ANALISIS SENTIMEN PENGGUNA TWITTER TERHADAP ACARA STANDUP COMEDY DENGAN MENGGUNAKAN SUPPORT VECTOR MACHINE DAN NAÏVE BAYES CLASSIFIER

Rina Priyani

Teknik Informatika - Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
E-mail : rinapriyani27@gmail.com

ABSTRAK

Salah satu acara *standup comedy* adalah *standup comedy* kompastv. Adanya akun resmi *standup comedy* kompastv di twitter, memungkinkan masyarakat yang menjadi followers akun tersebut melakukan *feedback* dengan menanggapi setiap postingan *tweet* dari akun tersebut. Karena ketersediaan sentimen untuk acara *standup comedy* kompastv sangat melimpah, sehingga *feedback* yang disampaikan oleh *followers* dapat digunakan untuk mengukur analisis sentimen *followers* terhadap acara *standup comedy* di kompastv. Banyaknya *tweet* yang masuk ke akun twitter *standup comedy* kompastv dapat digunakan untuk mengetahui sentimen *followers* terhadap acara *standup comedy* kompastv, akan tetapi dalam penyampaian sentimen tersebut terdapat sentimen yang bukan merupakan opini, maka dari itu sebelum sentimen diklasifikasikan ke dalam kategori positif dan negatif, sebuah data twitter harus dipisahkan ke dalam kategori relevan dan tidak relevan dengan menggunakan metode *Support Vector Machine*. Setelah data twitter masuk ke dalam kategori relevan, selanjutnya data twitter tersebut diklasifikasikan ke dalam kategori positif dan negatif dengan menggunakan metode *Naïve Bayes Classifier*.

Permasalahan analisis sentimen adalah bagaimana mengklasifikasikan sentimen kedalam kelas-kelas yang sudah ditentukan. Selanjutnya, untuk mengatasi permasalahan tersebut dilakukan pembelajaran mesin dengan menggunakan *Support Vector Machine* untuk pemisahan data relevan dan tidak relevan, dan *Naïve Bayes Classifier* untuk klasifikasi data positif dan negative. Penggunaan *Support Vector Machine* dan *Naïve Bayes Classifier* dikarenakan performasinya lebih baik untuk teks berbahasa Indonesia, sehingga analisis sentiment dapat diterapkan pada *tweet* yang memiliki nilai subyektif.

Dari hasil pengujian dengan metode 10-fold cross validation menunjukkan bahwa kedua metode yang digunakan memiliki tingkat akurasi yang cukup memuaskan yaitu 90.74% untuk metode *Support Vector Machine* dan 95.91% untuk metode *Naïve Bayes Classifier*.

Kata kunci: Analisis sentimen, *Support Vector Machine*, *Naïve Bayes Classifier*, *Cross Validation*, Twitter, *tweet*

1. PENDAHULUAN

Dewasa ini perkembangan pertelevisian Indonesia sangatlah pesat, sehingga banyak bermunculan acara-acara baru yang bervariasi. Mulai dari acara *news*, *talk show*, *variety show* hingga acara lawak atau komedi. Salah satu acara komedi adalah *standup comedy*, yaitu salah satu genre profesi melawak yang pelawaknya membawakan lawakannya di atas panggung seorang diri, dengan cara bermonolog mengenai suatu topik. Orang yang melakukan kegiatan ini disebut pelawak tunggal atau komik[1].

Salah satu televisi yang mempunyai acara *standup comedy* yaitu kompastv. Tak sekedar memberi hiburan komedi dengan komik yang sudah ada, kompastv pun beberapa kali menggelar acara pencarian bakat *standup comedy*, yang audisinya tersebar di seluruh Indonesia. Oleh karena antusias masyarakat yang cukup besar terhadap *standup comedy* ini, maka kompastv pun memiliki media sosial sebagai sarana promosi acara *standup comedy*, salah satunya yaitu twitter. Twitter digunakan kompastv untuk menampilkan jadwal acara standup comedy yang akan ditayangkan sehingga masyarakat dapat menonton acara tersebut. Adanya akun resmi *standup comedy* kompastv di twitter, memungkinkan masyarakat yang menjadi followers akun tersebut melakukan *feedback* dengan menanggapi setiap postingan *tweet* dari akun tersebut. Karena

ketersediaan sentimen untuk acara *standup comedy* kompastv sangat melimpah, sehingga *feedback* yang disampaikan oleh *followers* dapat digunakan untuk mengukur analisis sentimen *followers* terhadap acara *standup comedy* di kompastv.

