

# Analisis Sentimen Masyarakat Terhadap Mainan Anak “Lato-Lato” Menggunakan Metode Support Vector Machine (Svm) Pada Media Sosial Youtube

Ahmad Roshid<sup>1)</sup>, Erizal, S.Kom., M.Kom.<sup>2)</sup>, & Rahmi Imada, S.Kom., M.Kom.<sup>3)</sup>

<sup>1,2,3)</sup>Sistem Teknologi Informasi, Fakultas Teknologi Industri dan Informatika, Universitas Muhammadiyah Prof. Dr. Hamka  
Jl. Tanah Merdeka No.6, Kec. Pasar Rebo, DKI Jakarta 13830 Indonesia Telp : (021)87782739  
Website uhamka.ac.id, E-mail: <sup>1,2,3)</sup> [ahmadzeed.mulmed@gmail.com](mailto:ahmadzeed.mulmed@gmail.com), [erizal@uhamka.ac.id](mailto:erizal@uhamka.ac.id), [rahmi.imanda@uhamka.ac.id](mailto:rahmi.imanda@uhamka.ac.id)

## Abstrak

*Lato-lato adalah mainan anak yang sempat viral di Indonesia. Mainan anak ini cukup menuai kontroversi di masyarakat Indonesia karena mainan ini menciptakan suara yang cukup bising ketika dimainkan tetapi juga memiliki beberapa dampak yang positif bagi anak. Tujuan dari penelitian ini adalah untuk menganalisa sentiment masyarakat Indonesia terhadap mainan lato-lato dari komentar video di sosial media YouTube. Proses yang dilakukan adalah crawling data, preprocessing, normalisasi, pelabelan, dan klasifikasi SVM. Setelah dilakukan proses klasifikasi SVM dengan kernel linear, didapat akurasi sebesar 81%. Untuk sentiment positif didapat precision sebesar 80%, recall 85% dan f1-score 82%, untuk sentiment negatif didapat precision sebesar 83%, recall 77% dan f1-score sebesar 80%. Dari hasil analisis sentiment tersebut, lebih banyak masyarakat Indonesia yang menanggapi mainan lato-lato secara positif yang menandakan bahwa masyarakat Indonesia melihat mainan lato-lato ini bukanlah sebuah masalah dan banyak manfaat positif yang didapat dari mainan lato-lato.*

**Keyword:** analisis sentimen, support vector machine, youtube, lato-lato

## Abstract

*Lato-lato is a children's toy that has gone viral in Indonesia. This children's toy is quite controversial in Indonesian society because this toy creates quite a noise when played but also has several positive impacts for children. The purpose of this research is to analyze the sentiment of Indonesian people towards lato-lato toys from video comments on YouTube social media. The processes carried out are data crawling, preprocessing, normalization, labeling, and SVM classification. After the SVM classification process with a linear kernel, an accuracy of 81% was obtained. For positive sentiment, the precision is 80%, recall 85% and f1-score 82%, for negative sentiment, the precision is 83%, recall 77% and f1-score 80%. From the results of the sentiment analysis, more Indonesians respond positively to lato-lato, which indicates that Indonesians see lato-lato as not a problem and there are many positive benefits from lato-lato.*

**Kata kunci:** sentiment analisist, support vector machine, YouTube, clackers

## 1. PENDAHULUAN

Pada era digital seperti sekarang ini, penggunaan sosial media semakin meningkat. Menurut Data Indonesia, pada tahun 2022 jumlah pengguna sosial media youtube di Indonesia sebanyak 127 juta jiwa pengguna atau sebesar 46,2% dari total penduduk Indonesia [1].

Jenis video yang dapat di sajikan oleh youtube antara lain pengetahuan, *tutorial*, hiburan, dan lainnya. Masyarakat atau pengguna youtube dapat menyukai, membagikan serta dapat berkomentar pendapat pribadinya [2].

Lato-lato adalah salah satu bentuk permainan tradisional yang dapat ditemukan di Indonesia yang menjadi ikonik sejak era 1990-an, terutama di kalangan penduduk (Aditya, 2022). Lato-lato merupakan permainan yang berasal dari Amerika Serikat. Di negara asalnya, lato-lato dikenal dengan berbagai nama

seperti "clackers", "click-clacks", "knockers", "ker-bangers", atau "clankers"[3].

