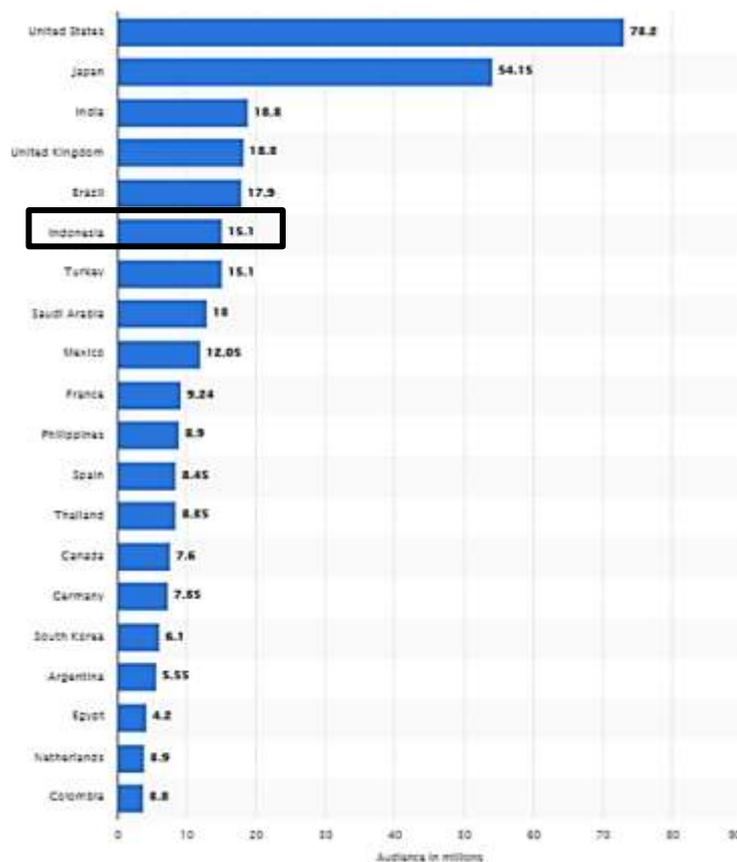


BAB II

LANDASAN TEORI

2.1 *Twitter*

Twitter merupakan layanan jejaring sosial yang berguna untuk saling menghubungkan antara pengguna satu dengan pengguna lainnya (Basri, 2017). Berdasarkan data yang dilansir oleh statista.com, bahwa pada kuartal ketiga tahun 2020, *twitter* memiliki 187 juta pengguna aktif harian yang dapat dimonetisasi di seluruh dunia serta jika dilihat dari data statistik bahwa Indonesia berada di peringkat ke-6 dengan jumlah pengguna *twitter* sebanyak 15,1 juta per April 2021 (Tankovka, 2021).



Gambar 2.1 Grafik Jumlah Pengguna Twitter per Negara 2021

CEO *Twitter* Jack Dorsey mengungkapkan dalam *coffarence call* dengan para investornya menyebutkan bahwa banyak orang menggunakan *twitter* untuk mendapat informasi baru dikarenakan peningkatan pengguna aktif harian setiap tahunnya. Pada tahun 2019 pengguna *twitter* meningkat sebanyak 24% dari 134 juta menjadi 166 juta, pernyataan tersebut dilansir inet.detik.com (Jati, 2020).

Berdasarkan Kementerian Komunikasi dan Informatika Republik Indonesia, kebanyakan pengguna *twitter* di Indonesia adalah konsumen, yaitu tidak memiliki *Blog* atau tidak pernah mengupload video di *Youtube* namun sering *update* status dan saling menimpali komentar di *twitter* (Kominfo, 2020).

2.2 Analisis Sentimen

Analisis sentimen termasuk dalam metode pemrosesan bahasa alami yang tujuannya adalah untuk mengetahui sentimen emosional dari teks yang dianalisis (Nimesh et al, 2019). Tugas utama dari analisis sentimen adalah mengklasifikasikan polaritas informasi dalam dokumen baik kalimat maupun kata-kata. Analisis sentimen bekerja secara sistematis untuk mengidentifikasi, mengekstrak, dan mempelajari kondisi dan informasi subjektif yang secara luas diterapkan pada analisis pendapat konsumen, ulasan, tanggapan *survey*, dan media sosial (Kim Schouten, 2016).

