Klasifikasi Sentimen Komentar Politik dari Facebook Page Menggunakan Naive Bayes

Antonius Rachmat C [1], Yuan Lukito[2]

***Abstrak*— Seiring maraknya situs media sosial yang digunakan sebagai sarana kampanye politik online maka makin banyak pula daukungan kampanye dari dunia maya melalui berbagai cara. Cara kampanye yang digunakan para politisi diantaranya adalah melalui Twitter hashtag, petisi di Facebook, atau pembuatan Facebook Page di mana komentarnya dapat di-like/disline oleh para pendukungnya. Permasalahan yang dibahas pada tulisan ini adalah belum banyaknya sistem yang dapat mengklasifikasikan pro kontra dari komentar-komentar yang terdapat pada Facebook Page. Pada tulisan ini akan dibahas penggunaan metode Naive Bayes untuk melakukan klasifikasi sentimen positif atau negatif terhadap komentar dari status kampanye politik dari Facebook Page. Studi kasus yang digunakan pada penelitian ini adalah status dan komentar terhadap Facebok Page calon presiden Republik Indonesia pada Pemilu tahun 2014. Tahapan penelitian dilakukan dengan pengumpulan data 68 status (3400 komentar) selama masa kampanye, dengan kegiatan preprosesing tokenisasi, stemming, pembobotan token, kemudian dilanjutkan klasifikasi, dan pengujian menggunakan confusion matrix. Dari hasil implementasi dan pengujian, metode Naive Bayes memiliki tingkat akurasi klasifikasi sentimen mencapai lebih dari 83%.**

***Kata Kunci*: Facebook Page, klasifikasi sentimen, Naive Bayes, PEMILU Indonesia 2014.**

***Abstract*— As the proliferation of social media sites being used as a means of online political campaigns so there are so many ways of campaign support using the cyberspace. Campaign ways are the use of Twitter hashtags, a petition on Facebook, or using Facebook Page where from the statuses comments can be like or disline by its supporters. The problem discussed in this paper is the absence of a system to classify the pros or cons of the comments contained in Facebook Page. In this paper we will discuss the use of Naive Bayes method to classify positive or negative sentiment from the comment on the status of the political campaign in Facebook Page. The case study used in this paper is the status and comments on Facebok Page presidential election of Indonesia in 2014. This research is done by collection of data 68 statuses (3400 comments) during the campaign period by using tokenization process, stemming, stopword, weighting token, and then classification and testing using confusion matrix. The result of the implementation and testing shows that Naive Bayes sentiment classification has accuracy more than 83%.**

Keywords: Facebook Page, sentiment classification, Naive Bayes, PEMILU Indonesia 2014.

I. PENDAHULUAN

Situs jejaring sosial banyak digunakan oleh berbagai kalangan tidak terkecuali para politikus ataupun partai politik. Para politikus menggunakan jejaring sosial untuk berbagai keperluan seperti berkampanye, pencitraan publik, memyampaikan ide/gagasan dengan menggunakan halaman khusus [1]. Salah satu contoh nyata adalah kampanye politik yang dilakukan pada saat masa pemilihan calon legislatif pada tahun 2014 di Indonesia. Para politikus sudah menggunakan situs jejaring sosial terutama Facebook dan Twitter untuk berkampanye mencari dukungan dari para pengguna kedua situs tersebut [2]. Facebook memiliki fitur yang dapat digunakan oleh para penggunanya yang cocok untuk para politikus yaitu Facebook Page, yaitu suatu halaman khusus yang dapat diklaim menjadi milik seseorang yang biasanya merupakan publik figur dan dapat digunakan untuk menyampaikan berbagai hal kepada para followernya. Para politikus menggunakan Facebook Page untuk melakukan berbagai kegiatannya seperti berpromosi sehingga pengguna Facebook dapat me-like page tersebut dan mengikuti semua berita (status) dari politikus yang difollow-nya tersebut.

Salah satu hal penting untuk dilihat dari Facebook Page adalah jumlah like atau dapat disebut sebagai dukungan dari para pengguna Facebook terhadap status ataupun komentar tertentu. Jumlah like dapat diketahui secara otomatis dari Facebook Page, namun dukungan yang berasal dari komentar-komentar yang terdapat pada status Facebook Page tidak bisa dilakukan secara otomatis. Facebook Page belum dapat mengetahui seberapa besar sentimen pengguna (pro atau kontra) dari komentar-komentar baik positif ataupun negatif secara otomatis, dengan demikian seberapa sentimen pengguna melalui Facebook Page belum dapat diketahui.

