MATH*unesa*
Jurnal Ilmiah Matematika *Volume 7 No.3 Tahun 2019*
ISSN 2301-9115

DETEKSI UJARAN KEBENCIAN PADA TWITTER MENJELANG PILPRES 2019 DENGAN MACHINE LEARNING

Dayang Putri Nur Lyrawati

Jurusan Matematika, FMIPA, Universitas Negeri Surabaya
e-mail : dayangl@mhs.unesa.ac.id

Abstrak

Ujaran kebencian merupakan salah satu dari analisis sentimen atau opinion mining. Salah satu data yang representative untuk studi ujaran kebencian adalah data Twitter. Untuk mendeteksi teks yang mengandung ujaran kebencian pada twitter, diperlukan sebuah algoritme pada machine learning yang dapat mengklasifikasikan data berbentuk teks. Digunakan algoritme SVM-RBF Kernel dengan data tweet menjelang debat pertama pilpres. Deteksi ujaran kebencian yang telah dilakukan menghasilkan akurasi tertinggi sebesar 61,667% dengan waktu 1,21 detik.

Kata kunci: Analisis sentimen, Twitter, SVM, Ujaran Kebencian

Abstract

Hate speech is one of sentiment analysis or opinion mining. One representative data for hate speech studies is Twitter data. To detect text containing hate speech on Twitter, an algorithm in machine learning is required which can classify data in the form of text. The SVM-RBF Kernel algorithm is used with tweet data ahead of the first election debate. Detection of hate speech that has been done produces the highest accuracy of 61.667% with 1.21 seconds.

Keywords : Hate Speech; SVM; Sentiment Analysis; Twitter

1. PENDAHULUAN

Indonesia tercatat sebagai pengguna internet terbanyak di dunia dengan 85% digunakan untuk mengakses media sosial (Kominfo, 2013). Salah satu situs media sosial yang sering diakses adalah twitter. Saat ini twitter merupakan sebuah indikator yang baik untuk memberikan pengaruh dalam penelitian (Putranti, Dwi, & Edi, 2014). Terutama di bidang analisis sentimen atau *opinion mining*. Analisis sentimen atau opinion mining merupakan proses memahami, mengekstrak dan mengolah data teks guna mendapatkan informasi sentimen yang terkandung dalam kalimat opini (Buntoro, 2016). Terdapat banyak metode yang bisa digunakan untuk sentimen analisis, pada paper ini metode yang akan digunakan adalah SVM-RBF Kernel. Beberapa penelitian yang telah dilakukan sebelumnya menyatakan bahwa kedua metode tersebut memiliki akurasi yang tinggi dalam klasifikasi data teks. Penelitian telah yang dilakukan oleh Bhumika dan Vimalkumar (2016) dengan menganalisis tiga dataset (Gold, Movie, Twitter) menggunakan metode SVM dan Naïve Bayes menghasilkan akurasi (Gold 72,74% ; Movie 74,73%; Twitter 76,92%) dan (Gold 69,10% ; Movie 74,55% ; Twitter 76,67%) membuktikan bahwa SVM memiliki akurasi yang lebih tinggi dari Naïve Bayes (Jadav & Vaghel, 2016). Selain itu, dengan metode yang sama penelitian yang telah dilakukan Ghulam Asrofi Buntoro pada Tahun 2017 menggunakan data sentimen ujaran kebencian twitter bahasa Indonesia menghasilkan akurasi sebesar 66,6% dan 63,7% (Buntoro, 2017). Analisis sentimen terutama ujaran kebencian ini sering sekali digunakan di bidang perdagangan, promosi, pendidikan, politik, pemerintahan maupun kampanye (Hasan, Moin, Karim, & Shamshirband, 2018). Bertepatan dengan saat ini merupakan tahun politik. Beberapa penelitian di bidang politik pemerintahan sudah dilakukan, salah satunya Budiharto and Meiliana, menggunakan analisis sentimen untuk memprediksi pilihan rakyat dalam pemilihan presiden 2019 menggunakan bahasa R dan *packages* hasil prediksi mengatakan Jokowi memimpin arus prediksi pemilihan dan terus meningkat hingga saat ini. Hasil prediksi ini sesuai ke empat lembaga survei di Indonesia; Indikator, Cyrus Networks, Litbang Kompas dan Poltracking