Analisis sentimen juga dapat didefinisikan sebagai proses analisa teks yang dilihat dari sudut pandang polaritas sentimen yang dimiliki. Setiap teks merupakan opini yang memiliki maksud sesuai dengan sumber penulisannya yang menunjukkan kesubjektifitasan, dimana subjektifitas mempengaruhi hasil penelitian dari orang atau alat yang memproses dan membaca teks tersebut.

Polaritas emosi ataupun polaritas dokumen dikelompokkan menjadi positif, negatif, dan netral. Polaritas emosi ditimbulkan dari pengalaman yang diperoleh kemudian dirasakan lalu dicurahkan dalam bentuk tulisan (Mustaqim, 2019).

Proses analisis sentimen memiliki beberapa pedekatan yaitu *Lexicon-Based* atau *Machine Learning* (Vishal A. Kharde, 2016). Terkait penelitian yang dilakukan ini, analisis sentimen dilakukan melalui pendekatan *Machine Learning* yang berarti melakukan analisa sentimen berdasarkan algoritma *Machine Learning* melalui *data training* yang telah dilabeli kemudian dilakukan *data testing* dengan tujuan akhir mengetahui hasil akurasi prediksi sentimen yang dilakukan.

2.3 Algoritma *Random Forest Classifier*

Random Forest Classifier dijelaskan sebagai kumpulan pengklasifikasi terstruktur yang merupakan versi lanjutan dari *Bagging* (Bahrawi, 2019). Dimana *Bagging* adalah metode kombinasi pembelajaran yang meningkatkan hasil secara keseluruhan (Donges, 2019).

Random Forest termasuk dalam algoritma *Machine Learning* yang dapat digunakan untuk melakukan analisis sentimen. *Random forest* dimulai dengan teknik penambangan data dasar, *decision tree*. Dimana kumpulan data *decision tree* digunakan untuk mengklasifikasikan data ke suatu kelas. Proses input variabel dari algoritma *decision tree* yang dilakuka secara acak untuk membuat masing-masing algoritma *decision tree* tidak saling tersambung dan mempengaruhi satu sama lainnya. Fungsi dari pengacakan tersebut juga untuk menghindari *error* dan kesalahan perhitungan akibat dari data yang homogen. Proses kalkulasi *random forest* mirip dengan penilaian manusia yang dalam hal ini dari beberapa algoritma

decision tree dengan tujuan menghindari kesalahan perhitungan dari salah satu individu algoritma *decision tree* (Mustaqim, 2019).

Random Forest Classifier juga merupakan algoritma yang terdiri dari sejumlah besar pohon keputusan individu yang beroperasi sebagai *ensemble*, Setiap pohon individu di *random forest* mengeluarkan prediksi kelas dan kelas dengan suara terbanyak menjadi model prediksi (Yiu, 2019). Sebagai metode *ensemble*, *random forest* menumbuhkan banyak pohon sebagai pembelajar dasar dan agregat untuk memprediksi serta sebagai model *ensemble* yang terdiri dari sejumlah model *decision tree* dimana setiap individunya menggunakan *subset bootstrap* (*bootstrapped subset*) yang diambil dari sampel melalui pemilihan secara acak. pengambilan keputusan mengenai kelas data baru x ditetapkan dengan persamaan (Heryadi et al, 2020) :

$$y(x) = \text{mode}\{C_1(x), C_2(x), \dots, C_m(x)\} \quad (2.1)$$

Dimana m adalah jumlah *decision tree* dari *random forest*, dan mode adalah operator untuk memilih *input* bilangan dengan frekuensi tertinggi.

