BAB II
LANDASAN TEORI

2.1 Penelitian Terkait

Berikut merupakan beberapa peneliti sebagai studi literatur dan memiliki keterkaitan terhadap penelitian yang akan dilakukan dimana dapat dijadikan sebagai pembanding, bahan acuan dan pembelajaran sehingga dapat memudahkan dalam menyelesaikan penelitian.

Tabel 2. 1 Penelitian Terkait

Peneliti/Tahun	Judul	Metode	Hasil
Harun and Ananda (2021)	Analisa Sentimen Opini Publik Tentang Vaksinasi Covid-19 di Indonesia Menggunakan *Naïve Bayes* dan *Decission Tree*	*Naïve Bayes* dan *Decission Tree*	Hasil penelitian analisa sentimen opini masyarakat tentang vaksinasi COVID-19 yang telah dilakukan, cenderung ke tanggapan negatif dengan nilai akurasi 100.00% menggunakan algoritma NBC dan 50.39% menggunakan algoritma Decision Tree
Fajriansyah and Siswanto (2018)	Analisis Sentimen Pengguna Twitter Terhadap Partai Politik Pendukung Calon Gubernur Di Jakarta Menggunakan Algoritma C4.5 Decision Tree Learning	Algoritma C4.5	Hasil dari pengujian akurasi klasifikasi menggunakan pohon keputusan yang dibangun algoritma C4.5 berdasarkan data latih dan 150 data uji yang digunakan menghasilkan rata rata error 36%. Kata
Kusuma and Nugroho (2021)	Analisa Sentimen Pada Twitter Terhadap Kenaikan Tarif Dasar Listrik Dengan Metode Naïve Bayes	Naive Bayes	Hasil penelitian yang telah dilakukan dapat diketahui bahwa sentimen negatif paling banyak terbentuk sekitar 60% dalam menanggapi isu kenaikan tarif dasar listrik
Rakhman and Tsani (2019)	Analisis Sentimen Review Media Massa	Algoritma C4.5	Hasil penelitian analisis sentimen review media massa diperoleh metode

Peneliti/Tahun	Judul	Metode	Hasil
	Menggunakan Metode C4.5 Berbasis Forward Selection		forward selection untuk seleksi fitur dan algoritma C4.5 menghasilkan akurasi yang lebih baik, dibandingkan pada penelitian sebelumnya dimana hasil akurasi tertinggi sebesar 80.00% dengan menggunakan forward selection sebesar 4.00%
Cahyaningtyas, Nataliani and Widiasari (2021)	Analisis sentimen pada rating aplikasi Shopee menggunakan metode Decision Tree berbasis SMOTE	Algoritma Klasifikasi *Decision Tree*	Hasil penelitian dengan menggunakan algoritma Decision Tree dengan SMOTE (Synthetic Minority Oversampling Technique) nilai accuracy-nya menghasilkan 99,91 persen, AUC (Area Under Curve) 0,999, recall 99,88 persen dan nilai precision 99,98 persen. Hasil menggunakan algoritma Decision Tree tanpa SMOTE nilai accuracynya menghasilkan 99,89 persen, AUC (Area Under Curve) 0,950, recall 99,88 persen dan nilai precision 99,98 persen.

Pada penelitan yang dilakukan oleh Harun and Ananda (2021) meneliti tentang Analisa Sentimen Opini Publik Tentang Vaksinasi Covid-19 di Indonesia Menggunakan Naïve Bayes dan Decission Tree. COVID-19 adalah penyakit baru yang dilaporkan di Wuhan China pada Desember 2019. Berdasarkan dari www.covid19.go.id, di Indonesia per tanggal 20 Januari 2021 terdapat 939 ribu lebih kasus dan sebanyak lebih dari 26 ribu kasus yang mengakibatakan kematian. Pesatnya penyebaran COVID-19 dan bahaya yang ditimbulkan, pemerintahan

Indonesia melakukan pencegahan dengan vaksinisasi yang informasinya sudah tersebar diberbagai media sosial, diantarnya ialah facebook page yang dimiliki oleh Kementrian Kesehatan. Facebook page yang memiliki fitur komentar pada postingannya, belum dapat menentukan besar sentimen pengguna tehadap komentar postif atau negatif secara otomatis. Analisa sentimen merupakan bagian dari teks mining untuk pengelompokan polaritas teks dalam mengetahui polaritas suatu opini yang diberikan bersifat positif atau negatif menggunakan algoritma tertentu. Penelitian ini bertujuan untuk memberikan hasil opini masyarakat tentang analisa sentimen vaksinasi COVID-19 menggunakan algoritma Naïve Bayes Classifier (NBC) dan Decision Tree serta membandingkan tingkat akurasi kedua algoritma tersebut. Hasil penelitian analisa sentimen opini masyarakat tentang vaksinasi COVID-19 yang telah dilakukan, cenderung ke tanggapan negatif dengan nilai akurasi 100.00% menggunakan algoritma NBC dan 50.39% menggunakan algoritma Decision Tree.

