yang dianggap penting seperti opini, judul, topik, dan kelas dalam proses klasifikasi dan per*etrieve*an dokumen nantinya.

2.4.2 Tokenisasi

Tokenisasi adalah tugas memisahkan deretan kata di dalam kalimat, paragraf atau halaman menjadi token atau potongan kata tunggal atau *termmed word*. Pada saat bersamaan, tokenisasi juga membuang beberapa karakter tertentu yang dianggap sebagai tanda baca.

2.4.3 *Case Folding*

Case-folding adalah proses penyamaan case dalam sebuah dokumen. Ini dilakukan untuk mempermudah pencarian.

2.4.4 Penghilangan *Stopword*

Stopword didefinisikan sebagai term yang tidak berhubungan (*irrelevant*) dengan subyek utama dari database meskipun kata tersebut sering kali hadir di dalam dokumen. Berikut ini adalah contoh *stopwords* dalam bahasa Indonesia: yang, juga, dari, dia, kami, kamu, aku, saya, ini, itu, atau, dan, tersebut, pada, dengan, adalah, yaitu, ke, tak, tidak, di, pada, jika, maka, ada, pun, lain, saja, hanya, namun, seperti, kemudian, dll.

2.4.5 *Stemming*

Kata-kata yang muncul di dalam dokumen sering mempunyai banyak varian morfologik. Karena itu, setiap kata yang bukan stop-words direduksi ke bentuk *stemmed word* (term) yang cocok. Kata tersebut distem untuk mendapatkan bentuk akarnya dengan menghilangkan awalan atau akhiran. Dengan cara ini, diperoleh kelompok kata yang mempunyai makna serupa tetapi berbeda wujud sintaktis satu dengan lainnya.

Kelompok tersebut dapat direpresentasikan oleh satu kata tertentu. Sebagai contoh, kata menyebutkan, tersebut, disebut dapat dikatakan serupa atau satu kelompok dan dapat diwakili oleh satu kata umum sebut

2.5 PembobotanTF IDF

.Term Frequency Inverse Document Frequency atau TF-IDF merupakan suatu metode yang digunakan dalam melakukan pembobotan terhadap kemunculan kata dalam suatu dokumen. TF menyatakan jumlah kata

yang muncul dalam suatu dokumen. Sedangkan IDF menunjukkan tingkat kepentingan suatu kata yang terdapat dalam kumpulan dokumen. Pada TF –IDF terdapat rumus untuk menghitung bobot (W) masing-masing dokumen terhadap kunci dengan rumus yaitu [**5**] :

(2.1)

–

(2.2)

Keterangan :

= bobot kata terhadap dokumen

= jumlah kemunculan kata dalam

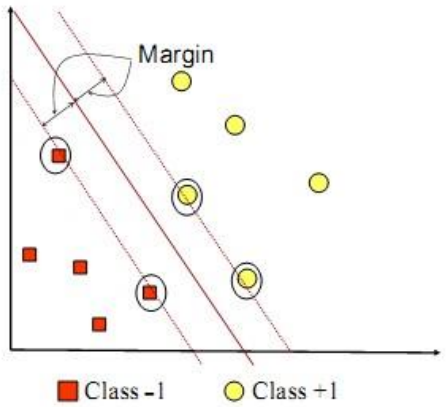
N = jumlah semua dokumen yang ada dalam kumpulan dokumen

n = jumlah dokumen yang mengandung kata (minimal ada satu kata yaitu term)

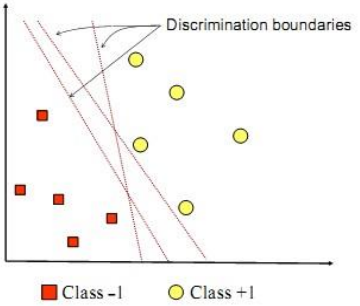
2.6 SVM

Support Vector Machine pertama kali diperkenalkan oleh Vapnik pada tahun 1992 sebagai rangkaian harmonis konsep-konsep unggulan dalam bidang pattern recognition [**6**]. SVM adalah metode *learning machine* yang bekerja atas prinsip *Structural Risk Minimization* (SRM) dengan tujuan menemukan hyperplane terbaik yang memisahkan dua buah class pada input space.

