Jurnal Ilmu Komputer Agri-Informatika
Volume 7 Nomor 1 halaman 31 – 40
eISSN: 2654-9735, pISSN: 2089-6026

Tersedia secara *online* di:
http://journal.ipb.ac.id/index.php/jika

Deteksi *Spam* pada Twitter Menggunakan Algoritme Naïve Bayes

Spam Detection on Twitter using Naïve Bayes Algorithm

ANDITA WAHYUNINGTYAS[1]*, IMAS SUKAESIH SITANGGANG[1], HUSNUL KHOTIMAH[1]

Abstrak

Di era berkembangnya penggunaan Internet, Twitter merupakan salah satu layanan jejaring sosial yang sering digunakan sebagai alat komunikasi yang saling menghubungkan antar pengguna. Selain itu Twitter juga dimanfaatkan sebagai media untuk promosi, kampanye politik, dan sarana protes. Twitter dihadapkan pada berbagai masalah seperti gangguan privasi pengguna dan *spam* pada Twitter. Dengan adanya masalah *spam*, perlu dilakukan klasifikasi untuk *tweet spam* dan bukan *spam*. Penelitian ini bertujuan untuk mendeteksi *tweet spam* dan bukan *spam*. Hal tersebut dapat dilakukan dengan klasifikasi, terdapat berbagai macam metode klasifikasi, salah satu metode dalam *data mining* untuk mengklasifikasikan *spam* dan bukan *spam* adalah Naïve Bayes. Naïve Bayes banyak digunakan karena kesederhanaan algoritme dan mudah untuk diimplementasikan. Penelitian ini mengumpulkan data *spam* dari Twitter dengan mengidentifikasi terlebih dahulu akun yang diduga sebagai *spammer*. Penelitian ini menggunakan 70% data latih dan 30% data uji dengan metode klasifikasi Naïve Bayes. Data Twitter yang diperoleh merupakan data teks yang masih banyak mengandung kata-kata yang tidak baku, sehingga dibutuhkan tahap praproses, tahap yang dilakukan adalah *tokenizing*, *filtering*, normalisasi kata, *stemming*. Akurasi hasil klasifikasi *tweet spam* dan bukan *spam* adalah 95.57%.

Kata Kunci**:** klasifikasi spam, Naïve Bayes, text mining, Twitter.

Abstract

In this era of the Internet, Twitter is one of the social networking services that is often used as a communication tool between users to connect with each other. Twitter is also used as a tool for promotions, political campaigns, and protests. Twitter faces with various problems such as interference with user privacy and spam messages. With the existence of spam problem, classification of spam and non-spam tweet needs to be done. This research aims to detect spam and non-spam tweet, which can be done by the classification task. One of the methods in data mining to classify spam and non-spam is Naïve Bayes algorithm. Naïve Bayes algorithm is widely used because of its simplicity and easy to implement. This research collects spam data from Twitter by identifying accounts that are suspected as spammers. The data are split to 70% for training data and 30% for test data. The data are text data that still contains many non-standard words, so a preprocessing stage is needed. The stage which is done are tokenizing, filtering, text normalization, and stemming. The accuracy of the classification of spam tweet and non-spam tweet is 95.57%.

*Keywords**:** Naïve Bayes, spam classification, text mining, Twitter.*

PENDAHULUAN

Di era berkembangnya penggunaan Internet, penggunaan Internet memberikan dampak langsung terhadap penggunaan media sosial. Pada tahun 2013, pengguna Internet di Indonesia mencapai 63 juta orang, 95 persen di antaranya menggunakan Internet untuk mengakses media sosial seperti Facebook dan Twitter (Kemenkominfo 2013).

Twitter merupakan salah satu layanan jejaring sosial yang sering digunakan sebagai alat komunikasi. Selain itu, Twitter juga dimanfaatkan sebagai media untuk promosi, kampanye

[1]Andita Wahyuningtyas, Departemen Ilmu Komputer IPB, 0251-8625584;
*Penulis Korespondensi: Tel/Faks: 0251-8625584; Surel: andita.wahyuningtyas@yahoo.com

politik ataupun sarana protes. Fitur-fitur pada Twitter yang dapat dimanfaatkan antara lain seperti *tweet*, *mention*, *reply*, *retweet*, *hashtag*, *following* dan *follower*, *direct massage*, dan *search*. *Tweet* merupakan kegiatan *post* dan *posting* yang dilakukan oleh pengguna Twitter, jika di dalam *tweet* terdapat *@username* hal tersebut disebut dengan *mention*, *mention* dapat terlihat oleh pengguna yang di-*mention* walaupun bukan sebagai *follower* dari pengirim. Dengan adanya fitur-fitur yang ada pada Twitter, penggunaan Twitter semakin berkembang, sehingga Twitter dihadapkan pada berbagai masalah seperti munculnya *spam*. *Spam* pada Twitter adalah konten yang dikirim oleh akun palsu yang dibuat oleh *spammer* atau akun yang *username* dan *password*-nya telah dicuri oleh *spammer* (McCord dan Chuah 2011).