Penelitain yang dilakukan oleh Fajriansyah and Siswanto (2018) meneliti tentang Analisis Sentimen Pengguna Twitter Terhadap Partai Politik Pendukung Calon Gubernur Di Jakarta Menggunakan Algoritma C4.5 *Decision Tree Learning*. Pada tahun 2017 dilaksanakan pemilihan calon gubernur yang menjadi perhatian masyarakat, informasi digunakan masyarakat untuk menilai partai politik perserta salah satunya opini negatif dan positif yang berasal dari twitter. Dalam menentukan polaritas positif atau negatif suatu opini dapat dilakukan secara manual, tetapi mempertimbangkan bertambahnya opini menjadi semakin banyak, tentunya banyak waktu yang akan semakin banyak terpakai. Maka diajukan sebuah metode machine learning untuk mengklasifikasikan konten opini dari sumber data yang sangat

banyak. Metode machine learning untuk melakukan analisis terhadap konten tweet yaitu menggunakan pohon keputusan atau decision tree learning yang dibangun dengan algoritma C 4.5. Pada penelitian ini akan dilakukan analisis tweet yang bersifat opini terhadap partai politik peserta pemilihan calon gubernur 2017. Analisis sentimen terhadap partai politik dilakukan untuk menganalisa opini negatif dan positif dari pengguna twitter terhadap partai peserta pemilihan calon gubernur 2017. Hasil dari pengujian alpha menunjukan bahwa sistem analisis sentimen menggunakan algoritma C4.5 bebas dari kesalahan sintak dan berjalan sesuai dengan fungsinya. Hasil pengujian aplikasi dapat disimpulkan bahwa aplikasi analisis sentimen ini baik untuk referensi tambahan masyarakat terhadap partai politik. Hasil dari pengujian akurasi klasifikasi menggunakan pohon keputusan yang dibangun algoritma C4.5 berdasarkan data latih dan 150 data uji yang digunakan menghasilkan rata rata error 36%.

Pada penelitian yang dilakukan oleh Kusuma and Nugroho (2021) meneliti tentang Analisa Sentimen Pada Twitter Terhadap Kenaikan Tarif Dasar Listrik Dengan Metode Naïve Bayes. Opini masyarakat yang tertuang dalam media sosial twitter berupa sebuah persepsi, baik itu positif maupun negatif. Melimpahnya opini masyarakat dapat dimanfaatkan sebagai bahan penelitian untuk mencari sebuah informasi. Pemanfaatan informasi tersebut membutuhkan teknik analisis yang tepat sehingga informasi yang dihasilkan mampu membantu banyak pihak dalam mengambil sebuah keputusan. Untuk mengatasi permasalahan di atas digunakan teknik data mining yang tepat yaitu sentimen analisis. Oleh sebab itu, pada penelitian ini mencoba melakukan analisa sentimen untuk melihat persepsi masyarakat terhadap isu kenaikan tarif dasar listrik pada media sosial twitter

menggunakan metode baïve bayes dengan mengklasifikasikan sentimen menjadi positif, negatif dan netral. Dari hasil penelitian yang telah dilakukan dapat diketahui bahwa sentimen negatif paling banyak terbentuk sekitar 60% dalam menanggapi isu kenaikan tarif dasar listrik.

Pada penelitian yang berkaitan dengan metode klasifikasi yang digunakan untuk proses *teks mining* yaitu Algoritma C4.5 atau disebut juga algoritma *decision tree* yang dilakukan Rakhman and Tsani (2019) meneliti tentang Analisis Sentimen Review Media Massa Menggunakan Metode C4.5 Berbasis *Forward Selection*. Hasil penelitian analisis sentimen review media massa diperoleh metode forward selection untuk seleksi fitur dan algoritma C4.5 menghasilkan akurasi yang lebih baik, dibandingkan pada penelitian sebelumnya dimana hasil akurasi tertinggi sebesar 80.00%. Pada analisis sentimen review media massa menggunakan C4.5 dengan seleksi fitur forward selection mendapatkan hasil akurasi sebesar 84,00%. Dengan demikian dapat disimpulkan bahwa penelitian klasifikasi dengan algoritma C4.5 pada analisis sentimen review media massa dapat ditingkatkan akurasinya dengan menggunakan forward selection sebesar 4.00%.

Pada penelitian yang dilakukan oleh Cahyaningtyas, Nataliani and Widiasari (2021) meneliti tentang Analisis sentimen pada rating aplikasi Shopee menggunakan metode Decision Tree berbasis SMOTE. Analisis sentimen adalah cabang penelitian text mining yang melakukan proses dalam klasifikasi pada dokumen teks. Analisis sentimen merupakan mengekstrasi pendapat, emosi dan evaluasi

seseorang yang tertulis mengenai suatu topik tertentu dengan memanfaatkan teknik pemrosesan bahasa alami. Peneliti melakukan penelitian tentang analisis

sentimen pada rating aplikasi Shopee dengan menggunakan metode Decision Tree. Tujuan penelitian ini untuk mengetahui tingkat keakurasian dan mengetahui pendapat pengguna mengenai aplikasi Shopee ini. Hasil penelitian dengan menggunakan algoritma Decision Tree dengan SMOTE (Synthetic Minority Oversampling Technique) nilai accuracy-nya menghasilkan 99,91 persen, AUC (Area Under Curve) 0,999, recall 99,88 persen dan nilai precision 99,98 persen. Hasil menggunakan algoritma Decision Tree tanpa SMOTE nilai accuracynya menghasilkan 99,89 persen, AUC (Area Under Curve) 0,950, recall 99,88 persen dan nilai precision 99,98 persen. Dari hasil evaluasi yang ada dapat ditarik kesimpulan SMOTE dapat berpengaruh terhadap nilai accuracy dan AUC (*Area Under Curve*), serta untuk nilai recall dan precision tidak berpengaruh atau hasilnya tetap sama walau menggunakan SMOTE atau tanpa SMOTE. Selisih nilai accuracy yang didapat adalah 0,02 persen dan untuk AUC-nya sebesar 0.049.

