Jurnal SimanteC
Vol. 11, No. 1 Desember 2022
P-ISSN : 2088-2130
E-ISSN : 2502-4884

DETEKSI *CYBERBULLYING* PADA DATA *TWEET* MENGGUNAKAN METODE *RANDOM FOREST* DAN SELEKSI FITUR *INFORMATION GAIN*

CYBERBULLYING DETECTION ON TWEET DATA USING RANDOM FOREST METHOD AND INFORMATION GAIN FEATURE SELECTION

Rachmad Masbadi Hatullah Nurnaryo[1)], Mulaab[2)], Ika Oktavia Suzanti[3)], Doni Abdul Fatah[4)], Andharini Dwi Cahyani[5)], Fifin Ayu Mufarroha[6)]

[1,2,3,5,6] Program Studi Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo

[4]Prodi Sistem Informasi, Fakultas Teknik, Universitas Trunojoyo

Jl. Raya Telang, PO BOX 2 Kamal, Bangkalan

E-mail : [1*]160411100049@student.trunojoyo.ac.id, [2] mulaab@trunojoyo.ac.id, [3]iosuzanti@trunojoyo.ac.id, [4]doni.fatah@trunojoyo.ac.id, [5]andharini.cahyani@trunojoyo.ac.id, [6]fifin.mufarroha@trunojoyo.ac.id

ABSTRAK

Indonesia merupakan salah satu negara dengan pengguna media sosial terbanyak. Dengan banyaknya pengguna media sosial, hal ini dapat memicu munculnya *cyberbullying*. *Cyberbullying* adalah tindakan berulang yang melecehkan, mempermalukan, mengancam, atau mengganggu orang lain melalui komputer, ponsel, dan perangkat elektronik lainnya, termasuk situs web jejaring sosial *online*. Twitter merupakan salah satu media sosial yang sering digunakan untuk melakukan *cyberbullying*. Deteksi *cyberbullying* merupakan langkah penting untuk membuat lingkungan yang baik dalam interaksi media sosial. Penelitian ini mendeteksi *cyberbullying* yang berasal dari *tweet* berbahasa Indonesia dengan menggunakan metode *Random Forest* sebagai pengklasifikasi. Seleksi fitur *information gain* juga digunakan untuk menyeleksi fitur yang berupa atribut. Penelitian ini bertujuan untuk mengetahui akurasi deteksi *cyberbullying* dari metode *Random Forest* dan memilih fitur penting untuk meningkatkan kinerja metode. Dari hasil pengujian, didapatkan nilai *Accuracy* tertinggi sebesar 72.1% dengan atribut berjumlah 1295 dari 2277 atribut. Hal ini berarti, pemilihan fitur yang baik dapat meningkatkan performa dari metode *machine learning*.

Kata kunci: *Cyberbullying, Information Gain, Random Forest, Tweet*

ABSTRACT

Indonesia is one of the countries with the most social media users. With the large number of social media users, this can trigger the emergence of cyberbullying. Cyberbullying is the repeated act of harassing, humiliating, threatening, or harassing others through computers, cell phones and other electronic devices, including online social networking websites. Twitter is one of the social media that is often used for cyberbullying. Detection of cyberbullying is an important step to create a good environment for social media interactions. This study detects cyberbullying originating from Indonesian-language tweets using the Random Forest method as a classifier. Information gain feature selection is also used to select features in the form of attributes. This study aims to determine the accuracy of cyberbullying detection from the Random Forest method and to select important features to improve the performance of the method. From the test results, obtained the highest Accuracy value of 72.1% with 1295 attributes of 2277 attributes. This means that good feature selection can improve the performance of machine learning method.

Keywords: *Cyberbullying, Information Gain, Random Forest, Tweet*

PENDAHULUAN

Situs jejaring sosial *online* telah menjadi sangat populer dalam beberapa tahun terakhir. Media sosial menjadi alat komunikasi dan promosi yang dinilai cukup efektif saat ini, mengingat tingginya populasi penggunanya. Saat ini, alih – alih maju dan menjadi sarana pertukaran informasi kesehatan, internet, khususnya media sosial, menjadi tempat saling menghina dan menghancurkan kehidupan satu sama lain [1].