Antonius Rachmat C & Yuan Lukito telah menghasilkan sebagian dataset status dan komentar dari dua calon presiden Republik Indonesia pada masa kampanye Pemilihan Umun tahun 2014 dari Facebook Page [3]. Dataset tersebut akan digunakan sebagai dasar untuk menganalisis sentimen melalui komentar yang ada terhadap status kedua calon presiden melalui metode klasifikasi analisis sentimen. Klasifikasi sentimen dapat menggunakan

[1] *Dosen Tetap, Program Studi Teknik Informatika, Universitas Kristen Duta Wacana, Jl. Dr. Wahidin Sudirohusodo 5-25 Yogyakarta (tlp: 0274-563929; e-mail: anton@ti.ukdw.ac.id)*

[2] *Dosen Tetap, Program Studi Teknik Informatika, Universitas Kristen Duta Wacana, Jl. Dr. Wahidin Sudirohusodo 5-25 Yogyakarta (tlp: 0274-563929; e-mail: yuanlukito@ti.ukdw.ac.id)*

berbagai algoritma klasifikasi pada bidang *text mining* seperti Naive Bayes, Decision Tree, C.45, k-NN dan lain sebagainya. Berdasarkan Troussas, algoritma Naive Bayes dapat digunakan dan memiliki hasil yang cukup baik dalam melakukan klasifikasi sentimen dibandingkan dengan algoritma Perceptron [4].

Masih sedikitnya penelitian mengenai penentuan dukungan pengguna terhadap publik figur politik seperti dukungan terhadap calon presiden melalui komentar-komentar pada media sosial membutuhkan penelitian untuk mengklasifikasikan komentar-komentar ke dalam jenis dukungan positif dan negatif. Penelitian ini akan dilaksanakan untuk menjawab permasalahan tingkat keakuratan metode Naive Bayes dalam menentukan klasifikasi sentimen data komentar pengguna media sosial Facebook terhadap data Pemilu Presiden Republik Indonesia 2014 ada pertanyaan lebih lanjut mengenai penulisan makalah yang tidak ada dalam panduan, bisa menghubungi panitia publikasi jurnal melalui email yang tertera pada situs web.

II. TINJAUAN PUSTAKA

A. *Text Mining*

Text mining merupakan bagian dari bidang ilmu *Data Mining* yang khusus pada bagian pencarian pola informasi yang relevan pada data teks / dokumen dalam skala besar s. Pada *text mining*, biasanya berisi informasi-informasi yang tidak terstruktur. Oleh karena itu, pada *text mining* diperlukan proses pengubahan bentuk dari data yang tidak terstruktur menjadi data yang terstruktur yang berupa nilai numerik [5]. Setelah data menjadi data terstruktur dan berupa nilai numerik maka data tersebut dapat diproses untuk mengekstrak informasi atau pengetahuan dari dokumen-dokumen teks yang dapat digunakan untuk analisis berbagai bidang multidisiplin seperti klasifikasi, klasterisasi, temu kembali informasi, ekstraksi informasi, visualisasi, teknologi database, machine learning atau berbagai analisis teks yang lainnya [6].

1) *Tokenisasi*

Proses tokenisasi pada data teks adalah melakukan memecah sekumpulan karakter (kalimat) menjadi potongan karakter atau kata-kata sesuai kebutuhan yang sering disebut toke [7]. Berikut adalah algoritma melakukan tokenisasi dan membangun token-token dari sebuah dokumen [5]:

```
Input:
ts, semua token dalam koleksi dokumen
k, jumlah feature yang diinginkan
Output:
fs, kumpulan k feature
Inisialiasasi:
hs := table kosong

for each tok in ts do
  if hs mengandung tok then
    i:=nilai dari tok dalam hs
    i = i + 1
  else
    i = 1
  endif
  simpan i sebagai value dari tok dalam
hs
endfor
sk := keys dalam hs diurutkan
berdasarkan pengurangan nilai
fs := top k keys in sk
Output fs
```

2) *Stemming*

Tahapan setelah tokenisasi adalah tahapan stemming. Tahap ini merupakan tahap untuk mengubah token-token menjadi token yang berupa kata dasar. Tujuan dari tahap stemming adalah mengurangi jumlah token yang terbentuk sehingga dari berbagai token akan menjadi token kata dasar yang sama [5]. Algoritma stemming biasanya menggunakan algoritma Porter. Algoritma Porter telah disesuaikan untuk banyak bahasa, tidak hanya bahasa Inggris tetapi juga termasuk bahasa Indonesia. Stemming bahasa Indonesia telah banyak dikembangkan oleh para peneliti, salah satunya adalah library Sastrawi Stemming (https://github.com/sastrawi/sastrawi). Pada penelitian ini digunakan library Sastrawi Stemming untuk bahasa pemrograman PHP.

3) *Stopwords Removal*

Stopwords merupakan kumpulan daftar kata-kata yang kemungkinan besar tidak akan memberikan pengaruh prediksi, seperti imbuhan dan pronoun seperti "it" dan "they". Kata-kata yang bersifat umum tersebut sebaiknya dibuang dari kamus sebelum diproses berikutnya [5]. Penggunaan *stopwords removal* terbukti dapat meningkatkan hasil akurasi sistem klasifikasi sentimen dibandingkan tanpa penggunaan *stopwords* [8].