sebagaimana disebutkan dalam Detik News (Budiharto & Meiliana, 2018). Sebelum itu, Josemar dkk pada tahun 2018 menganalisis homofilia politik Amerika Serikat dalam tiga skenario. Pertama, menganalisis *Twitter follow*, *mention* dan *retweet* koneksi baik searah dan timbal balik. Dalam skenario kedua, menganalisis koneksi multipleks dan yang ketiga, menganalisis pertemanan dengan yang serupa pidato. Hasil yang didapat adalah pengguna negatif, pengguna yang mendukung Trump, dan pengguna yang mendukung Hillary (Caetano, 2018). Berdasarkan dari beberapa literatur tersebut, akan dilakukan penelitian tentang deteksi ujaran kebencian pada twitter menjelang pilpres 2019 dengan menggunakan algoritme SVM.

2. KAJIAN TEORI

Hatespeech (Ujaran Kebencian)

Analisis sentimen dan pengembangan pendapat (*opinion mining)* adalah bidang studi yang menganalisis pendapat seseorang, sentimen seseorang, evaluasi seseorang, sikap seseorang dan emosi seseorang ke dalam bahasa tertulis. Ujaran kebencian dapat ditemukan dalam analisis sentimen. Ujaran Kebencian adalah "perkataan yang meremehkan orang atau grup berdasarkan beberapa karakteristik seperti ras, etnis, jenis kelamin, orientasi seksual, kebangsaan, agama, atau karakteristik lainnya“ (Alfina, Mulia, & Fanany, 2017). Dalam hukum ujaran kebencian (Hate Speech) adalah perkataan, perilaku, tulisan, ataupun pertunjukan yang dilarang karena dapat memicu terjadinya tindakan kekerasan dan sikap prasangka entah dari pihak pelaku pernyataan tersebut ataupun korban dari tindakan tersebut (Putra et al., 2018).

SVM (Support Vector Machine)

SVM (*Support Vector Machine*) adalah seperangkat metode *machine learning* yang dapat menganalisis dan mengenali pola. SVM digunakan untuk klasifikasi dan analisis regresi (Buntoro, 2017). SVM pertama kali dikenalkan oleh Vapnik, Boser dan Guyon tahun 1992 pada *Workshop on Computational Learning.* Metode yang digunakan pada SVM tergolong baru yaitu dengan mencari *hyperplane* terbaik pada ruang input. Konsep pada pembelajaran SVM secara sederhana sebagai usaha mencari *hyperplane* (*hyperplane* dalam ruang vektor berdimensi *d* adalah subruang afin berdimensi *d-1* yang membagi ruang vektor tersebut ke dalam dua bagian, yang masing-masing berkorespondensi pada kelas yang berbeda) terbaik yang berfungsi sebagai pemisah dua kelas pada ruang input (*input space)* (Ahmad & Ali, 2017). Hyperplane dapat berupa garis pada dimensi dua dan dapat berupa bidang datar pada *multiple dimension*.

Hyperplane pemisah terbaik antara kedua *kelas* dapat ditemukan dengan mengukur *margin* hyperplane tersebut dan mencari titik maksimalnya. Margin adalah jarak dari dua *support vector* kelas yang berbeda. *Pattern* yang paling dekat ini disebut support vector. SVM awalnya hanya fokus untuk klasifikasi data linier *(linear classifier),* kemudian SVM dikembangkan pada masalah non-linier dengan memasukkan fungsi kernel**.**

SVM yang digunakan untuk menyelesaikan masalah linear yang disebut *Linear Separable Data* yaitu data yang dapat dipisahkan secara linier dari 2 kelas yaitu kelas -1 dan kelas 1.

Diberikan himpunan data $D = \{X_i, y_i\}$ dimana X_i merupakan himpunan i tupel berurutan dan y_i merupakan label dari tupel berurutan tersebut. Persamaan hyperplane pemisah dapat ditulis :

$$\mathrm{W} \, . X + b = 0 \qquad (1)$$

Dengan, W= $\{w_1, w_2, w_{3,} \dots, w_n\}$ w_n adalah vektor bobot dari n atribut dan b adalah bias. Untuk memperoleh hasil pembelajan yang bagus, margin harus dimaksimalkan. Data yang masuk pada kelas -1 adalah data yang memenuhi persamaan (2), sedangkan data yang masuk kelas +1 adalah data yang memenuhi persamaan (3).

