Sistem Pendeteksi Pengirim Tweet dengan Metode Klasifikasi Naive Bayes

Maresha Caroline Wijanto

S1 Teknik Informatika Universitas Kristen Maranatha
Jl. Surya Sumantri no. 65, Bandung
maresha.caroline@itmaranatha.org

Abstract— **Until Januari 2015, social media users reached 29% of the world population. In Indonesia itself had 28% active users from total population of Indonesia. The usage of social media gives positives and negatives effect. The negatives effect are the increasing number of fraud by using SMS or social media, such as Twitter. Many people are deceived by the tweet messages sent from known user account when in fact the sender is other person. Because of that, there is a need to have a system to detect whetеher the tweet sender is the same person or not. Naive Bayes classifiers method is used to classify that. The data source is taken from tokens selected based on two models, the minimum n-time number of occurrences and the n-th highest number of occurrences. Each tweets also processed into six different types of tweets, such as formal tweet or lowercase tweet. The test uses tenfold cross-validation and measured by the value of accuracy, precision, recall, and F-score. The common result shows 82,145% level of accuracy. Second model to select the tokens shows consistency level of accuracy for each types of tweets. The fifth types of tweets also get the highest level of accuracy for both models to select the tokens.**

Keywords—**Classifiers, Detection, Naive Bayes, Tweet.**

I. PENDAHULUAN

Laporan dari We Are Sosial, sebuah badan yang meneliti tentang media sosial, menyebutkan bahwa jumlah pengguna internet sampai Januari 2015 adalah 7.210 milyar pengguna dan pengguna media sosial yang aktif berjumlah 2.078 milyar pengguna [1]. Pengguna media sosial ini mencapai 29% dari populasi penduduk dunia. Dari laporan tersebut juga tertulis bahwa di Indonesia sendiri terdapat 28% pengguna aktif media sosial dari populasi total (sekitar 72 juta pengguna). Per Januari 2015, media sosial yang paling banyak digunakan di seluruh dunia adalah Facebook (sekitar 1.366 juta pengguna), diikuti oleh Qzone (media sosial di China), Google+, Instagram, Twitter, dan Tumblr. Fenomena yang terjadi di Indonesia agak sedikit berbeda, media sosial yang paling banyak digunakan adalah Facebook, Twitter, Google+, LinkedIn, Instagram, dan Pinterest [1].

Penggunaan media sosial sendiri memberikan banyak manfaat, baik yang positif maupun yang negatif. Sebagai contoh penggunaan yang positif adalah memudahkan pengguna untuk berjualan (*e-commerce*), saling terhubung dengan pengguna lain di seluruh dunia, dan juga membantu menyebarkan informasi atau pengetahuan di seluruh dunia [2]. Tetapi selain hal positif tadi, media sosial juga memberikan dampak yang negatif, misalnya kejahatan penipuan yang berkedok keluarga mengalami kecelakaan [3] atau penawaran produk tertentu [4]. Berdasarkan data Polda Metro Jaya, kejahatan melalui internet telah mencapai 601 kasus pada 2013 dan pada 2014 sudah meningkat menjadi sekitar 70 kasus per bulan [5].

Kejahatan penipuan melalui media sosial ini disebabkan juga oleh orang-orang yang mudah percaya pada pesan yang dikirimkan. Apalagi bila pesan tersebut membawa kabar yang buruk terkait orang yang dikenal [3]. Seperti halnya pada kasus penipuan melalui SMS, orang seringkali tidak menyadari kalau pengirim pesan tersebut bukanlah orang yang sama. Orang hanya memperhatikan nama pengirim yang dikenal dan langsung menuruti pesan tersebut tanpa menyadari kalau sebenarnya sedang ditipu. Biasanya hal ini terjadi pada pesan singkat seperti SMS ataupun *tweet* (status dari Twitter). Dari hal ini lah, sistem untuk mendeteksi apakah pengirim pesan tersebut merupakan orang yang sama atau tidak menjadi sesuatu yang penting. Diharapkan sistem ini dapat membantu orang supaya tidak mudah tertipu.

Pada penelitian Stefanus dan Wijanto telah membuktikan bahwa metode klasifikasi Naive Bayes dapat digunakan untuk mendeteksi apakah pengirim pesan singkat (SMS) merupakan orang yang sama atau tidak. Metode ini memanfaatkan SMS-SMS yang telah dikirimkan sebelumnya [6]. Penelitian kali ini akan mencoba menggunakan metode klasifikasi Naive Bayes juga untuk mendeteksi apakah sebuah *tweet* tertentu memang benar dikirimkan oleh pengirim yang sama.

Penelitian ini akan mencoba menggunakan metode klasifikasi Naive Bayes dengan memanfaatkan fitur frekuensi kemunculan kata, *n-grams*, formalisasi kata, dan penggunaan huruf kecil semua maupun biasa. Setiap *tweet* yang diolah menggunakan bahasa Indonesia.

II. KAJIAN TEORI

Bab ini akan membahas tentang Twitter sebagai media sosial yang dipilih, metode klasifikasi Naive Bayes, dan penelitian sejenis di bidang ini.

A. Twitter

Twitter merupakan sebuah situs media sosial yang berbasis *microblogging*, dimana penggunanya dapat mengirimkan sebuah pesan yang disebut dengan *tweets*. Pengguna dibatasi hanya dapat mengirimkan pesan sebanyak 140 karakter saja. Tetapi hal ini juga yang membuat Twitter paling mudah digunakan oleh masyarakat. Pengguna dapat dengan mudah mengirimkan pesan dan membaca pesan-pesan dari orang lain dalam waktu yang singkat serta dapat tersebar luas dengan mudah juga [7].

Pengguna Twitter diidentifikasi melalui *username* (biasanya diawali dengan tanda @, misal @username), dimana *username* dapat berbeda dengan nama asli pengguna dan nama asli dari pengguna juga tidak harus dituliskan. Pengguna Twitter dapat mem-*follow* pengguna Twitter lainnya. Hal ini akan menyebabkan pengguna dapat melihat pesan atau *tweets* yang dituliskan oleh pengguna yang telah di-*follow*-nya. Pengguna yang telah di-*follow* dapat memilih apakah akan mem-*follow* juga atau tidak. *Tweets* dapat dikelompokkan berdasarkan *hashtag* tertentu, dimana dimulai dengan tanda # dan diikuti dengan kata yang populer. *Hashtag* membantu pengguna untuk dapat mengelompokkan *tweets* berdasarkan topik tertentu dan juga dapat menjadi ciri khas tersendiri dari setiap pengguna. Setiap pengguna dapat melakukan *retweet* sebuah *tweet* yang dituliskan oleh pengguna yang telah di-*follow* kepada pengguna yang telah mem-*follow* pengguna tadi (*following user*) [8].