Indonesia menjadi salah satu negara dengan rasio penggunaan media sosial yang tinggi. Berdasarkan data statista.com pengguna media sosial dari Indonesia sebanyak 193.43 juta. Salah satu dampak negatif dari fenomena ini adalah munculnya *cyberbullying* [2]. *Cyberbullying* adalah tindakan berulang yang melecehkan, mempermalukan, mengancam, atau mengganggu orang lain melalui komputer, ponsel, dan perangkat elektronik lainnya, termasuk situs web jejaring sosial *online* [3]. Twitter merupakan salah satu media yang sering digunakan untuk melakukan *cyberbullying*, karena orang dapat memposting tulisan kejam atau mengunggah foto yang berhubungan dengan individu lain dengan tujuan untuk mengintimidasi dan merusak reputasi korban sehingga korban merasa sakit hati dan malu, sedangkan pelakunya merasa puas dan senang karena cita – citanya telah tercapai [1]. Dengan semakin banyaknya pengguna sosial media saat ini menjadikan peluang terciptanya masalah *cyberbullying* semakin meningkat, hal ini membuat para peneliti tidak dapat mengabaikan aktivitas ini. Beberapa peneliti telah membuat penelitian untuk mengidentifikasi *cyberbullying*.

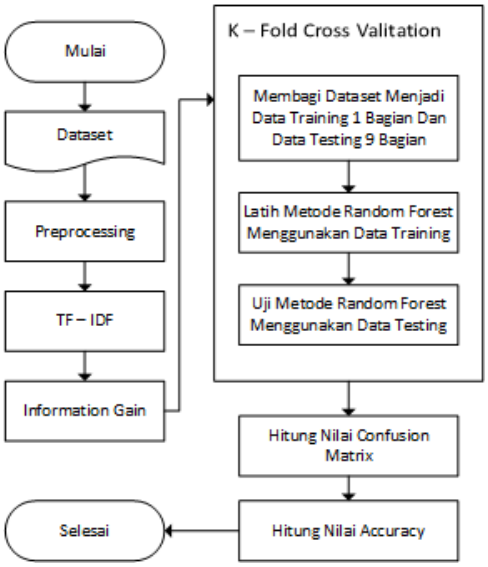
Penelitian untuk mengidentifikasi *cyberbullying* pernah dilakukan oleh Sharma dkk [4]. Mereka menganalisis dan bereksperimen dengan metode berbeda untuk menemukan cara yang layak dalam mengklasifikasikan teks *cyberbullying* dari *Kaggle, MySpace, Formspring,* Twitter dan YouTube. Penilitian ini menggunakan metode klasifikasi *Naive Bayes, Decision Trees, Random Forest* dan *Artificial Neural Network*. Al – Garadi dkk [5] mengusulkan deteksi teks *cyberbullying* menggunakan data dari twitter, awalnya mereka menggunakan berbagai fitur untuk mengembangkan *supervised machine learning*. Kemudian, tiga algoritma pemilihan fitur dipilih untuk menentukan fitur paling signifikan. Metode *Naive Bayes*, SVM (*Support Vector Machine*), *Random Forest* dan KNN (*K – Nearest Neighbor*) dipilih lalu diukur kinerjanya menggunakan empat tahapan pengaturan yang berbeda.

Saat proses pembentukan model dalam *machine learning* pada umumnya digunakan banyak fitur. Dengan banyak fitur tersebut terdapat beberapa fitur yang tidak relavan, hal ini mempengaruhi kinerja dari algoritma *machine learning*. Untuk memilih fitur yang relavan digunakan metode seleksi fitur. *Information gain* merupakan salah satu metode seleksi fitur yang digunakan untuk menyeleksi fitur.

Penelitian ini menggunakan metode *Random Forest* sebagai pengklasifikasi. Seleksi fitur *information gain* juga digunakan untuk menyeleksi fitur yang berupa atribut. Penelitian ini memiliki tujuan untuk mengetahui akurasi deteksi *cyberbullying* dari metode *Random Forest* dan memilih fitur penting untuk meningkatkan kinerja metode.

