Available at https://jtsiskom.undip.ac.id (31 October 2021)
DOI:10.14710/jtsiskom.2021.13907

Jurnal Teknologi dan Sistem Komputer, 9(4), 2021, 199-204

Algoritme decision tree untuk mendeteksi ujaran kebencian dan bahasa kasar multilabel pada Twitter berbahasa Indonesia

Decision tree algorithm for multi-label hate speech and abusive language detection in Indonesian Twitter

Fauzi Ihsan, Iwan Iskandar, Nazruddin Safaat Harahap, Surya Agustian*)

Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim Riau
Jl. H.R. Soebrantas km 11.5 Simpang Baru Panam, Pekanbaru, Riau 28293, Indonesia

Cara sitasi: F. Ihsan, I. Iskandar, N. S. Harahap, and S. Agustian, "Algoritme decision tree untuk mendeteksi ujaran kebencian dan bahasa kasar multilabel pada Twitter berbahasa Indonesia," *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 4, pp. 199-204, 2021. doi: 10.14710/jtsiskom.2021.13907, [Online].

***Abstract** – Hate speech and abusive language are easily found in written communications in social media like Twitter. They often cause a dispute between parties, the victims, and the first who write the tweet. However, it is also difficult to distinguish whether a tweet contains hate speech and/or abusive language for those who take sides. This research aims to develop a method to classify the tweets into abusive and/or contain hate speech classes. If hate speech is detected, then the system will measure the hardness level of hatred. The dataset includes 13,126 real tweets data. Word embeddings are used for featuring text input. For the tweets classification, we use a Decision Tree algorithm. Some engineering of features and parameters tuning has improved the classification of the three classes: hate speech class, abusive words, and hate speech level. The lexicon feature in the Decision Tree classification produces the highest accuracy for detecting the three classes rather than engineering special features and textual features. The average accuracy of the three classes increased from 69.77 % to 70.48 % for the training-testing composition of 90:10, and another 69.35 % to 69.54 % for 80:20 respectively.*

***Keywords** – hate speech; abusive language; decision tree; Twitter; word embeddings*

***Abstrak** - Ujaran kebencian dan bahasa kasar mudah ditemukan di dalam komunikasi tertulis di media sosial seperti Twitter, yang dapat memicu terjadinya persengketaan di antara korban dan pengujarnya. Bagaimanapun, akan sulit memeriksa apakah suatu twit mengandung ujaran kebencian dan/atau bahasa kasar bila seseorang berpihak. Penelitian ini bertujuan untuk membangun sistem untuk mengklasifikasi twit apakah mengandung ujaran kebencian dan kata-kata kasar. Apabila terdeteksi mengandung ujaran kebencian, maka level ujaran kebenciannya diukur. Dataset yang digunakan terdiri dari 13.126 twit asli dari Twitter. Word embedding digunakan untuk fitur dari teks. Algoritme Decision Tree digunakan untuk klasifikasi. Rekayasa fitur dan pengaturan parameter menunjukkan peningkatan performa deteksi. Fitur leksikon di klasifikasi Decision Tree menghasilkan akurasi tertinggi untuk deteksi ketiga kelas, yaitu kelas ujaran kebencian, kata-kata kasar dan level ujaran kebencian, daripada rekayasa fitur khusus dan fitur tekstual. Rata-rata akurasi dari ketiga kelas meningkat dari 69,77 % menjadi 70,48 % untuk komposisi data latih-uji 90:10, dan dari 69,35 % menjadi 69,54 % untuk komposisi 80:20.*

***Kata kunci** – ujaran kebencian; bahasa kasar; decision tree; Twitter; word embeddings*

I. PENDAHULUAN

Ujaran kebencian (*hate speech*) merupakan suatu ungkapan langsung maupun tidak langsung yang menuju kepada individu ataupun kelompok yang mengandung kebencian berdasarkan suatu hal yang melekat pada individu atau kelompok tersebut yang menyerang agama, etnis, gender, dan orientasi seksual. Ujaran kebencian adalah suatu perkataan, perilaku, tulisan ataupun suatu pertunjukan yang dilarang karena dapat memicu timbulnya tindakan kekerasan dan sikap prasangka, baik itu dari pihak pelaku yang memberikan pernyataan tersebut ataupun korban dari tindakan tersebut [1].