4) *Token Weightening*

Semua data token yang telah diciptakan pada akhir sub bab 3, selanjutnya akan diproses untuk dilakukan transformasi yang menghasilkan atribut-atribut dan nilainya agar data teks menjadi terstruktur. Atribut atau feaature tersebut harus diberi nilai bobot agar bisa dihitung tingkat kepentingan token-token penentu dalam klasifikasi sentimen. Pembobotan token menggunakan berbagai algoritma seperti *Term Frequency* (TF), TF-IDF dengan berbagai variasinya. Pada penelitian ini pembobotan yang digunakan adalah TF-IDF (*Term Frequency-Inverse Document Frequency*) dengan Persamaan (1) yang mengacu pada [5]:

$$TFIDF(t,d,D) = tf(t,d) * idf(t,D) \quad (1)$$

Di mana:

- tfidf(t,d,D) adalah bobot kepentingan suatu token yang muncul dalam suatu dokumen dalam seluruh dokumen-dokumen yang ada, dimana semakin sering muncul suatu token dalam suatu dokumen dan semakin banyak dokumen yang memilikinya akan semakin tidak penting karena token tersebut bersifat sangat umum.
- tf(t,d) adalah jumlah token yang terdapat pada satu dokumen idf(t,D) adalah *inverse document frequency*, yaitu log dari jumlah seluruh dokumen dibanding dengan jumlah seluruh dokumen dimana token tersebut muncul.

Sedangkan TF-Idf yang telah dinormaliasi dapat digunakan Persamaan (2) sebagai berikut:

$$\text{TFIDF-Norm}_{(j)} = \text{TF-IDF}_{(j)}/\text{SQRT}(\text{TF-IDF}_{(j)}^2) \ldots\ldots (2)$$

5) Features Selection

Features selection bertujuan untuk mengefisienkan proses klasifikasi dengan cara mengurangi jumlah token yang sudah diboboti sehingga analisis yang dilakukan menjadi lebih sedikit. Cara yang dilakukan menurut Patil [9]:

1. Mengidentifikasi bagian data hanya yang berkontribusi sentimen positif dan negatif saja.
2. Mengambil data-data yang memiliki bobot tertinggi saja beberapa bagian tertentu.

Pada penelitian ini tahap ini dilakukan dengan mengambil x persen dari seluruh token untuk mewakili atribut yang akan digunakan dalam proses klasifikasi.

6) Klasifikasi Sistem

Klasifikasi sentimen biasanya hanya menggunakan 2 *class*, yaitu *class* positif dan negatif saja. *Class* netral bisa dianggap masuk ke dalam *class* negatif [10]. Hal ini juga mempermudah pengklasifikasian data. Berbagai algoritma *supervised learning* yang dapat digunakan adalah algoritma Naïve Bayes, Rhoccio Feedback, atau Support Vector Machine. Troussas telah menggunakan algoritma Naïve Bayes pada klasifikasi status Facebook untuk pembelajaran bahasa dan berhasil dengan baik [4].

Klasifikasi sentimen berbeda dengan klasifikasi teks biasa, klasifikasi teks biasanya mengkategorikan teks menjadi topik tertentu seperti misalnya olahraga, musik, berita, atau hal lain, yang memiliki fitur token berupa kata-kata yang berelasi dengan topik yang akan diklasifikasikan. Klasifikasi sentimen menggunakan token yang mengidentifikasikan tentang opini positif atau negatif. Kata-kata seperti bagus, luar biasa, jelek, cantik, keren, mantap, buruk akan justru menjadi fitur tokennya [10].

7) Algoritma Naive Bayes

Algoritma Naive Bayes merupakan algoritma klasifikasi berdasarkan probabilitas dalam statistik yang dikemukakan oleh Thomas Bayes yang memprediksi peluang di masa depan berdasarkan peluang di masa sebelumnya (teorema Bayes). Metode ini kemudian dikombinasikan dengan "*naive*" dimana kondisi antar atribut saling bebas tidak berhubungan satu sama lain. Dalam dataset, setiap dataset memiliki atribut-atribut dan 1 (satu) label *class*, maka probabilitas suatu data masuk ke dalam suatu *label class* dapat didefinisikan pada dengan langkah-langkah sebagai berikut [11]:

1. Diketahui D adalah data pelatihan dan label *class*-nya. Setiap data direpresentasikan dalam bantuk n dimensi vektor atribut X = (x1, x2,... , xn)
2. Misalkan terdapat m jumlah class, C1, C2,... , Cm. Metode Naive Bayes akan memprediksi apakah X masuk dalam class yang memiliki nilai posterior probabilitas tertinggi. Naive Bayes akan memprediksi X akan masuk ke dalam kelas Ci jika dan hanya jika:P(Ci|X) > P(Cj|X) for $1 \leq j \leq m$, j <>i. Kemudian akan dimaksimumkan P(Ci|X).*Class* terbanyak dari Ci disebut dengan *maximum posteriori hypothesis* yang dihitung menggunakan Persamaan (3).

$$P(C_i|\boldsymbol{X}) = \frac{P(\boldsymbol{X}|C_i)P(C_i)}{P(\boldsymbol{X})}.$$

.....(3)