2.2 Tinjauan Pustaka

Dalam melakukan penelitian ini, penulis melakukan tinjauan pustaka sebagai studi literatur dan memiliki keterkaitan terhadap penelitian yang akan dilakukan dimana dapat dijadikan sebagai pembanding, bahan acuan dan pembelajaran sehingga dapat memudahkan dalam menyelesaikan penelitian. Studi literatur digunakan untuk memberikan gambaran serta mendukung teori ataupun konsep yang berkaitan dengan klasifikasi sentimen dengan algoritma C4.5 terhadap komentar kenaikan BBM. Studi literatur dalam penelitian ini diperoleh melalui jurnal online, buku, penelitian sebelumnya, dengan studi literatur tersebut peneliti mendapat referensi untuk menyelesaikan tujuan penelitian dalam pengumpulan data yang dibutuhkan.

2.2.1. *Sentimen*, Opini dan Analisis Sentimen

a. Sentimen

Menurut Kamus Besar Bahasa Indonesia (KBBI), sentimen berarti pendapat atau pandangan yang didasarkan pada perasaan yang berlebih-lebihan terhadap sesuatu. Sedangkan menurut *Merriam-Webster's Online Dictionary,* sentimen menunjukkan pendapat tetap(terus menerus) yang merefleksikan / mencerminkan perasaan seseorang.

b. Opini

Opini dan konsep terkait seperti sentimen, evaluasi, tingkah laku, dan emosi merupakan subjek studi dari analisis sentimen dan *opinion mining* (Bing Liu, 2012). Opini atau pendapat merupakan pusat hampir semua aktivitas manusia dan menjadi pengaruh utama dari perilaku. Persepsi terhadap realitas untuk mengevaluasi objek di sekitar.

Defenisi Opini menurut Hajmohammadi dkk dalam jurnal *Opinion Mining and Sentiment Analysis: A Survey* pada tahun 2012, yaitu:

1. Pandangan atau penilaian yang terbentuk tentang sesuatu, tidak selalu berdasarkan fakta atau pengetahuan.
2. Keyakinan atau pandangan dari sejumlah besar atau mayoritas orang-orang tentang hal tertentu.

Secara umum, opini mengacu pada apa yang orang pikirkan tentang sesuatu. Dengan kata lain, opini adalah keyakinan subjektif, dan merupakan hasil emosi atau interpretasi fakta.

c. Analisis Sentimen

Analisis sentimen disebut juga *opinion mining*, adalah bidang ilmu yang menganalisa pendapat, sentimen, evaluasi, penilaian, sikap dan emosi publik

terhadap entitas seperti produk, jasa, organisasi, individu, masalah, peristiwa, topik, dan atribut mereka (Hartanto, 2017). Analisis sentimen berfokus pada opini-opini yang mengekspresikan atau mengungkapkan sentimen positif atau negatif.

Secara umum analisis sentimen yang telah diteliti memiliki tiga tingkat (level), yaitu:

1. Level dokumen: mengklasifikasikan apakah seluruh dokumen opini mengungkapkan sentimen positif atau negatif. Analisis mengasumsikan bahwa setiap dokumen mengungkapkan opini yang objektif tentang suatu entitas tunggal (misalnya, produk tunggal).
2. Level kalimat: menentukan apakah setiap kalimat menyatakan opini positif, negatif, atau netral.
3. Level entitas dan aspek: Menemukan sentimen pada entitas dan / atau aspeknya. Sebagai contoh, kalimat "kualitas panggilan iPhone baik, tetapi daya tahan baterai pendek". Ada dua aspek evaluasi, kualitas panggilan dan baterai kehidupan, dari iPhone (entitas). Sentimen pada kualitas panggilan iPhone adalah positif, tapi sentimen pada hidup baterai negatif. Kualitas panggilan dan daya tahan baterai iPhone adalah target pendapat.

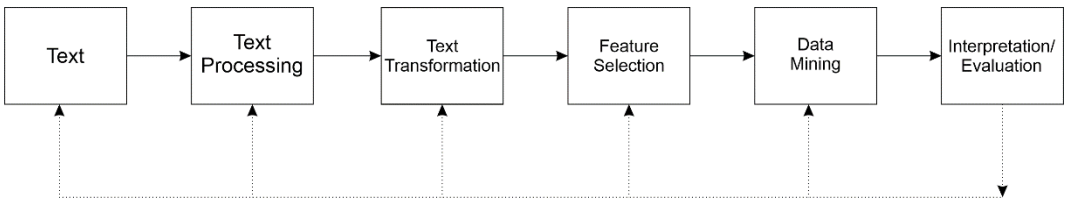
Analisis sentimen merupakan salah satu cabang penelitian *text mining* . Analisis sentimen hadir untuk menangani kondisi ledakan informasi teks yang tidak terstruktur.

2.2.2. *Text Mining*

Text mining adalah lintas disiplin ilmu yang mengacu pada pencarian informasi, data mining, *machine learning*, statistik, dan komputasi linguistik. *Text mining* juga dikenal dengan *text data mining* atau pencarian pengetahuan di basis data tekstual adalah proses yang semi otomatis melakukan ekstraksi dari pola data (Harun and Ananda, 2021).

Tipe pekerjaan *text mining* meliputi kategorisasi, *text clustering*, ekstraksi konsep/entitas, analisis sentimen, *document summarization*, dan *entity-relation modeling* (yaitu, hubungan pembelajaran antara entitas) (Harun and Ananda, 2021). Sumber data yang digunakan pada *text mining* adalah kumpulan teks yang memiliki format yang tidak terstruktur atau minimal semi terstruktur. Tujuan dari *text mining* adalah untuk mendapatkan informasi yang berguna dari sekumpulan dokumen.