Dalam kehidupan sehari-hari, media sosial menjadi tempat/wadah bagi individu ataupun kelompok dalam melakukan penyebaran ujaran kebencian dan sering juga disertai dengan bahasa kasar (*abusive language*) [2]. Bahasa kasar dalam bahasa Indonesia biasanya diucapkan dan dituliskan untuk menyerang pihak tertentu, mengungkapkan kekesalan, kekecewaan ataupun meluapkan emosi terhadap peristiwa tertentu. Salah satu pengungkapan kata kasar dapat diungkapkan dengan menyebutkan jenis hewan tertentu, seperti anjing, monyet dan sebagainya. Namun, tidak semua kalimat yang memuat jenis hewan anggap ke dalam bahasa kasar [3].

*) Penulis korespondensi (Surya Agustian)
Email: surya.agustian@uin-suska.ac.id

Submitted: 7 September 2020; Revised: 4 June 2021; Accepted: 8 August 2021; Published: 31 October 2021

Bentuk penyalahgunaan media sosial di antaranya adalah penipuan, penyebaran ujaran kebencian, dan bahasa kasar. Ujaran kebencian dan bahasa kasar mudah ditemukan di dalam komunikasi tertulis di media sosial, seperti Twitter, Facebook dan Instagram. Siapa saja bisa menyebarkan ujaran kebencian atau bahasa kasar kepada orang lain yang tidak disukainya. Terlebih lagi, beberapa ujaran kebencian menjadi viral setelah di-twit ulang oleh kelompok, robot, dan provokator. Kemudian, pertikaian sering terjadi antara kedua pihak, korban, dan orang yang pertama menuliskannya.

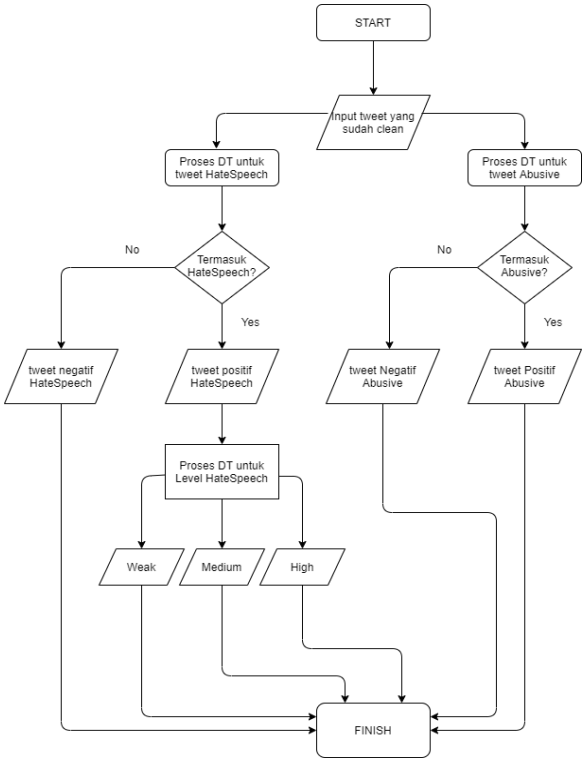
Kelebihan Twitter dibanding dengan media sosial lainnya di antaranya adalah jangkauannya luas [4]. Tidak hanya teman, Twitter juga mampu menjangkau publik figur, potensi periklanan di masa mendatang lebih besar, komunikasi terjadi sangat cepat (*update*), terhubung dengan banyak jaringan (*multi-link*) dan lebih terukur dari Facebook. Twitter membantu penyebaran informasi secara lebih cepat yang kemudian akan menjadi sebuah topik yang dibahas oleh para penggunanya. Media massa, seperti televisi, koran, majalah, dan tabloid, juga menggunakan Twitter sebagai penyampai berita-beritanya. Hal ini mempermudah masyarakat memperoleh informasi secara cepat dan terkini karena berita dapat diperbarui setiap saat oleh media massa melalui Twitter.

