

IMPLEMENTASI ALGORITMA *MULTINOMIAL NAIVE BAYES* DALAM ANALISIS SENTIMEN KOMENTAR *YOUTUBE* TERKAIT KEMAJUAN *ARTIFICIAL INTELLIGENCE*

Aulia Azzaskia¹, Mohammad Syafrullah^{2*}

^{1,2}Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur, Jakarta, Indonesia

Email: ¹2011501885@student.budiluhur.ac.id, ^{2*}mohammad.syafrullah@budiluhur.ac.id
(* : corresponding author)

Abstrak-Kemajuan pesat teknologi informasi dan komunikasi, terutama dalam bidang *Artificial Intelligence* (AI), telah memberikan dampak signifikan di berbagai sektor seperti pendidikan, kesehatan, dan industri. Meskipun kemajuan *Artificial Intelligence* (AI) meningkatkan efisiensi, otomatisasi pekerjaan, dan kemajuan teknologi di berbagai bidang, kekhawatiran masyarakat terkait hilangnya pekerjaan, isu privasi, dan etika masih menjadi tantangan. Penelitian ini bertujuan untuk menganalisis sentimen publik terhadap kemajuan *Artificial Intelligence* (AI) dengan menerapkan algoritma *Multinomial Naive Bayes* pada komentar video *YouTube*. *YouTube* menjadi salah satu media sosial yang memberi kesempatan bagi pengguna untuk mengunggah, menonton, berbagi video, dan berinteraksi melalui komentar, sehingga layak dijadikan bahan dalam analisis sentimen. Analisis sentimen merupakan sebuah proses dalam *text mining* yang berfokus pada identifikasi dan ekstraksi informasi subyektif dari teks, seperti opini, perasaan, hingga komentar pengguna terhadap suatu topik. Sebanyak 999 data komentar dikumpulkan, dengan 936 data setelah *preprocessing*, kemudian data dilabeli untuk mengkategorikan sentimen, menghasilkan 348 sentimen positif dan 369 sentimen negatif dari total *data labeling* sebanyak 717, pengurangan data ini dikarenakan penghapusan sentimen netral. Penerapan algoritma *Multinomial Naive Bayes* menunjukkan akurasi masing-masing 81.94%, 75.46%, dan 75.96% untuk rasio data 80:20, 70:30, dan 60:40. Hasil penelitian ini memberikan wawasan penting tentang persepsi publik terhadap *Artificial Intelligence* (AI), yang dapat menjadi panduan bagi pengembangan teknologi dan kebijakan di masa depan.

Kata Kunci: *Artificial Intelligence* (AI), *YouTube*, *Multinomial Naive Bayes*, *Text Mining*

IMPLEMENTATION OF THE *MULTINOMIAL NAIVE BAYES* ALGORITHM IN SENTIMENT ANALYSIS OF *YOUTUBE* COMMENTS REGARDING PROGRESS IN *ARTIFICIAL INTELLIGENCE*

Abstract-The rapid advancement of information and communication technology, especially in the field of *Artificial Intelligence* (AI), has had a significant impact in various sectors such as education, healthcare, and industry. Although advances in AI have increased efficiency, job automation, and technological advancements in various fields, public concerns regarding job loss, privacy issues, and ethics remain a challenge. This research aims to analyse public sentiment towards advances in AI by applying the *Multinomial Naive Bayes* algorithm to *YouTube* video comments. *YouTube* is one of the social media that provides opportunities for users to upload, watch, share videos, and interact through comments, making it worthy of being used as material in sentiment analysis. Sentiment analysis is a process in *text mining* that focuses on identifying and extracting subjective information from text, such as users' opinions, feelings, and comments on a topic. A total of 999 comment data was collected, with 936 data after *preprocessing*, then the data was labelled to categorise the sentiment, resulting in 348 positive sentiments and 369 negative sentiments out of a total of 717 labelled data, this data reduction was due to the removal of neutral sentiments. The application of the *Multinomial Naive Bayes* algorithm showed accuracies of 81.94%, 75.46%, and 75.96% for data ratios of 80:20, 70:30, and 60:40, respectively. The results of this study provide important insights into public perceptions of *Artificial Intelligence* (AI), which can guide future technology and policy development.