$$(w_i . x_i) + b = -1 \qquad (2)$$
$$(w_i . x_i) + b = +1 \qquad (3)$$

Terdapat kemungkinan bahwa titik pada dua kelas tidak dapat dipisah dengan sebuah *hyperplane* pada ruang yang sebenarnya. Untuk itu diperlukan ruang fitur yang lebih tinggi untuk mentransformasikan setiap data sehingga dapat dipisahkan secara linier oleh *hyperplane*. Masalah tersebut dapat dikatakan *linear separable* (non-linier) dan dapat diatasi dengan fungsi *kernel trick*. *Kernel trick* merupakan fungsi yang memetakan fitur yang berdimensi rendah ke berdimensi tinggi. Karena umumnya transformasi fungsi ini $\boldsymbol{\phi}$ dengan diketahui maka perhitungan hasil kali titik (*dot product)* sesuai teori Mercer dapat digantikan fungsi kernel $K(\vec{x}_i, \vec{x}_j)$ yang mendefinisikan secara implisit transformasi $\boldsymbol{\phi}$. Inilah yang disebut *Kernel Trick:*

$$\boldsymbol{K(\vec{x}_i, \vec{x}_j) = \phi(\vec{x}_i) . \phi(\vec{x}_j)} \qquad \boldsymbol{(4)}$$

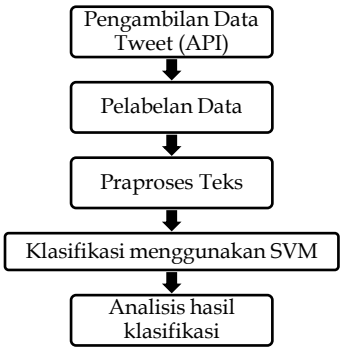
Kernel trick yang digunakan dalam penelitian ini yaitu SVM-RBF Kernel. (Mishra & Lotia, 2014)

$$K(x,y)=exp\ (\frac{-\|x-y\|^2}{2\sigma^2}) \qquad (5)$$

3. METODE

Penelitian ini terdiri dari dua proses yaitu praproses data dan pengklasifikasian data. Praproses data teks sangat diperlukan untuk menghilangkan informasi yang tidak dibutuhkan dalam klasifikasi. Praproses data teks akan diterapkan pada kedua data teks. Seluruh data teks tersebut akan melalui tahap tokenisasi (pemecahan kata), Cleansing (penghapusan simbol-simbol), Filtering (penghapusan kata-kata yang tidak diperlukan), Stemming (pengubahan menjadi kata dasar). Pada penelitian ini digunakan stemmer Sastrawi yang diakses

dari library python. Proses penelitian diilustrasikan pada Gambar 1.



Gambar 1. Bagan Proses Penelitian

1. Pengumpulan Data
   Data *twitter* yang digunakan sebagai sumber data dan dikumpulkan menggunakan *Twitter Streaming API* dengan mendapatkan *API Key* (twitterdev, n.d.) terlebih dahulu dan *tools Tweepy* pada *Pyhton 3.1* dengan menggunakan *source code* (computermacgyver, 2015) . Teks *tweet* yang digunakan mengacu pada kegiatan politik Pemilihan Umum 2019. Dataset yang digunakan adalah data yang didapat setelah dilakukan penyaringan teks yang duplikat maupun memiliki makna yang sama dan data yang tidak berkaitan dengan penelitian.
2. Pelabelan data
   Pada fase ini setiap dataset akan diberi label. Hanya menggunakan dua label. *Tweet* yang mengandung ujaran kebencian diberi label 'HS' dan *tweet* yang tidak mengandung ujaran kebencian diberi label 'Non_HS'. Pelabelan dilakukan dengan cara membuat *Google Forms* yang dapat diakses oleh mahasiswa dengan berbagai latar belakang agama, daerah serta jenis kelamin yang berbeda. Setiap *Google Forms* terdiri atas 40 tweets yang dapat diakses oleh 5 orang saja.
3. Praproses Teks
   Praproses data input sangat penting dalam tahap klasifikasi. Tujuan dari praproses dokumen adalah untuk menghilangkan *noise*, menyeragamkan kata, mengurangi volume kata dan memudahkan dalam mengklasifikasi data. Tahap praproses yang digunakan terdiri atas proses *Stemming*, *Tokenizing*, *Cleansing*, *Filtering.*
   1. *Stemming* yaitu proses mengubah kata berimbuhan menjadi kata dasar.
   2. *Tokenizing* atau tokenisasi dilakukan untuk memecah tweet menjadi beberapa kata atau kumpulan kata yang berdiri sendiri.
   3. *Cleansing* yaitu proses membersihkan simbol-simbol yang kurang penting dalam data tweet yang bisa mengganggu proses klasifikasi.
   4. *Filtering* dilakukan untuk menghapus kata-kata yang kurang relevan terhadap proses klasifikasi.
   5. *String Word to Vector* dilakukan untuk mengubah data teks menjadi bentuk vektor numeric yang selanjutnya akan diklasifikasi menggunakan matlab.
4. Klasifikasi menggunakan algoritme SVM
   Data yang sudah malalui tahap praproses, kemudian diklasifikasikan menggunakan algoritme SVM. Output hasil klasifikasi adalah *confusion matrix*, besar akurasi dan waktu hasil klasifikasi.