Masalah dari penelitian ini adalah bagaimana masyarakat menilai mainan anak lato-lato, apakah hasilnya lebih banyak sentimen positif atau sentimen negatif, serta bagaimana proses klasifikasi SVM dengan kernel linear untuk mendapatkan hasil akurasi, presisi, *recall*, dan *F1-Score* untuk sentimen positif dan negatif.

Tujuan dari penelitian ini adalah untuk mendapatkan hasil olah data sentimen positif dan negatif dalam proses pelabelan, serta mendapatkan nilai akurasi dan evaluasi dari SVM.

SVM adalah metode *supervised learning* dalam klasifikasi yang memproyeksikan kelas berdasarkan model atau pola yang ditemukan selama proses pelatihan. Klasifikasi ini dilakukan dengan mencari

*hyperplane* atau garis batas yang memisahkan satu kelas dari kelas lainnya [4].

Beberapa penelitian tentang analisis sentiment pada media sosial yang sudah dilakukan sebelumnya. Dilakukan analisis sentiment terhadap transportasi online di Indonesia menggunakan metode SVM dengan basis *Particle Swarm Optimization*. Menganalisis emosi atau sentimen dan meningkatkan ketepatan metode dapat dicapai dengan menerapkan metode klasifikasi SVM dan SVM-PSO. Dari hasil 10 *k-fold cross validation*, metode klasifikasi SVM menunjukkan akurasi sebesar 95,46% dengan nilai AUC 0,979 (klasifikasi sangat baik), sementara metode SVM-PSO mencapai akurasi 96,04% dengan AUC 0,993 (klasifikasi sangat baik). Pada penelitian ini Algoritma SVM dengan basis PSO mempunyai nilai akurasi lebih baik dibandingkan algoritma SVM biasa [5].

## 2. LANDASAN TEORI

### A. Analisis Sentimen

Analisis sentimen merupakan pendekatan gabungan antara bahasa, komputasi, dan text mining yang digunakan untuk menganalisa suatu topik atau kegiatan khusus terhadap emosional atau opini [6].

*Opinion mining* atau analisis sentimen merupakan salah satu cabang dalam bidang *data mining* yang dapat melakukan analisis terhadap pendapat, penilaian, sikap, emosi, dan evaluasi terkait dengan entitas seperti produk, topik, layanan, masalah, organisasi, peristiwa, serta atributnya [7].

Melalui analisis sentimen, opini yang mengandung polaritas dapat dianalisis, sehingga informasi yang dihasilkan dapat memiliki nilai positif, negatif, atau netral [8].

### B. Lato-lato

Lato-lato adalah salah satu bentuk permainan tradisional yang dapat ditemukan di Indonesia. Permainan ini telah menjadi ikonik sejak tahun 1990-an, terutama di kalangan penduduk pedesaan [9].

Di Amerika Serikat, lato-lato atau *clackers* mengalami penarikan dari pasar bersamaan dengan laporan insiden cedera pada anak-anak saat bermain. Hal ini disebabkan oleh bobot yang cukup berat dan gerakan cepat dari lato-lato yang terbuat dari plastik akrilik keras. Di Amerika Serikat, lato-lato dikategorikan sebagai "*mechanical hazard*" atau peringatan terhadap mainan yang dapat menimbulkan bahaya [10].

### C. Youtube

YouTube memiliki berbagai tujuan, seperti hiburan, pembelajaran, berbagi aktivitas sehari-hari, dan lain-lain. Pengguna YouTube dapat membuat, menonton, dan berbagi video secara gratis. Selain itu, YouTube menyediakan fitur komentar yang memungkinkan penonton memberikan respon terhadap konten yang mereka tonton, baik dalam bentuk positif maupun negatif. YouTube juga memungkinkan pengguna memberikan *feedback* berupa *like*, *dislike*, dan komentar terhadap video yang diunggah [11].

Maksud dari *text mining* adalah untuk memperoleh informasi berharga dari sekelompok dokumen. Dengan kata lain, data yang dimanfaatkan dalam *text mining* adalah sejumlah teks yang bersifat tidak terstruktur atau setidaknya semi terstruktur [12].

### D. Text Mining

Merupakan pengambilan informasi ketika pengguna berinteraksi dengan sekelompok dokumen menggunakan alat analisis yang terdiri dari komponen-komponen yang ditemukan dalam *data mining* [13].