Tweet terbatas hanya 140 karakter, keterbatasan tersebut memaksa pengguna Twitter untuk melakukan penyingkatan kata. Hal tersebut menyebabkan isi *tweet* mengandung banyak kata yang tidak baku. Hal tersebut menjadi tantangan untuk mengolah data *tweet* menjadi kata-kata yang baku. Untuk mengolah data *tweet* menjadi baku dibutuhkan beberapa tahapan praproses, yaitu, tokenisasi, normalisasi kata, *filtering*, dan *stemming*. Tokenisasi adalah proses pemotongan kata menjadi bagian-bagian kecil atau disebut *token*. Pada tokenisasi akan membuang karakter, seperti tanda baca dan angka. Normalisasi kata merupakan proses untuk penggantian kata yang tidak baku menjadi baku dan menghilangkan karakter berulang (Aziz 2013). *Filtering* adalah langkah untuk menghilangkan kata-kata yang jika dihilangkan, data *tweet* masih memiliki makna. *Stemming* adalah langkah untuk membuang imbuhan dan akhiran yang terdapat pada *token*.

Palupiningsih (2011) dan Makhtidi (2012) melakukan penelitian untuk menganalisis pola pada SMS untuk memprediksi apakah SMS berindikasi sebagai *spam* atau tidak. Dalam penelitian tersebut, Palupiningsih (2011) menggunakan algoritme Naïve Bayes dan berhasil mengklasifikasikan data *spam* dan bukan *spam* dengan akurasi sebesar 80.93%. McCord dan Chuah (2011) melakukan penelitian untuk klasifikasi *spam* pada Twitter. Penelitian ini menggunakan fitur yang diambil dari masing-masing akun pengguna Twitter. Fitur yang digunakan adalah fitur *user-based* dan fitur *content-based*, yang termasuk kategori fitur *user-based* adalah hubungan pengguna seperti orang-orang yang pengguna ikuti dan orang-orang yang mengikuti pengguna atau prilaku pengguna seperti periode dan frekuensi pengguna mengirim *tweet.* Tahap awal penelitian ini adalah memilih secara acak 1000 akun Twitter dan memeberi label *spam* dan *non spam* secara manual, setiap akun dievaluasi 20, 50, 100 *tweet* terbaru yang di-*posting* oleh pengguna. Menggunakan algoritme Random Forest dengan presisi sebesar 95.7% dan *F-measure* sebesar 95.7%. Prototipe deteksi akun spam pada Twitter dibangun oleh Wang (2012). Studi ini menunjukkan bahwa bayesian classifier menunjukkan kinerja yang baik dibandingkan classifier lainnya dalam mendeteksi akun spam pada Twitter. Selain Bayesian *classifier*, algoritme *machine learning* yang lain yang banyak digunakan dalam deteksi spam pada Twitter adalah Random Forest (Chakraborty *et al*. 2016), Support Vector Machine dan Neural Network (Gupta *et al.* 2018).

Dari permasalahan berkembangnya kemunculan *spam* pada Twitter, maka dibutuhkan penelitian untuk menganalisis pola *tweet* untuk memprediksi apakah *tweet* terindikasi sebagai *spam* atau bukan *spam*. Dalam mengklasifikasikan terdapat berbagai macam algoritme, salah satunya adalah Naïve Bayes, Naïve Bayes banyak digunakan karena kesederhanaan algoritme dan mudah untuk diimplementasikan (Metsis *et al*. 2006). Oleh karena itu, penelitian ini dibutuhkan untuk membangun model klasifikasi untuk mendeteksi *spam* dan bukan *spam* dengan menggunakan algoritme Naïve Bayes.

Tujuan penelitian ini adalah 1) membangun model klasifikasi menggunakan algoritme Naïve Bayes dalam mendeteksi *tweet* ke dalam kelas *spam* dan bukan *spam*, dan 2) menganalisis isi *tweet* untuk mengidentifikasi kata yang digunakan sebagai *spam*. Penelitian ini diharapkan dapat membantu pengguna Twitter untuk mengklasifikasikan *tweet spam* dan bukan *spam*, sehingga *tweet spam* tidak muncul pada akun pengguna