Di dalam sebuah klasifikasi dengan variabel target bersifat biner atau $t_i \in \{-1, 1\}$ maka :

$$y(x) = \text{sign}\left[\sum_{j=1, \dots, m} C_j(x)\right] = f(x) = \begin{cases} 1, & \sum_{j=1, \dots, m} C_j(x) \geq 0 \\ -1, & \text{otherwise} \end{cases} \quad (2.2)$$

Algoritma *random forest* dapat diringkas ke dalam beberapa urutan langkah sebagai berikut :

- 1) Buat sampel ukuran n secara acak dengan teknik sampling memilih n buah sampel dari data input dengan pengembalian (*replacement*).
- 2) Buat *decision tree* dari sampel. Pada setiap *node* lakukan hal berikut :

- a) Pilih fitur sebanyak d buah tanpa pengembalian (*replacement*)
- b) Pecahlah *node* menggunakan fitur memberikan *split* terbaik berdasarkan fungsi objektif yang digunakan misalnya *gain*.
- c) Ulangi langkah (1) dan (2) sebanyak k kali
- d) Gabungkan hasil prediksi dari setiap *decision tree* untuk membuat keputusan mengenai kelas data berdasarkan suara mayoritas (*majority vote*)

Algoritma *random forest* ada dua faktor yang dipilih secara acak yaitu :

- 1) Data sampel, setiap model *decision tree* akan mendapatkan *input* berupa sejumlah sampel (*subset*) data yang digunakan untuk membuat sebuah *classification tree* yang dipilih dengan teknik pengambilan sampel “acak dengan pemulihan” dari keseluruhan data.
- 2) Fitur data, sejumlah data yang menjadi input bagi setiap model *decision tree* hanya memiliki sebagian fitur yang dipilih secara acak dari keseluruhan fitur data.

2.4 Information Gain

Information gain merupakan salah satu teknik seleksi fitur yang dilakukan untuk mengurangi fitur yang tidak relevan dan mengurangi fitur pada data (Sari, 2016). *Information gain* juga merupakan algoritma terbaik yang digunakan sebagai *feature selection*.

Information Gain dari suatu atribut diperoleh dari nilai *entropy* sebelum pemisahan dikurangi dengan nilai *entropy* setelah pemisahan. Apabila atribut yang

tidak relevan akan menurunkan performa *machine learning*. Sedangkan atribut yang redundan akan membuat *machine learning* bekerja lebih lama (Astuti, 2017).

Seleksi fitur menggunakan *information gain* dilakukan dengan cara menghitung nilai *gain* setiap fitur. Ada tiga tahapan dalam pemilihan fitur menggunakan *Information Gain* diantaranya adalah sebagai berikut :

- 1) Hitung nilai *gain* informasi untuk setiap atribut dalam dataset asli
- 2) Buang semua atribbut yang tidak memenuhi kriteria yang ditentukan
- 3) Dataset direvisi

Untuk menghitung *Information Gain*, dihitung dengan rumus dibawah ini (Han et al, 2012):

$$Info(D) = - \sum_{i=1}^c p_i \log_2(p_i) \quad (2.3)$$

Keterangan dari rumus tersebut adalah :

c : jumlah nilai yang ada pada atribut target (jumlah kelas klasifikasi)

pi : jumlah sampe untuk kelas i

$$Info_A(D) = \sum_{j=1}^v \frac{|D_j|}{|D|} x Info(D_j) \quad (2.4)$$

Keterangan dari rumus tersebut adalah :

A : atribut

|D| : jumlah seluruh sampel data

|D_j| : jumlah sampel untuk nilai j

v : suatu nilai yang mungkin untuk atribut A

Kemudian nilai *information Gain* yang digunakan untuk mengukur efektifitas suatu atribut dalam pengklasifikasian data dapat dihitung dengan rumus di bawah ini :

$$Gain(A) = |Info(D) - Info_A(D)| \quad (2.5)$$

2.5 Literature Review

Sebelumnya para peneliti sudah melakukan beberapa penelitian terkait bidang penelitian dengan memasukkan unsur-unsur tambahan yang dapat memenuhi kebutuhan dengan fokus Analisis Sentimen.