Permasalahan analisis sentimen adalah bagaimana melakukan klasifikasi sentimen pada data twitter yang hasilnya dapat dimanfaatkan untuk evaluasi mengenai kualitas suatu acara televisi. Penggunaan *Support Vector Machine* dan *Naïve Bayes Classifier* tersebut dikarenakan performasinya lebih baik untuk klasifikasi teks berbahasa Indonesia, sehingga analisis sentimen dapat diterapkan pada *tweet* bahasa Indonesia yang memiliki nilai subyektif[2].

2. ANALISIS SENTIMEN PENGGUNA TWITTER TERHADAP ACARA STANDUP COMEDY DENGAN MENGGUNAKAN SUPPORT VECTOR MACHINE DAN NAÏVE BAYES CLASSIFIER

Kompastv merupakan sebuah stasiun televisi swasta yang memiliki program acara *standup comedy*, dan menggunakan Twitter sebagai salah satu sosial media untuk sarana promosi. Dengan adanya akun *standup comedy* kompastv dan jumlah postingan *tweet* yang banyak setiap harinya memungkinkan terdapat *feedback* dari penikmat acara standup comedy yang merupakan *followers* dari akun twitter tersebut, yang dapat dianggap sebagai ketersediaan sentimen yang dimanfaatkan untuk evaluasi kualitas program acara standup comedy kompastv.

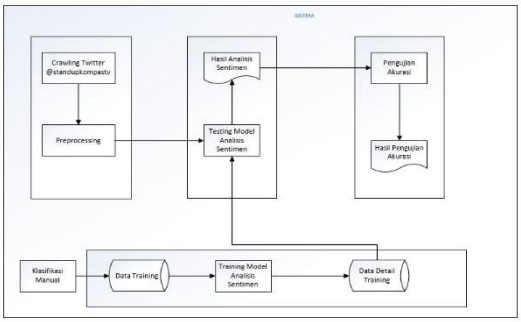
Namun permasalahan pada analisis sentimen adalah bagaimana melakukan klasifikasi sentimen pada data twitter yang hasilnya dapat dimanfaatkan untuk evaluasi program acara *standup comedy* di kompastv.

2.1 Analisis Sistem

Dalam mengimplementasikan metode *Support Vector Machine* dan *Naïve Bayes Classifier* akan dibangun sebuah prototype dengan gambaran sistem yang akan tertera pada Gambar 1.

Sistem yang akan dibangun memiliki 4 tahap yaitu praproses, training data, testing, dan *cross validation*. Untuk penjelasan setiap tahap dapat dilihat sebagai berikut:

1. Tahap pertama adalah tahap mengambil data twitter dengan crawling menggunakan stream API, kemudian data tersebut dilakukan proses *preprocessing* yang merupakan tahap dimana data yang akan digunakan sebagai data testing dibersihkan dari *noise* atau dari hal yang tidak mempunyai pengaruh dalam sentimen. *Preprocessing* yang dilakukan dalam tahap ini antara lain: *case folding*, *cleansing*, *stopword removal*, *convert emoticon*, *convert negation*, tokenizer.
2. Tahap yang kedua adalah tahap training data, pada tahap ini menggunakan dua data training, yaitu data training support vector machine dan data training naïve bayes, lalu kedua data training tersebut dilakukan tahap *preprocessing* dan diklasifikasikan secara manual kedalam empat kelas yaitu kelas relevan dan tidak relevan untuk data training support vector machine, dan kelas positif dan negatif untuk data training naïve bayes. Kemudian keempat kelas tersebut di simpan ke dalam database data training. Setelah itu *tweet* yang sudah diklasifikasikan secara manual dibuat model analisis sentimen yang nantinya akan disimpan kembali ke dalam database detail data training yang akan digunakan pada tahap testing sebagai model analisis sentimen.
3. Tahap yang ketiga yaitu tahap testing, pada tahap ini menggunakan data testing support vector machine dan data testing naïve bayes sebagai data masukan yang tentu saja sudah melalui tahap *preprocessing*. Kedua data testing tersebut diproses sehingga dapat dilihat kelas sentimennya.
4. Tahap keempat yaitu tahap pengujian menggunakan metode *k-fold cross validation* dimana data dibagi sejumlah n-*fold* yang diinginkan,kemudian proses testing dan training dilakukan sebanyak n kali sampai hasil akurasi keluar. Hasil akurasi tersebut didapat dari rata-rata akurasi pada setiap iterasi.



Gambar 1. Gambaran Sistem

2.2 Analisis Data Masukan

Data masukan yang digunakan adalah data *tweet* dari akun twitter resmi StandUp KompasTv. Data *tweet* tersebut didapat dengan memanfaatkan fitur *API* (*Application Interface*) yang telah disediakan oleh twitter untuk mendapakan *API key*, dan *access token* yang nantinya digunakan untuk pengambilan data *tweet.* Data yang diambil adalah data *tweet* yang mengandung kata "standupkompastv" atau data *tweet* yang terdapat pada akun resmi "@StandUpKompasTV", data tersebut dapat dianggap mewakili sentimen dari pengguna atau *followers* StandUp KompasTV. Data berupa sebuah kalimat dengan panjang maksimal

140 karakter. Contoh dari setiap *tweet* yang mengandung sentimen pada akun "@StandUpKompasTV" terdapat pada Gambar 2 dan Gambar 3.