Keywords: *Artificial Intelligence* (AI), *YouTube*, *Multinomial Naive Bayes*, *Text Mining*

1. PENDAHULUAN

Kemajuan teknologi informasi dan komunikasi telah membawa kemajuan pesat dalam berbagai bidang, salah satunya adalah *Artificial Intelligence* (AI). Sejak diperkenalkan oleh Alan Turing dan diformalisasi pada Konferensi Dartmouth tahun 1956. Pada era revolusi 4.0 ini, AI telah menjadi teknologi yang berperan penting dalam pendidikan, kesehatan, industri, dan kehidupan sehari-hari[1]. Implementasi AI seperti *Google Assistant* dan *Chat GPT* menunjukkan bagaimana AI kini telah menjadi bagian dari aktivitas kita. Namun demikian,

kemajuan *AI* tidak sepenuhnya mendapatkan opini yang positif dari masyarakat. Saat ini, masyarakat dapat meluapkan opini melalui media sosial, seperti *YouTube*.

Youtube menjadi salah satu media sosial populer tak berbayar yang mana para pengguna dapat melakukan pengunggahan video, menonton, berbagi video, hingga interaksi dalam kolom komentarnya[2]. Dengan memanfaatkan data komentar pada konten *YouTube* yang mengupas tentang kemajuan teknologi *AI*. Analisis sentimen dapat dilakukan untuk memahami opini, sikap, atau emosi masyarakat yang dicurahkan dalam teks perihal tersebut. Beberapa penelitian terdahulu melakukan pengambilan data komentar *YouTube* dengan bahasa pemrograman *python* yang telah terkoneksi dengan *API YouTube*.

Penelitian terdahulu yang dilakukan oleh Oktavia & Rahman Isnain pada 2024 yang membahas topik “Melakukan Analisa Sentimen Opini Mengenai *Tools Artificial Intelligence* (AI) Pada *Twitter* Dengan Metode *Multinomial Naive Bayes*”. Penelitian ini melakukan *scrapping* data *twitter* sebanyak 2119 data *tweets* dan pemodelan algoritma *Multinomial Naive Bayes* dengan hasil akurasi sebesar 79.41%[3]. Kemudian, penelitian terdahulu oleh Yasir & Suraji pada 2023 yang membahas topik “Perbandingan Metode Klasifikasi *Naive Bayes*, *Decision Tree*, dan *Random Forest* Terkait Analisis Sentimen *YouTube* Mengenai Kenaikan Biaya Haji 2023”. Penelitian ini melakukan *crawling* data sebanyak 1041 dari komentar *YouTube*. Penelitian ini dilakukan untuk menghasilkan kategori sentimen dan membandingkan hasil akurasi dari tiap algoritma. Maka dari itu, didapatkan akurasi tertinggi sebesar 90% dengan menggunakan algoritma *Naive Bayes*, sebesar 87% dengan algoritma *Random Forest*, dan sebesar 83% dengan algoritma *Decision Tree* [4].

Dalam penelitian ini, peneliti melakukan analisis sentimen komentar *YouTube* dengan menggunakan algoritma *Multinomial Naive Bayes*. Dilakukan pengambilan dataset dari komentar *YouTube* pada dua *channel YouTube*. Data diperoleh dengan menggunakan bahasa pemrograman *python* yang telah terkoneksi *API YouTube*. Penelitian ini memiliki tujuan untuk mengevaluasi opini masyarakat dalam komentar *YouTube* tersebut kedalam kategori positif dan negatif dengan *InSet Lexicon Method*.

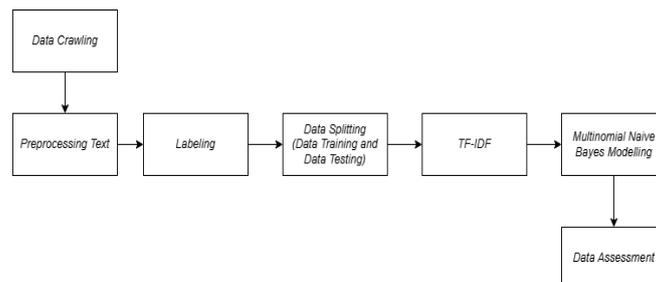
InSet Lexicon berisikan daftar kata yang berisi *labeling* positif maupun negatif dengan bobot nilai untuk per kata. Metode ini digunakan untuk membantu mengkategorikan sentimen secara otomatis berdasarkan kata-kata yang muncul dalam teks. Dengan menggunakan *InSet Lexicon*, analisis sentimen dapat dilakukan lebih cepat dan terstruktur. Namun, dapat terjadi pembatasan kamus, jika ada ditemukan kata yang tidak ada pada kamus walaupun bisa saja kata itu mengandung nilai sentimen[5].