3. Karena P(X) bersifat tetap untuk semua *class*, maka hanya P(X | Ci) P(Ci) yang harus dimaksimumkan. Jika *prior probabilities* dari *class* tidak diketahui, maka secara umum diasumsikan semua *class* sama P(C1) = P(C2) = ⋯ = P(Cm). Perlu diingat bahwa *prior probabilities* dari *class* diestimasi dengan P(Ci) = |Ci,D| / |D|, dimana |Ci,D| adalah jumlah *training data* yang termasuk dalam *class* Ci di dataset D.
4. Untuk dataset yang memiliki banyak atribut maka kompleksitas komputasi akan sangat tinggi, sehingga perlu direduksi dengan cara mengasumsikan semua kondisi *class* bersifat saling bebas (*independence*). Hal ini menganggap bahwa nilai antar atribut saling tidak mempengaruhi satu sama lain, sehingga dapat didefinisikan :
   Dimana probabilitas P(x1| Ci), P(x2| Ci),... , P(xn| Ci) dapat diperoleh dengan mudah dari data training. Dimana *xk* adalah nilai yang ada di atribut*Ak* untuk data*X*. Untuk setiap atribut, kita harus melihat apakah nilai atribut bersifat kategorikal atau nilai kontinu.
   - Jika*Ak* bersifat kategorikal, maka*P(xk|Ci)* adalah jumlah data xk yang memiliki *classCi* di data training *D* dibagi dengan |*Ci*,*D*|, jumlah seluruh data *classCi* didata training *D*.
   - Jika Ak bersifat kontinu, seperti misalnya data umur, data angka lainnya, yang tidak bisa dikategorikan, maka data tersebut harus dibuat dalam rentang nilai, menggunakan Gaussian Distribution dengan Persamaan (4) dan (5).

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

.....(4)

Sehingga diperoleh P(xk| Ci) = g(xk, μCi, σCi). .....(5)

5. Untuk memprediksi label *class* untuk data X, *P(X|Ci)P(Ci)* maka prediksi dilakukan untuk setiap *classCi*. Metode Naive Bayes akan memprediksi *class* untuk*X* adalah Ci jika dan hanya jika (Persamaan 6). *P(X|Ci)P(Ci) > P(X|Cj)P(Cj)* for $1 \leq j \leq m$, $j <> i$.....(6) Dalam arti prediksi terhadap *class* dengan probabilitas terbesar dihitung dengan Persamaan (7).

$$P(\boldsymbol{X}|C_i) = \prod_{k=1}^{n} P(x_k|C_i)$$
$$= P(x_1|C_i) \times P(x_2|C_i) \times \cdots \times P(x_n|C_i).$$

.....(7)

8) Confussion Matrix

Untuk melakukan pengujian terhadap sistem, dilakukan evaluasi akurasi sistem dalam mengklasifikasikan sentimen pada dataset dengan menggunakan *confusion matrix* [12] seperti pada Tabel 1 berikut.

TABEL I.
CONFUSION MATRIX

		Class Hasil Prediksi	
		Negatif	Positif
Class sebenarnya	Negatif	True Negatif (TN)	False Negatif (FN)
	Positif	False Positif (FP)	True Positif (TP)

Keterangan:

- True negatif = jumlah data negatif yang benar dikategorikan sebagai class negatif
- False negatif = jumlah data negatif yang dikategorikan sebagai class positif
- False positif = jumlah data positif yang dikategorikan sebagai class negatif
- True positif = jumlah data positif yang benar dikategorikan sebagai class positif

Dari confusion matrix pada Tabel I dapat dilakukan perhitungan lebih lanjut untuk mendapatkan tingkat akurasi (*accuracy*), *recall*, *precision* dan *f-measure* dengan Persamaan (8) – (13).

Accuracy = (TN + TP) / (TN + FP + FN + TP) ..... (8)

Recall / True Positive Rate = TP / (FP + TP) ..... (9)

False Positive Rate = FN / (TN + FN) ..... (10)

Specificiy / True Negative = FP / (FP + TP) ..... (11)

Precision = TP / (FN + TP) ..... (12)

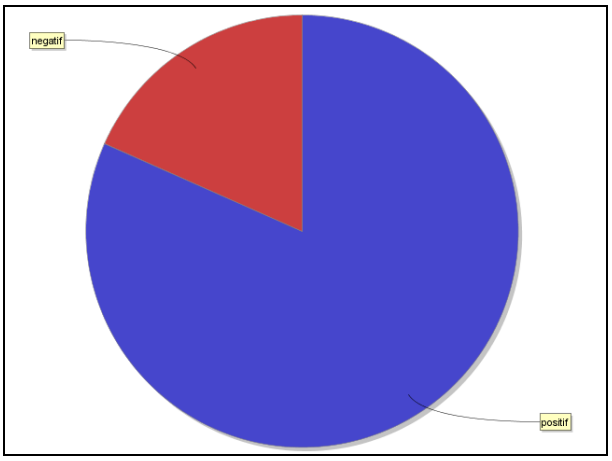
F-Measure = 2 * TP / (2 * TP + FP + FN ..... (13)

III. HASIL DAN PEMBAHASAN

A. *Profil Data*

Data diperoleh dari penelitian Rachmat C & Lukito (2015) dalam format .csv yang berupa data status dan komentar Pemilu Presiden tahun 2014 berjumlah 300 komentar dengan detail keterangan sebagai berikut: data status diperoleh dari 15 status Bpk Joko Widodo dan 15 status Bpk Prabowo Subianto, dimana masing-masing diambil 10 komentar secara acak, sehingga terbentuk 15 x 10 = 150 komentar. Dengan demikian 300 komentar berasal dari total 150 komentar status Bpk Joko Widodo ditambah dengan 150 komentar status Bpk Prabowo Subianto. Profil data dapat dilihat pada gambar 1 berikut ini. Data akan dianalisis menjadi 2 profil:

1. Data profil I memiliki profil positif berjumlah 1765, negatif berjumlah 258, dan netral berjumlah 77 data. Dari data ini label netral akan digabungkan ke dalam label negatif sehingga negatif akan berjumlah 335 data.
2. Data profil II memiliki profil positif berjumlah 1765, negatif berjumlah 258, dan netral berjumlah 77 data. Dari data ini label netral akan dihapus dan tidak digunakan.