Pada 30 Juni 2015, Twitter memiliki 316 juta pengguna yang aktif dalam 1 bulan. Sekitar 77% pengguna merupakan pengguna diluar Amerika Serikat dan terdapat 500juta *tweets* yang dibuat dalam satu hari [9]. Berikut ini merupakan istilah-istilah umum yang terdapat di Twitter [7] [8]:

- *Tweet*: sebuah pesan atau status dari pengguna Twitter, dengan maksimum 140 karakter. *Tweets* juga termasuk aktivitas pengguna, membagikan informasi, mem-*forward* status pengguna lain, atau berbincang dengan pengguna lain.
- *Tweeple*: sebutan untuk pengguna Twitter.
- *Tweeps*: sebutan untuk pengguna yang telah saling mem-*follow*.
- *ShortURL*: sebuah *Tweet* dapat berisi *link* ke *website*, *blog*, dan lainnya. Untuk mencegah panjangnya *link*, pengguna dapat memanfaatkan *shortURL* yang dibuat dari *service* lain. Contoh: http://bit.ly/IyBgIO.
- *Reply*: fitur untuk berkomunikasi dengan penulis *tweet* dengan menekan tombol '*Reply'* untuk merespon sebuah *tweet*. Fitur ini otomatis menambahkan *username* dari penulis *tweet* asal.
- *Hashtag*: dimulai dengan simbol # (contoh: #PersibDay). *Hashtag* menjadi dasar untuk topik yang ditentukan oleh pengguna, dan dapat membantu mengelompokkan pembicaraan dalam grup.
- *Retweet*: mem-*forward* sebuah *tweet* lain seperti *email forwarding*. Secara otomatis akan menambah kata 'RT' dan *username* dari penulis lalu dilanjutkan dengan *tweet* aslinya.
- *Mention*: menambahkan pengguna dengan simbol @ tanpa menggunakan fitur 'Reply'. Contoh: "Hello @_maresha. Apa kabar?".
- *Direct-Message*: fitur yang membuat pengguna dapat mengirimkan pesan khusus ke pengguna lain yang di *follow* dan juga telah mem-*follow* ulang.
- *Follower* / *Following*: dalam Twitter, blog, atau media sosial lain, *follower* adalah seorang yang mendaftar untuk menerima *update* dari pengguna tertentu. *Following* adalah pengguna yang akan dilihat statusnya pada *timeline* pengguna.

B. Tokenization dan N-Grams

Tokenizing adalah sebuah proses membentuk kata dari urutan karakter yang ada pada sebuah dokumen. Dalam beberapa sistem, *tokenizing* dianggap proses yang sederhana, tetapi sebenarnya prosesnya lebih rumit. Sebuah kata atau *token* dapat dianggap sebagai urutan dari karakter alfanumerik dengan panjang tiga karakter atau lebih yang dipisahkan oleh spasi atau karakter spesial lainnya [10].

Token atau kata yang dihasilkan dapat langsung digunakan atau bisa juga digunakan sebagai beberapa urutan kata. Urutan kata yang dimaksud adalah kata yang diikuti oleh kata lainnya atau urutan dari *n* kata [11]. Hal ini disebut juga sebagai *n-grams*. Urutan dari dua kata disebut sebagai *bigrams* dan urutan dari tiga kata disebut *trigrams*. Sedangkan untuk satu kata tersendiri disebut sebagai *unigrams* [10].

C. Naive Bayes

Teorema Bayes dinamakan dari Thomas Bayes, orang Inggris yang melakukan penelitian awal terkait probabilitas dan teori keputusan pada abad ke-18 [12]. Teorema Bayes memberikan sebuah cara untuk menghitung probabilitas dari sebuah hipotesis berdasarkan *prior probability*. Nilai probabilitas yang diyakini benar sebelum melakukan eksperimen terhadap sesuatu disebut *prior probability*. Teorema bayes dapat digunakan sebagai algoritma dasar yang menghitung probabilitas untuk setiap kemungkinan hipotesis dan mencari yang paling memungkinkan [13].

Metode klasifikasi Naive Bayes mengasumsikan bahwa efek dari satu nilai atribut pada kelas tertentu adalah *independent* terhadap nilai dari atribut lain. Asumsi ini disebut dengan *class-conditional independence*. Rumus teorema Bayes adalah sebagai berikut [12]:

$$P(H|X) = \frac{P(X|H)\,P(H)}{P(X)}$$

dimana,

- $P(H|X)$ adalah *posterior probability* dari kelas H terhadap X atau probabilitas dari *instances* X dalam kelas H
- $P(H)$adalah *prior probability* dari kelas H atau probabilitas dari kelas H
- $P(X|H)$ adalah *posterior probability* dari X dikondisikan pada kelas H atau probabilitas dari *instances* X jika terdapat pada kelas H
- $P(X)$adalah *prior probability* dari X atau probabilitas dari kelas X

Sedangkan cara kerja metode klasifikasi Naive Bayes adalah sebagai berikut [12]:

- Setiap data training disajikan dalam bentuk *n-dimensional attribute vector*, $X = (X_1, X_2, \dots, X_n)$, dimana *n* merupakan jumlah atribut, $A_1, A_2, \dots, A_n$
- Misal terdapat m kelas, $C_1, C_2, \dots, C_m$. Dengan data training X, sistem akan memprediksi bahwa X termasuk dalam kelas yang memiliki *posterior probability* tertinggi dengan kondisi X
- Sesuai Teorema Bayes, maka rumusnya berubah menjadi: $P(C_i|X) = \frac{P(X|C_i)\,P(C_i)}{P(X)}$
- Tetapi karena nilai $P(X)$ selalu konstan untuk semua kelas yang ada, maka pada rumus tadi $P(X)$ dapat dihilangkan

Dalam perhitungan semua kemungkinan yang terjadi, mungkin terdapat kata yang tidak pernah muncul di semua *data source*. Maka frekuensi kemunculan kata tersebut bernilai 0. Untuk mencegah terjadinya *division by zero*, dapat menggunakan *add-one* atau *Laplace Smoothing* yang menambahkan langsung 1 nilai untuk setiap kemungkinan yang ada [14].

D. Penelitian Sejenis

Sudah banyak penelitian lain yang pernah membahas hal sejenis terkait Twitter maupun metode klasifikasi Naive Bayes, contohnya:

1) Detection the Similarity of the Message Sender On Short Message Service (SMS): Stefanus dan Wijanto memanfaatkan metode klasifikasi Naive Bayes untuk menilai apakah sebuah SMS tertentu dikirim oleh orang yang sama atau tidak. Semua data SMS yang ada ditokenisasi lalu dihitung frekuensi kemunculannya. Fitur yang digunakan adalah pemanfaatan *unigram* dan *bigram* dari *token* yang diambil berdasarkanjumlah kemunculan minimum dan termasuk dalam jumlah kemunculan yang ke-*n* terbanyak. *Token* tersebut yang dijadikan dasar untuk pembentukan *training set* dan *test set*. Lalu dihitung nilai probabilitas yang paling tinggi antara dua kelas yang memungkinkan berdasarkan rumus Teorema Bayes [6].