METODE

Diagram alur sistem deteksi *cyberbullying* yang dibuat bisa dilihat pada Gambar 1.



Gambar 1. Sistem deteksi *cyberbullying*

Dataset berasal dari twitter berjumlah 1000 tweet dimana didalamnya terdapat 40 kata yang biasanya digunakan untuk mem*bully* dalam bahasa Indonesia. Tiap kata dicari sebanyak 25 tweet lalu dikumpulkan menjadi satu *dataset*. Daftar kata diperoleh dari https://www.femina.co.id/trending-topic/infografis-ini-jenis-hinaan-dan-kata-kata-yang-sering-digunakan-oleh-para-pelaku-bullying-di-media-sosial dan https://sites.google.com/site/catatancatatansaya/kata-kata-kasar-di-indonesia.

Pelabelan dilakukan secara manual oleh 3 anotator menjadi *cyberbullying* (CB) atau *non – cyberbullying* (NCB). Pada tabel 1 merupakan rincian dari dataset dan tabel 2 adalah daftar kata yang digunakan untuk melakukan bully.

Tabel 1. Rincian *Dataset*

Label	Jumlah Data
CB	562
NCB	438
Total	1000

Tabel 2. Daftar Kata Yang Digunakan Untuk Mem – *Bully*

No	Kata Untuk Mem – ***Bully***	Jumlah tweet
1	Anjing	25
2	Asu	25
3	Babi	25
4	Bajingan	25
5	Banci	25
6	Bangsat	25
7	Bego	25
8	Bejad	25
9	Bencong	25
10	Bolot	25
11	Brengsek	25
12	Budek	25
13	Buta	25
14	Geblek	25
15	Gembel	25
16	Gila	25
17	Goblok	25
18	Iblis	25
19	Idiot	25
20	Jablay	25
21	Jelek	25
22	Kampungan	25
23	Keparat	25
24	Kontol	25
25	Kunyuk	25
26	Memek	25
27	Monyet	25
28	Ngehe	25
29	Ngentot	25
30	Ngewe	25
31	Orang gila	25
32	Pecun	25
33	Perek	25
34	Sarap	25
35	Setan	25
36	Sinting	25
37	Sompret	25
38	Tai	25
39	Tolol	25
40	Udik	25

Untuk memasukkan data teks ke algoritma *machine learning*, dalam bentuk yang lebih baik daripada bentuk aslinya, digunakan teknik *preprocessing* [6]. *Preprocessing* digunakan untuk menghilangkan *noise* seperti: karakter – karakter tidak penting, kata yang diulang – ulang, kata yang tidak baku, huruf *single*, url, *mention*, *hastag*, kata dengan huruf kapital dan lain sebagainya. Tahapan *preprocessing* yang dilakukan dalam penelitian ini yaitu:

1. *Case Folding*, mengubah huruf kapital menjadi huruf kecil.
2. *Cleaning*, menghapus karakter yang tidak diperlukan.
3. Tokenisasi, memecah kalimat menjadi potongan kata.

4. Normalisasi, mengubah kata menjadi baku.
5. *Stemming*, mengubah kata menjadi kata akarnya.
6. *Stopword Removal*, menghilangkan kata yang sering muncul.

Setelah tahap *preprocessing*, bobot tiap kata dihitung menggunakan metode TF – IDF. Persamaan untuk menghitung nilai TF – IDF adalah sebagai berikut:

$$TF(t) = \frac{n_t}{n_d} \quad (1)$$

$$IDF(t) = log^{m_d}_{m_t} \quad (2)$$

$$TF\text{-}IDF(t) = TF(t).\ IDF(t) \quad (3)$$

Dimana:

n_t = Jumlah berapa kali *t* muncul dalam dokumen.

n_d = Jumlah total kata yang muncul dalam dokumen.

m_d = Jumlah total dokumen.

m_t = Jumlah dokumen yang mengandung kata *t*.