Text mining merupakan variasi dari *data mining* yang berusaha menemukan pola yang menarik dari sekumpulan data tekstual yang berjumlah besar. Perbedaan terletak pada pola yang digunakan, pola *text mining* diambil dari sekumpulan bahasa alami yang tidak terstruktur sedangkan dalam *data mining* pola diambil dari *database* testruktur. Beberapa tahapan proses pokok dalam *text mining*, yaitu pemrosesan awal teks (*text preprocessing*), transformasi teks (*text transformation*) atau (*Feature Generation*), pemilihan fitur (*feature selection*), dan penemuan pola *text* atau *data mining* (*pattern discovery*).



Gambar 2. 1 Proses Text Mining

a. *Text*

Sama halnya dengan permasalahan pada *data mining,* pada *text mining* data yang akan diolah jumlahnya sangat banyak, dimensi yang tinggi, data dan struktur yang terus berubah dan data *noise*. Perbedaan di antara keduanya adalah pada data yang digunakan. Pada *data mining,* data yang digunakan adalah *structured data,* sedangkan pada *text mining,* data yang digunakan *text mining* pada umumnya adalah *unstructured data*, atau *minimal semistructured*. Hal ini menyebabkan adanya tantangan tambahan pada *text mining* yaitu struktur text yang *complex* dan tidak lengkap, arti yang tidak jelas dan tidak standar, dan bahasa informal.

Data *training* dari teks komentar yang akan digunakan pada penelitian ini ditentukan berdasarkan beberapa ketentuan berdasarkan penelitian yang analisis sentimen yang dilakukan oleh Kusuma and Nugroho (2021) Berdasarkan dua penelitian tersebut suatu teks dapat diambil sebagai data *training* dikarenakan dua hal, yaitu adanya kata entitas/target diiringi minimal satu kata sentimen positif/negatif dan atau *emoticon*. Sementara untuk menentukan suatu kata sentimen tergolong positif atau negatif didasarkan pada penelitian yang dilakukan dengan mengacu pada bobot sentimen dari halaman website *Sentiwordnet*.

b. Text Preprocessing

Preprocessing dilakukan untuk menghindari data yang kurang sempurna, gangguan pada data, dan data-data yang tidak konsisten (Kusuma and Nugroho, 2021). Tahap *preprocessing* diperlukan untuk membersihkan data dari *noise,* menyeragamkan bentuk kata dan mengurangi volume kata.

Menurut Ma'rifah, Wibawa and Akbar (2020) umumnya ada empat tahap *preprocessing* untuk dokumen teks, yaitu cas*e folding, tokenizing, stopwords removal* dan *stemming*, penjelasan dapat dilihat sebagai berikut :

1. *Case Folding*

   Case folding yaitu penyeragaman bentuk huruf menjadi *lower case* atau *upper case* serta penghapusan angka dan tanda baca. Dalam hal ini yang digunakan hanya huruf latin antara "a" sampai dengan "z".

2. *Tokenizing*

   Tokenizing yaitu memenggal dokumen menjadi satuan kata. Tokenizing diperlukan untuk proses *stopwords removing* berbasis kamus yang berjalan dengan perulangan pada tiap-tiap kata dalam dokumen.

3. *Stopwords Removal*

   Stopword Removal merupakan proses *filtering*, pemilihan kata-kata penting dari hasil token yaitu membuang kata-kata yang sering muncul dan bersifat umum, kurang menunjukkan relevansinya dengan teks.

4. *Stemming*

   Stemming bertujuan untuk menyaring kata dasar dari setiap kata yang yang ada dalam dokumen. Sehingga setiap kata yang berbeda tetapi serupa misal seperti 'mengukur' dan 'pengukuruan' dianggap satu kata yang sama yaitu 'ukur'.

c. Featured Selection

Tahap *feature selection* merupakan tahap lanjut dari pengurangan dimensi pada proses transformasi teks (Fajriansyah and Siswanto, 2018). Pada tahap *feature selection* terbagi atas :

1. Filtering

Filtering adalah proses untuk memilih kata-kata penting dari hasil *tokenization. Filtering* dilakukan dengan menggunakan algoritma *stopword removal. Stopword removal* digunakan untuk membuang kata-kata yang sering muncul dan bersifat umum, kurang menunjukkan relevansinya dengan teks. Membuang kata-kata yang sering muncul namun tidak memiliki pengaruh apapun terhadap ekstraksi sentimen. Misalnya “di”, “oleh”, “pada”, “sebuah”, “karena” dan lain sebagainya. Kata-kata yang akan dibuang didefinisikan dalam *stopword list*.

Tabel 2. 2 *Stopword List*

Atau	Di	Oleh	Karena
Saya	Gue	Min	Dalam
Sih	Aja	Bagi	Iya
Dengan	Ke	Lu	Gaul
Cc	Punya	Gan	Aku

2. Stemming

Stemming adalah tahap membuat kata yang berimbuhan kembali ke bentuk asalnya (Sentiaji, 2014). Atau dengan kata lain, *stemming* merupakan proses mencari akar kata dan menghilangkan imbuhan pada kata. *Stemming* bertujuan mengurangi variasi kata yang memiliki kata dasar sama.