2) *Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets:* Gamallo dan Garcia menggunakan metode klasifikasi Naive Bayes untuk mendeteksi *polarity* pada *tweet* berbahasa Inggris. Hasil terbaik diperoleh dengan menggunakan *binary classifier* antara dua pola *polarity* yang paling jelas, yaitu *tweet* positif dan *tweet* negatif. Sistem ini juga menggunakan *rule-rule* yang mencari kata khusus pada *tweet* untuk mendeteksi *polarity*. Sistem ini memiliki 4 fitur utama, *lemmas*, *multiwords*, *polarity lexicons*, dan *valence shifters.* Hasil terbaik diperoleh dengan fitur *polarity lexicon* dan *multiwords* [15].

III. ANALISIS DAN PERANCANGAN

Sistem ini akan menggunakan metode klasifikasi Naive Bayes. Semua kumpulan *tweet* akan dijadikan sebagai *data source*, dan akan diambil 1 buah *tweet* yang nantinya akan dijadikan sebagai *test set*. Tabel I menampilkan kumpulan contoh *tweet* yang akan digunakan sebagai *data source*.

TABEL I
CONTOH DATA SOURCE

Contoh *Tweet* dari @basukibtp [16]
Kita akan cabut KJP peserta jika dipergunakan untuk kegiatan diluar keperluan sekolah
Semua terdeteksi dari sistem Bank DKI Untuk orangtua yg anaknya dapat KJP, belanjakanlah utk kebutuhan sekolah.
Di catatan sistem ada yg pakai utk beli emas, kosmetik dan karaoke
Semua camat, lurah, guru di DKI di harap saling koordinasi untuk mengetahui siapa saja warga yang layak menerima KJP.
Dengan metode ini diharapkan alokasi dana KJP bisa lebih tepat sasaran. Tepat penerimanya dan tepat juga pembelanjaannya
Di acara book fair tsb, pemegang KJP bisa belanja macam-macam perlengkapan sekolah dgn debit Bank DKI (non-cash transaction)Untuk keperluan lain seperti tas, buku dan alat tulis,
DKI bekerjasama dgn IKAPI mengadakan acara bookfair untuk para pemegang KJP
Jadi dana yg dicairkan per minggunya bisa benar-benar digunakan untuk biaya personal siswa, seperti transportasi ke sekolah
Karena itulah tahun ini ada kebijakan pembatasan pencairan tunai dana KJP. Orangtua/siswa tdk bisa lagi langsung cairkan seluruh dana KJP
Evaluasi kita selama ini, banyak anak didik yg menggunakan dana KJP utk hal diluar urusan pendidikan
Kebijakan KJP ini adalah untuk memastikan agar kendala biaya personal tidak jadi hambatan utk anak Jakarta bersekolah
Selamat malam, ingin berbagi tentang program pemprov DKI di bidang pendidikan, yaitu Kartu Jakarta Pintar (KJP)

Data source tadi ditokenisasi dan dihitung jumlah frekuensi kemunculannya. Contoh hasil tokenisasi beserta jumlah kemunculannya dapat dilihat pada Tabel II. Data ini telah diurutkan mulai dari yang paling sering muncul. Data ini nantinya dijadikan sebagai dasar pemilihan atribut apa saja yang akan digunakan untuk membentuk *training set* dan *test set*.

TABEL II
HASIL TOKENISASI

No.	Token	Jumlah Kemunculan
1.	KJP	11
2.	untuk	5
3.	dana	5
4.	bisa	4
5.	ini	4
6.	utk	4
7.	yg	4
8.	DKI	4

Token apa saja yang akan digunakan sebagai atribut dipilih berdasarkan dua cara, yaitu berdasarkan jumlah kemunculan minimum *n*-kali (contoh jumlah kemunculan minimal 5) atau termasuk dalam jumlah kemunculan yang ke-*n* terbanyak (contoh yang ke-5 terbanyak). Berdasarkan data pada Tabel II, atribut yang akan dipilih berdasarkan jumlah kemunculan minimum 5-kali adalah 'KPJ', 'untuk', dan 'dana'. Sedangkan contoh atribut yang akan dipilih berdasarkan jumlah kemunculan yang ke-5 terbanyak adalah 'KJP', 'untuk', 'dana', 'bisa', 'ini'.

Atribut ini yang akan menjadi dasar pembentukan *training set* berdasarkan setiap data yang ada dalam *data source*. *Training set* yang dibentuk berisi 1 apabila atribut tersebut ada dalam *instance* terkait dan berisi 0 apabila tidak ada. *Training set* yang dibentuk harus menggambarkan setiap kelas yang ada (pengirim yang sama atau pengirim yang berbeda). Tabel III menampilkan *training set* yang terbentuk dari 12 *instances* untuk kelas yang sama dan 12 *instances* untuk kelas yang berbeda. Pada kolom kelas, nilai T berarti *instance* tersebut menggambarkan pengirim yang sama dan nilai F menggambarkan pengirim yang berbeda.

TABEL III
HASIL TRAINING SET

KJP	untuk	dana	bisa	ini	Kelas
1	1	0	0	0	T
1	0	0	0	0	T
0	0	0	0	0	T
1	1	0	0	0	T
1	0	1	1	1	T
1	0	0	1	0	T
1	1	0	0	0	T
0	1	1	1	0	T
1	0	1	1	1	T
1	0	1	0	1	T
1	1	0	0	1	T
1	0	0	0	0	T
0	0	0	0	0	F
0	0	0	0	0	F
0	0	0	0	0	F
0	0	0	0	0	F
0	0	0	0	0	F
0	0	0	0	0	F
0	0	0	0	0	F
0	0	0	0	0	F
0	0	0	0	0	F
0	0	0	0	0	F

Test set yang dijadikan contoh adalah "Book fair tsb dibuka senin besok. Walau ini diperuntukkan utk pemegang KJP, tapi acaranya terbuka jg untuk umum". Berdasarkan *test set* tadi, sistem harus membentuk *instance* dengan atribut yang sama dengan *training set*. *Instance* yang dibuat harus menggambarkan kedua hipotesis atau kelas yang tersedia. Contoh hasil bentukan *instance* dari *test set* dapat dilihat pada Tabel IV.