Penelitian ini bertujuan mengevaluasi sentimen masyarakat terhadap kemajuan *AI* melalui komentar di *YouTube*, sekaligus memberikan kontribusi dalam analisis sentimen teks berbahasa Indonesia. Dengan menggunakan algoritma *Multinomial Naive Bayes* dan *InSet Lexicon*, penelitian ini mengungkap persepsi masyarakat yang dapat membantu pembuat kebijakan dan pengembang teknologi memahami dampak sosial *AI*. Hasilnya juga diharapkan mendukung perancangan strategi komunikasi yang lebih efektif. Selain itu, penelitian ini memperkaya literatur ilmiah tentang penggunaan *InSet Lexicon* dalam analisis sentimen bahasa Indonesia.

Perbedaan penelitian ini dengan penelitian terdahulu terletak pada fokus topik, metode pengkategorian sentimen, dan sumber data. Penelitian terdahulu oleh Oktavia & Rahman Isnain (2024) berfokus pada analisis sentimen mengenai *tools AI* di *Twitter* menggunakan algoritma *Multinomial Naive Bayes*, sedangkan penelitian oleh Yasir & Suraji (2023) membandingkan akurasi tiga algoritma klasifikasi (*Naive Bayes*, *Decision Tree*, dan *Random Forest*) dalam analisis sentimen komentar di *YouTube* terkait kenaikan biaya haji. Penelitian ini memfokuskan pada analisis sentimen komentar *YouTube* menggunakan metode *Multinomial Naive Bayes* dengan *InSet Lexicon* untuk mengklasifikasikan sentimen menjadi positif dan negatif. Selain itu, penelitian ini menggunakan dataset komentar dari dua *channel YouTube*, sehingga menawarkan pendekatan yang berbeda dalam mengevaluasi opini masyarakat pada platform, metode, dan fokus analisis yang berbeda.

2. METODE PENELITIAN

Dalam membangun sebuah sistem analisis sentimen diperlukan sebuah desain awal atau skema untuk memastikan sistem berjalan secara terstruktur dan efisien. Pada gambar 1. merupakan desain alur sistem analisis sentimen, sebagai berikut.



Gambar 1. Alur Sistem Analisis

Sketsa pada gambar 1. menjelaskan tahapan perjalanan sistem dimulai dari pengumpulan data menggunakan melalui *YouTube API* dengan bahasa pemrograman *python* yang kemudian disimpan kedalam *database* (basis data) dalam bentuk data mentah. Selanjutnya, dataset yang ada masuk kedalam tahapan *text preprocessing* untuk mendapatkan dataset yang lebih bersih. Jika tahapan ini selesai, dataset kembali disimpan dalam *database* dan diproses kembali untuk tahapan *labeling* berfungsi menghasilkan kategori sentimen, yaitu sentimen positif dan sentimen negatif. Ada dua cara dalam tahap *labeling*, pertama *labeling* secara otomatis dengan penerapan *Inset Lexicon* dan yang kedua, *labeling* manual oleh pakar (ahli dalam bidang tertentu). Setelah menghasilkan *labeling*, dataset kembali disimpan di *database* dan dilakukan tahapan *data splitting* untuk memproses perhitungan nilai akurasi menggunakan metode *Multinomial Naive Bayes*, presisi, *recall*, *f1-score* menggunakan metode *Confusion Matrix*, hingga perhitungan bobot kata menggunakan *TF-IDF*. Dalam penelitian ini *data training* dan *data testing* diuji dengan tiga perbandingan berbeda 80:20, 70:30, dan 60:40. Setelah semua nilai berhasil didapatkan, hasilnya ditampilkan pada layar visualisasi dalam bentuk grafik maupun *wordclouds*.