METODE

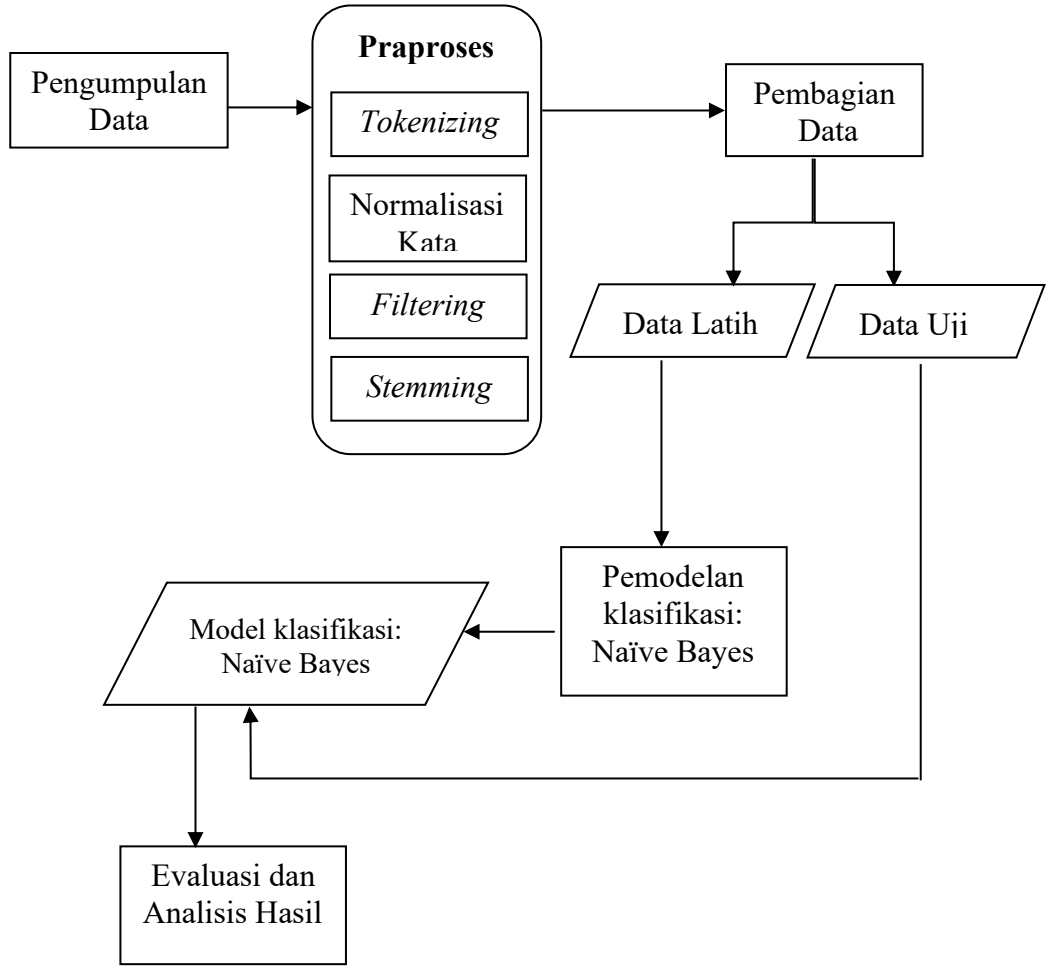
Data Penelitian

Data yang digunakan dalam penelitian ini adalah data *tweet,* yang didapat dari pengguna Twitter. Penelitian ini menggunakan 1000 data *tweet* yang diambil dari 25 akun *spammer* dan 25 akun bukan *spammer* yang telah ditentukan sebelumnya. Dari masing-masing akun diambil 20 *tweet* terbaru. Pengambilan data *tweet* diperoleh dalam rentang waktu Maret 2015-Agustus 2015. Data kedua yang dibutuhkan dalam penelitian adalah data *stopwords* yang digunakan untuk kebutuhan *filtering*, data *stopwords* dalam bahasa Indonesia yang digunakan dalam penelitian ini diperoleh dari (Tala 2003) yang berjumlah 759 kata. Data *stopwords* disimpan dalam format *Comma Separated Value* (CSV).

Data yang dibutuhkan selanjutnya adalah data kata dasar yang digunakan untuk kebutuhan *stemming*, Data kata dasar pada penelitian ini disimpan di dalam basis data dalam format *Structure Query Language* (SQL) yang berjumlah 28526 kata dasar. Data terakhir yang dibutuhkan dalam penelitian ini adalah data kata baku untuk normalisasi singkatan, data kata baku diperoleh dari penelitian (Aziz 2013) yang berjumlah 3719 kata baku, data kata baku disimpan di dalam basis data dalam format *Structure Query Language* (SQL).

Tahapan Penelitian

Penelitian ini terdiri atas beberapa tahapan, yaitu tahap pengumpulan data, tahap praproses, tahap pembagian data, tahap pemodelan klasifikasi, tahap pengujian, dan evaluasi hasil. Tahapan penelitian dapat dilihat pada Gambar 1.