TF – IDF akan mengubah kata pada tiap dokumen menjadi angka yang kemudian disusun menjadi sebuah matriks. Angka yang disusun dalam sebuah matriks berasal dari perhitungan nilai kemunculan kata (*Term Frequency*) pada tiap *document* (*Inverse Document Frequency*). Pada tabel 3 merupakan nilai-nilai TF-IDF dalam bentuk matriks.

Tabel 3. Nilai TF – IDF Tiap Kata Dalam Bentuk Matriks

Atribut Ke	1	2		2277
Doc	abang	abiz	.	zodiak
0	0.0000	0.0000	.	0.0000
1	0.0000	0.0000	.	0.0000
2	0.0000	0.0000	.	0.0000
…	…	…	.	…
999	0.0000	0.0000	.	0.0000

Setelah TF – IDF tahapan selanjutnya yaitu seleksi fitur menggunakan *information gain*. Seleksi fitur sendiri merupakan proses yang melibatkan penghapusan fitur yang tidak relevan dan berulang dari *dataset* untuk meningkatkan kinerja teknik *machine learning* dan aplikasinya [7]. *Information gain* juga merupakan metode seleksi fitur yang cara kerjanya melakukan perangkingan berdasarkan atribut [8]. *Information gain* antara fitur X yang diberikan dan fitur kategori terkait Y adalah sebagai berikut [9] (4).

$$IG(Y,X) = H(Y) - H(Y|X) \quad (4)$$

$IG(Y,X)$ = *Information gain* dari fitur *X* untuk kategori *Y*

$H(Y)$ = *Entropy* dari *Y*

$H(Y|X)$ = *Entropy* dari *Y* terhadap *X*

Untuk mencari nilai $H(Y)$ digunakan persamaan (5).

$$H(Y) = -\sum_{i=1}^{n} p(y_i) \log_2 p(y_i) \quad (5)$$

n = Jumlah kriteria pada kategori.

$p(y_i)$ = Rasio jumlah sampel di kelas y_i terhadap total sampel pada himpunan data

Untuk mencari nilai $H(Y|X)$ atau *entropy* bersyarat dari dua peristiwa X dan Y, ketika X memiliki nilai i digunakan persamaan (6).

$$H(Y|X) = \sum_{i=1}^{m} p(x_i) H(Y|X=x_i) \quad (6)$$

Pada tahapan *information gain*, nilai entropi tiap atribut dihitung lalu atribut diurutkan dari yang paling tinggi hingga ke rendah berdasarkan nilai entropi. Pada tabel 4 merupakan tabel nilai entropi tiap atribut.

Tabel 4. Nilai Entropi Tiap Atribut

No	Atribut	Nilai Entropi
1	kamu	0.05865
2	patrol	0.0515
3	kenyat	0.05136
4	chika	0.04976
5	tolong	0.04947
6	info	0.04905
7	ovo	0.04818
8	lau	0.04749
9	otak	0.04707
…	…	…
2277	abang	0

Metode *machine learning* yang digunakan dalam penelitian ini yaitu *Random Forest*. *Random Forest* adalah

metode berdasarkan *Decision Tree*, yang menggunakan aturan untuk membagi data dalam mode *biner* [10]. Untuk cara kerja metode *Random Forest* dalam penelitian ini adalah sebagai berikut:

1. *Dataset* yang telah dihitung nilai TF – IDF lalu atributnya telah diiurutkan berdasarkan nilai entropi digunakan untuk membuat *Bootsrap Sample*.
2. Pada tahap pembuatan *Bootsrap Sample*, data diambil secara acak dari *Dataset* dimana nantinya terdapat dokumen yang sama. Kemudian, *Bootsrap Sample* dibagi menjadi *Data Train* dan *Data Test*. Masing – masing *Data Train* dan *Data Test* digunakan untuk melatih dan menguji metode *decision tree*. *Bootsrap Sample* yang dibuat digunakan sebagai acuan untuk membuat *decision tree*.
3. *Decision tree* yang telah dibuat dari *Data Test* nantinya menghasilkan sebuah prediksi atau dalam hal ini disebut *Class*.
4. Langkah pada nomor 1 hingga 3 dilakukan hingga memenuhi kriteria atau dilakukan sebanyak K kali.
5. Tiap *Class* yang dihasilkan oleh *Bootsrap Sample* dikumpulkan dan digunakan *Majority Voting* untuk menentukan *Final Class*. *Final Class* digunakan untuk menentukan *Class* pada *Data Test*.