TABEL IV
HASIL INSTANCES DARI TEST SET

KJP	untuk	dana	bisa	ini	Hipotesis
1	1	0	0	1	T
1	1	0	0	1	F

Berdasarkan data pada Tabel IV, akan dihitung setiap kemungkinan yang dapat terjadi, baik untuk hipotesis bernilai benar (orang yang sama) atau hipotesis yang bernilai salah (orang yang berbeda). Dalam perhitungan ini, mungkin terdapat kata yang tidak pernah muncul sama sekali pada *data source*. Untuk mencegah terjadinya *division by zero*, digunakan *Laplace Smoothing* yang secara otomatis menambahkan 1 untuk setiap perhitungan. Berikut contoh perhitungan berdasarkan contoh data yang ada:

1. P (Value = T) = (12+1)/ (24+1) = 13/25
2. P (Value = F) = (12+1) /(24+1) = 13/25
3. P (KJP = 1 | Value = T) = (10+1) / (13+1) = 11/14
4. P (untuk = 1 | Value = T) = (5+1) / (13+1) = 6/14
5. P (dana = 0 | Value = T) = (8+1) / (13+1) = 9/14
6. P (bisa = 0 | Value = T) = (8+1) / (13+1) = 9/14
7. P (ini = 1 | Value = T) = (4+1) / (13+1) = 5/14
8. P (KJP = 1 | Value = F) = (0+1) / (13+1) = 1/14
9. P (untuk = 1 | Value = F) = (0+1) / (13+1) = 1/14
10. P (dana = 0 | Value = F) = (12+1) / (13+1) = 13/14
11. P (bisa = 0 | Value = F) = (12+1) / (13+1) = 13/14
12. P (utk = 1 | Value = F) = (0+1) / (13+1) = 1/14

Nilai probabilitas untuk kelas T diperoleh dari: 13/25 * 11/14 * 6/14 * 9/14 * 9/14 * 5/14 = 0,02584.
Nilai probabilitas untuk kelas F diperoleh dari: 13/25 * 1/14 * 1/14 * 13/14 * 13/14 * 1/14 = 0,00016.

Berdasarkan kedua nilai tersebut diatas, yang paling besar adalah nilai probabilitas untuk kelas T (nilai probabilitas untuk kelas T: 0,02584 dan nilai probabilitas untuk kelas F: 0,00016). Artinya *tweet* yang dijadikan *test set* merupakan *tweet* yang berasal dari pengirim yang sama dengan kumpulan *tweet* pada Tabel I.

IV. HASIL IMPLEMENTASI

Proses pengolahan dan rencana pengujian yang dilakukan pada penelitian ini akan adalah sebagai berikut:

A. Preprocessing Data

Proses ini berpengaruh pada pengujian yang akan dilakukan. Pengujian akan dilakukan untuk *tweet* asli apa adanya dan *tweet* yang sudah di-*preprocessing*. Proses *preprocessing* sendiri terdiri dari beberapa cara, yaitu penghilangan tanda baca, melalui proses *formalization* dan dibuat menjadi huruf kecil semua. Proses pertama setiap tanda baca dihapus, seperti tanda titik, tanda koma, ataupun tanda kurung. Tetapi tanda @ dan # tidak dihapus karena tanda @ menunjukkan pengguna Twitter lain dan tanda # menunjukkan pengelompokkan topik pada Twitter. Setelah penghapusan tanda baca, *tweet* tersebut akan melalui proses *formalization* (menghasilkan bentuk kata formal). Proses *formalization* ini menggunakan *tools* tambahan, InaNLP, melalui metode *IndonesianSentenceFormalization*. *Tools* ini merupakan sebuah *library natural language processing* untuk bahasa Indonesia yang dikembangkan oleh Tim Lab Grafika dan Intelegensia Buatan dari Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung.

Lalu *tweet* akan diubah menjadi huruf kecil semua. Proses ini akan diterapkan untuk *tweet* asli dan juga *tweet* formal. Jadi pengujian akan dilakukan dengan 6 jenis cara, yaitu *tweet* asli huruf normal, *tweet* asli huruf kecil semua, *tweet* tanpa tanda baca huruf normal, *tweet* tanpa tanda baca huruf kecil semua, *tweet* formal tanpa tanda baca huruf normal, dan*tweet* formal tanpa tanda baca huruf kecil semua.

B. Tokenization

Setelah *preprocessing* data selesai dilakukan, *tweet* tersebut akan dipecah menjadi *token-token*. *Token* inilah yang akan menjadi dasar perhitungan dalam metode Naive Bayes. Dalam penelitian kali ini hanya menggunakan token hasil *unigram* (hanya satu kata). *Token* yang nantinya digunakan menjadi atribut dipilih dengan menggunakan 2 model, yaitu berdasarkan jumlah kemunculan minimum *n*-kali (contoh jumlah kemunculan minimal 5-kali) atau termasuk dalam jumlah kemunculan yang ke-*n* terbanyak (contoh yang ke-5 terbanyak).

C. Perhitungan Naive Bayes dengan Weka

Setiap *tweet* yang tersedia yang telah diproses akan dilakukan perhitungan dengan menggunakan metode Naive Bayes pada Weka. Weka merupakan sebuah kumpulan algortima *machine learning* dan *tools* untuk *data preprocessing*. Weka dikembangkan oleh Universitas Waikato di New Zealand yang berbasis Java [17]. Weka menyediakan fitur pengujian dengan teknik statistik yaitu *cross-validation*. Metode ini akan membagi data dengan pembagian yang dapat ditentukan oleh pengguna, yang disebut sebagai *fold* atau partisi dari data. Contohnya *stratified tenfold cross-validation*. Data yang ada dibagi secara acak menjadi 10 bagian dimana *class* direpresentasikan dalam *full dataset* dengan proporsi yang hampir sama. Dan pengujian ini telah menjadi metode standar yang dilakukan [17].

Setiap pengujian akan diukur akurasinya dengan beberapa nilai ukuran. Ukuran tersebut antara lain *Accuracy*, *Precision*, *Recall*, dan *F-score*. Sebelum membahas masing-masing ukuran tadi, ada beberapa hal yang perlu dibahas, yaitu [12]:

- *Positives (P)*: semua data dari kelas utama (kelas orang yang sama)
- *Negatives (N)*: semua data dari kelas lainnya (kelas orang yang berbeda)
- *True Positives (TP)*: mengacu pada semua data *positives* yang dianggap *positives* juga oleh metode klasifikasi
- *True Negatives (TN)*: mengacu pada semua data *negatives* yang dianggap *negatives* juga oleh metode klasifikasi
- *False Positives (FP)*: mengacu pada semua data *negatives* yang dianggap *positives* oleh metode klasifikasi (contoh: data dari orang yang berbeda tetapi oleh metode klasifikasi dianggap dari orang yang sama)
- *False Negatives (FN)*: mengacu pada semua data *positives* yang dianggap *negatives* oleh metode klasifikasi (contoh: data dari orang yang sama tetapi oleh metode klasifikasi dianggap dari orang yang berbeda)

Dalam hasil perhitungan Weka, nilai tersebut akan digambarkan pada *confusion matrix*. Nilai tadi digunakan untuk menghitung ukuran pengujian, yaitu [12]:

- *Accuracy*: persentase dari data yang diklasifikasikan dengan benar oleh metode klasifikasidengan jumlah semua data

$$accuracy = \frac{TP + TN}{P + N}$$

- *Precision*: dianggap sebagai ukuran dari ketepatan (persentase dari data *positives* yang diklasifikasikan dengan benar)