### E. Data Mining

*Data mining* adalah hasil gabungan beberapa disiplin ilmu komputer. Definisi dari *data mining* adalah proses dari penemuan pola-pola baru dari kumpulan data yang sangat besar. Metode-metode yang digunakan dalam *data mining* mencakup aspek-aspek dari kecerdasan buatan, *machine leaning*, statistika, dan sistem basis data [13].

### F. Support Vector Machine (SVM)

Merupakan suatu metode pembelajaran terbimbing yang menganalisis data dan mengidentifikasi pola yang digunakan untuk keperluan klasifikasi [14].

Metode SVM memiliki keunggulan dalam kemampuannya mengenali *hyperplane* yang dapat memisahkan *margin* dengan maksimal, sehingga dapat memaksimalkan *margin* antara kelas yang berbeda. Namun, metode SVM juga memiliki kelemahan, masalah utamanya adalah di mana fitur-fitur yang sama dapat signifikan memengaruhi tingkat akurasi [8].

Untuk menemukan batas keputusan atau *hyperplane* optimal, langkahnya adalah mencari garis yang melalui titik-titik paling ekstrem dari kelas -1 dan +1 [12]. Berikut adalah rumusnya:

$$(W * X_i) + b \geq 1 \quad (1)$$

Sementara untuk menghitung *hyperplane* adalah dengan rumus berikut :

$$(W * X_i) + b = 0 \tag{2}$$

Keterangan:

$W$  = Bobot dari atribut

$X_i$  = Atribut ke- $i$

$b$  = Bias

*Kernel trick* adalah sebuah formula dari transformasi  $\phi$  yang sangat sulit untuk diketahui serta dipahami. Maka, dengan fungsi kernel  $Kx_i x_j$ , fungsi transformasi  $\phi$  dapat didefinisi [15].

$$K(x_i, x_j) = \phi_j(x_i) * \phi_j(x_j) \tag{3}$$

G. Pengujian Model Evaluasi

Salah satu cara untuk mengetahui apakah hasil dari implementasi algoritma SVM sudah benar dan sesuai dengan algoritmanya, yaitu dengan *Confusion Matrix*. Dengan metode ini, kita dapat mengevaluasi tingkat ketepatan dan hasil penerapan yang dapat dianalisis untuk menentukan sejauh mana klasifikasi berhasil mengidentifikasi *tuple* dari kelas yang berbeda [16]. Berikut ini adalah rumus untuk menghitung *accuracy*, *precision*, *recall*, dan *f1-score*-nya:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4}$$

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$F1 - Score = \frac{2 + Precision + Recall}{Precision + Recall} \tag{7}$$

Keterangan :

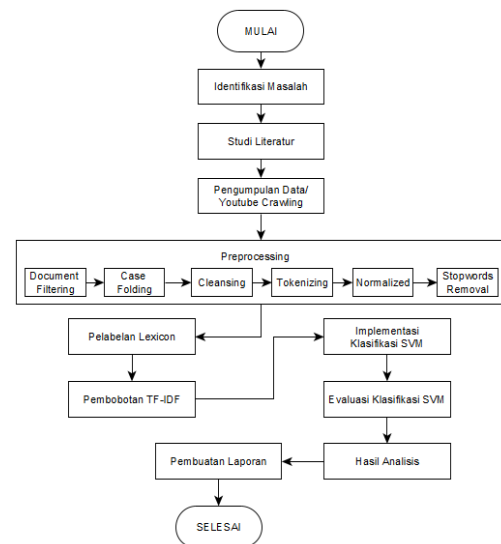
TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

3. METODOLOGI PENELITIAN



Gambar 1 Alur Penelitian

Gambar 1 merupakan tahapan yang dilakukan dalam penelitian ini. Penelitian dimulai dari tahapan identifikasi masalah, dilanjutkan dengan studi literature, pengumpulan data, *preprocessing*, pelabelan, pembobotan, klasifikasi SVM, evaluasi.

A. Identifikasi Masalah

Tahapan awal dalam metodologi penelitian adalah mengenali masalah, dengan tujuan untuk memahami, mengidentifikasi, dan menemukan solusi untuk suatu permasalahan serta mencapai tujuan penelitian [11].