Gambar 2.1 memperlihatkan beberapa pattern yang merupakan anggota dari dua buah class : +1 dan –1. Pattern yang tergabung pada class –1 disimbolkan dengan warna merah (kotak), sedangkan pattern pada class +1, disimbolkan dengan warna kuning (lingkaran). Problem klasifikasi dapat diterjemahkan dengan usaha menemukan garis (*hyperplane*) yang memisahkan antara kedua kelompok tersebut [**6**].



Gambar 2. 1 *SVM berusaha untuk menemukan hyperplane terbaik yang memisahkan kedua kelas -1 dan +1 [6]*

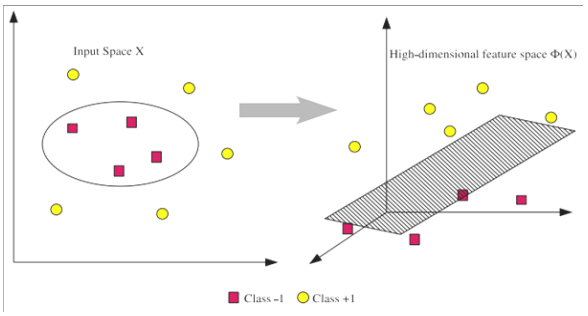


Gambar 2. 2 Hyperplane terbentuk diantara class-1 dan +1 [**6**]

Hyperplane pemisah terbaik antara kedua class dapat ditemukan dengan mengukur margin hyperplane tsb. dan mencari titik maksimalnya. *Margin* adalah jarak antara hyperplane tersebut dengan pattern terdekat dari masing-masing class. Pattern yang paling dekat ini disebut sebagai *support vector*. Garis solid pada gambar menunjukkan hyperplane yang terbaik, yaitu yang terletak tepat pada tengah-tengah kedua class, sedangkan titik merah dan kuning yang berada dalam lingkaran hitam adalah support vector. Usaha untuk mencari lokasi hyperplane ini merupakan inti dari proses pembelajaran pada SVM.

2.6.1 *Kernel Trick* dan *non linear* SVM

Pada umumnya masalah dalam domain dunia nyata (real world problem) jarang yang bersifat *linear* separable, kebanyakan bersifat non *linear*. Untuk menyelesaikan problem non *linear*, SVM dimodifikasi dengan memasukkan fungsi Kernel. Dalam non *linear* SVM, pertama-tama data x dipetakan oleh fungsi Φ(x) ke ruang vektor yang berdimensi lebih tinggi. Pada ruang vektor yang baru ini, *hyperplane* yang memisahkan kedua class tersebut dapat dikonstruksikan. Hal ini sejalan dengan teori Cover yang menyatakan *"Jika suatu transformasi bersifat non linear dan dimensi dari feature space cukup tinggi, maka data pada input space dapat dipetakan ke feature space yang baru, dimana pattern-pattern tersebut pada probabilitas tinggi dapat dipisahkan secara linear"*.



Gambar 2. 3 Pemetaan input space berdimensi dua dengan pemetaan ke dimensi tinggi

Ilustrasi dari konsep ini dapat dilihat pada gambar. Pada gambar di atas sisi kiri diperlihatkan data pada class kuning dan data pada class merah yang berada pada input space berdimensi dua tidak dapat dipisahkan secara *linear*. Selanjutnya gambar menunjukkan bahwa fungsi Φ memetakan tiap data pada input space tersebut ke ruang vektor baru yang berdimensi lebih tinggi (dimensi 3), dimana kedua class dapat dipisahkan secara *linear* oleh sebuah *hyperplane*. Notasi matematika dari mapping ini adalah sbb.

(2.11)

Pemetaan ini dilakukan dengan menjaga topologi data, dalam artian dua data yang berjarak dekat pada input space akan berjarak dekat juga pada feature space, sebaliknya dua data yang berjarak jauh pada input space akan juga berjarak jauh pada feature space.