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

- *Recall*: dianggap sebagai ukuran dari keberhasilan (persentase dari data *positives* yang diidentifikasi dengan benar)

$$recall = \frac{TP}{TP + FN} = \frac{TP}{P}$$

- *F-score*: *harmonic mean* dari *precision* dan *recall*

$$F - score = \frac{2\ x\ precision\ x\ recall}{precision + recall}$$

V. PENGUJIAN

Data *tweet* yang berhasil dikumpulkan berasal dari 15 pengguna Twitter dan masing-masing terdapat *tweet* sejumlah kurang lebih 150 *tweet*. Sehingga total *tweet* yang dijadikan uji coba berjumlah sekitar 2.250 *tweet*. Pengujian akan dilakukan dengan menggunakan *tenfold cross-validation.* Proses pengolahan dan rencana pengujian yang dilakukan pada penelitian ini dibagi menjadi 2 model pemilihan atribut, yaitu dengan jumlah kemunculan minimum *n*-kali dan jumlah kemunculan ke-*n* terbanyak. Setiap model akan diuji untuk setiap jenis *tweet* yang berbeda. Terdapat 6 jenis *tweet* yang digunakan, yaitu (1) *tweet* asli huruf normal, (2) *tweet* asli huruf kecil semua, (3) *tweet* tanpa tanda baca huruf normal, (4) *tweet* tanpa tanda baca huruf kecil semua, (5) *tweet* formal tanpa tanda baca huruf normal, dan (6) *tweet* formal tanpa tanda baca huruf kecil semua. Ukuran yang digunakan untuk setiap pengujian adalah *accuracy*, *precision*, *recall*, dan *F-score*. Hasilnya adalah sebagai berikut:

A. Jumlah Kemunculan Minimum n-kali

Pemilihan atribut untuk model ini menggunakan *token* yang memiliki jumlah kemunculan minimum *n*-kali. Penelitian ini akan menggunakan jumlah kemunculan minimum 5-kali, 7-kali, dan 11-kali.

1) Tweet asli huruf normal

Hasil pada Tabel V menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan minimum 5-kali. Hal yang sama juga terjadi untuk hasil *precision*, *recall*, dan *F-score*. Jumlah kemunculan minimum 5-kali membuat atribut yang menjadi dasar perhitungan semakin banyak.

TABEL V
HASIL PENGUJIAN MODEL 1 JENIS 1

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
Min. 5-kali	*accuracy*	**83,038%**
	precision	**0,823**
	recall	**0,830**
	F-score	**0,822**
Min. 7-kali	*accuracy*	82,596%
	precision	0,817
	recall	0,826
	F-score	0,814
Min. 11-kali	*accuracy*	80,826%
	precision	0,814
	recall	0,809
	F-score	0,772

2) Tweet asli huruf kecil semua

Hasil pada Tabel VI menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan minimum 5-kali. Hal yang sama juga terjadi untuk hasil *recall* dan *F-score*. Jumlah kemunculan minimum 5-kali membuat atribut yang menjadi dasar perhitungan semakin banyak. Nilai *precision* terbaik terjadi pada kemunculan minimum 11-kali, hal ini mungkin karena atribut lebih sedikit, kesalahan klasifikasi data lebih sedikit.

TABEL VI
HASIL PENGUJIAN MODEL 1 JENIS 2

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
Min. 5-kali	*accuracy*	**82,596%**
	precision	0,785
	recall	**0,771**
	F-score	**0,775**
Min. 7-kali	*accuracy*	81,268%
	precision	0,771
	recall	0,733
	F-score	0,743
Min. 11-kali	*accuracy*	81,416%
	precision	**0,830**
	recall	0,695
	F-score	0,710

3) Tweet tanpa tanda baca huruf normal

Hasil pada Tabel VII menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan minimum 5-kali. Hal yang sama juga terjadi untuk hasil *recall* dan *F-score*. Jumlah kemunculan minimum 5-kali membuat atribut yang menjadi dasar perhitungan semakin banyak. Nilai *precision* terbaik terjadi pada kemunculan minimum 11-kali, hal ini mungkin karena atribut lebih sedikit, kesalahan klasifikasi data lebih sedikit.

TABEL VII
HASIL PENGUJIAN MODEL 1 JENIS 3

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
Min. 5-kali	*accuracy*	**84,071%**
	precision	0,815
	recall	**0,779**
	F-score	**0,792**
Min. 7-kali	*accuracy*	83,186%
	precision	0,808
	recall	0,762
	F-score	0,774
Min. 11-kali	*accuracy*	80,973%
	precision	**0,831**
	recall	0,672
	F-score	0,689

4) Tweet tanpa tanda baca huruf kecil semua

Hasil pada Tabel VIII menunjukkan bahwa nilai akurasi tertinggi diperoleh melalui atribut dengan *token* yang diambil dari jumlah kemunculan minimum 5-kali dan 11-kali. Hal mungkin dikarekan jenis *tweet* yang sudah dihapus

tanda bacanya dan menjadi huruf kecil semua. Sehingga atribut yang jumlah kemunculannya sedikit tidak terlalu berpengaruh.

TABEL VIII
HASIL PENGUJIAN MODEL 1 JENIS 4

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
Min. 5-kali	*accuracy*	**82,448%**
	precision	0,787
	recall	**0,788**
	F-score	**0,784**
Min. 7-kali	*accuracy*	81,858%
	precision	0,782
	recall	0,762
	F-score	0,767
Min. 11-kali	*accuracy*	**82,448%**
	precision	**0,833**
	recall	0,736
	F-score	0,753

5) Tweet formal tanpa tanda baca huruf normal

Hasil pada Tabel IX menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan minimum 5-kali. Hal yang sama juga terjadi untuk hasil *recall* dan *F-score*. Jumlah kemunculan minimum 5-kali membuat atribut yang menjadi dasar perhitungan semakin banyak. Nilai *precision* terbaik terjadi pada kemunculan minimum 11-kali, hal ini mungkin karena atribut lebih sedikit, kesalahan klasifikasi data lebih sedikit.

TABEL IX
HASIL PENGUJIAN MODEL 1 JENIS 5

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
Min. 5-kali	*accuracy*	**84,218%**
	precision	0,805
	recall	**0,782**
	F-score	**0,790**
Min. 7-kali	*accuracy*	83,776%
	precision	0,792
	recall	0,772
	F-score	0,777
Min. 11-kali	*accuracy*	83,038%
	precision	**0,859**
	recall	0,692
	F-score	0,701

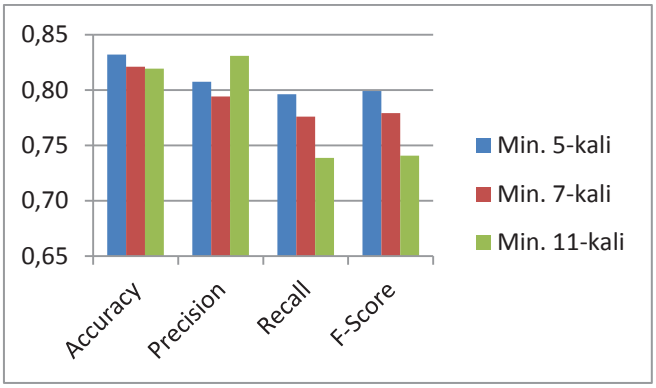
6) Tweet formal tanpa tanda baca huruf kecil semua

Hasil pada Tabel X menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan minimum 5-kali. Hal yang sama juga terjadi untuk hasil *precision*, *recall*, dan *F-score*. Jumlah kemunculan minimum 5-kali membuat atribut yang menjadi dasar perhitungan semakin banyak.