B. Studi Literatur

Langkah literatur adalah bagian dari proses setelah mengidentifikasi masalah. Pada tahap ini, dijelaskan bagaimana data dan informasi dikumpulkan dari berbagai sumber, termasuk jurnal, buku, *e-book*, media massa, dan referensi lain yang berhubungan dengan penelitian, baik dalam bentuk teori maupun penelitian serupa yang telah dilakukan sebelumnya [11].

C. Pengumpulan Data/ Youtube Crawling

Pengambilan data ini menggunakan *library* Youtube Data API v3 dengan meminta API key dari *google developers*. Untuk mendapatkan API key-nya harus melakukan *create credentials* di dalam menu *Enable APIs & Services*. Untuk melakukan crawling data, digunakan *Google Colabs* serta *python* untuk bahasa pemrogramannya [17].

#### D. *Preprocessing*

Pra-pemrosesan (*preprocessing*) merupakan langkah untuk mempersiapkan dataset sehingga memudahkan analisis menggunakan algoritma klasifikasi. *Preprocessing* bertujuan untuk menghilangkan data teks yang tidak relevan selama analisis. Terdapat berbagai metode pra-pemrosesan, dan setiap tahap memiliki tujuan yang spesifik [17].

1. *Document Filtering* merupakan langkah untuk menghapus bagian-bagian dari dokumen mentah yang tidak relevan atau tidak memiliki makna sama sekali dalam konteks proses klasifikasi [18].
2. *Case Folding* merupakan langkah untuk menyamakan gaya penulisan dalam dokumen. Tujuan dari proses ini adalah mempermudah pencarian, karena tidak semua teks konsisten dalam penggunaan huruf besar (kapital) [18].
3. *Cleansing* adalah proses pembersihan hal-hal yang tidak diperlukan, seperti URL, *username*, *emoticon*, *symbol*, dan *hashtag*.
4. *Tokenizing* merupakan langkah pembagian teks yang berasal dari kalimat atau paragraf menjadi bagian-bagian khusus. Langkah ini akan memisahkan setiap kata berdasarkan pemisah token yang ditetapkan, seperti spasi dan tanda baca. Contohnya adalah pada kalimat "aku mau makan nasi goreng" menghasilkan lima token yaitu "aku", "mau", "makan", "nasi", "goreng" [18].
5. Normalisasi bertujuan untuk meningkatkan mutu representasi kata dalam dokumen teks, sehingga analisis dapat dilakukan dengan lebih akurat dan efisien. Langkah-langkah normalisasi dilakukan pada dataset dengan menggunakan file normalisasi yang kemudian diterapkan pada dataset [11].
6. *Stopwords Removal* adalah proses penghapusan kata penghubung yang kurang atau tidak sama sekali memiliki arti. *Stopword*, atau kata penghubung, dapat diartikan sebagai kata yang muncul sangat sering dalam sebuah dokumen teks dan memiliki kontribusi makna yang kurang signifikan terhadap isi dokumen [18].

#### E. Pelabelan *Lexicon Based*

Setiap kata positif yang muncul dalam sebuah komentar akan mendapatkan penambahan skor +1. Di sisi lain, setiap kata negatif yang terdapat dalam komentar akan mengurangi skor sebesar -1. Skor keseluruhan kemudian dihitung dengan menjumlahkan semua skor tersebut. Jika total skor > 0 (lebih dari nol), maka komentar

diklasifikasikan sebagai positif. Sebaliknya, jika total skor < 0 (kurang dari atau sama dengan nol), komentar diklasifikasikan sebagai negatif [19].

#### F. Pembobotan TF-IDF

Pembobotan atau transformasi dengan *Term Frequency* (TF) dan *Inverse Document Filtering* (IDF) atau TF-IDF. TF berfungsi menghitung frekuensi banyaknya kata yang muncul pada data, sementara IDF akan menghitung setiap token pada setiap dokumen dalam korpus. Saat ini TF adalah skema pembobotan yang cukup populer, akurat dan efisien [20]. Berikut adalah rumus menghitung bobot token (kata)  $t$  di dokumen  $d$  :

$$W_{dt} = t_{fdt} * IDF_t \quad (8)$$

Keterangan :

$d$  : dokumen ke- $d$

$t$  : kata ke- $t$  dari kata kunci

$W$  : bobot dokumen ke- $d$  terhadap kata ke- $t$

$tf$  : banyaknya kata yang dicari

IDF: *Inverse Document Frequency*

Berikut adalah rumus untuk mencari nilai IDF:

$$IDF = \log_2 \left( \frac{D}{df} \right) \quad (9)$$

Keterangan :

$D$  : total dokumen

$df$  : banyaknya dokumen yang mengandung token yang dicari

#### G. Implementasi Klasifikasi SVM

SVM adalah teori pembelajaran statistika yang diharapkan dapat menghasilkan kinerja yang lebih baik jika dibandingkan dengan metode alternatif. [21].