2.1 Data Crawling

Data crawling menjadi langkah awal untuk mengambil dataset dengan bahasa pemrograman *python* yang tekoneksi *YouTube API*, kemudian dataset disimpan kedalam *database*.

2.2 Text Preprocessing

Pemrosesan teks atau preprocessing text merupakan bagian mengubah atau memperbaiki data awal agar dapat diproses menjadi data yang siap dipakai[5]. Dalam bagian ini ada beberapa tahapan, meliputi pembersihan teks dari simbol atau karakter khusus yang tidak relevan, menormalkan kata dengan mengedit semua teks menjadi huruf kecil saja, menghapus kata yang tidak sesuai, mengganti kata yang berimbuhan menjadi kata dasar. Terdapat beberapa tahapan dalam proses *preprocessing text* yang dilakukan dalam keperluan penelitian, diantaranya:

- a. *Case Folding*, melakukan konversi keseluruhan teks menjadi huruf kecil[6]. Hal ini berfungsi agar dataset analisis teks yang diproses dianggap similar atau sama.
- b. *Cleaning*, melakukan pembersihan atau membuang noise simbol atau karakter yang ada pada dataset. Semua elemen disesuaikan agar tidak merusak pemrosesan data selanjutnya. Seperti, spasi ekstra, emoji, simbol, dan lain sebagainya.
- c. *Tokenizing*, proses pembentukan teks atau pemisahan kata yang menghasilkan unit-unit lebih kecil (minor)[7].
- d. Normalisasi, melakukan perubahan kata menjadi bentuk kata dasar yang menyesuaikan dengan standar konsisten atau Ejaan Bahasa Indonesia Yang Disesuaikan (EYD).
- e. *Stemming*, proses memperkecil kata-kata infleksi kembali menjadi bentuk kata dasar atau akar kata.
- f. Menghilangkan *Stopword*, melakukan filtrasi dalam menghilangkan semua kata yang kurang penting dengan menerapkan *stoplist*[8].

2.3 Labeling

Labeling menjadi tahap pelabelan hasil dua kategori sentimen, yaitu positif dan negatif. Tahapan ini ada dua cara yang dilakukan, langkah pertama *labeling* secara otomatis dengan menerapkan *InSet lexicon*. *InSet Lexicon* berisikan daftar kata yang berisi *labeling* positif maupun negatif dengan bobot nilai untuk per kata. Namun, dapat terjadi pembatasan kamus, jika ada ditemukan kata yang tidak ada pada kamus walaupun bisa sajan kata itu mengandung nilai sentimen[6].

2.4 Data Splitting

Data Splitting merupakan pembagian data, menjadi data latih (training) dan data uji (testing). *Data Training* berfungsi membuat data menjadi konsisten dalam melatih *text preprocessing*. Keseluruhan dataset dihitung bobot perkata dengan menerapkan perhitungan *TF-IDF* agar menghasilkan pemodelan *data training*. Sedangkan, *data testing* berfungsi menguji *data training* dengan menciptakan *predict evaluation* pada kategori sentimen. Dalam proses ini dilakukan tiga kali percobaan dengan tiga rasio perbandingan yang berbeda, diantaranya 80:20, 70:30, serta 60:40.

2.5 TF-IDF

Perhitungan bobot kata atau biasa disebut juga dengan nilai *TF-IDF* merupakan pengukuran pentingnya kata dalam dokumen relatif terhadap keseluruhan dokumen yang tersedia. Langkah awal merupakan perhitungan nilai TF atau Term Frequency dengan mengukur kemunculan kata yang sering. Langkah kedua merupakan perhitungan IDF atau Inverse Document Frequency (IDF) dengan mengukur pentingnya kata. Kemudian langkah akhir merupakan perhitungan gabungan nilai *TF-IDF* dalam memberikan nilai bobot kata yang kemunculannya sering dalam sebuah dokumen tapi jarang muncul pada dokumen lain. Persamaan ini menjabarkan perhitungan nilai *TF-IDF* dengan tiga tahapan[9].