5. DATASET

Data yang akan digunakan dalam penelitian ini diambil langsung dari *Tweepy API* yang mengandung kata "#pilpres2019", "#2019gantipresiden", "#debatcapres", "debatpilpres" dan "#jokowi2periode". Menggunakan tagar sebagai kata kunci dalam mengekstraksi data dari Twitter adalah cukup umum (Alfina, Sigmawaty, Nurhidayati, & Hidayanto, 2017). Data yang diambil mulai tujuh hari sebelum debat pilpres hingga tujuh hari setelah debat berlangsung yaitu tanggal 10 Januari 2019 sampai 24 Januari 2019. Berdasarkan analisis data dari SocialBarel yang dicantumkan dalam web, waktu terbaik untuk memposting sesuatu di twitter adalah setelah jam 11 siang dan mulai tumbuh mencapai puncak pada jam 3 sore dan menghindari memposting diatas jam 8 malam (Marikxon, n.d.). Maka dari itu pengambilan data teks twitter sekitar jam 11 hingga sekitar jam 7 malam. Data yang didapat sebanyak 54650 selanjutnya data teks twitter akan disaring. Untuk data yang berduplikat, memiliki arti yang sama, maupun tidak ada kaitan dengan penelitian tidak akan dijadikan dataset. Setelah itu, dataset akan dilakukan pelabelan. Data yang akan masuk dalam proses pelabelan sebanyak 597. Dengan 320 berlabel HS dan 277 berlabel Non_HS.

6. PEMBAHASAN

Dataset yang telah diperoleh, diubah ke format csv untuk dapat diinputkan pada Weka. Data yang telah terinput pada weka kelas tweet terbaca sebagai tipe nominal sedangkan pada kelas label terbaca sebagai tipe String. Untuk dapat melakukan preprocessing data, data Tweet harus diubah ke tipe String. Maka dari itu data harus disimpan ke bentuk format ARFF (*Attribute Relation File Format*).

Setelah data bertipe string, dilakukan preproses data yaitu Stemming untuk menyesuaikan stemmer yang ada pada Weka data di translate ke dalam Bahasa Inggris. Preproses data yang digunakan pada studi ini menggunakan tools menu yang ada di Weka yaitu StringWordToVector. Pada tahap Stemming, yang digunakan adalah *Stemmer online* yang bersumber dari sastrawi, sedangkan pada tahap Tokenizing menggunakan default Weka. Selanjutnya, Cleansing juga dilakukan pada menu yang sama hanya mengganti delimiters dengan symbol-simbol yang tidak digunakan. Seperti angka dan symbol berikut ",.';:/\)(*&^%$#@+-=. Tahap terakhir dari preproses data adalah filtering. Setelah itu, data diklasifikasikan menggunakan menu pada Weka.

Akurasi dihitung menggunakan rumus:

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

Hasil akurasi dan waktu yang dihasilkan dari penelitian ini disajikan pada Tabel 1. Hasil output yang dihasilkan pada Weka tidak hanya akurasi, tetapi nilai TP, FP, Precision, dan Recall. *True Positive* (TP) adalah banyaknya klasifikasi benar yang diklasifikasi ke kelas 1 (*HS*), *False Positive* (FP) adalah banyaknya klasifikasi salah yang diklasifikasi ke kelas 1, True Negative (TN) adalah banyaknya klasifikasi benar yang diklasifikasi ke kelas 0 (Non_HS), dan False Negative (FN) adalah banyaknya klasifikasi salah yang diklasifikasi ke kelas 0 (Arora, 2012).