Gambar 2. Tweet dengan sentimen positif



Gambar 3. Tweet dengan sentimen negatif

Data *tweet* yang diambil dari akun resmi Standup comedy kompastv masih berupa data mentah dan terdapat noise atau ciri-ciri yang tidak mempunyai pengaruh pada klasifikasi sentimen seperti link, "@", *stopword*, *hashtag* yang ditandai dengan munculnya karakter "#". Contoh *tweet* mentah tersebut ada pada Gambbar 4.



Gambar 4. Contoh data tweet

Selain itu, karateristik *tweet* yang dirasa mempunyai pengaruh pada penentuan sentimen adalah terdapat *emoticon* yang merupakan salah satu cara untuk mengekspresikan ungkapan persetujuan atau pertidaksetujuan pada suatu kalimat atau *tweets.* Contoh *tweet* mentah yang terdapat *emoticon* ada pada Gambar 5.



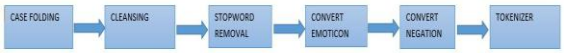
Gambar 5. Contoh data tweet dengan emoticon

Karakteristik *tweet* selanjutnya yaitu ada terdapat kata negasi yang perlu diperhatikan dalam analisis sentimen karena dapat merubah nilai sentimen suatu *tweet*. Kata yang bersifat negasi seperti "tidak", "bukan", "ga", "jangan", "nggak", "tak", "tdk", dan "gak". Contohnya kata "lelet" yang merupakan kata dengan sentimen negatif, karena di depan kata "lelet" terdapat kata negasi "tidak" maka sentimen dari gabungan kata tersebut adalah positif. Contoh *tweet* yang terdapat kata negasi ada padaGambar 6.



Gambar 6. Contoh data tweet dengan negasi

Agar data-data *tweets* tersebut dapat dimanfaatkan dengan baik untuk mengklasifikasikan sentimen maka diperlukan proses *preprocessing* . Pada proses ini data tweet yang digunakan untuk data training data data testing dibersikan dari *noise* atau ciri-ciri yang tidak berpengaruh pada klasifikasi sentimen seperti link, *"@", "RT", stopword,* dan lain sebagainya. Gambaran proses *preprocessing* dapat dilihat pada Gambar 7.



Gambar 7. Tahapan Preprocessing

1. *Case Folding*

Pada proses *case folding* huruf besar atau *uppercase* yang terdapat pada *tweet* diubah menjadi *lowercase* atau huruf kecil.

2. *Cleansing*

Tweet yang terdapat pada akun resmi StandUp KompasTV memiliki berbagai komponen atau karakteristik tweet yang khas seperti "@" yang diidentifikasi sebagai komponen *username*, URL yang dikenal melalui operasi regular, hashtag yang menandakan kata sebagai topik yang sedang dibicarakan, dan "RT" yang diidentifikasi sebagai mengulang kembali *tweet* yang telah diposting. Komponen-komponen tersebut tidak memiliki pengaruh apapun terhadap sentimen, maka akan dibuang.

3. *Stopword Removal*

Data *tweet* yang sudah melalui proses sebelumnya masih mengandung kata yang dianggap tidak dapat memberikan pengaruh dalam menentukan suatu kategori sentimen. Kata-kata tersebut dimasukkan kedalam daftar stopword yang biasanya berupa kata ganti orang, kata ganti penghubung, pronomial penunjuk, dan lain sebagainya.

4. *Convert Emoticon*

Convert *emoticon* adalah proses mengkonversikan *emoticon* kedalam string yang sesuai dengan ekspresi *emoticon* itu sendiri. Convert *emoticon* dilakukan karena pada data *tweet* yang diambil dari standup comedy kompastv terdapat *emoticon* yang merupakan salah satu cara mengekspresikan persetujuan atau pertidaksetujuan dalam suatu *tweet*. Hal ini dirasa mempunyai pengaruh terhadap pengklasifikasian sentimen, oleh karena itu *convert emoticon* digunakan.