a. Nilai TF

$$TF_{(k,n)} = \frac{f_{k,n}}{T_n} \quad (1)$$

b. Nilai IDF

$$IDF_{(k,N)} = \log\left(\frac{T}{|\{n \in N : k \in n\}|}\right) \quad (2)$$

c. Nilai $TF-IDF$

$$TF - IDF_{(k,n,N)} = TF_{(k,n)} \times IDF_{(k,N)} \quad (3)$$

Keterangan :

k	: Kata tertentu
n	: Dokumen tertentu
$f_{k,n}$: Frekuensi kemunculan kata k pada dokumen n
T_n	: Total kata dalam dokumen n
N	: Keseluruhan dokumen
T	: Total jumlah dokumen dalam kumpulan
$ \{n \in N : k \in n\} $: Jumlah dokumen terdapat kata-k
$TF_{(k,n)}$: Nilai TF
$IDF_{(k,N)}$: Nilai IDF

2.6 Multinomial Naïve Bayes

Metode *Multinomial Naïve Bayes* merupakan sebuah metode khusus klasifikasi yang berproses berdasarkan kemungkinan yang kondisional, metode ini juga melakukan prediksi yang ada pada setiap kelas dalam dataset[10]. Metode ini sering kali diimplementasikan pada *text mining* untuk mengklasifikasi teks berdasarkan frekuensi kemunculan serangkaian kata dengan ketentuan (term) dalam teks. Setiap ketentuan (term) diperlakukan secara independen, sehingga peluang kemunculannya tidak bergantung pada ketentuan (term) lainnya. Algoritma *Multinomial Naïve Bayes* mudah di implementasikan dan memiliki nilai akurasi yang tinggi[11].

Penelitian diawali dengan proses *data crawling* untuk pengambilan data sebagai dataset dalam proses selanjutnya, kemudian *preprocessing text, labeling* dengan *InSet Lexicon Method*, serta perhitungan $TF-IDF$. Setelah semua ini dilakukan, dilanjutkan perhitungan prior atau peluang awal kemunculan suatu kategori terhadap data latih. Kemudian, dilakukan perhitungan probabilitas kondisional dengan *smoothing laplace* umumnya nilai yang digunakan dalam parameter *smoothing laplace* ($\alpha=1$). Hal ini dilakukan agar *probability* tidak bernilai 0. Setelah mendapatkan nilai *conditional probability*, dapat dilanjutkan perhitungan skor tiap kelas atau class dengan rumus, sebagai berikut [9].

$$\text{Nilai Class}(q) = \log P(q) + \sum_{w_p \in d} (\log P(w|q) \times tfidf(w, t)) \quad (4)$$

Keterangan:

$\log P(w q)$: Kemunculan <i>probability class-w</i> pada <i>class-q</i>
$tfidf(w, t)$: Hasil $TF-IDF$ dari <i>class-w</i> dalam data t

2.7 Confusion Matrix

Dalam penerapan algoritma *Multinomial Naive Bayes* terdapat tahapan akhir untuk mengevaluasi nilai prediksi yaitu dengan *confusion matrix method*. Disisi lain, metode ini dapat digunakan untuk melakukan perhitungan nilai presisi, *recall*, serta *f1-score*. Dilakukan perhitungan akurasi, presisi, *recall*, dan *f1-score* sebagai hasil validasi[6], sebagai berikut.

a. Rumus perhitungan nilai akurasi *Accuracy*

$$A = \frac{TP + TN}{TP + FP + TN + FN} \times 100\% \quad (5)$$

b. Rumus perhitungan nilai presisi

$$P = \frac{TP}{TP + FP} \times 100\% \quad (6)$$

c. Rumus perhitungan nilai *recall*

$$Re = \frac{TP}{TP + FN} \times 100\% \quad (7)$$

d. Rumus perhitungan nilai *f1-score*

$$f1 - score = 2 \times \frac{R \times P}{R + P} \times 100\% \quad (8)$$

Keterangan:

TP : *TruePositive* (Prediksi benar positif)
 TN : *TrueNegative* (Prediksi benar negatif)
 FP : *FalsePositive* (Prediksi salah positif)
 FN : *FalseNegative* (Prediksi salah negatif)

3. HASIL DAN PEMBAHASAN

Dalam melakukan penelitian analisis sentimen ini, terdapat langkah-langkah berurutan dalam penerapan metode tersebut, mulai dari langkah pengambilan data hingga pembagian data. Di bawah ini adalah penjelasan keseluruhan langkah yang berurutan, sebagai berikut.