Gambar 1 Tahapan penelitian

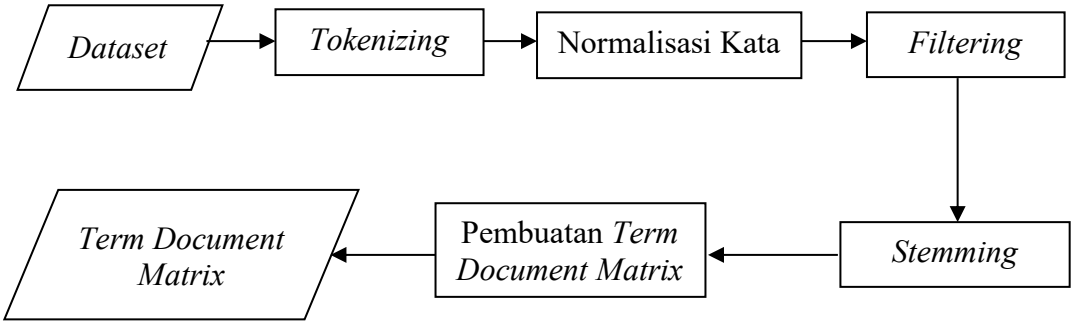
Pengumpulan Data

Tahap pengumpulan data dibutuhkan untuk mendapatkan isi *tweet* yang diambil dari Twitter, data diperoleh menggunakan Twitter API (*Application Programming Interface*). Data yang didapatkan masih berupa data campuran yaitu *spam* dan bukan *spam*. Data yang digunakan sebanyak 1000 *tweet* yang diambil dari 25 akun *spammer* dan 25 akun bukan *spammer*. Masing-masing akun diambil 20 *tweet* terbaru. Langkah yang dilakukan pada tahap pengumpulan data yaitu:

1 Mengumpulkan akun Twitter
2 Memberi label untuk akun *spammer* dan bukan *spammer*. Akun *spammer* dan akun bukan *spammer* ditentukan secara manual, akun yang dikategorikan sebagai akun *spammer* adalah akun yang mengirim *tweet* ke banyak pengguna Twitter dengan isi *tweet* yang sama atau hampir sama dalam waktu yang berdekatan.
3 Mengambil data *tweet* dari Twitter API dengan menggunakan *package* "twitteR" pada R dan menggunakan fungsi *userTimeline*.
4 Setiap data *tweet* diberi label sesuai dengan label akun, untuk akun *spammer* diberi label *spam* sedangkan untuk akun bukan *spammer* diberi label bukan *spam*.

Praproses

Setelah dilakukan tahap pengumpulan data berupa isi *tweet* pengguna Twitter, tahap selanjutnya adalah tahap praproses. Tahap praproses dibutuhkan karena data *tweet* yang tidak baku. Tahapan praproses dapat dilihat pada Gambar 2.



Gambar 10 Tahapan praproses data *tweet.*

- *Tokenizing:* Tahap pembuatan tokenisasi proses pertama yang dilakukan adalah membuat semua huruf pada *tweet* menjadi huruf kecil, proses selanjutnya adalah menghapus *mention*, URL, dan *hashtag* yang ada pada *tweet*, kemudian proses selanjutnya adalah penghapusan tanda baca dan angka.
- Normalisasi Kata: Normalisasi merupakan proses untuk penggantian kata yang tidak baku menjadi baku dan menghilangkan karakter berulang (Aziz 2013).
- *Filtering:* Pada tahap *filtering*, dilakukan penghapusan *stopwords* pada data *tweet*. Pengapusan *stopwords* dilakukan untuk menghilangkan kata- kata yang jika dihilangkan, data *tweet* masih memiliki makna. Penghapusan *stopwords* dilakukan pada *tweet* berbahasa Indonesia dan berbahasa Inggris. Penghapusan *stopwords* untuk bahasa indonesia menggunakan *database stopwords* Tala (2003) yang berjumlah 759 kata.
- *Stemming:* Proses *stemming* dilakukan menggunakan perangkat lunak Rstudio. Proses *stemming* dapat menggunakan *package* "tm" dan "snowball" yang tersedia pada R untuk beberapa bahasa. Namun, *package* tersebut belum mendukung untuk *stemming* teks berbahasa Indonesia. Sehingga dalam penelitian ini akan dilakukan *stemming* dengan menggunakan algoritme Nazief dan Adriani (1996).
- Pembuatan *Term Document Matrix* (TDM): Tahap selanjutnya adalah pembuatan *term document matrix* (TDM). TDM adalah matriks jumlah kemuculan suatu kata pada dokumen. Baris matriks menunjukkan kata yang ada pada data *tweet*, sedangkan kolom pada matriks

menunjukkan *tweet* yang ada pada data tersebut.

Pembagian Data

Pada tahap pembagian data, data yang yang telah dilakukan praproses, kemudian data dibagi menjadi data latih dan data uji. Pembagian data uji dan data latih adalah 70 persen untuk data latih dan 30 persen untuk data uji.