Tabel 1. Konversi Emoticon[3]

Emoticon	**Konversi**
>:] :-) :) :o) :] :3 :c) :> =] 8) =) :} :^)	esenang
>:D :-D :D &-D 8D x-D xD X-D XD =-D =D =-3 =3	etertawa
>:] :-(:(:-c :c :-< :< :-[:[:{	esedih

<><<><	
D:< D: D8 D; D= DX v.v D-‘:	ehoror
>:P :-P :P X-P x-p xp XP :-p :p =p :-b :b	elidah
>:o >:O :-O :O o_O o.O 8-0	ekaget
>:\ >:/ :-/ :-. :/ :\ =/ =\ :S	ekesal
:| :-|	edatar

5. *Convert Negation*

Convert negation merupakan proses konversi kata-kata negasi yang terdapat pada suatu *tweet*, karena kata negasi mempunyai pengaruh dalam merubah nilai sentimen pada suatu *tweet*. Kata negasi yang terdapat pada suatu *tweet* akan dihilangkan, dan diberikan penanda . Jika terdapat kata negasi makan akan disatukan dengan kata setelahnya. Kata-kata negasi tersebut meliputi kata “bkn”, “bukan”, “tida”, “tak”, “ga”, “enggak”, “g”, “jangan”, dan “tidak”.

6. *Tokenizer*

Pada proses tokenizer setiap kata pada *tweet* dipisahkan, pada proses ini tahap yang dilakukan adalah memisahkan setiap kata yang dipisahkan oleh spasi, selanjutnya bagian *tweet* yang memiliki karakter selain alphabet, angka, dan garis bawah akan dipecah sesuai posisi karakter tersebut dan bagian yang hanya memiliki satu karakter non alphabet dan angka akan dibuang .

2.3 Pembobotan TF-IDF

Term weighting ialah proses memberikan bobot terhadap semua kata pada dokumen, metode *Term weighting* yang digunakan pada penelitian ini adalah TF IDF. *Term frequency* adalah salah satu metode pembobotan yang paling sederhana. Pada metode ini setiap *term* diasumsikan memiliki proporsi kepentingan sesuai dengan jumlah terjadinya (munculnya) *term* tersebut dalam dokumen. Persamaan TF adalah sebagai berikut:

$$W\ d,t\ = TF(d,t)$$

Dimana TF(d,t) adalah frekuensi kemunculan *term* t pada dokumen d.

Inverse Document Frequency memperhatikan kemunculan *term* pada kumpulan dokumen. Pada metode ini, *term* yang dianggap bernilai adalah *term* yang jarang muncul pada kumpulan dokumen. Persamaan IDF adalah sebagai berikut:

$$IDF\ t\ = log\frac{N}{df(t)}$$

Dimana df(t) adalah banyak dokumen yang mengandung *term* t. TF*IDF merupakan kombinasi metode TF dengan IDF. Sehingga persamaan TF*IDF adalah sebagai berikut:

$$TF * IDF\ d,t\ = TF\ d,t\ * IDF(t)$$

2.4 Support Vector Machine

Support Vector Machine (SVM) merupakan salah satu metode *machine learning* yang mengubah *text* menjadi data *vector*. *Vector* dalam penelitian ini memiliki dua komponen yaitu dimensi (*word id*) dan bobot. Konsep SVM dapat dijelaskan secara sederhana sebagai usaha mencari *hyperplane* terbaik yang berfungsi sebagai pemisah dua buah *class* pada *input space*[4]. Usaha untuk mencari lokasi *hyperplane* ini merupakan inti dari proses pembelajaran pada SVM. Klasifikasi pada SVM dibagi menjadi 2, yaitu linier dan nonlinier.

Dimulai dengan kasus klasifikasi secara linier, fungsi ini dapat didefinisikan sebagai.

$$g\ x\ := sgn(f\ x\ )$$

Dengan $f\ x\ = wx + b$

$$\text{Atau } g\ x\ = \begin{matrix} +1, jika\ \ wx + b\ \geq +1 \\ -1, jika\ \ wx + b\ \leq -1 \end{matrix}$$

Input pada penelitian SVM terdiri dari poin-poin yang merupakan vektor dari angka-angka *real* data yang tersedia dinotasikan sebagai $xi \in R^d$ sedangkan label masing-masing dinotasikan sebagai $yi \in \{-1, +1\}$ untuk i = 1, 2, …,1 dimana 1 adalah banyaknya data. Diasumsikan kedua kelas -1 (negatif) dan +1 (positif) dapat terpisah secara sempurna oleh hyperplane berdimensi d, yang didefinisikan,

Dengan $wx + b = 0$

Sebuah *pattern* xi yang termasuk kelas -1 (sampel negatif) dapat dirumuskan sebagai *pattern* yang memenuhi pertidaksamaan:

Dengan $wx + b \leq -1$

Sedangkan *pattern* xi yang termasuk kelas +1 (sampel positif):

Dengan $wx + b \geq +1$

Margin terbesar dapat ditemukan dengan memaksimalkan nilai jarak antara *hyperplane* dan titik terdekatnya, yaitu $\frac{1}{\|w\|}$.