Proses pembelajaran pada SVM dalam menemukan titik-titik *support vector*, hanya bergantung pada dot product dari data yang sudah ditransformasikan pada ruang baru yang berdimensi lebih tinggi, yaitu ( ) ( )

Karena umumnya transformasi Φ ini tidak diketahui, dan sangat sulit untuk difahami secara mudah, maka perhitungan dot product tersebut sesuai teori Mercer dapat digantikan dengan fungsi kernel ( ) yang mendefinisikan secara implicit transformasi Φ. Hal ini disebut sebagai Kernel Trick [11], yang dirumuskan:

( ) ( ) ( ) (2.12)

Kernel trick memberikan berbagai kemudahan, karena dalam proses pembelajaran SVM, untuk menentukan support vector, kita hanya cukup mengetahui fungsi kernel yang dipakai, dan tidak perlu mengetahui wujud dari fungsi non *linear* Φ. Berbagai jenis fungsi kernel dikenal, sebagaimana dirangkumkan pada tabel.

Tabel kernel yang umum digunakan

Jenis Kernel	**Definisi**
Polynomial	$K(\vec{x}_i, \vec{x}_j) = (\vec{x}_i, \vec{x}_j + 1)^p$
Gaussian RBF	$K(\vec{x}_i, \vec{x}_j) = \exp\left(-\frac{\left\|\vec{x} - \vec{x}_j\right\|^2}{2\sigma^2}\right)$
Sigmoid	$K(\vec{x}_i, \vec{x}_j) = \tanh(\alpha\vec{x}_i, \vec{x}_j + \beta)$
Linear	$K(\vec{x}_i, \vec{x}_j) = \vec{x}_i^{\,t}\vec{x}_j$

Selanjutnya hasil klasifikasi dari data xr diperoleh dari persamaan berikut

$$f(\Phi(\vec{x})) = \vec{w}.\Phi(\vec{x}) + b \quad (2.13)$$

$$= \sum_{i=1,\vec{x}_i \in SV}^{n} \alpha_i y_i \Phi(\vec{x}).\Phi(\vec{x}_i) + b \quad (2.14)$$

$$= \sum_{i=1,\vec{x}_i \in SV}^{n} \alpha_i y_i K(\vec{x}, \vec{x}_i) + b \quad (2.15)$$

Support vector pada persamaan di atas dimaksudkan dengan subset dari training set yang terpilih sebagai *support vector*, dengan kata lain data x_i yang berkorespondensi pada $\alpha_i \geq 0$.

2.6.2 Gaussian Kernel

Penggunaan kernel merupakan salah satu faktor dalam keberhasilan dari banyaknya algoritma klasifikasi untuk permukaan non*linear*. Gaussian Kernel adalah pilihan kernel yang menjanjikan. Kernel ini secara non*linear* memetakan sampel ke dalam ruang dimensi yang lebih tinggi, sehingga tidak seperti kernel *linear*, kernel ini dapt menangani kasus ketika hubungan antara label kelas dan atributnya tidak *linear*. Alasan kedua adalah pada kernel Gaussian Kernel, kompleksitas hyperparameter-nya lebih sedikit dibandingkan dengan kernel non*linear* lain seperti kernel polinomial [7] dengan persamaan:

( ) (‖ ‖) (2.16)

Dimana niali $\gamma \geq 0$ adalah sebuah parameter yang mengontrol besarnya fleksibilitas persamaan Gaussian ini. Dapat dilihat bahwa data yang bersifat *linear* dapat di kernelisasikan, dengan syarat bahwa independensi data hanya menggunakan dot products.

2.7 Visualisasi Google Maps

Google Maps merupakan layanan gratis Google yang cukup popular. Didalam google maps dapat ditambahkan fitur membuat google maps sendiri dengan memanfaatkan Google MapsAPI. Google MapsAPI adalah library JavaScript. Untuk dapat mengimplementasikannya dibutuhkan kemampuan tentang Bahasa pemrograman HTML dan javascript. Dengan adanya google maps API memudahkan dalam pembangunan suatu peta yang terfokus pada data yang diinginkan saja.

Cara penulisan program Google Map API :

1. Memasukkan Maps API JavaScript ke dalam HTML telah dibuat.
2. Membuat element div dengan nama map_canvas untuk menampilkan peta.
3. Membuat beberapa objek literal untuk menyimpan property -properti pada peta.
4. Menuliskan fungsi JavaScript untuk membuat objek peta.
5. Melakukan inisiasi peta dalam tag body HTML dengan event onload

2.8 Pengukuran Evaluasi

Dalam menguji keefektifan suatu klasifikasi dibutuhkan suatu pengukuran evaluasi. Pengukuran tersebut didapatkan dalam sebuah set *confusion matrix* [**8**]. *Confusion matrix* merupakan sebuah tabel (Tabel 2-1) yang terdiri atas banyaknya baris data uji yang diprediksi benar dan tidak benar oleh model klasifikasi, digunakan untuk menentukan kinerja suatu model klasifikasi.