Pembuatan Model Klasifikasi Menggunakan Algoritme Naïve Bayes

Klasifikasi adalah proses menemukan model atau fungsi yang menggambarkan dan membedakan kelas data model. Model ini digunakan untuk memprediksi kelas dari objek yang belum diketahui kelasnya. Proses klasifikasi dibagi menjadi 2 tahapan yaitu *learning* dan *test*, sebagian data yang telah diketahui kelas datanya (*training set*) digunakan untuk membentuk model dan pada tahap *test*, model yang sudah terbentuk kemudian diuji menggunakan sebagian data lainnya untuk mengetahui akurasi model tersebut (Han *et al.* 2012). Naïve Bayes merupakan metode yang digunakan untuk pemodelan klasifikasi, *supervised learning* merupakan pemodelan klasifikasi berbasis peluang, perhitungan peluang tersebut berdasarkan kaidah peluang Naïve Bayes dapat dilihat pada persamaan 1.

$$P(c|d) \propto P(c) \prod_{1 \le k < nd} P(t_k|c) \quad (1)$$

dengan parameter $P(c)$ adalah peluang pada kelas c, $P(t_k|c)$ adalah peluang *token* t_k muncul pada *c* dan *nd* jumlah *token* unik. Pendugaan parameter $\hat{P}(c)$ dan $\hat{P}(t_k|c)$ dapat dilihat pada persamaan 2.

$$\hat{P}(c) = \frac{N_c}{N}, \quad \hat{P}(t_k|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}} \quad (2)$$

dengan N_c adalah banyaknya peluang pada kelas c, N adalah total dokumen, T_{ct} adalah banyaknya *token* t dalam dokumen data latih dari kelas c (Manning *et al*. 2009).

Tahap klasifikasi menggunakan algoritme Naïve Bayes. Setelah dilakukan tahap praproses, dari hasil praproses yang didapat, kata yang termasuk *spam* dan bukan *spam*, kemudian dihitung bobot untuk dapat diklasifikasikan menjadi *spam* dan bukan *spam*. Setelah dilakukan pemodelan klasifikasi menggunakan Naïve Bayes, tahap selanjutnya adalah dilakukan pengujian terhadap pemodelan menggunakan data uji yang telah tersedia, data tersebut juga dilakukan praproses seperti data latih.

Evaluasi dan Analisis Hasil

Proses evaluasi dapat dilakukan dengan cara membandingkan kelas aktual dari data uji dan kelas hasil prediksi dengan menggunakan *confusion matrix*.

HASIL DAN PEMBAHASAN

Pengumpulan Data

Pada tahap pengumpulan data, terdapat beberapa langkah. Berikut langkah- langkah pengumpulan data:

- Mengumpulkan akun Twitter. Langkah pertama dalam pengumpulan data adalah mengumpulkan akun Twitter, akun Twitter dipilih secara acak dari akun peneliti dan akun yang dipilih hanya akun yang bersifat *public* (bukan *private*).
- Memberi label untuk akun *spammer* dan bukan *spammer*. Setelah mengumpulkan akun Twitter, langkah selanjutnya adalah memberi label untuk akun *spammer* dan bukan *spammer*, pelabelan dilakukan secara manual, dengan cara melihat *content*. Jika, *content*

termasuk dalam kategori *spammer*, akun akan diberi label sebagai *spammer* sedangkan jika *content* tidak termasuk kedalam kategori *spammer*, akun akan diberi label bukan *spammer*.

- Mengambil data *tweet* menggunakan fungsi *userTimeline* yang ada pada *package* "twitteR", Fungsi *userTimeline*, antaralain: *text*, *favorite*, *favoritedCount*, *replyToSN*, *created*, *truncated*, *replyToSID*, *statusSource*, *screenname*, *retweetCount*, *isRetweet*, *retweeted*, *longitude*, *latitude*, dan URL. Banyaknya atribut pada pengambilan data Twitter yang termasuk fungsi *UserTimeline*, pada penelitian ini atribut yang digunakan hanya atribut *text*. Data didapatkan menggunakan Twitter API (*Application Programming Interface*) kemudian data tersebut disimpan dalam format file .csv.
- Setiap data *tweet* diberi label sesuai dengan label akun. Setelah dilakukan pelabelan pada akun dan dilakukan pengambilan data *tweet*, langkah selanjutnya adalah pelabelan pada data *tweet*.