Tabel 2.1 State Of The Art

No	Pengarang	Judul	Metode	Hasil
1.	(Aditya et al, 2020)	Analisis Sentimen Penggunaan <i>Twitter</i> Terhadap Penggunaan Cairan Desinfektan Menggunakan Metode <i>Term Frequency-Inverse Document Frequency</i> dan <i>Support Vector Machine</i>	Metode <i>Term Frequency-Inverse Document Frequency</i> dan <i>Support Vector Machine</i>	- Algoritma SVM menghasilkan nilai <i>accuracy</i> 78.93%, <i>precision</i> 45.83% dan <i>recall</i> 6.41%. - Hasil dari metode TF-IDF dan SVM bahwa dalam masa pandemik Covid-19 masyarakat secara umum masih berpendapat jika penggunaan cairan desinfektan dapat digunakan sebagai salah satu alat untuk memutus mata rantai persebaran virus corona.
2.	(Ruhiana, 2019)	Analisis Sentimen Terhadap Penerapan Sistem Plat Nomor Ganjil/Genap Pada <i>Twitter</i> Dengan Metode Klasifikasi <i>Naïve Bayes</i>	<i>Naïve Bayes</i>	Penelitian ini menghasilkan <i>accuracy</i> 86,67%, <i>precision</i> 71,43% dan <i>recall</i> 80,00% dalam bentuk positif dan negatif untuk penerapan lalu lintas ganjin genap.
3.	(Fauziyyah et al, 2020)	Analisis Sentimen Pandemi Covid-19 Pada Streaming <i>Twitter</i>	<i>Text Mining (Streaming data tweet)</i>	Penelitian ini menghasilkan nilai polaritas analisis

No	Pengarang	Judul	Metode	Hasil
		Dengan <i>Text Mining Python</i>		sentimen kategori netral paling tinggi yaitu 58,94% untuk sentimen COVID-19 dan 55,10% untuk variabel <i>coronavirus</i> .
4.	(Ananta et al, 2020)	Implementasi <i>Naïve Bayes</i> Dan <i>Pos Tagging</i> Menggunakan Metode <i>Hidden Markov Model Viterbi</i> Pada Analisa Sentimen Terhadap Akun <i>Twitter</i> Presiden Joko Widodo Di Saat Pandemi Covid-19	<i>Naïve Bayes</i> dan <i>Pos Tagging Hidden Markov Model Viterbi</i>	<ul style="list-style-type: none"> - Metode <i>Pos Tagging Hidden Markov Model Viterbi</i> menghasilkan akurasi sebesar 69%, 70%, dan 71 % - Metode <i>Naïve Bayes Classifier</i> menghasilkan akurasi sebesar 80%, 81% dan 83%.
5.	(Aprianto et al, 2016)	Analisis Sentimen Dan Peringkasan Opini Pada Ulasan Produk Menggunakan Algoritma <i>Random Forest</i>	<i>Random Forest</i>	<ul style="list-style-type: none"> - Kasus dengan <i>dataset</i> yang memiliki mayoritas kalimat yang kompleks seleksi kata benda dan frasa kata benda ini kurang sesuai karena hanya memperoleh nilai evaluasi sekitar 20%-40%. - Ekstraksi dengan <i>prunning</i> dapat meningkatkan nilai <i>F-score</i> dengan menggunakan parameter yang sesuai.
6.	(Deviyanto et al, 2018)	Penerapan Analisis Sentimen Pada Pengguna <i>Twitter</i> Menggunakan Metode <i>K-Nearest Neighbor</i> .	<i>K-Nearest Neighbor</i>	Hasil pengujian diketahui bahwa nilai akurasi terbesar adalah 67,2% ketika k=5, presisi tertinggi 56,94% ketika k=5, dan <i>recall</i> 78,24% dengan k=15.
7.	(Somantri et al, 2018)	<i>Support Vector Machine</i> Berbasis <i>Feature Selection</i> Untuk <i>Sentiment Analysis</i> Kepuasan Pelanggan Terhadap Pelayanan Warung Dan Restoran Kuliner Kota Tegal.	<i>Support Vector Machine</i> Berbasis <i>Feature Selection</i>	<ul style="list-style-type: none"> - Model <i>SVM-IG</i> dengan tingkat akurasi terbaik sebesar 72,45% mengalami peningkatan sekitar 3,08% yang awalnya hanya 69.36%. - Selisih rata-rata yang dihasilkan setelah dilakukannya optimasi