3. Convert negation

Convert negation adalah proses mengganti negasi yang terdapat dalam komentar. Negasi adalah sesuatu yang dikenal dalam semua bahasa dan biasanya negasi digunakan untuk mengubah polaritas dari suatu

pernyataan. Kata-kata yang bersifat negasi adalah “kurang”, “tidak”, “enggak”, “ga”, “nggak”, “tak”, dan “gak”.

d. Interpretation* atau *evaluation

Interpretation atau *evaluation*, hasil dari proses *mining* akan diinterpretasikan kedalam bentuk tertentu untuk kemudian dilakukan proses evaluasi. Apabila hasil keluaran dari penemuan pola belum sesuai untuk aplikasi, dilanjutkan evaluasi dengan melakukan iterasi ke satu atau beberapa tahap sebelumnya. Sebelum masuk pada pengembangan model, penting dilakukan evaluasi dari model yang dihasilkan proses *mining*. Meninjau kembali langkah-langkah yang dilakukan dalam membangun model untuk memastikan itu telah mencapai tujuan bisnis (Jackson, 2002). Interpretasi yakni, penafsiran pola *mining* agar lebih dimengerti oleh pengguna. Hasil interpretasi merupakan tahap akhir dari proses *text mining* dan akan disajikan ke pengguna dalam bentuk *summarization* dan visualisasi.

2.2.3 Algoritma *C4.5*

Algoritma C4.5 merupakan algoritma yang digunakan untuk membentuk pohon keputusan. Pohon keputusan merupakan metode klasifikasi dan prediksi yang sangat kuat dan terkenal. Metode pohon keputusan mengubah fakta yang sangat besar menjadi pohon keputusan yang merepresentasikan aturan. Aturan dapat dengan mudah dipahami dengan bahasa alami. Dan mereka juga dapat diekspresikan dalam bentuk bahasa basis data seperti *Structured Query Language* untuk mencari *record* pada kategori tertentu (Rakhman and Tsani, 2019)

Algoritma C4.5 merupakan kelompok algoritma *Decision Tree*. Algoritma ini mempunyai *input* berupa *training samples* dan *samples*. *Training samples* berupa

data contoh yang akan digunakan untuk membangun sebuah *tree* yang telah diuji kebenarannya. Sedangkan *samples* merupakan *field-field* data yang nantinya akan digunakan sebagai parameter dalam melakukan klasifikasi data (Umam, Puspitasari and Nurhadi, 2020)

Menurut (Umam, Puspitasari and Nurhadi, 2020) Algoritma C4.5 adalah salah satu metode untuk membuat *decision tree* berdasarkan *training* data yang telah disediakan. Algoritma C 4.5 dibuat oleh Ross Quinlan yang merupakan pengembangan dari ID3 yang juga dibuat oleh Quinlan. Beberapa pengembangan yang dilakukan pada C4.5 adalah sebagai antara lain bisa mengatasi *missing value*, bisa mengatasi *continue* data, dan *pruning*. Pohon keputusan banyak sekali perkembangan tetapi yang sering dipakai adalah ID3 dan C4.5. Keduanya mempunyai prinsip yang sama dikarenakan Algoritma C4.5 merupakan pengembangan dari ID3, tetapi mempunyai perbedaan utama yaitu :

1. C4.5 dapat menangani atribut yang kontinu dan diskrit dan juga dapat menangani data *training* dengan nilai yang hilang atau data yang kosong.
2. Hasil yang didapat dari Algoritma C4.5 akan terpangkas setelah dibentuk
3. Pemilihan atribut yang dilakukan dengan menggunakan *Gain Ratio*. Algoritma C4.5 merupakan perbaikan dari ID3 menggunakan *Gain Ratio* untuk diperbaharui *information gain.*

Secara umum alur proses algoritma C4.5 untuk membangun pohon keputusan dalam *data mining* adalah sebagai berikut : (Umam, Puspitasari and Nurhadi, 2020)

a. Pilih atribut sebagai akar

b. Buat cabang untuk tiap-tiap nilai

c. Bagi kasus dalam cabang

d. Ulangi proses untuk setiap cabang sampai semua kasus pada cabang memiliki kelas yang sama.

a. Perhitungan *Information Gain*

Perhitungan *information gain* dilakukan untuk *attribute selection measure* yang digunakan untuk memilih atribut pada setiap simpul pada pohon keputusan. Atribut dengan *information gain* tertinggi atau nilai pengurangan *entropy* yang terbesar dipilih sebagai tes atribut pada simpul. Untuk menghitung gain digunakan rumus seperti tertera dalam persamaan (2.1**)**

$$Gain(S, A) = Entropy(S) - (\frac{S_0}{S} \times Entropy(S_0) + \frac{S_1}{S} \times Entropy(S_1)) \qquad (2.1)$$

Keterangan :

S : Jumlah seluruh kasus

S_0: Jumlah kasus dengan nilai 0

S_1: Jumlah kasus dengan nilai 1

A : atribut

Perhitungan nilai *entropy* dapat dilihat pada persamaan (2.2)

$$Entropy(S) = \frac{-Pos(S)}{S} \times log_2 \frac{Pos(S)}{S} + \frac{Neg(S)}{S} \times log_2 \frac{Neg(S)}{S} \qquad (2.2)$$

Keterangan :

S : Jumlah seluruh kasus

Neg(S) : Jumlah kasus dengan kelas negatif

Pos(S) : Jumlah kasus dengan kelas positif

b. Pembentukan Pohon Keputusan

Algoritma C4.5 membangun pohon keputusan dari *training set* yang telah ditentukan menggunakan konsep *information entropy*. Secara umum langkah-langkah algoritma C4.5 untuk membangun pohon keputusan adalah sebagai berikut:

1. Terdapat masukan berupa *training set* yang setiap sampelnya telah diberi kelas atau kategori.
2. Jika seluruh sampel pada *training set* memiliki kelas yang sama maka pohon keputusan akan memiliki satu *node* berupa *leaf node* yang diberi label kelas yang terdapat pada semua sampel dalam *training set*.
3. Jika seluruh sampel tidak dalam satu kelas yang sama maka akan dicari *gain* tertinggi dari seluruh atribut untuk memilih atribut yang paling berpengaruh pada *training set*, dan akan dijadikan atribut penguji pada *node* tersebut.
4. Jumlah cabang setiap *node* dibangun berdasarkan partisi nilai dari atribut pengujian. Jika ada partisi nilai yang memiliki nilai *entropy* nol, maka cabang dari partisi nilai tersebut menjadi *leaf node* yang diisi kelas yang memiliki jumlah kemunculan tertinggi pada *training* data.
5. Jika ada cabang dari *node* yang dibentuk pada langkah sebelumnya belum mencapai *leaf node*, maka akan dicari nilai gain seperti pada langkah nomor 3 dimulai dari cabang paling kiri yang belum mencapai *leaf node*.

6. Jika seluruh cabang dari *node* yang dibentuk pada langkah sebelumnya telah mencapai *leaf node*, maka akan dicek cabang dari *node* diatas dari *node* yang dibentuk pada langkah sebelumnya, jika cabang tersebut belum mencapai *leaf node* maka akan dicari nilai gain seperti pada langkah nomor 3.
7. Proses yang sama akan dilakukan secara *rekursif* untuk membentuk pohon keputusan dari setiap sampel.
8. Proses *rekursif* akan berhenti jika semua sampel pada *node* memiliki kelas yang sama, semua simpul sudah mencapai *leaf node* atau semua atribut telah digunakan untuk memartisi sampel. Berikut ini adalah algoritma pembentuk pohon.

Algorima : Generate_decision_tree
Narasi : Membuat pohon keputusan
Masukkan : Sampel data pelatihan, *samples*, yang di presentasikan dengan atribut bernilai diskret,
Keluaran : Pohon keputusan

1) Buat simpul *N*;
2) If *samples* semua memiliki kelas yang sama, C, then
3) Return *N* sebagai simpul daun dengan label kelas C,
4) If *attribute-list* kosong then
5) Return *N* sebagai simpul daun dengan label kelas terbanyak di *samples*
6) Pilih *test-attribute*, yaitu salah satu atribut dari *attribute-list* dengan *gain-ratio* terbesar
7) Beri label pada simpul *N* dengan test-attribute;
8) For setiap nilai ??$_{??}$ pada *test-attribute*;
9) Tambahkan cabang pada simpul *N* untuk kondisi test-attribute = a;
10) Buat partisi sampel ??$_{??}$ dari samples dimana test-attribut =a;
11) If ???? kosong then
12) Templekan daun yang diberi label dengan kelas terbanyak di *samples*;
13) Else tempelkan simpul yang dibuat oleh Generate_decision_tree (??$_{??}$, *Attribute-list-test-attrribute*);

Gambar 2. 2 Algoritma Pembentukan Pohon

c. Klasifikasi Menggunakan Pohon Keputusan

Model pohon keputusan yang telah dibentuk menggunakan algoritma C4.5 selanjutnya digunakan untuk mengklasifikasikan sebuah kalimat opini. Kata-kata yang akan diuji adalah kata-kata yang menjadi atribut penguji pada pohon keputusan, jika pada kalimat opini terdapat kata yang tidak menjadi atribut penguji pada pohon keputusan maka kata tersebut tidak akan diuji. Jika pada kalimat terdapat kata yang menjadi atribut penguji simpul yang sedang diuji maka simpul berikutnya yang diuji adalah simpul ruas kanan dan seluruh simpul ruas kiri tidak akan diuji begitu juga sebaliknya. Berikut adalah langkah-langkah klasifikasi menggunakan pohon keputusan :

1. Simpul akar akan menjadi simpul pertama yang akan diuji,
2. jika pada kalimat terdapat kata yang menjadi atribut penguji pada simpul maka selanjutnya akan diuji simpul pada ruas kanan yang merupakan cabang dari simpul yang diuji sebelumnya. Jika pada kalimat tidak terdapat kata yang menjadi atribut penguji pada simpul tersebut maka selanjutnya akan diuji simpul pada ruas kiri yang merupakan cabang dari simpul yang diuji sebelumnya.
3. Jika pada simpul yang diuji bukan merupakan *leaf node* maka akan dilakukan langkah sebelumnya yaitu langkah 2 pada simpul yang sedang diuji.
4. Proses akan berhenti jika simpul yang diuji merupakan *leaf node.*

2.2.4. *K-Fold Cross Validation*

K-fold cross validation digunakan untuk mengestimasi kesalahan prediksi dalam mengevaluasi kinerja model. Data dibagi menjadi himpunan bagian k

berjumlah hampir sama. Model dalam klasifikasi dilatih dan diuji sebanyak k. Disetiap pengulangan, salah satu himpunan bagian akan digunakan sebagai data training dan data testing (Mardiana, Kusnandar and Satyahadewi, 2022). Langkah-langkah dari k fold cross validation yaitu:

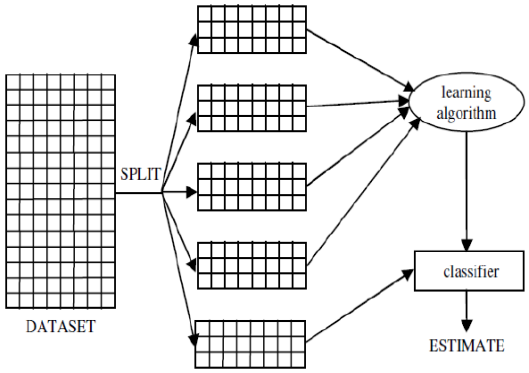
1. Total data dibagi menjadi k bagian.
2. Fold ke-1 adalah ketika bagian ke-1 menjadi data uji (testing data) dan sisanya menjadi data latih (training data). kemudian, hitung akurasi atau kesamaan atau kedekatan suatu hasil pengukuran dengan angka atau data yang sebenarnya berdasarkan porsi data tersebut. Perhitungan akurasi tersebut menggunakan persamaan sebagai berikut.