Gambar 1. Profil Data

B. *Implementasi Sistem*

Arsitektur Sistem yang dibangun dibagi menjadi dua bagian, bagian pelatihan dan bagian pengujian. Bagian pelatihan memiliki urutan bagian proses sebagai berikut: input data latih, cleaning, tokenisasi, case folding, konversi, stopword removal, stemming, token weightening TF-IDF, dan penyimpanan data latih. Bagian pengujian memiliki urutan bagian proses sebagai berikut: input data uji, cleaning, tokenisasi, case folding, konversi, stopword

removal, stemming, token weightening TF-IDF, dan klasifikasi menggunakan Naïve Bayes.

Implementasi metode dilakukan sebagai berikut: tokenisasi diimplementasikan dengan cara memecah string input komentar baik satu atau lebih per spasi, sehingga menjadi banyak kata (token) dalam bentuk array of string. Penanganan karakter khusus dilakukan dengan cara memeriksa satu per satu karakter yang termasuk dalam karakter khusus untuk dibuang karena karakter-karakter tersebut tidak berguna. Karakter yang termasuk dalam karakter khusus adalah '#', '?', '.', ',', ':', '!', '%', '=', dan '@'. Penanganan konversi smile dilakukan terhadap berbagai karakter pembentuk smile yang bersifat umum seperti ':)', '(y)', '~_~', '(Y)', ':*', 'like', 'Like', '=))', ':D', 'YES', 'yes', dan 'Yes', yang kemudian dikonversi menjadi kata-kata sesuai dengan jenis karakter smilenya seperti pada Tabel 2 berikut:

TABEL 2.
KONVERSI KARAKTER KHUSUS

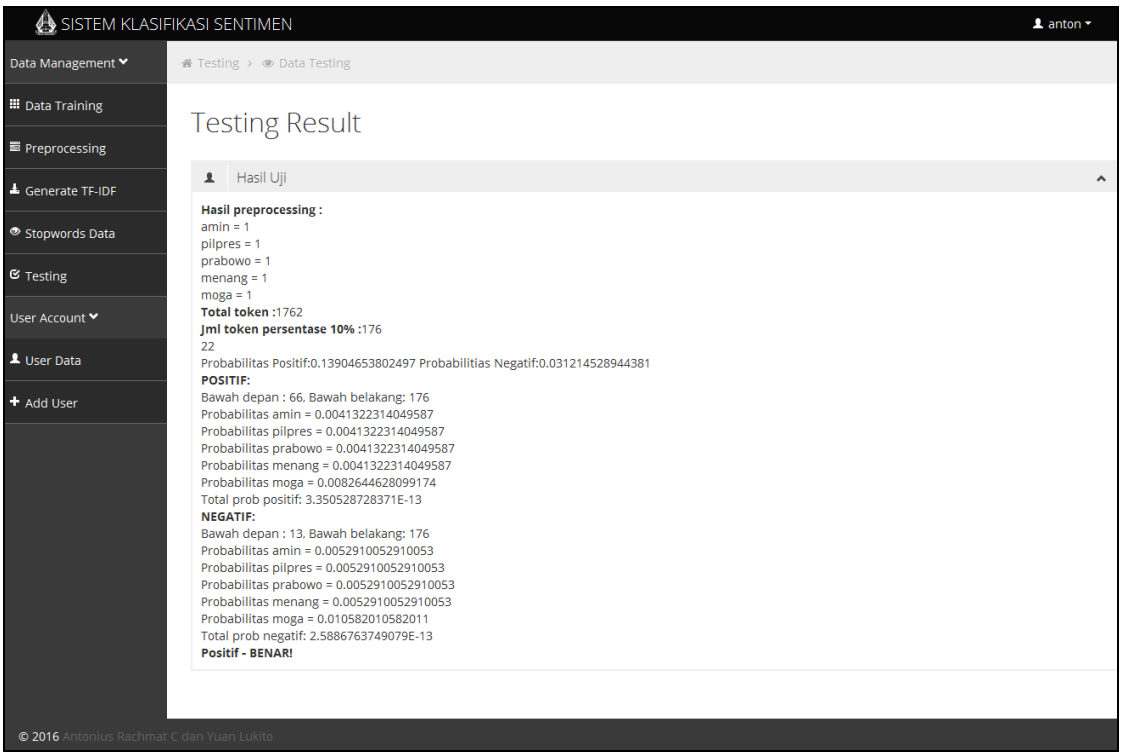
No.	Karakter Smile	Konversi
1	:)	Senyum
2	(y)	Suka
3	~_~	Netral
4	(Y)	Suka
5	:*	Cium
6	=))	Senyum
7	:D	Senyum
8	Like atau like	Suka
9	YES, Yes, atau yes	Suka