TP *Rate* atau disebut dengan *sensitivity* dirumuskan sebagai berikut:

$$TP\ Rate = \frac{TP}{TP+FN} \quad (7)$$

FP *Rate* dirumuskan sebagai berikut:

$$FP\ Rate = \frac{FP}{TN+FP} \quad (8)$$

Precision adalah tingkat ketepatan antara informasi dengan prediksi, dirumuskan sebagai berikut:

$$Precision = \frac{TP}{TP+FP} \quad (9)$$

Recall sama dengan TP *Rate* yaitu tingkat keberhasilan dalam melakukan klasifikasi pada suatu kelas, memiliki formula sebagai berikut:

$$Recall = \frac{TP}{TP+FN} \quad (10)$$

Nilai TP Rate, FP Rate, Precision dan Recall yang telah dihasilkan tertera dalam Tabel 7. Selain keempat nilai tersebut, output yang dihasilkan pada Weka juga menghasilkan nilai F-Measure, MCC, ROC dan PRC. Keempat nilai yang lainnya disajikan dalam Tabel 8.
F-*Measure* adalah *harmonic mean* dari *precision* dan *recall*, dirumuskan sebagai berikut:

$$F = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall} \quad (11)$$

Matthews correlation coefficient (MCC) adalah koefisien korelasi yang dihitung dari keempat *confusion matrix*, dirumuskan sebagai berikut (Saito & Rehmsmeier, 2015):

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{((TP+FP)(TP+FN)(TN+FP)(TN+FN))^{1/2}} \quad (12)$$

Kurva *Receiver Operating Characteristic* (ROC) dibuat berdasarkan nilai telah didapatkan pada perhitungan antara FP *Rate* (sumbu x) dengan TP *Rate* (sumbu y). *Baseline* pada ROC adalah garis lurus diagonal dari (0,0) ke (1,1). Klasifikasi dikatakan kurang baik apabila mendekati titik (0,0) dan klasifikasi dikatakan baik ketika mendekati titik (1,1) (Saito & Rehmsmeier, 2015).

Kurva *Precision-Recall* (PRC) dibuat berdasarkan nilai telah didapatkan pada perhitungan antara *Precision* (sumbu y) dan *Recall* (sumbu x). *Baseline* pada PRC ditentukan dengan rasio data pada tiap kelas. Jika data seimbang maka *baseline* yang digunakan adalah y=0.5 dan ketika data tidak seimbang maka *baseline* yang digunakan adalah y=0.1(Saito & Rehmsmeier, 2015).

Hasil studi yang telah dilakukan pada data teks twitter menjelang pemilihan presiden menggunakan algoritme SVM-RBF Kernel disajikan pada Tabel 1.

Tabel 1. Hasil Klasifikasi dengan parameter epsilon 1.00E-12

C	Rasio	Akurasi (%)	Waktu (s)
1	9:1	**61,667**	1,21
	8:2	57,1429	1,47
	7:3	59,7765	0,98
5	9:1	60	1,77
	8:2	57,1429	1,08
	7:3	57,1429	0,97
10	9:1	60	1,15
	8:2	57,9832	0,91
	7:3	60,3352	1,09

Pada Tabel 1, Tabel 2, Tabel 3, terdapat dua nilai parameter yaitu C dan ε. Parameter C menyatakan threshold, sedangkan nilai ε menyatakan level akurasi dari fungsi aproksimasi yang dapat dapat berpengaruh terhadap penggunaan support vector. Dari Tabel 1 didapat nilai akurasi terbesar pada rasio 9:1 yaitu sebesar 61,667 % dicapai saat nilai C=1, ε = 1.00E-12 dengan waktu 1,21 detik. Sama seperti yang dihasilkan pada Tabel 2 dengan parameter ε = 1.00E-09, akurasi terbesar dicapai saat parameter C=1 dengan waktu 1,04 detik.