Tabel 2-1: Confusion Tabel

		Kelas Prediksi	
		Positif	Negatif
Observasi	Positif	TP	FN
	Negatif	FP	TN

Keterangan :

TP (True Positive) adalah kelas yang diprediksi positif dan benar.

TN (True Negatif) adalah kelas yang diprediksi negatif dan benar.

FP (False Positive) adalah kelas yang diprediksi positif dan salah.

FN (False Negatif) adalah kelas yang diprediksi negatif dan salah.

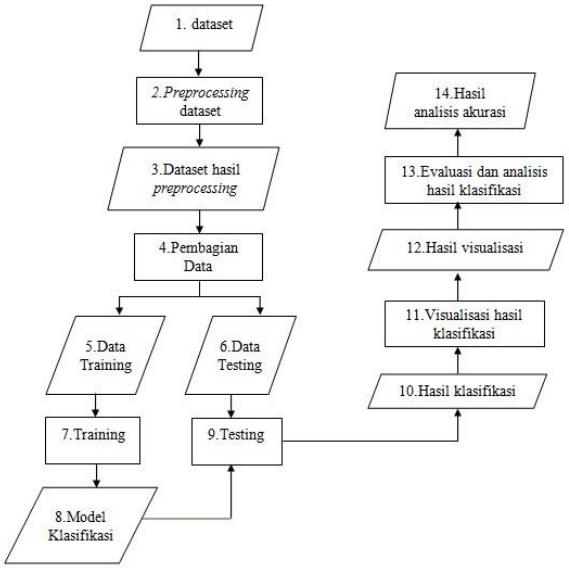
Sehingga akurasi dari klasifikasi dapat diperoleh dari penjumlahan true positif dan true negatif

dibagi total untuk melihat kinerja secara keseluruhan dengan rumus berikut:

$$akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.15)$$

3 Deskripsi Umum Sistem

Dalam penelitian ini penulis membuat sistem yang dapat melakukan klasifikasi terkait kondisi sebuah jalan ke dalam macet atau tidak berdasarkan tweet. Gambaran umum sistem yang akan dibuat dalam penelitian tugas akhir ini adalah



Gambar Gambaran Sistem secara global

4 Pengujian dan Analisis Sistem

4.1 Pengujian Sistem

Pengujian sistem yang dilakukan pada tugas akhir ini untuk melihat performansi metode Support Vector Machine dalam melakukan prediksi terhadap data testing. Performansi diukur dengan melakukan perbandingan antara hasil testing yang diklasifikasikan oleh sistem dengan data testing yang sebelumnya telah diberi label.

4.1.1 Tujuan Pengujian

Tujuan dilakukannya pengujian ini adalah:

a. Menganalisis pengaruh perbandingan data training dan data testing terhadap performansi sistem
b. Menganalis pengaruh jumlah dataset dalam pembentukan model klasifikasi
c. Menganalisis hasil klasifikasi yang didapatkan

4.2 Dataset

Dataset yang digunakan dalam pengerjaan tugas akhir ini adalah dokumen berbahasa Indonesia. Data diambil dari twitter.com menggunakan *twitter* API. Data yang diambil dari twitter berupa text, ur, mention, hashtag, retweet dan waktu. Data diambil berdasarkan kata kunci tertentu yaitu gatsu, mh thamrin, sudirman, tb simatupang, @lewatmana, @TMCPoldaMetro. Tweet berisi maksimal 140 karakter. Dan dataset yang digunakan adalah teks biasa dengan karakter a-z. Jumlah total dataset yang digunakan adalah 934 tweet. Dataset terdiri dari 7 jalan raya yang terdapat di Jakarta dengan masing masing jumlah pada setiap jalan terdapat pada table 4.1

Tabel 4. 1 Jumlah Dataset

Nama Jalan	Jumlah
MH Thamrin	175 tweet
Veteran	92 Tweet
Gatot Subroto	207 Tweet
Sudirman	48 Tweet
HR Rasuna Said	38 Tweer
MT Haryono	175 Tweet
TB Simatupang	199 Tweet