Implementasi penggunaan stopwords dilakukan dengan menggunakan kamus kata-kata yang harus dibuang. Jika token berada di dalam daftar kata yang masuk dalam tabel stopword, maka token tersebut tidak diikutsertakan dalam proses selanjutnya, alias dibuang. Data diperoleh dalam bentuk teks file dan kemudian dimasukkan ke dalam tabel basis data. Jumlah total stopwords adalah 1.326 kata. Implementasi stemming dilakukan dengan menggunakan library open source Sastrawi Stemming. Kemampuan Sastrawi stemming tergolong baik dalam mengambil kata dasar dari setiap kata-kata yang menjadi data masukan. Instalasi Sastrawi harus menggunakan PHPComposer sehingga library Sastrawi dapat terdownload dan disimpan pada folder vendor pada struktur folder CodeIgniter.

Implementasi Pembobotan TF-IDF dilakukan pada data sesuai dengan rumus 1 yang telah disebutkan sebelumnya. Setelah itu dilakukan penghitungan normalisasi TF-IDF sesuai dengan rumus 2. Implementasi Feature Selection dilakukan dengan cara mengambil sejumlah token sesuai dengan aturan berikut:

- Token terlebih dahulu diurutkan secara descending berdasarkan bobot TF-IDF dari besar ke kecil
- Kemudian diambil sejumlah % token sesuai dengan yang diinputkan oleh pengguna. Misalnya 10%, 30%, 60% atau 100%. Dalam hal ini jika pengguna mengambil 100% token berarti semua token dalam data tabel token tfidf terambil semua untuk digunakan dalam tahapan klasifikasi selanjutnya.

Bagian akhir dari implementasi adalah dilakukan klasifikasi data menggunakan algoritma Naive Bayes. Implementasinya dilakukan sesuai Persamaan (3). Implementasi sistem ini dapat dilihat pada gambar 3.



Gambar 2. Hasil Implementasi Sistem

C. *Pengujian dan Analisis Sistem*

Pada tahap ini dilakukan pengujian terhadap sistem klasifikasi sentimen yang telah diimplementasikan pada tahap pertama. Pengujian dilakukan menggunakan metode k-folding, di mana data pengujian dibagi ke dalam k = 2,3,5, dan 10, dan untuk masing-masing k tersebut dilakukan pengujian juga terhadap feature selection 10%, 30%, 60%, dan 100%. Harapan dari pengujian ini adalah menguji seberapa feature selection yang sudah dapat mewakili sistem dalam melakukan klasifikasi yang akurat, sekaligus menguji akurasi sistem dalam mengklasifikasikan dalam bentuk Accuracy, Sensitifity, Specificity, Precision, dan F-Measure.

Proses klasifikasi dan pengujian dilakukan menggunakan RapidMiner dengan konfigurasi sebagai berikut:

1. Komponen membaca database
2. Konversi nilai basis data dari nominal ke teks
3. Proses dokumen terdiri atas:
   a. Ekstrak konten teks
      Ekstrak konten teks akan mengambil data teks dari sumber data dan mengambil semua data teksnya (ASCII).
   b. *Transform case*
      Transform case akan mengubah data teks menjadi huruf kecil semua.
   c. *Replace token*
      Pada tahap ini semua token-token seperti emoticon dan karakter-karakter khusus lainnya akan diubah menjadi huruf yang bermakna.
   d. Tokenisasi
      Pada tahap ini akan dilakukan pemotongan kalimat menjadi token-token kata bermakna.
   e. Stopwords bahasa Indonesia
      Pada tahap ini akan dibuang tokent-token yang tidak penting dan masuk dalam kategori stopwords seperti yang telah didefinisikan.
   f. Filter token
      Pada tahap ini akan dilakukan pengambilan token yang memenuhi syarat seperti minimal huruf dalam kata minimal 4 huruf.
   g. Stemming (opsional)
      Pada tahap ini akan dilakukan pengambilan dan pengubahan kata dasar menggunakan algoritma Sastrawi Stemming
   h. Set Role label
      Pada bagian ini akan diset mana field yang berupa label untuk pelatihan sistem.
   i. Implementasi Naïve Bayes
      Pada tahap ini akan dilakukan implementasi klasifikasi Naive Bayes sehingga dapat digunakan untuk menentukan setimen politik.
   j. K-Fold Validasi

Pada tahap ini akan dilakukan validasi uji hasil klasifikasi yang telah dihasilkan menggunakan metode pembagian data uji K-Fold dengan k=10.