Praproses

Tahap praproses dibutuhkan karena data *tweet* yang tidak baku. Tahap praproses terbagi atas empat tahap yaitu, *tokenizing*, *filtering*, normalisasi kata, dan *stemming*. Tahap tokenisasi dilakukan menggunakan *package* "twitteR" dan *package* "tm" yang tersedia dalam R. Pada normalisasi kata dilakukan perbaikan dari kata tidak baku menjadi kata baku. Pada penelitian ini kode program diperoleh dari penelitian sebelumnya dilakukan oleh Khotimah (2014) dengan menggunakan kata baku yang ada pada penelitian Aziz (2013). Pada tahap *filtering* dilakukan penghapusan *stopword* pada data *tweet*. Penghapusan *stopword* dilakukan dengan menggunakan *database stopword* (Tala 2003) yang berjumlah 759 kata dan ditambah dengan *stopword* untuk menghilangkan term "rt". Pada tahap *filtering*, kata yang termasuk dalam *stopwords* seperti ke, di, ada, dan rt akan dihapus. Dalam penelitian ini, *stemming* dilakukan menggunakan algoritme Nazief dan Adriani (1996). Kode program yang digunakan dalam penelitian ini untuk melakukan *stemming* berbahasa Indonesia menggunakan kode program yang diikuti dari penelitian Khotimah (2014).

Setelah dilakukan tahapan praproses sehingga didapat hasil *token* yang akan menjadi *term*. *Term* adalah *token* unik. Dapat dilihat perbedaan sebelum dan sesudah dilakukan praproses, kata sebelum dilakukan praproses tersebut direpresentasi dalam bentuk *wordcloud* yang dapat dilihat pada Gambar 3 dan hasil dari praproses tersebut direpresentasi dalam bentuk *wordcloud* yang dapat dilihat pada Gambar 4.

Gambar 3 Representasi *wordcloud* untuk term sebelum dilakukan praproses.

Gambar 4 Representasi *wordcloud* untuk *term* sesudah dilakukan praproses.

Dari Gambar 3 dapat dilihat bahwa *term-term* yang muncul sebelum dilakukan praproses masih mengandung angka dan tanda baca sedangkan Gambar 4 menunjukkan *term-term* yang

muncul setelah dilakukan praproses, sudah tidak ada lagi angka dan tanda baca. Hasil dari praproses tersebut menghasilkan sebanyak 1648 *terms* dengan *sparsity* 100% dan menyisakan 977 *tweet*, *sparsity* menggambarkan banyaknya angka nol dalam matriks. Pengurangan *tweet* yang sebelumnya terdapat 1000 data *tweet* dan menyisakan 977 *tweet* terjadi karena telah dilakukan tahapan praproses, karena terdapat *tweet* yang hanya berisi URL atau terdapat *tweet* yang hanya berisi angka saja. Setelah dilakukan praproses, *matrix document-term* dibuat untuk mengetahui frekuensi kemunculan *term* pada dokumen. Jumlah *term* yang muncul pada *tweet spam* sebanyak 197 *terms* dan jumlah *term* yang muncul pada *tweet* bukan *spam* sebanyak 1130 *terms*. *Term-term* yang muncul pada *tweet spam* dan *tweet* bukan *spam* dapat direpresentasikan dengan *wordcloud*. Hasil *term* pada kelas *spam* dapat dilihat pada Gambar 5 dan *term* pada kelas bukan *spam* dapat dilihat pada Gambar 6. Pada Gambar 5 dapat diketahui, *term-term* yang sering muncul pada kelas *spam* adalah bahasa, informasi, inggris dan pada Gambar 6 dapat diketahui *term-term* yang sering muncul pada kelas bukan *spam* memiliki frekuensi kemunculan yang hampir sama.

Banyaknya *term* yang dihasilkan membuat dimensi *matrix* menjadi besar, untuk memperkecil dimensi *matrix* dapat dilakukan dengan cara mereduksi *terms* yang memiliki tingkat kemunculan yang rendah dapat menggunakan fungsi removeSparse*Term*() yang terapat pada *package* "tm". Nilai *sparse* adalah nilai numerik untuk sparsity maksimum yang dibolehkan dalam dokumen dengan rentang 0 – 1. *Sparsity* 90% *term* yang memiliki paling sedikit 90% *empty element*, *empty element* adalah *term* yang muncul 0 kali dalam dokumen. adalah Hasil nilai *sparse* dan jumlah *term* yang dihasilkan dapat dilihat pada Tabel 1. Dari hasil nilai reduksi yang dapat dilihat pada Tabel 1, semakin kecil presentase *term* yang digunakan akan menghasilkan *term* yang semakin sedikit.

Gambar 5 Hasil *term* pada kelas *spam* dari data latih.

Gambar 6 Hasil *term* pada kelas bukan *spam* dari data latih

Tabel 1 Hasil reduksi *terms*

Persentase *term* yang digunakan	Jumlah *term* yang dihasilkan
0.10-0.70	0
0.75	3
0.80	5
0.85	6
0.90	9
0.95	24
1	1648

Pembagian Data

Pembagian data yang dilakukan adalah 70% untuk data latih dan 30% untuk data uji sehingga didapat 683 data latih dan 294 data uji.