2.5 Klasifikasi dengan Naïve Bayes

Pada pengklasifikasian menggunakan *naïve bayes* dibagi kedalam 2 proses, yaitu proses training dan testing. Proses training digunakan untuk menghasilkan model analisis sentimen yang nantinya akan digunakan sebagai acuan untuk mengklasifikasikan sentiment dengan data testing atau data mentah yang baru. Berikut adalah algoritma klasifikasi sentimen menggunakan *Naïve Bayes Classifier*:

1. Proses Training
   a. Hitung p (c_i).
   b. Hitung p $w_k\ c_i)$ untuk setiap kata w_k pada model.
2. Proses Testing
   a. Hitung $_k$p w_k c) × p (c)untuk setiap kategori.

b. Tentukan kategori dengan nilai $_k p\ \ w_k\ c\ ) \times p\ (c)$ maksimal.

Pemberian kategori dari sebuah dokumen dilakukan dengan memilih nilai *c* yang memiliki nilai $p(C = c_i | D = d_j)$ maksimum, dan dinyatakan dengan:

$$c^* = arg\ max_{c\ \in C}\quad _k p\ \ w_k\ c\ ) \times p\ (c) \quad (1)$$

Dimana :

$_k p\ \ w_k\ \ C = c_i)$ adalah hasil perkalian dari probabilitas kemunculan semua kata pada dokumen *dj*.

$p\ \ c$ adalah hasil probabilitas dari semua kategori pada data training.

Kategori *c** merupakan kategori yang memiliki nilai $p(C = c_i \mid D = d_j)$ maksimum. Nilai $p(D = d_j)$ tidak mempengaruhi perbandingan karena untuk setiap k ategori nilainya akan sama.

3. IMPLEMENTASI DAN PENGUJIAN

3.1 Lingkungan Implementasi

Perangkat keras yang digunakan dalam proses perancangan dan implementasi sistem memiliki beberapa spesifikasi sebagai berikut:

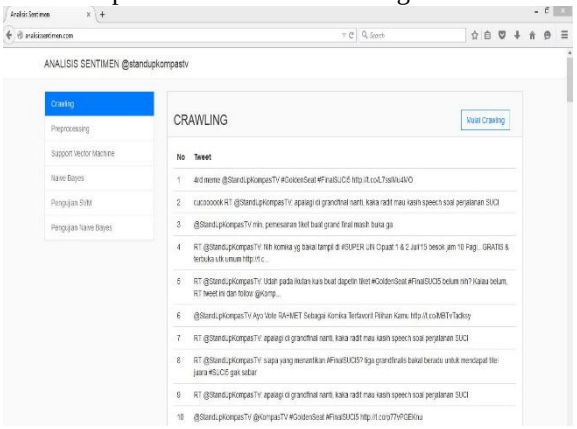
1. Processor 2.13GHz
2. RAM 1024MB
3. Monitor dengan resolusi 1366x768 pixels
4. Keyboard
5. Mouse

Perangkat lunak yang digunakan dalam proses implementasi sistem ini adalah sebagai berikut:

1. Sistem Operasi Windows 8 Starter 32 bit
2. Bahasa pemograman PHP.
3. Web server XAMPP.
4. Code editor berupa SublimeText3Portable.
5. DBMS : MySQL 5.5.27
6. Web Browser: Google Chrome 34.0.1847.137

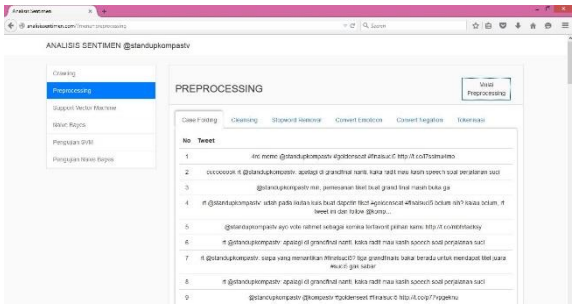
3.2 Impelementasi Antarmuka (Interface)

Pada *sub*-bab ini akan dijelaskan mengenai implementasi antar muka yang telah dirancang sebelumnya. Hasil perancangan antarmuka yang telah diimplementasikan adalah sebagai berikut:



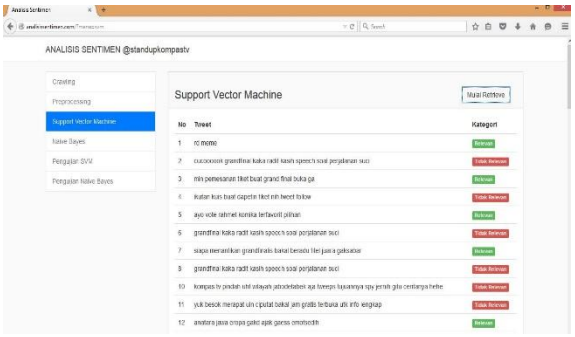
Gambar 8. Antarmuka Crawling Tweet

Gambar 8. merupakan antarmuka *Crawling Tweet* yang diimplementasikan dalam sistem. Tab ini berfungsi untuk *crawling* data *tweet* dari akun resmi standup comedy kompastv, yang nantinya akan digunakan sebagai data testing untuk diklasifikasikan kedalam beberapa kategori sentimen.