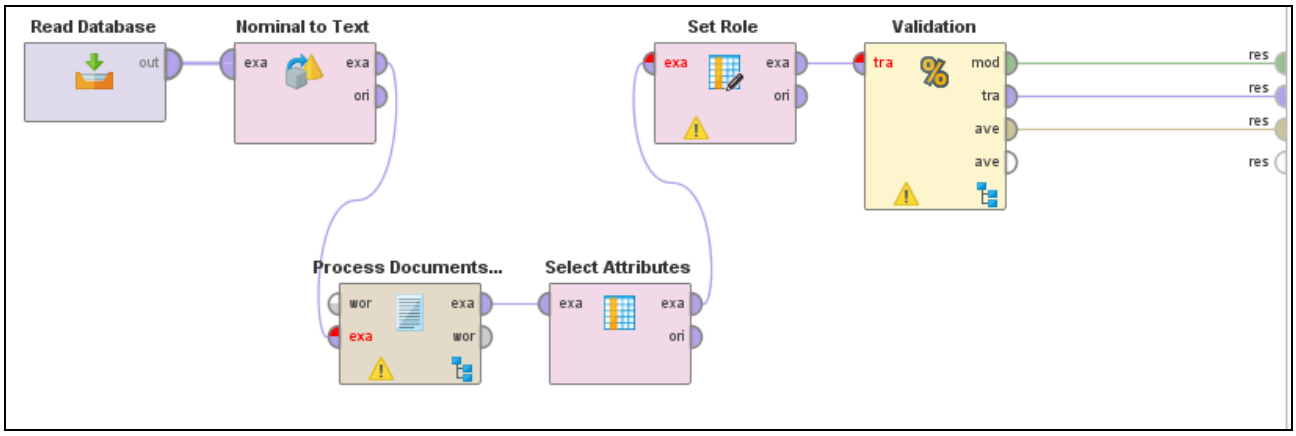
k. Hasil

Pada tahap ini akan diperoleh hasil akhir klasifikasi dan pengujian akurasi sistem.

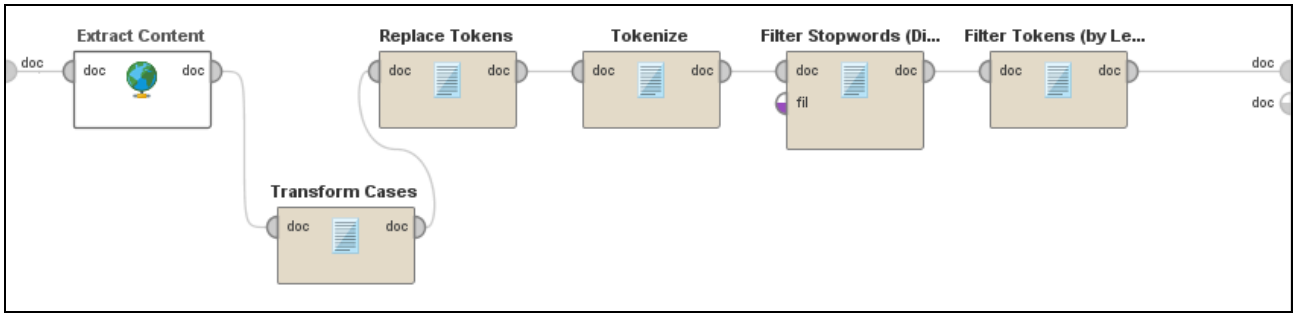
TABEL 3.
PARAMETER PENGUJIAN

No.	K-Fold	Feature Selection
1.	2	10%
2.	2	30%
3.	2	60%
4.	2	100%
5.	3	10%
6.	3	30%
7.	3	60%
8.	3	100%
9.	5	10%
10.	5	30%
11.	5	60%
12.	5	100%
13.	10	10%
14.	10	30%
15.	10	60%
16.	10	100%

Berdasarkan parameter pengujian pada Tabel 3 di atas, maka Tabel 4 adalah hasil pengujian yang telah dilakukan terhadap sistem klasifikasi sentimen komentar politik dari Facebook menggunakan metode Naïve Bayes pada RapidMiner.



Gambar 3. Proses Utama Pada RapidMiner



Gambar 4. Pemrosesan Documents Pada RapidMiner

TABEL 4.
HASIL PENGUJIAN SISTEM MENGGUNAKAN K-FOLD VALIDATION

No.	K-Fold	Feature Selection	Accuracy	Sensitifity	Specificity	Precision	F-Measure
1.	2	10%	0,8066666667	0,9676482372	0,1060606061	0,8265811717	0,8907091467
2.	2	30%	0,813333	0,979734	0,090909	0,825576	0,895217
3.	2	60%	0,816667	0,987914	0,068182	0,823745	0,897629
4.	2	100%	0,82	0,99182	0,068182	0,824193	0,899594

5.	3	10%	0,8	0,941145	0,16205534	0,8333804	0,88255095
6.	3	30%	0,816667	0,98385135	0,07378129	0,84319045	0,89675282
7.	3	60%	0,823333	0,99192568	0,07378129	0,84481293	0,90082358
8.	3	100%	0,826667	0,995671	0,07378129	0,84532828	0,90270093
9.	5	10%	0,8	0,8	0,8	0,8	0,8
10.	5	30%	0,941145	0,941145	0,941145	0,941145	0,941145
11.	5	60%	0,16205534	0,16205534	0,16205534	0,16205534	0,16205534
12.	5	100%	0,8333804	0,8333804	0,8333804	0,8333804	0,8333804
13.	10	10%	0,88255095	0,88255095	0,88255095	0,88255095	0,88255095
14.	10	30%	0,816667	0,816667	0,816667	0,816667	0,816667
15.	10	60%	0.806667	0.991238	0.113778	0.82913	0.900222
16.	10	100%	0,823333	0,995238	0,077778	0,824253	0,898921
Rata-rata			**0,833333**	**0,995238**	**0,077778**	**0,834253**	**0,898921**
Persentase			**83.3%**	**99.5%**	**7.8%**	**83.4%**	**89.8%**