Contoh data tweet yang digunakan dalam penelitian ditunjukkan pada tabel 4.2 sebagai berikut:

Tabel 4. 2 contoh dataset

No	Tweet	Label
1	Lalin di Jalan TB Simatupang ke Cilandak Lancar: Arus lalulintas di Jalan ... http://t.co/kZMNO9lIQy #TeamAyana	-1
2	RT @UcupBengsin: Kemacetan panjang arah fatmawati dari ps.rebo menumpuk d dpn Pom Bensin PP TB Simatupang. http://t.co/QsRâ€	1
3	#Macet #Jokowi cctv @lewatmana: Arteri MT Haryono dari Pancoran menuju Cawang macet	1
4	RT @TMCPoldaMetro: RT @tweetanuan: Kondisi lalulintas MT Haryono arah Kp Melayu ramai lancar. http://t.co/7wkUW9zZzg	-1
5	cctv @lewatmana: Arteri MT Haryono dari Pancoran menuju Cawang macet sebaliknya tersendat. http://t.co/cbdmzjwTxq	1
6	#JMRLalin MT Haryono arah suhat ramai lancar arah dinoyo padat merambat | http://t.co/24uYAXVvCI	-1
7	Jl.MT haryono dpn ktr BNN Cawang arah Pancoran apbl melintas harap hati2 lalin padat	1
8	Kecelakaan Bus di Jl MT Haryono Lalin Arah Pancoran Padat http://t.co/DctSR0W8h6	1
9	#Tol_JLJ Serpong - Veteran - TMII - Cikunir - Semper LANCAR.	-1

4.3 Skenario Pengujian

a. Perbandingan jumlah dataset
Pengujian perbandingan jumlah dataset dilakukan dengan mengubah jumlah dataset yang digunakan pada setiap

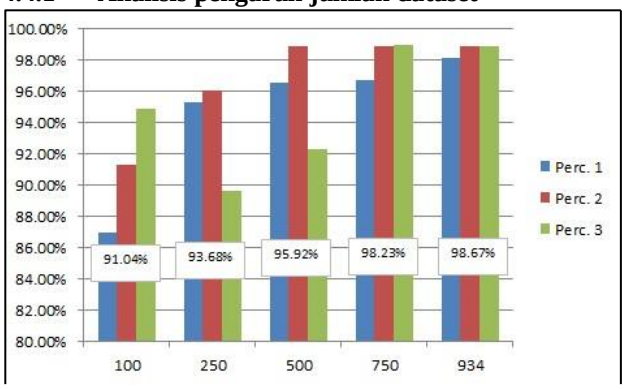
pengujian. Analisa dilakukan dengan membandingkan akurasi yang didapat pada setiap model dengan jumlah dataset yang berbeda.

b. Perbandingan komposisi data training dan data testing
Pengujian perbandingan komposisi data training dan data testing dilakukan untuk melihat pengaruh pada model yang dibuat oleh sistem klasifikasi. Pengujian ini dilakukan dengan mengubah komposisi data training dan data testing kemudian membandingkan dan menganalisa hasil akurasi dari masing-masing komposisi.