3.1 *Crawling Data*

Dilakukan pengambilan dataset Sebanyak 999 data dari komentar *YouTube* pada dua *channel*. Pertama, *channel Kok Bisa?* dengan judul video “AI Sama Sekali Enggak Seperti yang Kalian Kira (ft. Maudy Ayunda)”, yang dapat di akses dengan *url YouTube* https://www.YouTube.com/watch?v=_FcHPPrY5zg. Kedua, *channel Dr. Indrawan Nugroho* dengan judul video “Peluang dan Ancaman Kecerdasan Buatan (AI)”, yang dapat di akses dengan *url YouTube* <https://www.YouTube.com/watch?v=qAe8oGKiRbc&t=154s>. Data diperoleh dengan menggunakan bahasa pemrograman *python* yang telah terkoneksi *API YouTube*. Penelitian ini memiliki tujuan untuk mengevaluasi opini masyarakat dalam komentar *YouTube* tersebut kedalam kategori positif dan negatif dengan *InSet Lexicon Method*. *Data crawling* menjadi langkah awal untuk mengambil dataset dengan bahasa pemrograman pemrograman *python* terkoneksi *YouTube API* yang kemudian dataset disimpan kedalam *database*. Seluruh dataset yang didapat berisikan informasi berupa *id*, *time*, *user*, dan *comment_user* yang kemudian disimpan dalam database atau basis data. Hasil dari *data crawling* dapat dilihat pada tabel 1.

Tabel 1. *Data Crawling*

Komentar <i>YouTube</i>	Username	Time
Tapi enak misalkan dia kerja di industri kita. Kita tinggal ngambil untung nya aja 😊😊	@Dzakygamespoter	08 Maret 2024
Tetap sngat berbahaya bro , kalo pengembangan AI ntinya di masa depan sudah sempurna!!!	@fiandp89	22 Juli

3.2 *Text Preprocessing*

Text preprocessing diterapkan untuk melakukan pembersihan dataset agar pelaksanaan klasifikasi sentimen memiliki hasil akurasi yang pasti, tahapan ini meliputi enam tahapan, sebagai berikut.

3.2.1 *Case Folding*

Melakukan konversi secara merata agar teks menjadi huruf kecil. Hal ini berfungsi agar dataset analisis teks yang diproses dianggap similar atau sama. Pada tabel 2. terdapat kata " Tapi" dan "tapi" dianggap similar atau sama setelah dilakukan tahapan ini.

Tabel 2. *Case Folding*

Sebelum	Sesudah
Tapi enak misalkan dia kerja di industri kita. Kita tinggal ngambil untung nya aja 😊😊	tapi enak misalkan dia kerja di industri kita. kita tinggal ngambil untung nya aja 😊😊

3.2.2 *Cleaning*

Cleaning dataset diterapkan untuk melakukan pembersihan atau membuang noise simbol atau karakter yang ada pada dataset. Seperti pada tabel 3. Sebagai berikut.

Tabel 3. *Cleaning*

Sebelum	Sesudah
Tapi enak misalkan dia kerja di industri kita. Kita tinggal ngambil untung nya aja 😊😊	tapi enak misalkan dia kerja di industri kita kita tinggal ngambil untung nya aja

3.2.3 *Tokenizing*

Tahapan ini diterapkan untuk memproses teks yang menghasilkan unit-unit lebih kecil atau minor. Seperti pada tabel 4. Sebagai berikut.

Tabel 4. *Tokenizing*

Sebelum	Sesudah
tapi enak misalkan dia kerja di industri kita kita tinggal ngambil untung nya aja	['tapi', 'enak', 'misalkan', 'dia', 'kerja', 'di', 'industri', 'kita', 'kita', 'tinggal', 'ngambil', 'untung', 'nya', 'aja']

3.2.4 Normalisasi

Dalam tahap ini, dilakukan penyesuaian kata yang berstandar KBBI. Selain itu, dilakukan representasi dan transmudasi kata untuk mengurangi variasi kata tak sesuai. Pada tabel 5. terdapat kata "yg" menjadi "yang", "ga" menjadi "tidak", "bs" menjadi "bisa", dan sebagainya.