Beberapa kajian telah dilakukan untuk menyelesaikan kasus klasifikasi ujaran menggunakan beragam algoritme. Metode klasifikasi yang cukup populer dan banyak digunakan di antaranya adalah Decision Tree, Naive Bayes Classifier (NBC) dan k-Nearest Neighboor (k-NN) [5]. Romadloni dkk. [6] melakukan perbandingan metode NBC, k-NN dan Decision Tree untuk menyelesaikan analisis sentimen transportasi KRL *commuter line*. Luqyana dkk. [7] melakukan analisis sentimen *cyberbullying* pada komentar Instagram dengan metode klasifikasi Support Vector Machine (SVM) dan memperoleh akurasi terbaik sebesar 90 %. Hakiem dan Fauzi [8] melakukan klasifikasi ujaran kebencian pada Twitter menggunakan NBC berbasis N-Gram dengan seleksi fitur *information gain* dan memperoleh akurasi terbaik sebesar 84 %.

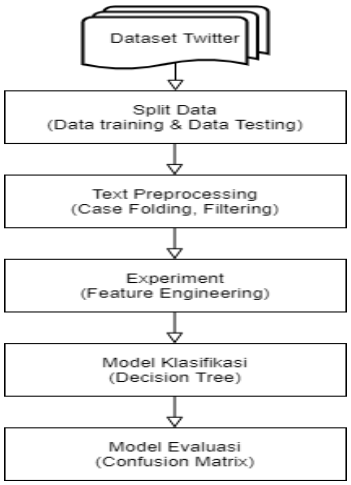
Kebanyakan algoritme yang digunakan pada penelitan tersebut di atas masih merupakan algoritme standar (*baseline*) yang belum dioptimalkan untuk meningkatkan akurasinya. Banyak cara yang bisa dilakukan untuk meningkatkan akurasi sebuah algoritme *machine learning* untuk klasifikasi, salah satunya adalah dengan menggunakan teknik rekayasa fitur (*feature engineering*). Penelitian ini mengusulkan suatu algoritme Decision Tree untuk klasifikasi ujaran kebencian dan bahasa kasar pada Twitter Bahasa Indonesia, dengan merekayasa fitur dan pengaturan parameter untuk meningkatkan hasil akurasi klasifikasi.

II. METODE PENELITIAN

Kelas yang diklasifikasi dalam masalah penelitian ini berjumlah tiga, yaitu kelas HateSpeech untuk ujaran kebencian, kelas Abusive untuk bahasa kasar dan Level HateSpeech untuk level ujaran kebencian. Alur klasifikasi pengembangan sistem deteksi multikelas pada penelitian ini ditunjukkan pada Gambar 1. Empat tahap untuk proses klasifikasi meliputi tahap persiapan data, pelatihan model bahasa, tahap rekayasa fitur, dan tahap klasifikasi seperti ditunjukkan pada Gambar 2.



Gambar 1. Alur klasifikasi keseluruhan



Gambar 2. Tahapan penelitian

A. Dataset

Data yang dipakai dalam penelitan ini diperoleh dari Ibrohim dan Budi [9]. Dataset berupa teks twit di Twitter dan dikumpulkan dengan menggunakan teknik perambanan (*crawling*). Proses pelabelan data dilakukan oleh 30 anotator dengan berbagai latar belakang usia, pendidikan terakhir, pekerjaan, etnis dan agama.

Dataset terdiri dari 13.169 dengan 12 label data, namun terdapat data *noise* sebanyak 43 data sehingga jumlah data yang digunakan pada penelitian ini adalah 13.126. Label yang digunakan terdiri dari tiga, yaitu HateSpeech, Abusive, dan Level HateSpeech. Distribusi kelas dan label pada dataset dapat dilihat pada Tabel 2.