Tabel 2. Hasil Klasifikasi dengan parameter epsilon 1.00E-09

C	Rasio	Akurasi (%)	Waktu (s)
1	9:1	**61,667**	1,04
	8:2	57,9832	0,42
	7:3	59,7765	0,92
5	9:1	60	1,34
	8:2	57,1429	1,31
	7:3	60,3352	1,37
10	9:1	60	1,13
	8:2	57,1429	1,06
	7:3	60,3352	1,2

Pada Tabel 3 dengan parameter ε=1.00E-06 dan C=1 didapat nilai akurasi terbesar 61,667% terletak pada rasio 9:1 dengan waktu 0,97 detik. Berdasarkan hasil deteksi ujaran kebencian pada ketiga tabel diatas, menunjukan untuk rasio 9 : 1 akurasi tertinggi dicapai saat parameter seluruh nilai *epsilon* dan C = 1 yaitu sebesar 61,667%. Hal ini dikarenakan pembagian data latih yang lebih banyak dibanding data uji.

Tabel 3. Hasil Klasifikasi dengan parameter epsilon 1.0E-06

C	Rasio	Akurasi (%)	Waktu (s)
1	9:1	**61,667**	0,97
	8:2	58,8235	0,94
	7:3	59,2179	1,36
5	9:1	60	1,13
	8:2	57,1429	1,12
	7:3	60,3352	0,45
10	9:1	60	0,96
	8:2	57,1429	0,97
	7:3	60,3352	1,1

Hasil akurasi yang disajikan pada ketiga tabel diatas, didapat dari nilai TP, FP, TN, FN yang ada pada *Confusion Matrix.* Confusion matriks yang didapat disajikan dalam Tabel 4, Tabel 5, dan Tabel 6. Nilai TP pada Tabel 4 sebesar 30, pada Tabel 5 sebesar 58 dan pada Tabel 5 sebesar 69. Sedangkan, untuk Nilai TN pada Tabel 4 sebesar 7, Tabel 5 sebesar 11, Tabel 6 sebesar 16.

Tabel 4. Confusion Matrix pada rasio 9:1, C=1 dan ε=1.00E-12

		Prediksi	
		1	0
Kelas	1	30	0
	0	23	7

Tabel 5. Confusion Matriks pada rasio 8:2 C=1 dan ε=1.00E-12

		Prediksi	
		1	0
Kelas	1	58	2
	0	48	11

Tabel 6. Confusion Matriks pada rasio 7:3 C=1 dan ε=1.00E-12

		Prediksi	
		1	0
Kelas	1	91	3
	0	69	16

Deteksi ujaran kebencian dengan algoritme SVM menghasilkan confusion matriks seperti yang ada pada ketiga tabel diatas. Dengan hasil deteksi terbaik ada pada rasio 9:1, hal itu terlihat pada nilai TP dan TN yang dihasilkan yaitu kelas 1 terklasifikasikan secara benar seluruhnya. Sedangkan untuk nilai TN terklasifikasikan secara benar hanya kurang dari setengah data yang ada pada kelas 0.

Tabel 7. Nilai TP Rate, FP Rate, Precision dan Recall

Rasio	TP Rate	FP Rate	Precision	Recall	Class
9:1	1,000	0,767	0,566	1,000	1
	0,233	0,000	1,000	0,233	0
	0,617	0,383	0,783	0,617	Avg
8:2	0,967	0,814	0,547	0,967	1
	0,186	0,033	0,846	0,186	0
	0,580	0,427	0,695	0,580	Avg
7:3	1,000	0,800	0,556	1,000	1
	0,200	0,000	1,000	0,200	0
	0,600	0,400	0,778	0,600	Avg

Tabel 7 dan Tabel 8 menunjukkan hasil dari deteksi ujaran kebencian yang tertera pada Weka. Dengan menggunakan persamaan yang sudah disebutkan diatas yaitu Persamaan (7), Persamaan (8), Persamaan (9), Persamaan (10), Persamaan (11) dan Persamaan (12). Untuk nilai ROC area dan PRC area, akan digunakan untuk membangun model dari hasil klasifikasi. Berikut Tabel 8 untuk mendukung hasil dari deteksi :