Tabel 5. Normalisasi

Sebelum	Sesudah
yg pasti mereka ga bs kuasai atau gantiin kerjaan pelaut	yang pasti mereka tidak bisa kuasai atau gantiin kerjaan pelaut

3.2.5 Stemming

Tahapan ini diterapkan untuk memperkecil semua kata kembali menjadi akar kata atau kata dasar. Pada tabel 6. terdapat kata, “bersaing” menjadi “saing” dan lain sebagainya.

Tabel 6. *Stemming*

Sebelum	Sesudah
semoga saya bisa bersaing dengan ai	moga saya bisa saing dengan ai

3.2.6 Menghilangkan *Stopword*

Tahapan ini diterapkan untuk melakukan penghilangan semua kata yang kurang penting dengan menggunakan *stoplist*. Pada tabel 7. terdapat kata "saya", "bisa", "dengan", dan sebagainya. Hal ini dilakukan dengan memanfaatkan daftar kata atau *stopword list*.

Tabel 7. Menghilangkan *Stopword*

Sebelum	Sesudah
moga saya bisa saing dengan ai	moga saing ai

3.3 Labeling

Dalam proses *labeling* ini, label diberikan pada data yang telah melalui proses *preprocessing* untuk memberikan sentimen positif atau negatif. Dalam penelitian ini, terdapat 717 dataset yang diberi hasil *text preprocessing* yang diberi label, 348 dari dataset tersebut berkelas positif dan 369 dari dataset tersebut berkelas negatif.

3.4 Data Splitting

Dalam proses *splitting data* atau pembagian data, dua kategori dataset yang telah dihasilkan dibagi menjadi data latih (*training*) dan data uji (*testing*). Dilakukan percobaan penelitian sejumlah tiga kali dengan tiga rasio perbandingan data latih (*training*) dan data uji (*testing*) yang berbeda, diantaranya 80:20, 70:30, dan 60:40.

3.5 TF-IDF

Pembobotan kata dihitung melalui penerapan *TF-IDF Method* menggunakan *data training*. Pertama, pada tabel 8. terdapat menghitung frekuensi kemunculan kata pada dokumen (TF).

Tabel 8. Nilai TF

Teks	Kata	TF
“aiancam ganti manusia”	ai	0.25
	ancam	0.25
	ganti	0.25
	manusia	0.25

Tahap kedua yang dilakukan pada tabel 9. terdapat perhitungan nilai *IDF* dengan menghitung *log* dari hasil pembagian total dokumen dan jumlah dokumen kata tertentu.

Tabel 9. Nilai *IDF*

Teks	Kata	<i>IDF</i>
"ai ancam ganti manusia"	ai	$\log(6/1) = 0.77815$
	ancam	$\log(6/1) = 0.77815$
	ganti	$\log(6/1) = 0.77815$
	manusia	$\log(6/1) = 0.77815$

Tahap akhir adalah perhitungan nilai *TF-IDF*, pada tabel 10. terdapat perhitungan dari hasil kali dari nilai *TF* dan *IDF*.

Tabel 10. Nilai *TF-IDF*

Teks	Kata	<i>TF-IDF</i>
"ai ancam ganti manusia"	ai	0,04402
	ancam	0,19454
	ganti	0,11928
	manusia	0,07526

Pada tabel 11. berisikan kamus kata *data training* yang didapatkan dari perhitungan frekuensinya dengan menggunakan metode *TF-IDF*.

Tabel 11. Kamus Kata

Kamus Kata	Jumlah
kontrol', 'ai', 'kenal', 'buruk', 'malah', 'empati', 'kaya', 'bikin', 'mimpi', 'nikola', 'manusia', 'euforia', 'harga', 'cerdas', 'ahli', 'please', 'robot', 'review', 'google', 'wkwwk', 'iya', 'emosi', 'tidak', 'nyangka', 'takluk', 'butuh', 'andai', 'sumpah', 'kayak', 'ancam', 'saing', 'tesla', 'pribadi', 'olah', 'canggih', 'banget', 'kreatifitas', 'ganti', 'tenaga', 'temu', 'dok', 'konsultan', 'chatgpt', 'asisten'	44

3.6 Multinomial Naïve Bayes

Dalam proses ini dilakukan perhitungan dataset dengan menerapkan algoritma *Multinomial Naive Bayes*. Pertama, pada tabel 12. terdapat perhitungan prior kemunculan pada sentimen kategori positif dan kategori negatif.