Tabel 2. Distribusi dataset

No	Kelas	Label	Jumlah	Total
1	HateSpeech	HateSpeech	5.552 (42.3%)	13.126
		Netral	7.574 (57.7%)	
2	Abusive	Abusive	5.034 (38.35%)	13.126
		Netral	8.092 (61.65%)	
3	Level HateSpeech*)	Lemah	3.375 (60.79%)	5.552
		Sedang	1.704 (30.69%)	
		Kuat	473 (8.52%)	

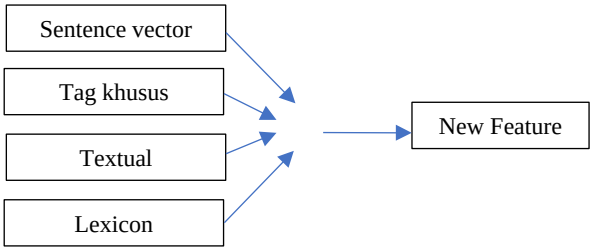
*) jika twit termasuk kelas HateSpeech

B. Metode

Sesuai dengan Gambar 2, prapengolahan teks dilakukan pada tahap awal terhadap data twit yang sudah dipecah dengan komposisi data latih dan data uji. Tahap prapengolahan teks yang dilakukan pada penelitian ini meliputi *case folding*, *tokenizing*, dan *filtering* sebagaimana pada [10] yang menggunakan data Facebook. *Case folding* mengubah teks menjadi huruf kecil semuanya, sedangkan *tokenizing* bertujuan untuk memecah teks menjadi token-token. Penelitian ini menggunakan kata sebagai token. Proses *filtering* dilakukan untuk menyaring dan memilih tanda-tanda baca (seperti tanda titik, tanda seru, tanda tanya, tanda *mention*, *hashtag*) dan *stopword* (kata-kata yang sering muncul dan tidak signifikan artinya) yang perlu dihapus atau dipertahankan sebagai fitur tambahan.

Tahap selanjutnya adalah pembentukan vektor kata. Penelitian ini memilih teknik *word embeddings* yang memiliki dimensi vektor lebih kecil dibandingkan *bag-of-word*, dan lebih sederhana dalam pembentukan fitur dibandingkan *language model*. *Word embeddings* menjadi populer sejak pertama diperkenalkan oleh Mikolov dkk. [11] karena dapat menemukan relasi antara kata-kata yang berhubungan makna atau yang memiliki arti yang mirip. Menurut Antariksa dkk. [12], *word2vec* yang digunakan sebagai fitur dapat menghasilkan akurasi yang tinggi untuk ketiga model *machine learning* yang digunakan. Teknik *word2vec* dapat berhasil baik pada Naïve Bayes untuk kasus kategorisasi Bahasa Indonesia pada [13]. Namun, word2vec tidak mampu menangani kata yang tidak ada di dalam corpus *(unseen words).* Teknik *word embeddings FastText* digunakan untuk menangani kata yang tidak pernah dijumpai sebelumnya, yang dikembangkan berdasarkan *word2vec* oleh Facebook [14].

Sebagai fitur yang akan menjadi masukan, vektor *word embeddings* di dalam twit merupakan elemen terpenting yang perlu dikembangkan. Karena *word embeddings* Bahasa Indonesia yang telah dilatih tidak tersedia, baik dengan *word2vec* maupun *FastText*, penelitian ini dilakukan pembentukan *word embeddings* dari dataset Twitter ini dengan menggunakan *FastText.* Pada tahap ini, kata akan diubah ke dalam *word vector* dengan dimensi 128. *Sentence embedding* dikonstruksi dengan menghitung vektor resultan dari *word embedding* kata-kata penyusun twit, dengan variasi *stemming*, *stopword*, dan *case folding*. Selanjutnya, setiap elemen dari *sentence vector* ini menjadi fitur masukan untuk sistem Decision Tree.