Confusion Matrix pada pengujian Data Profil I menggunakan Naive Bayes dapat dilihat pada gambar 5 sebagai berikut:

accuracy: 80.00% +/- 2.40% (mikro: 80.00%)

	true positif	true negatif	class precision
pred. positif	1626	281	85.26%
pred. negatif	139	54	27.98%
class recall	92.12%	16.12%	

Gambar 5. Confusion Matrix Data Profil I

Confusion Matrix pada pengujian Data Profil II menggunakan Naive Bayes dapat dilihat pada gambar 6 sebagai berikut:

accuracy: 83.29% +/- 1.90% (mikro: 83.29%)

	true positif	true negatif	class precision
pred. positif	1650	223	88.09%
pred. negatif	115	35	23.33%
class recall	93.48%	13.57%	

Gambar 6. Confusion Matrix Data Profil II

Dari tabel Confussion Matrix tersebut dapat diperoleh peningkatan akurasi dari 80% menjadi 83%, yang berarti terjadi peningkatan sebesar 3% menggunakan data pada profil II.

IV. KESIMPULAN

Dari implementasi dan pengujian yang telah dilakukan didapat kesimpulan sebagai berikut:

a. Penelitian ini telah mampu mengembangkan sistem klasifikasi sentimen berdasarkan data Pemilu Presiden Indonesia 2014 dari Facebook Page menggunakan Naive Bayes.
b. Dalam implementasinya, sistem klasifikasi yang berasal dari data media sosial berbahasa Indonesia membutuhkan preprosesing terutama dalam hal konversi singkatan dan emoticon.
c. Algoritma Naive Bayes mampu mengklasifikasikan sentimen dengan tingkat akurasi rata-rata tertinggi 82%.
d. Peningkatan hasil akurasi terjadi karena data netral tidak digunakan dalam sistem.

V. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Lembaga Penelitian dan Pengabdian Masyarakat Universitas Kristen Duta Wacana Yogyakarta atas bantuan dana dan dukungan penelitian yang telah diberikan melalui kontrak penelitian No. 067/D.01/LPPM.2016.

VI. DAFTAR PUSTAKA

[1] F. Amirullah, S. Komp and Y. Nurhadryani, "Campaign 2 . 0 : An Analyze of the Utilization Social," *2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2013.

[2] D. A. Ramadhan, Y. Nurhadryani and I. Hermadi, "Campaign 2.0: Analysis of social media utilization in 2014 Jakarta legislative election," *2014 International Conference on Advanced Computer Science and Information System*, 2014.

[3] A. Rachmat C. and Y. Lukito, "Implementasi Crowdsourced Labelling Berbasis Web," *Ultima InfoSys*, vol. 6, no. 2, 2015.

[4] C. Troussas, M. Virvou, K. J. Espinosa, K. Llaguno and J. Caro, "Sentiment analysis of Facebook statuses using Naive Bayes classifier for language learning," *IISA 2013*, 2013.

[5] S. M. Weiss, N. Indurkhya, T. Zhang and F. Damerau , Text mining: Predictive Methods for Analyzing Unstructured Information, New York: Springer, 2005.

[6] A. Akilan, "Text mining: Challenges and future directions," *2015 2Nd International Conference*, 2015.

[7] C. D. Manning, P. Raghavan and H. Schütze, Introduction to information retrieval, New York: Cambridge University Press, 2008.

[8] K. V. Ghag and K. Shah, "Comparative analysis of effect of stopwords removal on sentiment," *2015 International Conference on Computer, Communication and Control (IC4)*, 2015.

[9] G. Patil, V. Galande, V. Kekan and K. Dange, "Sentiment Analysis Using Support Vector Machine," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, no. 1, pp. 2607-2612, 2014.

[10] B. Liu, Sentiment analysis: mining opinions, sentiments, and emotions, New York: Cambridge University Press, 2015.

[11] J. Han, M. Kamber and J. Pei, Classification: basic concepts. In Data mining Concepts and techniques, Amsterdam: Elsevier, 2012.

[12] H. Hamilton, "www2.cs.uregina.ca," Computer Science Uregina, 2009. [Online]. Available: http://www2.cs.uregina.ca/~dbd/cs831/notes/confusion_matrix/confusion_matrix.html. [Accessed 4 February 2016].

[13] B. Pang, L. Lee and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - EMNLP '02*, vol. 10, pp. 79-86, 2002.

[14] N. Zainuddin and A. Selamat, "Sentiment analysis using Support Vector Machine," *2014 International Conference on Computer, Communications, and Control Technology (I4CT)*.

[15] V. K. Verma, M. Ranjan and P. Mishra, "Text mining and information professionals: Role, issues and challenges," *Emerging Trends and Technologies in Libraries and Information Services (ETTLIS)*, 2015.