TABEL X
HASIL PENGUJIAN MODEL 1 JENIS 6

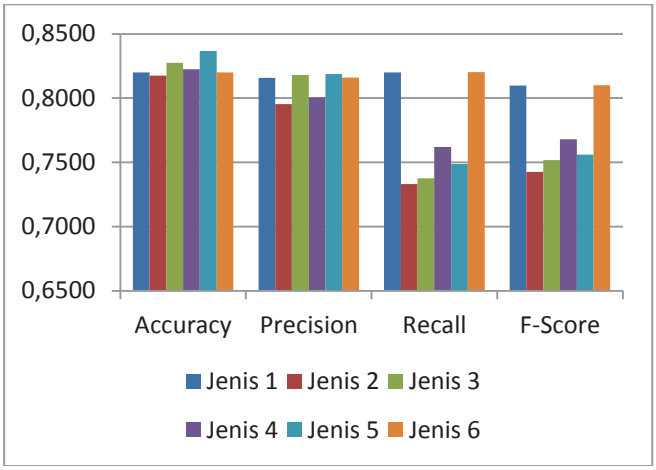
Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
Min. 5-kali	*accuracy*	**82,891%**
	precision	**0,826**
	recall	**0,829**
	F-score	**0,827**
Min. 7-kali	*accuracy*	81,268%
	precision	0,806
	recall	0,813
	F-score	0,807
Min. 11-kali	*accuracy*	81,858%
	precision	0,816
	recall	0,819
	F-score	0,796

Gambar 1 menunjukkan hasil keseluruhan pengujian untuk masing-masing jumlah kemunculan. Secara rata-rata, nilai akurasi, *recall*, dan *F-score* terbaik diperoleh ketika jumlah kemunculan min. 5-kali. Sedangkan untuk nilai *precision* terbaik diperoleh ketika jumlah kemunculan min. 11-kali. Pada model ini, jumlah kemunculan minimum 5-kali merupakan batasan yang paling baik untuk memperoleh nilai akurasi yang terbaik.



Gambar 1. Rata-rata hasil pengujian untuk semua jenis *tweet* terhadap jumlah kemunculan minimum *n*-kali.

Gambar 2 menunjukkan hasil keseluruhan pengujian untuk setiap jenis *tweet*, agak sedikit berbeda dengan Gambar 1. Nilai akurasi dan *precision* terbaik diperoleh pada *tweet* dengan jenis 5. Sedangkan nilai *recall* dan *F-score* terbaik diperoleh pada *tweet* dengan jenis 6. Hal ini mungkin dikarenakan pada ke-2 jenis tersebut, *tweet* sudah melalui tahap formalisasi dan penghapusan tanda baca. Kondisi ini membuat pemilihan atribut menjadi lebih seragam dan akurat.



Gambar 2. Rata-rata hasil pengujian untuk semua jumlah kemunculan minimum *n*-kali terhadap jenis *tweet*.

B. Jumlah Kemunculan ke-n Terbanyak

Pemilihan atribut untuk model ini menggunakan *token* dengan jumlah kemunculan ke-*n* terbanyak. Penelitian ini akan menggunakan jumlah kemunculan ke-5, ke-8, dan ke-12 terbanyak.

1) Tweet asli huruf normal

Hasil pada Tabel XI menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan ke-12 terbanyak. Hal yang sama juga terjadi untuk hasil *precision*, *recall*, dan *F-score*. Jumlah *token* yang dijadikan atribut lebih banyak dibanding lainnya sehingga lebih banyak nilai yang dapat ikut dihitung.

TABEL XI
HASIL PENGUJIAN MODEL 2 JENIS 1

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
ke-5 terbanyak	*accuracy*	78,614%
	precision	0,782
	recall	0,786
	F-score	0,738
ke-8 terbanyak	*accuracy*	82,301%
	precision	0,815
	recall	0,823
	F-score	0,807
ke-12 terbanyak	*accuracy*	**82,596%**
	precision	**0,817**
	recall	**0,826**
	F-score	**0,814**

2) Tweet asli huruf kecil semua

Hasil pada Tabel XII menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan ke-12 terbanyak. Hal yang sama juga terjadi untuk hasil *precision*, *recall*, dan *F-score*. Akan tetapi di jenis ini, nilai akurasi untuk *token* dengan jumlah kemunculan ke-8 terbanyak juga sama besarnya. Hal ini mungkin dikarenakan *tweet* telah diproses menjadi huruf kecil semua, sehingga menyebabkan *token* tidak sevariatif jenis 1.

TABEL XII
HASIL PENGUJIAN MODEL 2 JENIS 2

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
ke-5 terbanyak	*accuracy*	78,614%
	precision	0,778
	recall	0,786
	F-score	0,740
ke-8 terbanyak	*accuracy*	**81,268%**
	precision	0,801
	recall	0,813
	F-score	0,796
ke-12 terbanyak	*accuracy*	**81,268%**
	precision	**0,802**
	recall	**0,813**
	F-score	**0,803**

3) Tweet tanpa tanda baca huruf normal

Hasil pada Tabel XIII menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan ke-12 terbanyak. Hal yang sama juga terjadi untuk hasil *precision*, *recall*, dan *F-score*. Jumlah *token* yang dijadikan atribut lebih banyak dibanding lainnya sehingga lebih banyak nilai yang dapat ikut dihitung.

TABEL XIII
HASIL PENGUJIAN MODEL 2 JENIS 3

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
ke-5 terbanyak	*accuracy*	81,268%
	precision	0,817
	recall	0,813
	F-score	0,780
ke-8 terbanyak	*accuracy*	82,596%
	precision	0,819
	recall	0,826
	F-score	0,810
ke-12 terbanyak	*accuracy*	**83,038%**
	precision	**0,823**
	recall	**0,831**
	F-score	**0,819**

4) Tweet tanpa tanda baca huruf kecil semua

Hasil pada Tabel XIV menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan ke-12 terbanyak. Hal yang sama juga terjadi untuk hasil *precision*, *recall*, dan *F-score*. Pada tweet jenis ini, nilai akurasi, *precision*, dan *recall* untuk *token* dengan jumlah kemunculan ke-8 terbanyak juga

sama besarnya. Seperti pada *tweet* jenis 2, hal ini mungkin dikarenakan *tweet* telah diproses menjadi huruf kecil semua, sehingga menyebabkan *token* tidak sevariatif jenis 1. Selain itu, jenis ini juga sudah menghapus semua tanda baca yang ada pada *tweet*.