Gambar 3. Operasi penggabungan fitur

Tahap rekayasa fitur dilakukan untuk memilih, merekayasa, dan mengombinasikan fitur-fitur dasar yang berupa vektor kalimat (*sentence vector*) dengan lainnya. Rekayasa fitur yang dilakukan adalah dengan menambahkan fitur-fitur lain sebagaimana dapat dilihat pada Tabel 1. Gambar 3 menunjukkan bentuk fitur masukan untuk metode Decision Tree setelah melalui proses rekayasa fitur dengan proses penggabungan (*concatenation*) dengan komponen fitur-fitur tambahan. Dalam eksperimen, rekayasa fitur yang dihasilkan adalah kombinasi dari vektor kalimat sebagai fitur dasar dengan salah satu dari ketiga fitur lainnya (tag khusus, tekstual dan leksikon). Vektor masukan dari Decision Tree kemudian disusun menggunakan (1)-(4).

$$Baseline1 = [\, x_1,\ x_2,, \ldots, x_{128}\,] \quad (1)$$

Tabel 1. Rekayasa fitur yang akan dihitung frekuensinya

Kategori Fitur	Fitur	Keterangan
Khusus Twitter	F1	Jumlah tag dalam twit
	F2	Jumlah kata dalam twit
	F3	Jumlah tanda seru dalam twit
Tekstual	F4	Jumlah tanda tanya dalam twit
	F5	Jumlah kata huruf kapital dalam twit
	F6	Jumlah kata huruf kecil dalam twit
	F7	Kata berkonotasi positif
Leksikon	F8	Kata berkonotasi negatif
	F9	Kata yang mengandung kosakata bahasa kasar

Tabel 3. Percobaan tanpa rekayasa fitur pada komposisi data latih dan data uji 90:10

Variasi fitur pada prapengolahan			Akurasi (%)			
Case folding	**Stopword**	**Punctuation**	**HateSpeech**	**Abusive**	**Level**	**Rata-rata**
Ya	Ya	Ya	68,47	74,79	64,39	69,22
Ya	Ya	Tidak	71,06	74,18	61,87	69,04
Ya	Tidak	Ya	70,07	76,16	60,43	68,89
Ya	Tidak	Tidak	71,52	75,55	62,23	69,77
Tidak	Ya	Ya	68,55	72,35	59,35	66,75
Tidak	Ya	Tidak	67,10	73,95	62,59	67,88
Tidak	Tidak	Ya	68,93	73,72	60,43	67,69
Tidak	Tidak	Tidak	71,97	74,41	59,53	68,64

Tabel 4. Percobaan tanpa rekayasa fitur pada komposisi data latih dan data uji 80:20

Variasi fitur pada prapengolahan			Akurasi (%)			
Case folding	**Stopword**	**Punctuation**	**HateSpeech**	**Abusive**	**Level**	**Rata-Rata**
Ya	Ya	Ya	69,46	72,35	63,01	68,27
Ya	Ya	Tidak	70,94	73,27	63,01	69,07
Ya	Tidak	Ya	71,25	73,53	63,28	69,35
Ya	Tidak	Tidak	69,84	72,35	63,28	68,49
Tidak	Ya	Ya	67,10	72,24	64,09	67,81
Tidak	Ya	Tidak	69,61	71,74	59,23	66,86
Tidak	Tidak	Ya	68,39	73,23	63,10	68,24
Tidak	Tidak	Tidak	69,23	70,94	63,10	67,76

$$Khusus = [\, x_1,\ x_2,, \ldots, x_{128}\,]\ \ [F_1] \quad (2)$$

$$Tekstual = [\, x_1,\ x_2,, \ldots, x_{128}\,]\ \ [F_2, \ldots, F_6] \quad (3)$$

$$Leksikon = [\, x_1,\ x_2,, \ldots, x_{128}\,]\ \ [F_7, F_8, F_9] \quad (4)$$

Evaluasi model pada penelitan ini bertujuan untuk menghitung akurasi algoritme Decision Tree dalam melakukan klasifikasi. Matriks konfusi digunakan untuk pengukuran unjuk kerja sistem pada masing-masing kelas (Abusive, HateSpeech, dan Level HateSpeech), meliputi akurasi, presisi, dan sensitivitas (*recall*) [15]. Percobaan dilakukan dengan komposisi data latih dan data uji masing-masing 90:10 dan 80:20.