$$Akurasi = \frac{\sum data\ uji\ benar\ klsifikasi}{\sum total\ data\ uji}\ x\ 100\%$$

3. Fold ke-2 adalah ketika bagian ke-2 menjadi data uji (testing data) dan sisanya menjadi data latih (training data). kemudian hitung akurasi berdasarkan porsi data tersebut.
4. Demikian seterusnya hingga mencapai fold ke-k. Hitung rata-rata akurasi dari k buah akurasi diatas. Rata-rata akurasi ini menjadi akurasi final.

Pada proses evaluasi *k-fold cross validation*, perlu dibentuk k subset dari data sets yang ada. Misalnya seperti penelitian Fuadah *et* al (2022), 5-*fold cross validation* berarti 4 *subsets* digunakan sebagai data training dan 2 subset digunakan sebagai data testing, dilakukan 5 kali iterasi. Hasil pengukuran adalah nilai rata-rata dari 5 kali pengujian. Alasan peneliti menimplementsikan 5-*fold cross validation* untuk meningkatkan performansi sistem dalam mengklasifikasikan menjadi lima kategori. Pada tahap awal akan mengalami proses *preprocessing* yang terdiri dari

augmentasi untuk memperbanyak data dan Resize untuk menyeragamkan data. Implementasi 5-*fold cross validation* dilakukan pada tahap pelatihan model menggunakan data latih dan data validasi untuk mendapatkan model terbaik. seperti ilustrasi pada Gambar 2.3 berikut.



Gambar 2. 3 Prosedur *5-fold cross validation* (Arifin and Fitrianah, 2018)

2.2.5. Evaluasi *Confusion Matrix*

Salah satu metode evaluasi yang digunakan untuk klasifikasi dengan algoritma C4.5 adalah *confusion matrix*. *Confusion matrix* adalah salah satu *tools* penting dalam metode visualisasi yang digunakan pada mesin pembelajaran yang biasanya memuat dua kategori atau lebih. Sebanyak setengah atau dua pertiga dari data keseluruhan digunakan untuk keperluan proses *training* sedangkan sisanya digunakan untuk keperluan *testing*. Untuk memperoleh informasi hasil pencarian akurasi, dilakukan perhitungan recall dan precision. *Precision* dapat dianggap sebagai ukuran ketepatan atau ketelitian, sedangkan *recall* adalah kesempurnaan (Arifin and Ariesta, 2019).

Confusion matrix merupakan tabel yang digunakan untuk mengevaluasi kinerja dari suatu model klasifikasi. Tabel terdiri atas banyaknya baris data uji yang

diprediksi bernar atau tidak benar dari model klasifikasi. Berikut contoh perhitungan akurasi tabel *confusion matrix:*

Tabel 2. 3 Contoh Tabel *Confusion Matrix* Prediksi Dua Kelas

		Prediksi	
		Positif	Negatif
Aktual	Positif	TP	FN
	Negatif	FP	TN

Dimana TP merupakan jumlah prediksi yang benar dari contoh negatif, FN adalah jumlah prediksi yang salah dari contoh positif, FP adalah jumlah prediksi yang salah prediksi dari contoh negatif dan TN adalah jumlah prediksi yang benar dari contoh positif.

Rumus *Accuracy* (AC) adalah jumlah prediksi yang benar. Ini ditentukan dengan persamaan :

$$\text{AC} = \frac{TP+TN}{TP+FP+TN+FN} \dots\dots\dots\dots\dots\dots\dots\dots\dots \quad (2.3)$$

Recall Recall atau sensitivity: menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi, rumus mencari *Recall*:

$$True\ Positive/Recall = \frac{TP}{TP+FN} \quad \dots\dots\dots\dots\dots\dots\dots \quad (2.4)$$

Precision menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model, rumus :

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (2.5)$$

2.2.6. *Machine Learning*

Machine learning adalah cabang dari kecerdasan buatan, merupakan disiplin ilmu yang mencakup perancangan dan pengembangan algoritma yang memungkinkan komputer untuk mengembangkan perilaku yang didasarkan kepada data empiris, seperti dari sensor data pada basis data (Arifin and Ariesta, 2019). Sistem pembelajaran dapat memanfaatkan contoh (data) untuk menangkap ciri yang diperlukan dari probabilitas yang mendasarinya (yang tidak diketahui). Data dapat dilihat sebagai contoh yang menggambarkan hubungan antara variabel yang diamati. Fokus besar penelitian *Machine learning* adalah bagaimana mengenali secara otomatis pola kompleks dan membuat keputusan cerdas berdasarkan data. Kesukarannya terjadi karena himpunan semua perilaku yang mungkin, dari semua masukan yang dimungkinkan, terlalu besar untuk diliput oleh himpunan contoh pengamatan (data pelatihan). Karena itu *Machine learning* harus merampatkan (generalisasi) perilaku dari contoh yang ada untuk menghasilkan keluaran yang berguna dalam kasus-kasus baru.