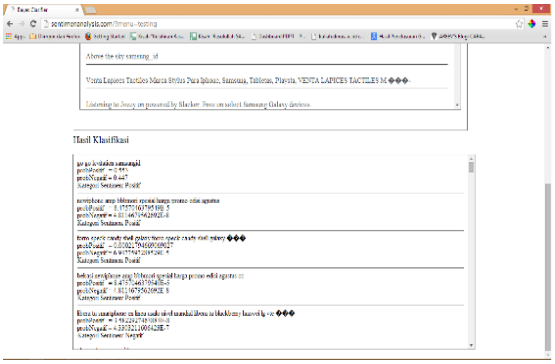
Gambar 9. Antarmuka *Preprocessing*

Gambar 9. merupakan antarmuka *preprocessing* yang berfungsi yang berfungsi untuk menampilkan hasil dari setiap *preprocessing* data testing sebelum data tersebut di klasfikasi, seperti *case folding, cleansing, stopword removal, convert emoticon, convert negation, tokenizer*.



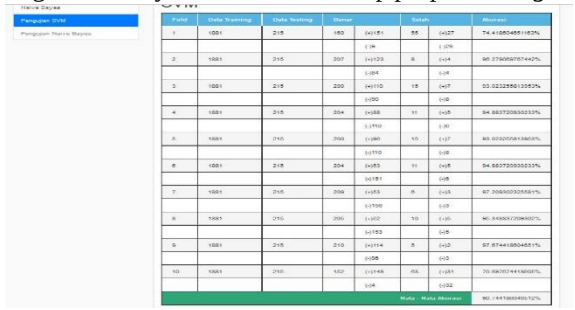
Gambar 10. Antarmuka Support vector machine

Gambar 10. merupakan antarmuka support vector machine yang berfungsi untuk menampilkan data tweet yang sudah dikelompokkan ke dalam kategori relevan dan tidak relevan.



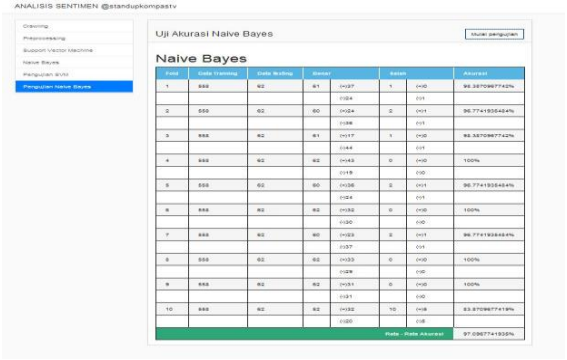
Gambar 11. Antarmuka Naïve Bayes

Gambar 11. merupakan antarmuka naïve bayes yang berfungsi untuk menampilkan data tweet yang sudah diklasifikasi ke dalam kategori positif dan negatif. hasil testing atau hasil klasifikasi data testing, klasifikasi sentimen dari setiap *tweet* pada data testing didapat dari perhitungan *naïve bayes* yang sebelumnya telah melalui tahap *preprocessing*.



Gambar 12. Antarmuka Pengujian Support Vector Machine

Gambar 12. merupakan antarmuka pengujian support vector menggunakan *cross validation* yang berfungsi untuk menguji performa klasifikasi dari sistem yang di bangun .



Gambar 13. Antarmuka Pengujian Naïve Bayes Classifier

Gambar 13. merupakan antarmuka pengujian naïve bayes menggunakan *cross validation* yang berfungsi untuk menguji performa klasifikasi dari sistem yang di bangun. Performa klasifikasi dapat dilihat dari hasil perhitungan akurasi dengan metode *cross validation*.

3.3 Pengujian Akurasi Sistem dengan Metode 10-*fold cross validation*

Jumlah data memegang peranan penting di dalam algoritma *machine learning*. Jumlah data yang sedikit (<100 instance) mungkin membuat algoritma *machine learning* tidak akurat. Algoritma *machine learning* merekomendasikan jumlah instance yang banyak (>1000 instance) namun data itu sendiri tidak mudah untuk diperoleh.