No	Pengarang	Judul	Metode	Hasil
				SVM dengan <i>feature selection</i> adalah 2,51% kenaikan tingkat akurasi. - <i>Information Gain</i> (IG) (SVM-IG) memiliki tingkat akurasi lebih baik apabila dibandingkan SVM dan <i>Chi Squared</i> (SVM-CS).
8.	(Wijaya, 2017)	Penerapan <i>Information Gain</i> Guna Meningkatkan Akurasi Pada Algoritma <i>Support Vector Machine</i> dan <i>Naive Bayes</i> Dalam Mendiagnosa <i>Chronic Kidney Disease</i>	<i>Information Gain</i> , Algoritma <i>Support Vector Machine</i> dan <i>Naive Bayes</i>	- algoritma SVM menunjukkan tingkat akurasi meningkat 0,3333% dari 98% menjadi 98,3333%. - algoritma <i>Naive Bayes</i> menunjukkan peningkatan sebesar 1,0075% dari 98,4925% menjadi 99,5%.
9.	(Alita et al, 2020)	Pendeteksian Sarkasme Pada Proses Sentimen Menggunakan <i>Random Forest Classifier</i>	<i>Random Forest Classifier</i>	- peningkatan nilai rata-rata akurasi sebesar 16,61 %, nilai presisi sebesar 5,45 %, nilai <i>recall</i> sebesar 9,64% dan kenaikan nilai <i>F1score</i> sebesar 11,27%. - didapatkan dari tweet dengan label positif yang kemudian diberikan label sarkasme atau tidak sarkasme dan didapat hasil label dengan jumlah keseluruhan berlabel sarkasme berjumlah 354 dan tidak sarkasme berjumlah 669.
10.	(Evita Fitri et al, 2020)	Analisis Sentimen Terhadap Aplikasi Ruangguru Menggunakan Algoritma <i>Naive Bayes</i> , <i>Random Forest</i> , Dan <i>Support Vector Machine</i>	<i>Naive Bayes</i> , <i>Random Forest</i> , Dan <i>Support Vector Machine</i>	Hasil performa menunjukkan <i>Random Forest</i> memiliki nilai akurasi tertinggi dari kedua algoritma lainnya dengan

No	Pengarang	Judul	Metode	Hasil
				kenaikan akurasi sebesar 7,16% yaitu Random Forest sebesar 97,16%, sedangkan SVM sebesar 96,01% dan Naïve Bayes sebesar 94,16%.
11.	(Mustaqim T. , 2020)	<i>Sentiment Analysis Opini Pelantikan Kabinet Pemerintah Indonesia Tahun 2019 Menggunakan Vader Dan Random Forest</i>	<i>Vader Dan Random Forest</i>	Menghasilkan tingkat akurasi tertinggi diperoleh oleh kombinasi model algoritma dari <i>random forest, count vectorizer</i> dan <i>n-gram</i> berjumlah 3 gram mencapai 89,19%.
12.	(Singhal et al, 2019)	<i>Bagged Random Forest Approach to Classify Sentiment Based on Technical Words</i>	<i>Bagged Random Forest Approach</i>	Pendekatan <i>Random Forest and bagging</i> menghasilkan akurasi 80,64% dalam hal persentase split menggunakan 66 dan 34 sebagai pelatihan dan pengujian persen dan 86,81% untuk model <i>cross validation</i> .
13.	(Kumar et al, 2020)	<i>Random Forest-Based Sarcastic Tweet Classification Using Multiple Feature Collection</i>	<i>Random Forest dan Multiple Feature Collection</i>	Di antara uji klasifikasi <i>Random Forest-Based Sarcastic Tweet Classification algorithm</i> , dimana Hasil observasi <i>Random Forest</i> dengan <i>overall accuracy</i> (84,7%) dibandingkan dengan model <i>supervised classification</i> lainnya yaitu SVM (78,6%), <i>Logistic Regression</i> (80,5%), dan KNN (73,1%)