Klasifikasi Menggunakan Algoritme Naïve Bayes

Pada tahap klasifikasi, penelitian ini menggunakan algoritme Naiıve Bayes. sehingga dari hasil praproses, kata yang termasuk *spam* dan bukan *spam*, kemudian dihitung bobot per *term* untuk dapat diklasifikasikan menjadi *spam* dan bukan *spam*. Contoh hasil perhitungan peluang *term* dapat diklasifikasikan sebagai *spam* dan bukan *spam* dapat dilihat pada Tabel 2.

Tabel 2 Contoh peluang *term* yang didapat menggunakan algoritme Naïve Bayes

Term	Bukan *Spam*	*Spam*
Coba	0.0029	0.0057
Daftar	0.0089	0
Detik	0.0059	0.0401
Difolbek	0	0.0143
Disc	0.0119	0.0000
Edukasi	0	0.0315
Efektif	0	0.0200
Follow	0.0059	0.3094
Foto	0.0149	0
Gabung	0.0209	0

Setelah dilakukan klasifikasi data menggunakan Naïve Bayes, tahap selanjutnya adalah dilakukan pengujian terhadap hasil klasifikasi menggunakan data uji yang telah tersedia. Contoh hasil prediksi dapat dilihat pada Tabel 3.

Tabel 3 Contoh hasil prediksi

Teks	Aktual	Prediksi	Hasil Prediksi
wajib baca tips alami mutih kulit tubuh cepat	Spam	Bukan Spam	Salah
born game with gigabyte gaming graphics cards	Bukan Spam	Spam	Salah
informasi nich kak ajar bahasa inggris tingkat toefl kursus klik"	Spam	Spam	Benar
jepret by terimakasih point photography nmc	Bukan Spam	Bukan Spam	Benar
informasi nich kak ajar bahasa inggris tingkat toefl kursus klik	Spam	Spam	Benar

Evaluasi dan Analisis Hasil

Dari hasil nilai reduksi yang dapat dilihat pada Tabel 1, semakin kecil presentase *term* yang digunakan akan menghasilkan *term* yang semakin sedikit sehingga model tidak dapat memprediksi dengan lebih baik lagi dan menghasilkan akurasi yang kecil. Hasil akurasi dari hasil reduksi *term* dapat dilihat pada Tabel 4. Banyaknya duplikat *tweet* juga mempengaruhi tingginya akurasi.

Tabel 4 Hasil akurasi dari hasil reduksi *term*

Persentase *term* yang digunakan	Jumlah *term* yang dihasilkan	Akurasi (%)
0.10-0.70	0	-
0.75	3	85.71
0.80	5	85.37
0.85	6	93.87
0.90	9	94.55
0.95	24	94.55
1	1648	95.57

Perhitungan evaluasi dengan cara membandingkan kelas aktual dari data uji dan kelas hasil prediksi dengan menggunakan *confusion matrix*. C*onfusion matrix* dapat dilihat pada Tabel 5.

Tabel 5. *Confusion matrix*

Prediksi	Aktual	
	Bukan *Spam*	*Spam*
Bukan *Spam*	139	9
Spam	4	142

Dari *confusion matrix* pada Tabel 5, dapat diketahui bahwa 142 data dengan kelas aktual *spam* dan benar diprediksi sebagai *spam*. Sementara data aktual *spam* yang salah diprediksi sebagai bukan *spam* sebanyak 9 data. Untuk data dengan kelas aktualnya bukan *spam* tetapi diprediksi sebagai *spam* sebanyak 4 data, sementara data yang benar sebagai bukan *spam* dan diprediksi sebagai bukan *spam* ada sebanyak 139 data. *Tweet* yang salah prediksi dapat dilihat pada Tabel 6. Akurasi hasil klasifikasi yang didapat adalah 95.57%.

Tabel 6. Data *tweet* yang salah prediksi

No.	*Text*	Aktual	Prediksi
1	wajib baca tips alami mutih kulit tubuh cepat	*Spam*	Bukan *Spam*
2	sebab hilang komedo alami	*Spam*	Bukan *Spam*
3	rahasia kecil paha betis	*Spam*	Bukan *Spam*
4	hilang ketombe rambut alami	*Spam*	Bukan *Spam*
5	tips atas rambut rontok alami	*Spam*	Bukan *Spam*
6	tips atas rambut rontok alami	*Spam*	Bukan *Spam*
7	urun berat badan minggu alami	*Spam*	Bukan *Spam*
8	halo kejar toefl score atas baca	*Spam*	Bukan *Spam*
9	halo kejar toefl score atas baca	*Spam*	Bukan *Spam*
10	tips rawat rambut kering alami	*Spam*	Bukan *Spam*
11	malas kl tugas isi bahasa inggris video ppt text book	Bukan *Spam*	*Spam*
12	hallo sajadah alas kasur ready stok nih	Bukan S*pam*	*Spam*
13	kak madam listrik kp nanggoh ds cimande c caringin kab bogor ya penyebabnya terimakasih	Bukan *Spam*	*Spam*