c. Analisis hasil klasifikasi pada sistem dan visualisasi di google maps

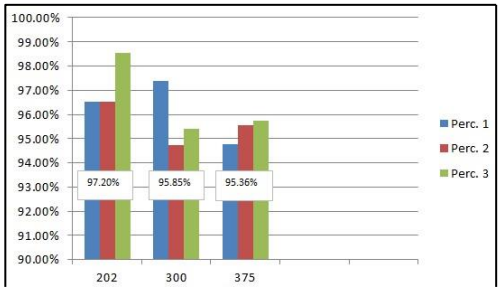
4.4 Analisis Hasil Pengujian

4.4.1 Analisis pengaruh jumlah dataset



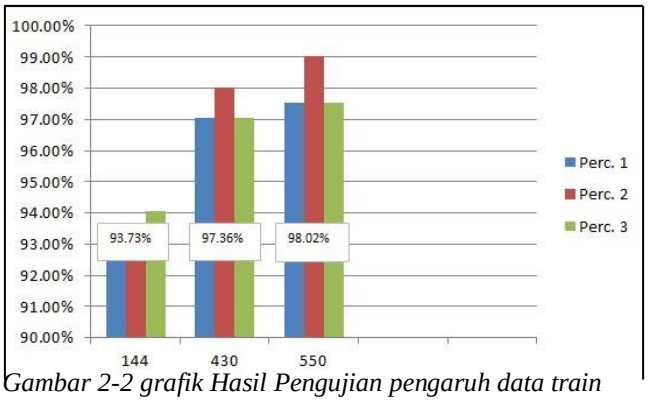
Dari hasil skenario yang telah dilakukan dapat dilihat bahwa tinggi rendahnya akurasi dipengaruhi oleh jumlah dataset yang digunakan. Rata rata dari hasil pengujian yang dilakukan semakin banyak dataset yang dipakai maka hasil akurasi yang didapatkan juga semakin tinggi. Hal ini dikarenakan dengan semakin banyak data yang dipakai maka semakin banyak keberagaman atau jenis data yang dapat ditangani oleh sistem sehingga ketika melakukan klasifikasi secara otomatis model mampu menangani berbagai macam pola. Dari data hasil pengujian didapatkan rata-rata akurasi tertinggi yaitu 98.67% pada data dengan jumlah dataset 934. Pada setiap percobaan akurasi yang didapatkan tidak selalu sama dikarenakan data yang terdapat pada data test dan data train merupakan hasil random pada saat melakukan pemecahan data.

4.4.2 Pengaruh komposisi data *training* dan data *testing*



Gambar 2-1 Grafik Hasil Pengujian pengaruh data test

Dari hasil scenario pertama yang telah dilakukan pada gambar 4.2 dapat dilihat semakin banyak data test yang dipakai ketika melakukan testing pada jumlah data train yang sama didapatkan hasil akurasi yang semakin mengecil. Hal ini dikarenakan sistem tidak mampu menangani keberagaman data yang semakin beragam pada data test yang terus meningkat.



Gambar 2-2 grafik Hasil Pengujian pengaruh data train

Semakin banyak jumlah data yang diberikan pada data train maka semakin tinggi akurasi yang didapatkan. Hal tersebut terjadi karena dengan banyaknya data train, model yang terbentuk dapat menangani lebih banyak keberagaman data yang ada pada dataset sehingga mampu mengklasifikasikan dengan lebih baik ketika melakukan testing.

Berdasarkan hasil pengujian baik dari scenario pertama maupun kedua, data dengan data test mendapatkan akurasi tertinggi sebesar 100%. Hal ini karena pada data yang di train untuk menghasilkan model klasifikasi sangat beragam dan hampir mewakili keseluruhan dari dataset yang ada sehingga seluruh keberagaman data saat melakukan proses pelabelan data testing dapat ditangani oleh model yang dibentuk. Meskipun hasil pengujian dengan jumlah datatrain berjumlah 144 dan data test sebanyak 202 memiliki akurasi paling kecil, tetapi rata rata akurasi yang didapatkan tergolong tinggi yaitu 93,73%. Hal ini dikarenakan ketika jumlah data train yang digunakan sedikit tetapi karakteristik data train tersebut baik (data memiliki perbedaan yang jelas untuk dapat diklasifikasi) maka model yang dibentuk juga akan menghasilkan model yang baik juga sehingga akurasi yang didapatkan tinggi.

4.4.3 Analisis hasil Klasifikasi

Pada beberapa skenario, tidak semua data testing mampu diklasifikasikan secara benar oleh sistem. Penyebab kesalahan sistem yang pertama adalah data yang digunakan pada data testing tidak terdapat pada data training sehingga sistem tidak mampu menemukan pola yang sesuai dengan data yang ada. Pada proses training sistem akan melakukan pembelajaran sesuai dengan data yang ada misalkan terdapat suatu kata kunci pada *tweet* tertentu maka sistem akan melakukan klasifikasi sesuai kata kunci yang ada. Sehingga jika terdapat kata kunci baru untuk suatu emosi yang tidak terdapat pada data *training* maka akan membuat sistem salah dalam melakukan klasifikasi.

Kesalahan yang kedua adalah dalam satu tweet terdapat dua atau lebih kata kunci yang berbeda dimana

kata kunci tersebut memiliki jenis klasifikasi yang berbeda pula. Contoh pada kasus ini terdapat pada data tweet ke-273 dan 247 pada table 4.5 diatas. Pada tweet tersebut terdapat dua kata kunci terhadap kelas macet (padat) dan ramai lancar. Oleh karena itu ketika suatu tweet memuat dua atau lebih kata kunci untuk jenis kelas yang berbeda menyebabkan suatu tweet tidak dominan pada salah satu kelas sehingga kesalahan sistem dalam melakukan klasifikasi akan tinggi.