Sejak pertama kali komputer diciptakan manusia sudah memikirkan bagaimana caranya agar komputer dapat belajar dari pengalaman. Hal tersebut terbukti pada tahun 1952, Arthur Samuel menciptakan sebuah program, *game of checkers*, pada sebuah komputer IBM. Program tersebut dapat mempelajari gerakan untuk memenangkan permainan *checkers* dan menyimpan gerakan tersebut kedalam memorinya Istilah *machine learning* pada dasarnya adalah proses komputer untuk belajar dari data (*learn from data*). Tanpa adanya data, komputer tidak akan bisa belajar apa-apa. Oleh karena itu jika kita ingin belajar *machine learning*, pasti akan terus berinteraksi dengan data. Semua pengetahuan *machine*

learning pasti akan melibatkan data. Data bisa saja sama, akan tetapi algoritma dan pendekatannya berbeda-beda untuk mendapatkan hasil yang optimal.

2.2.7. Bagian Machine Learning

Ada beberapa bagian pada *machine learning*, sistem pembelajaran mesin terdiri dari tiga bagian utama, yaitu (Arifin and Ariesta, 2019)

1. Model: sistem yang membentuk prediksi atau identifikasi.
2. Parameter: sinyal atau faktor yang digunakan oleh model untuk membentuk keputusannya.
3. Pembelajaran: sistem yang menyesuaikan parameter dan model dalam prediksi versus hasil aktual.

Machine learning merupakan salah satu cabang dari disiplin ilmu Kecerdasan Buatan (*Artificial Intellegence*) yang membahas mengenai pembangunan sistem yang berdasarkan pada data. Banyak hal yang dipelajari, akan tetapi pada dasarnya ada 4 hal pokok yang dipelajari dalam *machine learning*:

1. *Supervised machine learning algorithms*

   Supervised machine learning adalah algoritma *machine learning* yang dapat menerapkan informasi yang telah ada pada data dengan memberikan label tertentu, misalnya data yang telah diklasifikasikan sebelumnya (terarah). Algoritma ini mampu memberikan target terhadap *output* yang dilakukan dengan membandingkan pengalaman belajar di masa lalu.

2. *Unsupervised machine learning algorithms*

   Unsupervised machine learning adalah algoritma *machine learning* yang digunakan pada data yang tidak mempunyai informasi yang dapat

diterapkan secara langsung (tidak terarah). Algoritma ini diharapkan mampu menemukan struktur tersembunyi pada data yang tidak berlabel.

3. *Semi-supervised machine learning algorithms*

   Semi-supervised machine learning adalah algoritma yang digunakan untuk melakukan pembelajaran data berlabel dan tanpa label. Sistem yang menggunakan metode ini dapat meningkatkan efisiensi *output* yang dihasilkan.

4. *Reinforcement machine learning algorithms*

   Reinforcement machine learning adalah algoritma yang mempunyai kemampuan untuk berinteraksi dengan proses belajar yang dilakukan, algoritma ini akan memberikan poin (*reward*) saat model yang diberikan semakin baik atau mengurangi poin (*error*) saat model yang dihasilkan semakin buruk. Salah satu penerapannya adalah pada mesin pencari.

2.2.8. *RapidMiner*

RapidMiner merupakan perangakat lunak yang bersifat terbuka (*open source*). RapidMiner adalah sebuah solusi untuk melakukan analisis terhadap *data mining, text mining* dan analisis prediksi. Berbagai teknik deskriptif dan prediksi digunakan RapidMiner untuk memberikan kepada kepada pengguna sehingga dapat membuat keputusan yang paling baik. Terdapat kurang lebih 500 operator *data mining* yang dimiliki RapidMiner termasuk operator untuk *input*, *output*, *datapreprocessing* dan *visualisasi*. RapidMiner merupakan *software* yang berdiri sendiri untuk analisis data dan sebagai mesin *data mining*yang dapat diintegrasikanpada produknya sendiri. RapidMiner ditulis dengan menggunakan bahasa java sehingga dapat bekerja di semua sistem operasi (Triyansyah and

Fitrianah, 2018). Dalam penelitian ini memilih aplikasi RapidMiner dikarenakan aplikasi RapidMiner memiliki tingkat akurasi kurang dari 100%. Sehingga, dapat membantu dalam menentukan dan melihat seberapa kasus sesuai dengan kebutuhan dan keinginan, Rapid Miner juga sangat efektif dalam perhitungan di berbagai metode salah satunya *Decision Tree* (C4.5) (Mardiana, Kusnandar and Satyahadewi, 2022).

RapidMiner memiliki beberapa sifat sebagai berikut:

1. Ditulis dengan bahasa pemrograman Java sehinggadapat dijalankan di berbagai sistem operasi.
2. Konsep multi-layer untuk menjamin tampilan datayang efisien dan menjamin penanganan data.
3. Memiliki GUI, command line mode, dan Java APIyang dapat dipanggil dari program lain.

Beberapa Fitur dari RapidMiner, antara lain:

1. Banyaknya algoritma *data mining*, seperti *decision tree* dan *self-organization map*.
2. Bentuk grafis yang canggih, seperti tumpang tindihdiagram histogram, *tree chart* dan 3D *Scatter plots*.
3. Banyaknya variasi *plugin*, seperti *text plugin* untukmelakukan analisis teks.
4. Menyediakan prosedur *data mining* dan *machine learning* termasuk: ETL (*extraction, transformation,loading*), data *preprocessing*, *visualisasi*, *modelling*dan evaluasi.

5. Proses *data mining* tersusun atas operator-operatoryang *nestable*, dideskripsikan dengan XML, dandibuat dengan GUI
6. Mengintegrasikan proyek data mining Weka danstatistika R.