K-fold cross validation adalah teknik yang dapat digunakan apabila memiliki jumlah data yang terbatas (jumlah instance tidak banyak)[5]. *K-fold cross validation* merupakan salah satu metode yang digunakan untuk mengetahui rata-rata keberhasilan dari suatu sistem dengan cara melakukan perulangan dengan mengacak atribut masukan sehingga sistem tersebut teruji untuk beberapa atribut input yang acak. *K-foldcross validation* diawali dengan membagi data sejumlah *n-fold* yang diinginkan. Dalam proses *cross validation* data akan dibagi dalam *n* buah partisi dengan ukuran yang sama $D_1,D_2,D_3..D_n$ selanjutnya proses testing dan training dilakukan sebanyak n kali. Dalam iterasi ke-$_i$ partisi D_i akan menjadi data testing dan sisanya akan menjadi data training. Untuk penggunaan jumlah *fold* terbaik untuk uji validitas, dianjurkan menggunakan 10-*fold cross validation* dalam model.

Skenario pengujian merupakan tahap penentuan pengujian yang dilakukan. Pengujian dilakukan untuk metode support vector machine dan naïve bayes. Pengujian dilakukan menggunakan metode *k-cross validation* dengan nilai *k* sebanyak 10 *fold*, pengujian ini bertujuan untuk mengetahui akurasi metode support vector machine dan *naïve bayes classifier* yang diterapkan pada analisis sentimen jika diuji dengan data training dan data testing yang berbeda. Tahap pengujian dengan menggunakan metode 10-*fold cross validation* membagi dataset yang awalnya berjumlah 1881 data akan dibagi menjadi 10 *subset* data untuk pengujian support vector machine. Sedangkan untuk naïve bayes classifier tahap pengujian dengan menggunakan metode 10-*fold cross validation* membagi dataset yang awalnya berjumlah 549 data akan dibagi menjadi 10 *subset* data. Pada *fold* pertama terdapat kombinasi 9 *subset* yang berbeda digabung dan digunakan sebagai data training, sedangkan 1 *subset* (sisa) digunakan sebagai data testing, selanjutnya proses training dan testing dilakukan sampai *fold* kesepuluh. Skenario uji akurasi dengan metode 10-*fold cross validation* dapat dilihat pada Tabel 2.

Tabel 2. Skenario Uji Stabilitas dengan 10-*fold Cross validation*

3.4 Hasil dan Analisis Pengujian

Pada skenario uji dengan jumlah 10-*fold cross validation* dan iterasi yang dilakukan sebanyak 10 kali iterasi untuk support vector machine rata-rata yang dihasilkan yaitu 90.74%. Iterasi pertama pada *fold* pertama menggunakan 1881 data, proses training dan proses testing dilakukan sampai iterasi ke-10. Detail hasil dari akurasi pengujian skenario untuk support vector machine pada masing-masing *fold*nya dapat dilihat pada Tabel 3.

Tabel 3. Hasil Uji Stabilitas dengan 10-*fold cross validation* untuk support vector machine

Fold	Data Training	Data Testing	Akurasi
1	1881	215	74.42%
2	1881	215	96.28%
3	1881	215	93.02%
4	1881	215	94.89%
5	1881	215	93.02%
6	1881	215	94.89%
7	1881	215	97.21%
8	1881	215	95.35%
9	1881	215	97.67%
10	1881	215	70.70%
Rata-rata akurasi			90.74%

Pada skenario uji dengan jumlah 10-*fold cross validation* dan iterasi yang dilakukan sebanyak 10 kali iterasi untuk naïve bayes rata-rata yang dihasilkan yaitu 95.91%. Iterasi pertama pada *fold* pertama menggunakan 549 data, proses training dan proses testing dilakukan sampai iterasi ke-10. Detail hasil dari akurasi pengujian skenario untuk naïve bayes pada masing-masing *fold*nya dapat dilihat pada Tabel 4.

Tabel 4. Hasil Uji Stabilitas dengan 10-*fold cross validation* untuk support vector machine

Fold	Data Training	Data Testing	Akurasi
1	549	66	96.97%
2	549	66	95.45%
3	549	66	96.97%
4	549	66	98.49%
5	549	66	95.45%
6	549	66	98.49%
7	549	66	95.45%
8	549	66	98.49%
9	549	66	98.49%
10	549	66	84.85%
Rata-rata akurasi			95.91%