Tabel 12. Nilai Prior

Prior	Hasil
<i>Positive</i>	$3/6 = 0.5$
<i>Negative</i>	$3/6 = 0.5$

Dilanjutkan perhitungan pada tabel 13. untuk mendapatkan nilai *conditional probability* menggunakan *smoothing laplace*.

Tabel 13. *Conditional Probability*

Kata	Kata	Kelas	Nilai
"google salah"	google	Positif	$\frac{0,07782 + 1}{(2,05399) + 1(44)} = 0.0234$
		Negatif	$\frac{0 + 1}{(1,7268) + 1(44)} = 0.02567$
	salah	Positif	$\frac{0 + 1}{(2,05399) + 1(44)} = 0.02171$
		Negatif	$\frac{0 + 1}{(1,7268) + 1(44)} = 0.0218$

Tahapan akhir, pada tabel 14. terdapat perhitungan nilai *class positive* dan *class negative* dalam menentukan kategori sebuah dokumen.

Tabel 14. *Score Positive dan Negative*

<i>Data Testing</i>	=	"google saing"	
<i>Class Positive</i>	=	$\log_{(Positive)} + ((\log(0,0234) \times (0,09727)) + (\log(0,02379) \times (0,05964)) + ((\log(0,02171) \times (0)) + ((\log(0,02312) \times (0,09727)) + ((\log(0,02171) \times (0)) + ((\log(0,02171) \times (0)) + ((\log(0,02171) \times (0)) + ((\log(0,02171) \times (0,09727))))))$	= -0,87739
<i>Class Negative</i>	=	$\log_{(Negative)} + ((\log(0,02184) \times (0,09727)) + (\log(0,02357) \times (0,05964)) + ((\log(0,02171) \times (0)) + ((\log(0,02312) \times (0,09727)) + ((\log(0,02171) \times (0)) + ((\log(0,02171) \times (0)) + ((\log(0,02171) \times (0,09727))))))$	= -0,88025

- [7] Y. Akbar and T. Sugiharto, “Analisis Sentimen Pengguna Twitter di Indonesia Terhadap ChatGPT Menggunakan Algoritma C4.5 dan Naïve,” *Jurnal Sains dan Teknologi*, vol. 5, no. 1, pp. 115–122, 2023, doi: 10.55338/saintek.v4i3.1368.
- [8] A. A. Ningtyas, A. Solichin, and R. Pradana, “Analisis Sentimen Komentar Youtube Tentang Prediksi Resesi Ekonomi Tahun 2023 Menggunakan Algoritme Naïve Bayes,” *Bit (Fakultas Teknologi Informasi Universitas Budi Luhur)*, vol. 20, no. 1, pp. 9-16, 2023.
- [9] M. Y. Pratama, U. A. Putri, P. A. D. Angraini, D. Puspita, and F. Kurniawan, “Analisis Sentimen Chat GPT sebagai Masa Depan Pekerja pada Media Sosial Youtube menggunakan Algoritma Klasifikasi Naive Bayes,” *Explore: Jurnal Sistem Informasi dan Telematika*, vol. 14, no. 2, pp. 193-200, 2023, doi: 10.36448/jsit.v14i2.3391.
- [10] R. Asrianto and M. Herwinanda, “Analisis Sentimen Kenaikan Harga Kebutuhan Pokok Dimedia Sosial Youtube Menggunakan Algoritma Support Vector Machine,” *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 3, no. 3, pp. 431–440, 2022, doi: 10.37859/coscitech.v3i3.4368.
- [11] C. A. Misrun, E. Haerani, M. Fikry, and E. Budianita, “Analisis Sentimen Komentar Youtube Terhadap Anies Baswedan Sebagai Bakal Calon Presiden 2024 Menggunakan Metode Naive Bayes Classifier,” *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 4, no. 1, pp. 207–215, 2023, doi: 10.37859/coscitech.v4i1.4790.