III. HASIL DAN PEMBAHASAN

Pengujian dilakukan dalam dua skema, yaitu skema seleksi fitur dan skema rekayasa fitur. Pada pengujian seleksi fitur, eksperimen dilakukan dengan berbagai perpaduan seleksi fitur, yaitu *case folding*, *punctuation*, dan *stopword*. Hasil eksperimen dengan komposisi data latih dan data uji 90:10 ditunjukkan pada Tabel 3, yaitu berupa data nilai akurasi untuk berbagai kombinasi fitur. Baris pertama adalah metode dasar (*baseline*) yang menerapkan semua kombinasi prapengolahan, yaitu mengubah teks ke huruf kecil (*case folding*), penghapusan *stopword*, dan penghilangan tanda baca (*punctuation*), dan menghasilkan akurasi rata-rata 69,22 %. Akurasi rata-rata tertinggi yang diperoleh adalah sebesar 69,77 % menggunakan fitur *case folding* saja (*stopword* dan *punctuation* tetap dipertahankan di dalam teks).

Tabel 4 menunjukkan hasil akurasi rata-rata pada komposisi data latih dan data uji 80:20 dengan *baseline* pada baris pertama dan kombinasi fitur pada baris selanjutnya. Akurasi tertinggi sebesar 69,35 % dengan fitur *case folding* dan penghilangan *punctuation*. Selain *word2vec* yang digunakan dalam [12], [13] yang berhasil dengan baik untuk kategorisasi berbahasa Indonesia dengan Naive Bayes, *FastText* dapat digunakan untuk pembentukan *word embeddings* untuk Decision Tree dalam kategorisasi twit berbahasa Indonesia.

Tabel 5. Performa rekayasa fitur pada komposisi data latih dan data uji 90:10

Parameter	Performa (%)		
	Akurasi	**Presisi**	**Sensitivitas**
Rekayasa fitur khusus			
HateSpeech	72,20	68,16	65,36
Abusive	74,18	65,58	62,45
Level	62,77	62,77	62,77
Rata-rata	69,72		
Rekayasa fitur tekstual			
HateSpeech	71,36	67,49	63,39
Abusive	74,03	65,70	61,20
Level	62,95	62,95	62,95
Rata-rata	69,45		
Rekayasa fitur leksikon			
HateSpeech	71,67	67,22	65,54
Abusive	77,91	75,13	59,54
Level	61,87	61,87	61,87
Rata-rata	70,48		

Dari hasil akurasi terbaik dari dari fitur yang telah diseleksi, tahap skema eksperimen kedua dilanjutkan, yaitu dengan menerapkan rekayasa fitur. Pengetahuan empiris digunakan untuk memilih objek yang bisa dijadikan fitur tambahan (Gambar 3). Hasil eksperimen

Tabel 6. Perbandingan akurasi tanpa dan dengan fitur tambahan pada komposisi data latih dan data uji 90:10

Parameter	Akurasi rata-rata (%)			
	Tanpa	Khusus	Tekstual	Leksikon
HateSpeech	71,52	72,20	71,36	71,67
Abusive	75,55	74,18	74,03	77,91
Level	62,25	62,77	62,95	61,87
Rata-rata	69,77	69,72	69,45	70,48

untuk beragam rekayasa fitur pada komposisi data latih dan data uji 90:10 disajikan pada Tabel 5. Hasil rekayasa fitur leksikon mempunyai nilai akurasi rata-rata paling tinggi sebesar 70,48 % dibandingkan dengan rekayasa fitur lainnya. Selain itu, akurasi rata-rata hasil rekayasa fitur leksikon ini lebih tinggi dibandingkan tanpa rekayasa fitur (Tabel 6). Akurasi sebelum dilakukan rekayasa fitur sebesar 69,77 %, sedangkan akurasi setelah dilakukan rekayasa fitur leksikon menjadi 70,48 %. Akurasi pada label HateSpeech dan Abusive mengalami kenaikan, sedangkan pada label Level mengalami penurunan, namun secara rata-rata akurasi bertambah sekitar 0,30 %.