No	Pengarang	Judul	Metode	Hasil
14.	(Bouazizi et al, 2019)	<i>Multiclass Sentiment Analysis on Twitter: Classification Performance and Challenge</i>	<i>Multiclass Classification</i>	Pendekatan yang diusulkan untuk <i>Multiclass Classification</i> mencapai akurasi 60,2% untuk 7 kelas sentimen yang berbeda dibandingkan dengan akurasi 81,3% untuk <i>binary classification</i> .
15.	(Bouazizi et al, 2016)	<i>Sentiment Analysis in Twitter: From Classification to Quantification of Sentiment within Tweets</i>	<i>features and special unigram-based features along with the other basic features to perform the ternary classification</i>	Hasil klasifikasi <i>overall accuracy</i> yaitu 83,0% dan hasil dari metrik, M2 memungkinkan pengukuran deteksi perbedaan sentimen yang ada secara berurutan dan nilai yang diperoleh yaitu 47,0% adalah relatif tinggi untuk metrik yang sangat terbatas dibandingkan dengan M1
16.	(Rai et al, 2019)	<i>Sentiment Analysis Using Machine Learning Classifier: Evaluation of Performance</i>	<i>Machine Learning Classifier (Naïve Bayes, Random Forest, Support Vector Machine)</i>	Waktu eksekusi RF jauh lebih tinggi dibandingkan dengan SVM dan NB.
17.	(Gaikwad et al, 2015)	<i>Multiclass Mood Classification on Twitter Using Lexicon Dictionary and Machine Learning Algorithms</i>	<i>Machine Learning Algorithms (SVM, KNN, Naïve Bayesian)</i>	- Hasil SVM mencapai akurasi maksimum 82% dan terendah 77%, <i>Naïve Bayesian</i> mencapai akurasi maksimum 64% dan terendah 57%, KNN mencapai akurasi

No	Pengarang	Judul	Metode	Hasil
				maksimum 73% dan terendah 71% - SVM menghasilkan tingkat akurasi tertinggi untuk kelas <i>Funny, Happy, Sad</i> , dan <i>Angry</i> . Sedangkan <i>Naïve Bayesian</i> menghasilkan akurasi untuk kelas 'None' dan untuk kelas 'Sad' paling rendah sedangkan kelas 'Funny' yang tertinggi.
18.	(Januarsyah et al, 2019)	Perbandingan Algoritma <i>Random Forest, Decision Stump, Naïve Bayes, Bayesian Network</i> , dan Algoritma C4.5 Untuk Prediksi Pola Kartu Poker	<i>Random Forest, Decision Stump, Naïve Bayes, Bayesian Network</i> , dan Algoritma C4.5	Hasil akurasi tertinggi sebesar 74, 2863% yaitu Algoritma <i>Random Forest</i> , sedangkan <i>Decision Stump</i> memiliki akurasi sebesar 49,952%, <i>Naïve Bayes</i> sebesar 49,0644%, <i>Bayesian Network</i> sebesar 48,0764% dan C4.5 sebesar 57,537%

Penelitian yang akan dilakukan memiliki keterkaitan dengan penelitian-penelitian sebelumnya mengenai analisis sentimen pada sosial media yaitu *Twitter*. Penelitian ini bertujuan untuk menganalisis dan memprediksi pendapat dan opini masyarakat terkait hal, isu ataupun tema baik itu tema serius seperti politik, keamanan, pertahanan negara, sampai hal-hal ringan seperti dunia hiburan, *travelling* dan kuliner. Termasuk pada saat ramai isu pandemik Covid-19 (Aditya et al, 2020).

Selain itu penelitian ini juga bertujuan untuk menguji performansi dari algoritma *Random Forest Classifier* dengan menerapkan *Information Gain*. Dimana algoritma *Random Forest* memiliki tingkat akurasi yang lebih tinggi dari

algoritma klasifikasi lainnya. Dapat dibuktikan dengan penelitian yang dilakukan oleh Fitri et al, 2020 dalam penelitiannya menggunakan algoritma *Naïve Bayes*, *Random Forest*, dan *Support Vector Machine* untuk mengetahui algoritma klasifikasi yang memiliki performa tertinggi dan didapatkan hasil algoritma *Random Forest* dengan jumlah akurasi sebesar 97,16% serta nilai AUC 0,996. Algoritma *Support Vector Machine* yang menghasilkan akurasi sebesar 96,01% dengan nilai AUC sebesar 0,543. Algoritma *Naive Bayes* dengan nilai akurasi sebesar 94,16% serta nilai AUC sebesar 0,999. Hasil penelitian tersebut membuktikan bahwa algoritma *Random Forest* memiliki performa nilai akurasi tertinggi dari kedua algoritma lainnya dengan kenaikan akurasi sebesar 7,16%.