Dari Tabel 6, diketahui bahwa ada 13 *tweet* yang diprediksi salah, yaitu 10 data yang kelas sebenarnya adalah *spam* tetapi diprediksi sebagai bukan *spam* dan 3 data yang kelas sebenarnya bukan *spam* tetapi diprediksi sebagai *spam*, hal tersebut disebabkan banyaknya kata-kata pada *tweet* bukan *spam* muncul pada korpus data *spam*, sehingga *tweet* tersebut diprediksi sebagai *spam* dan sebaliknya banyak kata-kata pada *tweet spam* yang diindikasi sebagai bukan *spam*, sehingga *tweet* tersebut diprediksi sebagai bukan *spam*. Hal ini salah satunya dapat dipengaruhi oleh *content spam* yang didominasi oleh *tweet* promosi

SIMPULAN

Penelitian ini menggunakan algoritme Naïve Bayes untuk mengklasifikasikan *spam* dan bukan *spam* dengan menggunakan data Twitter, algoritme Naïve Bayes dapat dikatakan baik untuk klasifikasi *spam* dan bukan *spam* pada Twitter karena memiliki akurasi yang tinggi sebesar 95.57%, salah satu faktor yang mempengaruhi tingginya akurasi adalah karena penelitian ini mengabaikan duplikat *tweet* sehingga setiap *tweet* yang memiliki isi yang sama tetap dianggap sebagai *tweet* yang berbeda.

Dari penelitian ini juga dapat diketahui *term-term* yang termasuk *spam* dan bukan *spam*, term-term yang sering muncul pada kelas *spam* adalah bahasa, follow, inggris, sehingga dapat diketahui pada penelitian ini *tweet spam* didominasi dengan konten *tweet* promosi bahasa Inggris.

DAFTAR PUSTAKA

Adriani M, Asian J, Nazief B, Tahaghoghi SM, Wiliams HE. 1996. Stemming indonesian: a confix-stripping approach. *ACM Transactions on Asian Language Information Processing*. 6(4):1-33

Aziz ATA. 2013. Sistem pengklasifikasian entitas pada pesan twitter menggunakan ekspresi reguler dan Naïve Bayes [skripsi]. Bogor (ID): Institut Pertanian Bogor.

Chakraborty M, Pal S, Pramanik R, Chowdary CR. 2016. Recent developments in social spam detection and combating techniques: A survey. *Information Processing & Management*. 52(6):1053-1073.

Gupta H, Jamal MS, Madisetty S and Desarkar MS. 2018. A framework for real-time spam detection in Twitter. Di dalam: *2018 10th International Conference on Communication Systems & Networks (COMSNETS)*; Bengaluru, 2018 Jan 3-7. Bengaluru (IN). hlm: 380-383.

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

Kemenkominfo, Kementerian Komunikasi Informatika. 2013. Kominfo:pengguna internet di Indonesia 63 juta orang [internet]. [Diunduh 2015 Jun 27]; http://kominfo.go.id/index.php/content/detail/3415/%20Kominfo+:+Pengguna+Internet+di+Indonesia+63+Juta+Orang/0/berita_satker

Khotimah H. 2014. Pemodelan hybrid tourism recommendation menggunakan hidden markov model dan text mining berbasis data sosial media [tesis]. Bogor (ID): Institut Pertanian Bogor.

Makhtidi K. 2012. Sistem SMS *spam* detector untuk sms Berbahasa Indonesia pada smartphone android [skripsi]. Bogor (ID): Institut Pertanian Bogor.

Manning CD, Raghavan P, Schütze H. 2009. *An Introduction to Information Retrieval*. Cambridge (UK): Cambridge University press.

McCord M and Chuah M. 2011. Spam detection on twitter using traditional classifier. Di dalam: M Jose, editor. *Autonomic and Trusted Computing: 8th International Conference*; Banff, 2011 Sep 2-4. San Francisco(US): Springer. hlm: 175-186.

Metsis V, Androutsopolous I, Paliouras G. 2006. Spam Filtering With Naïve Bayes –Which Naïve Bayes?. Di dalam: *Third Conference on Email and Anti-Spam*; Mountain View, 2006 Jul 27-28. California(US).

Palupiningsih P. 2011. Sistem pengklasifikasian entitas pada pesan Twitter menggunakan ekspresi regular dan Naïve Bayes [tesis]. Bogor (ID): Institut Pertanian Bogor.

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

Wang A H. 2012. Machine Learning for the Detection of Spam in Twitter Networks. Di dalam: *Obaidat M S, Tsihrintzis GA, Filipe J. (eds) e-Business and Telecommunications. ICETE 2010*; Heidelberg, 2010 Jul 26-28. Berlin (DE): Springer. hlm: 319-333.