TABEL XIV
HASIL PENGUJIAN MODEL 2 JENIS 4

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
ke-5 terbanyak	*accuracy*	80,826%
	precision	0,798
	recall	0,808
	F-score	0,774
ke-8 terbanyak	*accuracy*	**82,153%**
	precision	**0,813**
	recall	**0,822**
	F-score	0,804
ke-12 terbanyak	*accuracy*	**82,153%**
	precision	**0,813**
	recall	**0,822**
	F-score	**0,814**

5) Tweet formal tanpa tanda baca huruf normal

Hasil pada Tabel XV menunjukkan bahwa akurasi terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan ke-8 terbanyak. Hal yang sama juga terjadi untuk nilai *recall* dan *F-score*. Sedangkan untuk nilai *precision* terbaik diperoleh pada jumlah kemunculan ke-5 terbanyak. Mulai *tweet* jenis ini, *tweet* telah melalui proses formalisasi dan tanda baca sudah dihapus semua. Hal ini mungkin menyebabkan variasi *tweet* lebih seragam.

TABEL XV HASIL PENGUJIAN MODEL 2 JENIS 5

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
ke-5 terbanyak	*accuracy*	83,038%
	precision	**0,859**
	recall	0,831
	F-score	0,795
ke-8 terbanyak	*accuracy*	**84,513%**
	precision	0,847
	recall	**0,845**
	F-score	**0,828**
ke-12 terbanyak	*accuracy*	83,481%
	precision	0,831
	recall	0,835
	F-score	0,822

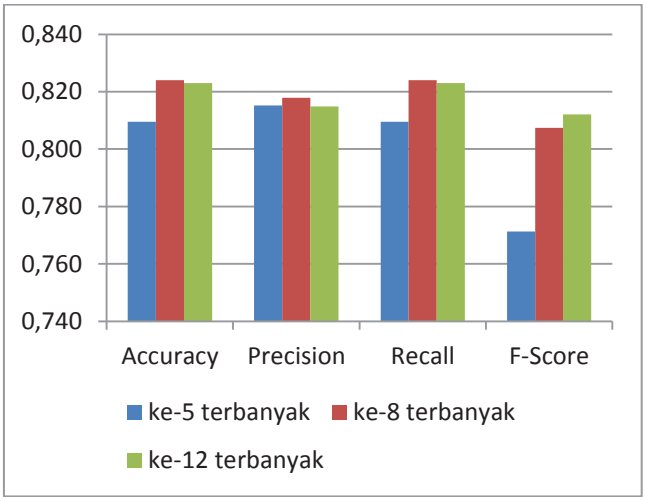
6) Tweet formal tanpa tanda baca huruf kecil semua

Hasil pada Tabel XVI menunjukkan bahwa nilai akurasi, *precision*, dan *recall* terbaik diperoleh melalui atribut dengan *token* yang diambil berdasarkan jumlah kemunculan ke-5 terbanyak. Sedangkan untuk nilai *F-score* terbaik diperoleh pada jumlah kemunculan ke-12 terbanyak.

TABEL XVI
HASIL PENGUJIAN MODEL 2 JENIS 6

Jumlah Kemunculan	Ukuran Pengujian	Rata-rata
ke-5 terbanyak	*accuracy*	**83,333%**
	precision	**0,858**
	recall	**0,834**
	F-score	0,801
ke-8 terbanyak	*accuracy*	81,563%
	precision	0,814
	recall	0,816
	F-score	0,801
ke-12 terbanyak	*accuracy*	81,268%
	precision	0,804
	recall	0,813
	F-score	**0,802**

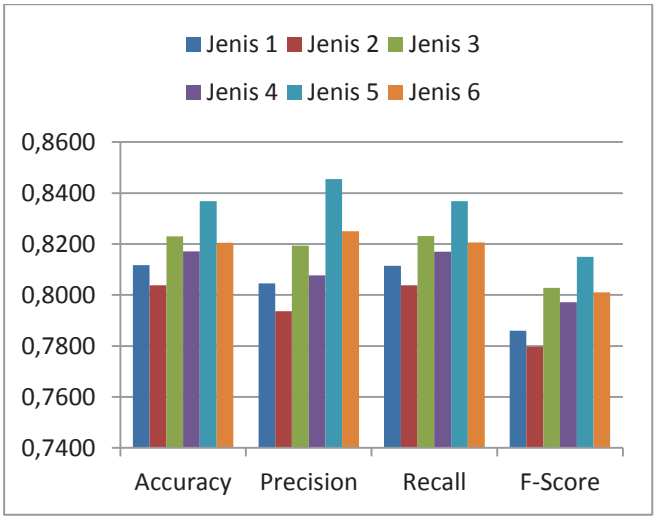
Gambar 3 menunjukkan hasil keseluruhan pengujian untuk masing-masing jumlah kemunculan. Secara rata-rata, nilai akurasi, *precision*, dan *recall* terbaik diperoleh ketika atribut diambil dari *token* dengan jumlah kemunculan ke-8 terbanyak. Sedangkan untuk nilai *F-score* terbaik diperoleh ketika jumlah kemunculan ke-12 terbanyak. Secara umum dapat dianggap bahwa jumlah kemunculan ke-8 terbanyak merupakan jumlah yang paling baik pada model ini untuk mendapatkan akurasi yang paling baik.



Gambar 3. Rata-rata hasil pengujian untuk semua jenis *tweet* terhadap jumlah kemunculan ke-*n* terbanyak.

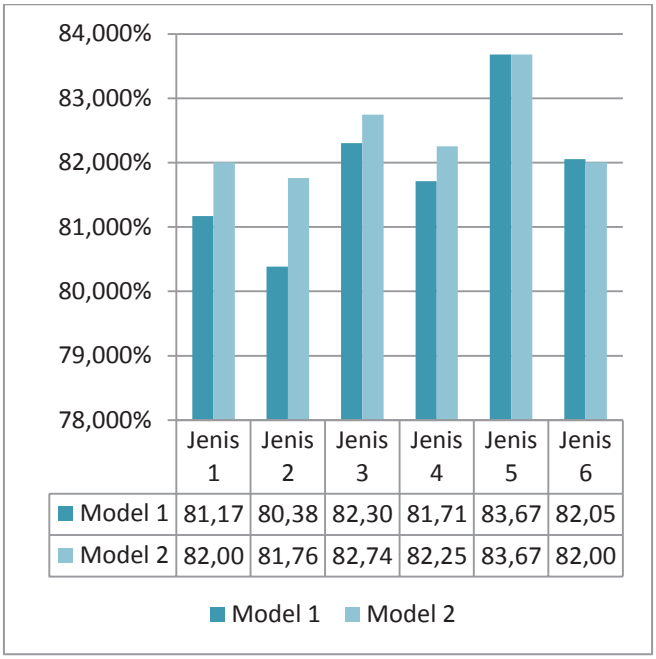
Gambar 4 menunjukkan hasil keseluruhan pengujian untuk setiap jenis *tweet*. Nilai akurasi, *precision*, *recall*, dan *precision* terbaik diperoleh pada *tweet* dengan jenis 5. Sama seperti hasil dari model 1, hal ini mungkin dikarenakan pada jenis ke-5, *tweet* sudah melalui tahap formalisasi dan

penghapusan tanda baca. Kondisi ini membuat pemilihan atribut menjadi lebih seragam dan akurat.