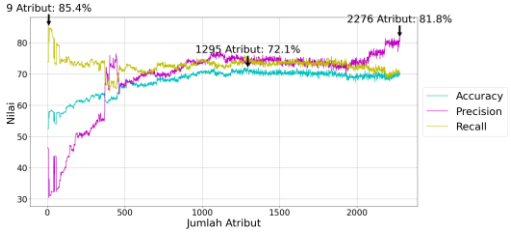
Dataset yang telah dihitung nilai TF – IDF dan atributnya telah diurutkan dari yang tertinggi hingga ke kerendah berdasarkan nilai entropi digunakan sebagai input untuk proses *K – Fold Cross Validation*. *K – Fold Cross Validation* menggunakan K sebanyak 10. Dimana nantinya terdapat 10 (*fold*) skenario pengujian dan *dataset* sebanyak 1000 dibagi menjadi 10 bagian. Pada *fold* ke – 1 *dataset* dibagi menjadi data *Training* sebanyak 9 bagian dan data *Testing* sebanyak 1 bagian. Pada data *Training* metode *Random Forest* digunakan untuk melatih data. Setelah melatih data, data *Testing* digunakan untuk menguji performa metode *Random Forest*. Kemudian nilai *Accuracy*, *Precicsion* dan *Recall* dihitung. Hal yang sama dilakukan untuk *fold* ke 2 hingga ke 10, namun untuk pembagian data *Training* dan data *Testing* pada tiap *fold* tidak sama, hal tersebut bertujuan untuk menemukan kombinasi terbaik. Lalu nilai *Accuracy*, *Precicsion* dan *Recall* tiap *fold* dirata – rata. Setelah perhitungan *K – Fold Cross Validation* pertama, atribut dikurangi satu lalu dilakukan lagi perhitungan *K – Fold Cross Validation* kedua. Hal tersebut dilakukan hingga jumlah atribut menjadi satu. Pengurangan dilakukan mulai dari nilai entropi atribut terendah hingga ke tinggi. Hasil akhir untuk skenario pengujian pertama ditentukan dengan memilih rata – rata *Accuracy*, *Precicsion* dan *Recall* tertinggi dari perhitungan pengurangan atribut. Pada tabel 5 adalah dataset yang telah diurutkan berdasarkan nilai entropi.

Tabel 5. Dataset Yang Telah Diurutkan Berdasarkan Nilai Entropi

Atribut Ke	969	1600		1
Doc	kamu	patrol	.	abang
0	0.0000	0.0000	.	0.0000
1	0.0000	0.0000	.	0.0000
2	0.0000	0.0000	.	0.0000
…	…	…	.	…
999	0.0000	0.0000	.	0.0000

HASIL DAN PEMBAHASAN

Hasil percobaan bisa dilihat pada gambar 2. Untuk hasil *confusion matrix* tiap *fold* pada *Accuracy*, *Precision* dan *Recall* tertinggi bisa dilihat pada tabel 6, 7 dan 8.



Gambar 2. Grafik Performansi Nilai Rata – Rata Accuracy, Precision dan Recall terhadap Jumlah Atribut

Tabel 6. Nilai *Confusion Matrix* Tiap *Fold* Pada Rata – Rata *Accuracy* Tertinggi

Jumlah Atribut	*Fold*	TP	FP	TN	FN	*Acuracy* (%)
1295	1	**53**	**8**	**27**	**12**	80