Kemudian diperkuat dalam penelitian Januarsyah et al, 2019 dalam membandingkan performa dari algoritma *Random Forest*, *Decision Stump*, *Naïve Bayes*, *Bayesian Network* dan algoritma *C4.5* untuk mengetahui algoritma yang memiliki akurasi tertinggi dan didapatkan hasil algoritma *C4.5* dengan jumlah akurasi sebesar 57,537%. Algoritma *Decision Stump* menghasilkan akurasi sebesar 49,952%. Algoritma *Naïve Bayes* dengan nilai akurasi sebesar 49,0644%. Algoritma *Bayesian Network* dengan nilai akurasi sebesar 48,0764%. Algoritma *Random Forest* sebesar 74,2863%. Hasil penelitian tersebut membuktikan bahwa algoritma *Random Forest* memiliki nilai akurasi tertinggi dari algoritma lainnya.

Information Gain yang merupakan salah satu metode seleksi fitur ini dapat meningkatkan performa algoritma yang digunakan ataupun yang dipasangkan dengannya. Seperti pada penelitian yang dilakukan oleh Somantri et al, 2018 untuk mengetahui performa algoritma *Support Vector Machine* setelah dan sebelum

diterapkan *Information Gain* dan didapatkan hasil tingkat akurasi terbaik sebesar 72,45% mengalami peningkatan sekitar 3,08% yang awalnya hanya 69.36%.

Kemudian, diperkuat dalam penelitian Wijaya, 2017 dalam mengkomparasi performa dari algoritma SVM dan *Naïve Bayes* setelah dan sebelum menggunakan *Information Gain* dan didapatkan hasil algoritma SVM menunjukkan tingkat akurasi meningkat 0,3333% dari 98% menjadi 98,3333% dan algoritma *Naïve Bayes* menunjukkan peningkatan sebesar 1,0075% dari 98,4925% menjadi 99,5%.

Berdasarkan penelitian-penelitian yang dilakukan sebelumnya, penelitian ini akan menggunakan algoritma *Random Forest Classifier* dengan penambahan fitur seleksi berupa *Information Gain* sebagai keterbaruan dari penelitian ini yang berfungsi untuk meningkatkan akurasi pada algoritma tersebut.

No	Penulis/Tahun Penelitian	Lingkup Penelitian																
		Metode											Optimasi					
		SV M	SV M + TF-IDF	Naïve Bayes	Random Forest	K-NN	Decision Stump	Bayesian Network	C4.5	Multiclass classification	Ternary Classification	Text Mining	Pos Tagging Hidden Markov	Information Gain	Vader	Chisquare	Bagging	Multiple feature selection
8.	Januarsyah et al, 2019	√	-	√	√		√	√	√	-	-	-	-	-	-	-	-	-
9.	Rai et al, 2019	√	-	√	√	-	-	-	-	-	-	-	-	-	-	-	-	-
10.	Ruhyana, 2019	-	-	√	-	-	-	-	-	-	-	-	-	-	-	-	-	-
11.	Sighal et al, 2019	-	-	-	√	-	-	-	-	-	-	-	-	-	-	√	-	-
12.	Aditya et al, 2020	-	√	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
13.	Alita et al, 2020	-	-	-	√	-	-	-	-	-	-	-	-	-	-	-	-	-
14.	Ananta et al, 2020	-	-	√	-	-	-	-	-	-	-	√	-	-	-	-	-	-
15.	Fauziyah et al, 2020	-	-	-	-	-	-	-	-	-	√	-	-	-	-	-	-	-
16.	Fitri et al, 2020	√	-	√	√	-	-	-	-	-	-	-	-	-	-	-	-	-
17.	Kumar et al, 2020	-	-	-	√	-	-	-	-	-	-	-	-	-	-	-	√	-
18.	Mustaqim, 2020	-	-	-	√	-	-	-	-	-	-	-	-	√	-	-	-	-
19.	Mitha Maharani Wahyudi , 2021	-	-	-	√	-	-	-	-	-	-	-	√	-	-	-	-	-