Tabel 7 menunjukkan performa akurasi untuk komposisi data latih dan data uji 80:20 dengan penambahan vektor masukan Decision Tree (rekayasa fitur). Fitur leksikon juga masih konsisten sebagai akurasi tertinggi jika dibandingkan dengan rekayasa fitur lain dan tanpa rekayasa, yaitu 69,54 %, seperti dinyatakan pada Tabel 8.

Fitur leksikon memiliki presisi tertinggi dibandingkan rekayasa fitur lainnya untuk keseluruhan kelas (HateSpeech, Abusive, dan Level) pada perbandingan data latih dan data uji 80:20. Namun untuk perbandingan data 90:10, fitur leksikon hanya tertinggi pada deteksi kelas Abusive saja. Hal ini karena fitur leksikon menyimpan kata-kata yang spesifik dari bahasa kasar, yang umumnya terdapat dalam twit berbahasa kasar maupun ujaran kebencian. Namun, tidak sedikit juga ujaran kebencian yang tidak berisi kata-kata kasar sehingga akurasi deteksi dapat saja lebih rendah dari fitur lainnya. Dari segi sensitivitas, hasilnya juga mirip dengan presisi, di mana untuk komposisi data 80:20, fitur leksikon memiliki sensitivitas tertinggi, sedangkan untuk 90:10, hanya kelas HateSpeech saja yang tertinggi dibandingkan fitur rekayasa lainnya.

Penerapan rekayasa fitur, terutama fitur leksikon, yang dilakukan telah berhasil untuk menaikkan akurasi dalam deteksi multi kelas seperti dalam [9]. Model Decision Tree dapat diterapkan secara terpisah dengan fitur yang berbeda-beda untuk mendapatkan hasil akurasi tertinggi dari masing-masing kelas. Misalnya, pada saat pemasangan sistem, dengan mengambil komposisi data latih terbanyak (90%) dan saat fokus pada deteksi ujaran kebencian, model dengan komposisi fitur yang perlu digunakan adalah penerapan *case folding*, dan fitur jumlah tag dalam twit (fitur khusus). Di sisi lain, untuk fokus deteksi bahasa kasar, model yang perlu digunakan adalah dengan komposisi fitur penerapan *case folding* dan fitur leksikon. Untuk mendeteksi level, fitur *case folding* dan fitur tekstual dapat digunakan. Hal tersebut bertujuan agar proses klasifikasi dapat memperoleh performa yang terbaik dari setiap kelas yang dideteksi.

Tabel 7. Performa rekayasa fitur pada komposisi data latih dan data uji 80:20

Parameter	Performa (%)		
	Akurasi	Presisi	Sensitivitas
Rekayasa fitur khusus			
HateSpeech	71,17	66,99	62,58
Abusive	73,57	65,37	64,57
Level	60,94	56,49	61,81
Rata-rata	68,56		
Rekayasa fitur tekstual			
HateSpeech	70,34	72,66	68,81
Abusive	72,32	65,41	64,45
Level	61,21	56,59	62,08
Rata-rata	67,96		
Rekayasa fitur leksikon			
HateSpeech	71,10	77,65	70,31
Abusive	77,38	72,58	66,48
Level	60,13	64,35	65,24
Rata-rata	69,54		

Tabel 8. Perbandingan akurasi tanpa dan dengan fitur tambahan pada komposisi data latih dan data uji 90:10

Parameter	Akurasi rata-rata (%)			
	Tanpa	Khusus	Tekstual	Leksikon
HateSpeech	71,25	71,17	70,34	71,10
Abusive	73,53	73,57	72,32	77,38
Level	63,28	60,94	61,21	60,13
Rata-rata	69,35	68,56	67,96	69,54

IV. KESIMPULAN

Penerapan seleksi dan rekayasa fitur dapat diterapkan untuk meningkatkan hasil akurasi klasifikasi algoritme Decision Tree dalam melakukan klasifikasi teks Twitter untuk deteksi multi kelas yang berbeda. Model yang terbentuk pada penelitian ini dapat digunakan dalam kasus klasifikasi Twitter untuk mendeteksi twit yang mengandung ujaran kebencian (kelas HateSpeech), bahasa kasar (kelas Abusive) dan mengukur level dari ujaran kebencian (kelas Level).