1295	2	**39**	**15**	**34**	**12**	73
1295	3	**37**	**13**	**36**	**14**	73
1295	4	**36**	**17**	**37**	**10**	73
1295	5	**44**	**24**	**20**	**12**	64
1295	6	**41**	**9**	**31**	**19**	72
1295	7	**42**	**18**	**27**	**13**	69
1295	8	**45**	**17**	**25**	**13**	70
1295	9	**40**	**10**	**33**	**17**	73
1295	10	**42**	**12**	**32**	**14**	74

Pada gambar 2 grafik performansi untuk bagian *Accuracy*, semakin banyak atribut yang digunakan maka semakin tinggi rata – rata nilai *Accuracy* – nya. Hal tersebut dikarenakan nilai TP dan TN pada tiap *fold* sebanyak 90% memilki nilai lebih tinggi dari FP dan FN. Namun saat rata – rata *Accuracy* mencapai nilai yang paling tinggi, rata – rata nilai *Accuracy* yang dihasilkan selanjutnya tidak mengalami kenaikan atau penurunan secara signifikan.

Tabel 7. Nilai *Confusion Matrix* Tiap *Fold* Pada Rata – Rata *Precision* Tertinggi

Jumlah Atribut	*Fold*	TP	FP	TN	FN	*Precision* (%)
2276	1	**49**	**12**	24	15	73
2276	2	**44**	**10**	30	16	74
2276	3	**43**	**7**	27	23	70
2276	4	**46**	**7**	31	16	77
2276	5	**54**	**14**	19	13	73
2276	6	**44**	**6**	27	23	71
2276	7	**39**	**21**	20	20	59
2276	8	**50**	**12**	25	13	75
2276	9	**43**	**7**	21	29	64
2276	10	**46**	**8**	23	23	69

Pada gambar 2 grafik performansi untuk bagian *Precision*, semakin banyak atribut yang digunakan maka semakin tinggi nilai *Precision* – nya. Hal tersebut dikarenakan nilai TP pada tiap *fold* lebih tinggi dari FP.

Tabel 8. Nilai *Confusion Matrix* Tiap *Fold* Pada Rata – Rata *Recall* Tertinggi

Jumlah Atribut	*Fold*	TP	FP	TN	FN	*Recall* (%)
9	1	**22**	39	**36**	3	58
9	2	**7**	47	**44**	2	51
9	3	**15**	35	**42**	8	57
9	4	**13**	40	**46**	1	59
9	5	**21**	47	**31**	1	52
9	6	**18**	32	**48**	2	66
9	7	**10**	50	**37**	3	47
9	8	**20**	42	**37**	1	57
9	9	**19**	31	**48**	2	67
9	10	**25**	29	**41**	5	66

Pada gambar 2 grafik performansi untuk bagian *Recall*, semakin banyak atribut yang digunakan maka semakin rendah nilai *Recall* – nya. Hal tersebut dikarenakan nilai TN pada tiap *fold* lebih tinggi dari TP.

SIMPULAN

1. Metode Random Forest menghasilkan *Accuracy* sebesar 72.1% dengan atribut berjumlah 1295.
2. Penerapan seleksi fitur yang tepat dapat meningkatkan kinerja algoritma *machine learning*. Hal tersebut bisa dilihat pada skenario pengujian yang menghasilkan nilai *Accuracy* sebesar 72.1% dengan atribut berjumlah 1295 dari total 2277.

SARAN

1. Dikarenakan penlitian ini hanya menggunakan kata kasar berdasarkan dari internet, disarankan menambahkan kata kasar yang berbeda untuk *dataset*.
2. Menggunakan metode *machine learning* yang berbeda dari penelitian ini, karena *Accuracy* yang dihasilkan hanya 72.1%.
3. Mencoba Menggunakan metode seleksi fitur yang berbeda dari penelitian ini.

DAFTAR PUSTAKA

[1] T. Febriana and A. Budiarto, "Twitter Dataset for Hate Speech and Cyberbullying Detection in Indonesian Language," *Proc. 2019 Int. Conf. Inf. Manag. Technol. ICIMTech 2019*, vol. 1, no. August, pp. 379–382, 2019, doi: 10.1109/ICIMTech.2019.8843722.