Gambar 4. Rata-rata hasil pengujian untuk semua jumlah kemunculan ke-*n* terbanyak terhadap jenis *tweet*.

Apabila dilihat secara keseluruhan, nilai akurasi terbaik untuk semua jenis *tweet*, sebagian besar diperoleh dari pemilihan atribut dengan model ke-2 (jumlah kemunculan ke-*n* terbanyak). Seperti terlihat pada Gambar 5, untuk *tweet* jenis 1 sampai 4, nilai akurasi tertinggi diperoleh dari model 2. Sedangkan untuk jenis 5, kedua model menghasilkan nilai yang sama. Untuk jenis 6, model 1 memiliki nilai akurasi lebih tinggi. Akan tetapi perbedaan nilainya juga hanya 0,05% saja.



Gambar 5. Perbandingan nilai akurasi model pemilihan token atribut dengan setiap jenis *tweet* yang ada.

VI. Simpulan

Berdasarkan hasil pengujian yang berhasil dilakukan, metode klasifikasi Naive Bayes dapat mendeteksi pengirim *tweet* merupakan orang yang sama atau tidak berdasarkan *tweet-tweet* sebelumnya. Nilai akurasi secara keseluruhan yang diperoleh juga cukup baik, yaitu 82,145%. Pengujian berhasil dilakukan untuk 2 model pemilihan atribut dan 6 jenis *tweet* yang berbeda. Hasil pengujian menunjukkan model 2 (pemilihan atribut berdasarkan token dengan jumlah kemunculan ke-*n* terbanyak) konsisten memiliki nilai akurasi yang lebih tinggi untuk setiap jenis *tweet*. Jenis *tweet* ke-5 (*tweet* formal tanpa tanda baca huruf normal) juga memperoleh nilai akurasi yang tertinggi untuk kedua model pemilihan atribut.

Token yang diperoleh masih dalam bentuk *unigram* (satu kata). Penelitian berikutnya mungkin dapat menguji apakah hasil tokenisasi dengan *bigram* memiliki nilai akurasi yang lebih baik atau tidak. Metode klasifikasi Naive Bayes yang digunakan pada penelitian ini masih *Binomial* Naive Bayes. Penelitian berikutnya juga mungkin dapat mencoba membandingkan dengan hasil dari metode *Multinomial* Naive Bayes.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Tim Lab Grafika dan Intelegensia Buatan dari Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung untuk sistem InaNLP yang digunakan untuk proses formalisasi.

Daftar Pustaka

[1] S. Kemp, "Digital, Social & Mobile in 2015," We Are Social, Singapore, 2015.

[2] N. I. Setyani, S. Hastjarjo and N. N. Amal, "Penggunaan Media Sosial Sebagai Sarana Komunikasi Bagi Komunitas (Studi Deskriptif Kualitatif Penggunaan Media Sosial Twitter, Facebook, dan Blog sebagai Sarana Komunikasi bagi Komunitas Akademi Berbagi Surakarta)," Program Studi Ilmu Komunikasi Fakultas Ilmu Sosial dan Ilmu Politik Universitas Sebelas Maret, Surakarta, 2013.

[3] A. H. Manumoyoso, "Penipu Lewat SMS Ditangkap di Jakarta - Kompas.com," 7 Oktober 2013. [Online]. Available: http://megapolitan.kompas.com/read/2013/10/07/2129518/Penipu.Lewat.SMS.Ditangkap.di.Jakarta. [Accessed 4 Agustus 2015].

[4] R. K., "Waspada, Modus Penipuan Via Twitter," 10 April 2012. [Online]. Available: http://tekno.tempo.co/read/news/2012/04/10/072395905/waspada-modus-penipuan-via-twitter. [Accessed 7 Agustus 2015].

[5] H. Liauw, "Kejahatan di Dunia Maya Kian Berbahaya - Kompas.com," 2014 Oktober 2014. [Online]. Available: http://megapolitan.kompas.com/read/2014/10/24/17561041/Kejahatan.di.Dunia.Maya.Kian.Berbahaya. [Accessed 4 Agustus 2015].

[6] W. Stefanus and M. C. Wijanto, "Detection the Similarity of the Message Sender On Short Message Service (SMS)," in *PACLING 2015 (Pacific Association for Computational Linguistics Conference)*, Bali, 2015.

[7] H. Purohit, A. Hampton, V. L. Shalin, A. P. Sheth, J. Flach and S. Bhatt, "What Kind of #Conversation is Twitter? Mining #Psycholinguistic Cues for Emergency Coordination," *Computers in Human Behavior,* vol. 29, no. 6, pp. 2438-2447, November 2013.

[8] T. O. Ugheoke, "Detecting the Gender of a Tweet Sender," M.Sc. Project Report, Department of Computer Science, University of Regina, Regina, 2014.

[9] "Company | About," 30 Juni 2015. [Online]. Available: https://about.twitter.com/company. [Accessed 9 Agustus 2015].

[10] W. B. Croft, D. Metzler and T. Strohman, Search Engines Information Retrieval in Practice, Boston: Pearson Education, Inc., 2010.

[11] D. Jurafski and J. H. Martin, Speech and Language Processing, 2nd Edition ed., New Jersey: Prentice Hall, 2009.

[12] J. Han, M. Kamber and J. Pei, Data Mining Concepts and Techniques, Massachusetts: Morgan Kaufmann, 2012.

[13] T. M. Mitchell, Machine Learning, United States of America: McGraw-Hill, 1997.

[14] C. D. Manning, P. Raghavan and H. Schutze, Introduction to Information Retrieval, Cambridge: Cambridge University Press, 2008.

[15] P. Gamallo and M. Garcia, "Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets," in *8th International Workshop on Semantic Evaluation (SemEval 2014)*, Dublin, 2014.

[16] B. T. Purnama, "Twitter.com," [Online]. Available: http://www.twitter.com/@basukibtp. [Accessed 1 Agustus 2015].

[17] I. H. Witten, E. Frank and M. A. Hall, Data Mining Pratical Machine Learning Tools and Techniques, Massachusetts: Morgan Kaufmann, 2011.