DAFTAR PUSTAKA

[1] M. Febriyani, "Analisis faktor penyebab pelaku melakukan ujaran kebencian (hate speech) dalam media sosial," *Poenale: Jurnal Bagian Hukum Pidana*, vol. 3, no. 2, pp. 139–157, 2018.

[2] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," arXiv:1703.04009v1 [cs.CL], 2017.

[3] A. F. Hidayatullah, A. A. Fadila, K. P. Juwairi, and R. A. Nayoan, “Identifikasi konten kasar pada tweet bahasa Indonesia,” *Jurnal Linguistik Komputasional*, vol. 2, no. 1, pp. 1-5, 2019. doi: 10.26418/jlk.v2i1.15

[4] E. D. Putra, *Menguak jejaring sosial*. Tangerang, 2014.

[5] F. Gorunescu, *Data mining: Concepts, models and techniques*. Berlin: Springer, 2011.

[6] N. T. Romadloni, I. Santoso, and S. Budilaksono, “Perbandingan metode naive bayes, knn dan decision tree terhadap analisis sentimen transportasi commuter line,” *Jurnal Komputer dan Informatika*, vol. 3, no. 2, pp. 1–9, 2019.

[7] W. A. Luqyana, I. Cholissodin, and R. S. Perdana, “Analisis sentimen cyberbullying pada komentar instagram dengan metode klasifikasi support vector machine,” *Jurnal Pengembangan Teknlogi Informasi dan Ilmu Komputer*, vol. 2, no. 11, pp. 4704–4713, 2018.

[8] M. Hakiem and M. A. Fauzi, “Klasifikasi ujaran kebencian pada twitter menggunakan metode naïve bayes berbasis n-gram dengan seleksi fitur information gain,” *Jurnal Pengembangan Teknlogi Informasi dan Ilmu Komputer*, vol. 3, no. 3, pp. 2443–2451, 2019.

[9] M. O. Ibrohim and I. Budi, “Multi-label hate speech and abusive language detection in Indonesian Twitter,” in the *Third Workshop on Abusive Language Online*, Florence, Italy, Aug. 2019, pp. 46–57. doi: 10.18653/v1/W19-3506

[10] A. K. B. A. Putra, M. A. Fauzi, B. D. Setiawan, and E. Setiawati, “Identifikasi ujaran kebencian pada Facebook dengan metode ensemble feature dan support vector machine,” *Jurnal Pengembangan Teknlogi Informasi dan Ilmu Komputer*, vol. 2, no. 12, 2018.

[11] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *International Conference on Learning Representations*, Arizona, USA, May 2013, pp. 1-12.

[12] K. Antariksa, Y. S. Purnomo, and D. Ernawati, “Klasifikasi ujaran kebencian pada cuitan dalam bahasa Indonesia,” *Jurnal Buana Informatika*, vol. 10, no. 2, pp. 164–171, 2019. doi: 10.24002/jbi.v10i2.2451

[13] S. Santoso, A. Dewa, B. Soetiono, E. Setyati, and E. M. Yuniarno, “Self-training naive bayes berbasis word2vec untuk kategorisasi berita bahasa Indonesia,” *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 7, no. 2, pp. 158–166, 2018. doi: 10.22146/jnteti.v7i2.418

[14] Z. A. Arliyanti Nurdin, Bernadus Anggo Seno Aji, Anugrayani Bustamin, “Perbandingan kinerja word embedding word2vec, Glove dan FastText pada klasifikasi teks,” *Jurnal Teknokompak*, vol. 14, no. 2, pp. 74--79, 2020. doi: 10.33365/jtk.v14i2.732

[15] D. M. W. Powers, "Evaluation: from precision, recall and f-measure to ROC, informedness, markedness & correlation," *Journal of Machine Learning Technologies.* vol. 2, no. 1, pp. 37–63, 2011.