[2] L. Anindyati, A. Purwarianti, and A. Nursanti, "Optimizing Deep Learning for Detection Cyberbullying Text in Indonesian Language," *Proc. - 2019 Int. Conf. Adv. Informatics Concepts, Theory, Appl. ICAICTA 2019*, pp. 1–5, 2019, doi: 10.1109/ICAICTA.2019.8904108.

[3] H. Nurrahmi and D. Nurjanah, "Indonesian Twitter Cyberbullying Detection using Text Classification and User Credibility," *2018 Int. Conf. Inf. Commun. Technol. ICOIACT 2018*, vol. 2018-Janua, pp. 543–548, 2018, doi: 10.1109/ICOIACT.2018.8350758.

[4] H. K. Sharma, K. Kshitiz, and Shailendra, "NLP and Machine Learning Techniques for Detecting Insulting Comments on Social Networking Platforms," *2018 Int. Conf. Adv. Comput. Commun. Eng.*, no. June, pp. 265–272, 2018.

[5] M. A. Al-garadi, K. D. Varathan, and S. D. Ravana, "Computers in Human Behavior Cybercrime detection in online communications : The experimental case of cyberbullying detection in the Twitter network," *Comput. Human Behav.*, vol. 63, pp. 433–443, 2016, doi: 10.1016/j.chb.2016.05.051.

[6] D. Ramachandran and R. Parvathi, "ScienceDirect Analysis Analysis of of Twitter Twitter Specific Specific Preprocessing Preprocessing Technique Technique for for Tweets Tweets," *Procedia Comput. Sci.*, vol. 165, pp. 245–251, 2020, doi: 10.1016/j.procs.2020.01.083.

[7] E. Odhiambo Omuya, G. Onyango Okeyo, and M. Waema Kimwele, "Feature Selection for Classification using Principal Component Analysis and Information Gain," *Expert Syst. Appl.*, vol. 174, no. February, p. 114765, 2021, doi: 10.1016/j.eswa.2021.114765.

[8] S. Chormunge and S. Jena, "Efficient feature subset selection algorithm for high dimensional data," *Int. J. Electr. Comput. Eng.*, vol. 6, no. 4, pp. 1880–1888, 2016, doi: 10.11591/ijece.v6i4.9800.

[9] Y. Zhang, X. Ren, and J. Zhang, "Intrusion detection method based on information gain and ReliefF feature selection," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2019-July, no. July, pp. 1–5, 2019, doi: 10.1109/IJCNN.2019.8851756.

[10] X. Ji, B. Yang, and Q. Tang, "Seabed sediment classification using multibeam backscatter data based on the selecting optimal random forest model," *Appl. Acoust.*, vol. 167, p. 107387, 2020, doi: 10.1016/j.apacoust.2020.107387.

[11] R. R. Dalvi, S. Baliram Chavan, and A. Halbe, "Detecting A Twitter Cyberbullying Using Machine Learning," *Proc. Int. Conf. Intell. Comput. Control Syst. ICICCS 2020*, no. Iciccs, pp. 297–301, 2020, doi: 10.1109/ICICCS48265.2020.9120893.

[12] M. Fortunatus, P. Anthony, and S. Charters, "Combining textual features to detect cyberbullying in social media Combining textual features to detect cyberbullying in social media posts posts," *Procedia Comput. Sci.*, vol. 176, pp. 612–621, 2020, doi: 10.1016/j.procs.2020.08.063.

[13] M. Z. Islam, J. Liu, J. Li, L. Liu, and W. Kang, "A Semantics Aware Random Forest for Text Classification," *CIKM*, vol. 19, pp. 1061–1070, 2019, doi: 10.1145/3357384.3357891.

[14] P. Jiang and J. Chen, "Neurocomputing Displacement

prediction of landslide based on generalized regression neural networks with K -fold cross-validation," *Neurocomputing*, pp. 1–8, 2016, doi: 10.1016/j.neucom.2015.08.118.

[15] D. Kim, D. Seo, S. Cho, and P. Kang, "Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec," *Inf. Sci. (Ny).*, 2018, doi: 10.1016/j.ins.2018.10.006