Tabel 8. Nilai F-Measure, ROC Area dan PRC Area

Rasio	F-Measure	MCC	ROC Area	PRC Area	Class
9:1	0,723	0,363	0,617	0,566	1
	0,378	0,363	0,617	0,617	0
	0,551	0,363	0,617	0,591	Avg
8:2	0,699	0,245	0,577	0,546	1
	0,306	0,245	0,577	0,561	0
	0,504	0,245	0,577	0,553	Avg
7:3	0,714	0,333	0,600	0,556	1
	0,333	0,333	0,600	0,600	0
	0,524	0,333	0,600	0,578	Avg

7. KESIMPULAN

Pada penelitian ini, deteksi ujaran kebencian pada data twitter menjelang pemilihan presiden 2019 dengan machine learning menggunakan algoritme SVM-RBF Kernel. Parameter yang digunakan pada algoritme SVM-RBF Kernel adalah C dengan nilai 1, 5, dan 10. Serta epsilon (ε) dengan nilai 10^{-12}, 10^{-9} dan 10^{-6}. Berdasarkan penelitian yang telah dilakukan, dihasilkan akurasi tertinggi sebesar 61,667% dicapai saat rasio 9:1 dengan ketiga nilai epsilon yang berbeda dengan C=1.

DAFTAR PUSTAKA

Ahmad, M., & Ali, I. (2017). Sentiment Analysis of Tweets using SVM, *177*(5), 25–29.

Alfina, I., Mulia, R., & Fanany, M. I. (2017). Hate

Speech Detection in the Indonesian Language : A Dataset and Preliminary Study, (October). https://doi.org/10.1109/ICACSIS.2017.8355039

Alfina, I., Sigmawaty, D., Nurhidayati, F., & Hidayanto, A. N. (2017). Utilizing Hashtags for Sentiment Analysis of Tweets in The Political Domain, 43–47.

Arora, R. (2012). Comparative Analysis of Classification Algorithms on Different Datasets using WEKA, *54*(13), 21–25.

Budiharto, W., & Meiliana, M. (2018). Prediction and analysis of Indonesia Presidential election from Twitter using sentiment analysis. *Journal of Big Data*, 1–10. https://doi.org/10.1186/s40537-018-0164-1

Buntoro, G. A. (2016). Analisis Sentiment Hatespeech pada Twitter dengan Metode Naive Bayes Classifier dan Support Vector Machine, *5*(September).

Buntoro, G. A. (2017). Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter, *2*(1), 32–41.

Caetano, J. (2018). Using sentiment analysis to define twitter political users ' classes and their homophily during the 2016 American presidential election.

computermacgyver. (2015). twitter-python. Retrieved December 24, 2018, from https://github.com/computermacgyver/twitter-python

Hasan, A., Moin, S., Karim, A., & Shamshirband, S. (2018). Machine Learning-Based Sentiment Analysis for Twitter Accounts. https://doi.org/10.3390/mca23010011

Jadav, B., & Vaghel, V. (2016). Sentiment Analysis using Support Vector Machine based on Feature Selection and Semantic Analysis. *International Journal of Computer Applications*, *146*(13), 26–30.

Kominfo. (2013). Kementerian Komunikasi dan Informatika. Retrieved January 30, 2019, from https://www.kominfo.go.id/content/detail/3415/kominfo-pengguna-internet-di-indonesia-63-juta-orang/0/berita_satker

Marikxon. (n.d.). Kapan Waktu Terbaik Untuk Posting di Media Sosial? Retrieved January 28, 2019, from https://www.maxmanroe.com/kapan-waktu-terbaik-untuk-posting-di-media-sosial.html

Mishra, P., & Lotia, P. (2014). Comparative Performance Analysis of SVM Speaker Verification System using Confusion Matrix, *3*(12), 2012–2015.

Putra, B. P., Irawan, B., Setianingsih, C., Elektro, F. T., Telkom, U., & Learning, D. (2018). Deteksi Ujaran Kebencian Dengan Menggunakan Algoritma Convolutional Neural Network Pada Gambar Hatespeech Detection Using Convolutional Neural Network Algorithm Based on Image, *5*(2), 2395–2402.

Putranti, Dwi, N., & Edi, W. (2014). Analisis Sentimen Twitter untuk Teks Berbahasa Indonesia dengan Maximum Entropy dan Support Vector Machine, *8*(1), 91–100.

Saito, T., & Rehmsmeier, M. (2015). The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets, 1–21. https://doi.org/10.1371/journal.pone.0118432

twitterdev. (n.d.). Developer Twitter. Retrieved from https://developer.twitter.com/en/apps