5. Penutup

Kesimpulan

Berdasarkan analisis terhadap hasil pengujian yang telah dilakukan pada Tugas Akhir ini, dapat disimpulkan bahwa :

1. Metode Support Vector Machine dapat diimplementasikan untuk melakukan klasifikasi data kemacetan yang terdapat pada Twitter .
2. Semakin banyak karakteristik data yang digunakan dalam sistem semakin tinggi nilai akurasi yang didapatkan. Hasil rata-rata percobaan pada jumlah dataset menunjukkan hasil tertinggi pada dataset berjumlah 934 tweet dengan akurasi 98.67%
3. Semakin banyak data yang dipakai ketika melakukan proses training semakin tinggi akurasi yang dihasilkan oleh sistem dalam melakukan klasifikasi.

Saran

Saran yang diperlukan dari tugas akir ini untuk pengembangan sistem yang lebih lanjut adalah sebagai berikut :

1. Membuat sistem yang mampu menangani penambahan kata tidak baku secara otomatis mengingat tweet bahasa Indonesia memiliki karakteristik data yang beragam.
2. Mampu melakukan online learning sehingga data dapat diambil secara real time.

Daftar Pustaka

[1] Andreas M., Michael Haenlein Kaplan, *Users of the world, unite! The challenges and opportunities of Social Media*.: Business Horizons, 2010.

[2] Hepburn, "Infographic : Twitter Statistic, Facts & Figures, http://visual.ly/twitter-facts-and-figures-2014," Diakses tanggal 18 Maret 2014.

[3] Ariief Budi Witarto, dan Dwi Handoko Nugroho Satriyo Anto, *Support Vector Machine- Teori dan Implementasinya dalam Bioinformatika*., 2003.

[4] Ni Wayan Sumartini, "Text Mining dengan Metode Naive Bayes Classifier dan Support Vector Machines untuk Sentiment Analysis," Juli 2011.

[5] J. Han dan M.Kamber, *Data Mining : Concepts and Techniques*.: Series Editor Morgan Kaufmann, 2000.

[6] James Sanger Ronen Feldman, *The Text Mining Handbook*. New York: Cambridge University Press, 2007.

[7] Alif Septian Nurdianto, *Klasifikasi Emosi pada Twitter menggunakan Metode Multiclass SVM*. Bandung: Universitas Telkom, 2014.

[8] Chin Wei(et. al.) Hsu, *A Practical Guide to Support vector Classification*. Taiwan: Department of Computer Science National Taiwan University, 2010.

[9] Max Bramer, *Principles of Data Mining*. London: Springer, 2007.

[10] A.F., Purwarianti, A Wicaksono, "HMM Based POS Tagger for Bahasa Indonesia," *On Proceedings of 4th International MALINDO (Malay - Indonesian Language) Workshop*, 2nd August 2010.

[11] http://lucene.apache.org/core/3_0_3/api/contrib-snowball/index.html?overview-summary.html.

[12] Sandi Fajar Rodiyansyah, "Klasifikasi Posting Kemacetan Lalu Lintas Kota Bandung menggunakan Naive Bayes Classifier," 2012.

[13] Patrick Ozer, "Data Mining Algorithms for Classification," 2008.

[14] V. Kumar, J. Ross Quinlan, J. Gosh, Q. Yang, H. Motoda X. Wu, "Top 10 Algorithm in Data Mining," 2007.

[15] D.T Larose, *Discovering Knowledge in Data*. New Jersey: John Willey & Sons, Inc, 2005.

[16] A. Rajaraman, *Mining of Massive Datasets*., 2011.

[17] Sari Khrisna Dini Yunita, *Text Categorization with Support Vector Machine (SVM) Classification Method*. Bandung: Institut Teknologi Telkom, 2006.

[18] Rinaldi Munir, *Diktat Kuliah Strategi Algoritmik*. Bandung, 2005.

[19] http://lucene.apache.org/core/3_0_3/api/contrib-snowball/index.html?overview-summary.html.