Fold	**Data**	***Subset***
Fold 1	Training Testing	$S_2, S_3, S_4, S_5, S_6, S_7 S_8, S_9, S_{10}$ S_1
Fold 2	Training Testing	$S_1, S_3, S_4, S_5, S_6, S_7 S_8, S_9, S_{10}$ S_2
Fold 3	Training Testing	$S_1,S_2, S_4, S_5, S_6, S_7 S_8, S_9, S_{10}$ S_3
Fold 4	Training Testing	$S_1,S_2, S_3, S_5, S_6, S_7 S_8, S_9, S_{10}$ S_4
Fold 5	Training Testing	$S_1,S_2, S_3, S_4, S_6, S_7 S_8, S_9, S_{10}$ S_5
Fold 6	Training Testing	$S_1,S_2, S_3, S_4, S_5, S_7, S_8, S_9, S_{10}$ S_6
Fold 7	Training Testing	$S_1,S_2, S_3, S_4, S_5, S_6, S_8, S_9, S_{10}$ S_7
Fold 8	Training Testing	$S_1,S_2, S_3, S_4, S_5, S_6, S_7,S_9, S_{10}$ S_8
Fold 9	Training Testing	$S_1,S_2, S_3, S_4, S_5, S_6, S_7$, S_{10} S_9
Fold 10	Training Testing	$S_1,S_2, S_3, S_4, S_5, S_6, S_7 S_8, S_9$ S_{10}

Tabel 3. mengindikasikan bahwa nilai akurasi pada tiap fold sudah sangat stabil. Percobaan pada *fold* ke-1 memperlihatkan hasil yang paling rendah diantara 9 *fold* lainnya yaitu ≈ 74.42%, sedangkan *fold* sisanya menghasilkan akurasi yang lebih besar dengan aurasi paling tinggi ada pada *fold* 1, 2, 3, 5, 7, dan 10. Sedangkan untuk Tabel 4. juga mengindikasikan bahwa nilai akurasi tiap fold sudah sangat stabil. Percobaan pada fold-10 memperlihatkan hasil yang rendah diantara 9 fold lainnya yaitu ≈ 84.85%. Jika dianalisis performa klasifikasi dengan Support Vector Machine dan *Naïve Bayes* yang dihasilkan sudah bisa dikatakan akurat, serta kinerja model klasifikasi sudah bisa dikatakan baik dengan melihat rata-rata akurasi dari setiap *fold* yang cukup besar yaitu ≈ 90.74% untuk support vector machine dan ≈ 95.91% untuk Naïve Bayes pada *10 fold cross validation* dengan data training yang digunakan sebanyak 1881 data untuk support vector machine dan 549 data untuk naïve bayes, karena secara umum semakin besar data training maka akan semakin baik kinerja model klasifikasi suatu sistem.

4 PENUTUP

4.1 Kesimpulan

Dari proses pengujian dan analisis yang telah dilakukan, kesimpulan yang dapat diambil antara lain:

1. Metode *Support Vector Machine* dapat diterapkan pada analisis sentimen dalam retrieval data untuk mengelompokkan data relevan dan tidak relevan.
2. Metode *Naïve Bayes Classifier* dapat diterapkan pada analisis sentimen dalam mengklasifikasikan sentimen positif dan negatif.
3. Hasil pengujian stabilitas performa klasifikasi dengan menggunakan metode 10-*fold cross validation* menghasilkan akurasi yang cukup tinggi yaitu 90.74% untuk Support Vector Machine dan 95.91% untuk Naïve Bayes Classifier. Hal ini menunjukan bahwa kinerja model klasifikasi sudah bisa dikatakan baik dan stabil.

4.2 Saran

Dari hasil pengujian , analisis dan kesimpulan yang telah dirumuskan, terdapat hal yang disarankan untuk penelitian selanjutnya, yaitu analisis sentimen pada penelitian ini belum memperhatikan *semantic* yaitu makna kata dan kalimat. Penelitian selanjutnya diharapkan dapat menggunakan makna kata dan kalimat untuk menentukan kategori sentimen pada suatu dokumen agar mendapatkan hasil yang lebih baik.

DAFTAR PUSTAKA

[1] [Online]. Available : http://suc.metrotvnews.com/article/ensiklopedia/13. [Accessed 12 Februari 2014].

[2] Ni Wayan Sumartini Saraswati, "pps.unud.ac.id," [Online]. Available: www.pps.unud.ac.id/thesis/pdf_thesis/unud-209-236721286-tesis.pdf. [Accessed 10 Februari 2014]

[3] I. Sunni and D. H. Widyantoro, "Analisis Sentimen dan Ekstraksi Topik Penentu Sentimen pada Opini Terhdap Tokoh Publik," *Jurnal Sarjana Institut Teknologi Bandung Bidang Elektro dan Informatika*, vol. 1, pp. 200 - 206, 2012.

[4] Nugroho A.S., Arief Budi, and Dwi Handoko. 2003. Support Vector Machine Teori dan Aplikasinya dalam Bioinformatika, kuliah umum IlmuKomputer.Com.

[5] R. Kohavi, "A study of Cross-Validation and Bootstrap for Accuracy Estimation andModelSelection,"1995. [Online].Available:http://ijcai.org/Past%20Proceedings/IJCAI-95-VOL2/PDF/016.pdf. [Accessed 18 Desember 2014].