Untuk Klasifikasi SVM ini menggunakan *kernel linear* karena berdasarkan riset yang peneliti lakukan terhadap referensi dari penelitian serupa lainnya, *kernel linear* memiliki hasil yang lebih baik daripada kernel RBF, Sigmoid ataupun Polynomial. Berikut adalah rumus SVM dengan *kernel linear* :

$$K(x, x_k) = x_k^T x \quad (10)$$

#### H. Evaluasi Klasifikasi

Penelitian ini memeriksa kinerja model dengan tujuan mengevaluasi seberapa baik model tersebut dalam memprediksi data uji yang tidak dimasukkan dalam proses pembangunan model. Evaluasi ini melibatkan perhitungan nilai akurasi, presisi, *recall*, dan *F1-score* berdasarkan *confusion matrix* [19].

I. Hasil Analisis Data

Adalah hasil dari seluruh penelitian yang sudah dilakukan dari proses awal sampai akhir, dengan hasil analisa berdasarkan pengujian yang telah dilakukan. Serta mendapatkan kesimpulan dari hasil olah data yang telah dilakukan.

4. HASIL DAN PEMBAHASAN

A. Youtube Crawling

Pada proses ini dilakukan pengumpulan atau pengambilan data komentar masyarakat dari empat belas video *YouTube* dengan menggunakan API dari *Google* yaitu *Youtube Data API v3*. Untuk mengambil data tersebut, digunakan *Google Colabs* dan bahasa pemrograman *python*. Dibawah ini adalah gambar dari program *crawling data* dengan API key di samarkan karena APIs *Youtube* ini merupakan *credentials* masing-masing pengguna *Google*.

```
df = pd.DataFrame(comments, columns=['publishedAt', 'authorDisplayName', 'Comment_YT', 'likeCount'])
df
```

	publishedAt	authorDisplayName	Comment_YT	likeCount
0	2023-01-16T15:19:52Z	TRIVA SIMANJUNTAK	Aq juga benci lato2, berisik bgj mungkin or...	1
1	2023-01-17T10:28:58Z	Antin Dami	Online seperti solot lebih parah,	0
2	2023-01-15T13:09:57Z	mohamad pamiuji	Narkoba sehat bagi Amerika	0
3	2023-01-13T11:02:23Z	Amron Amar	Semakin dilarang semakin nafsu main nya... 🤔🤔🤔	0
4	2023-01-13T06:29:56Z	msh phone	ya di larang nyar HP nya ngak lakuuuu cian deh as	0
...	...	...	...	...
110	2023-01-11T03:21:00Z	Anglosaxon Abyss	Katanya Amerika negara Freedom? Takut amat sm ...	8
111	2023-01-11T03:54:22Z	Ahsin Mudawwam	What about gun?	0
112	2023-01-11T03:56:24Z	Some Some	Gangguan suara.  Warga bisa protes dan pang...	4
113	2023-01-11T04:24:46Z	Anglosaxon Abyss	@Some Some jg gw bebas wild parties sini gk ad...	0
114	2023-01-11T04:43:03Z	Some Some	@Anglosaxon Abyss bisa.  Di ruang kedap sua...	0

Gambar 2 Hasil Youtube Crawling

Dari dua belas video *YouTube* yang telah dicari, peneliti mendapatkan sebanyak 2.987 data. Berikut adalah contoh data yang telah di ambil menggunakan *Youtube Crawler*.

B. Preprocessing

Pada proses ini akan dilakukan pembersihan data dari karakter, symbol, angka, serta kata sambung yang tidak digunakan dalam proses pelabelan dan proses klasifikasi. Serta dilakukan proses mengubah kata singkatan atau kata yang tidak baku menjadi kata baku.

1. Document Filtering

Penghapusan data ini dilakukan secara manual, yang sebelumnya ada sebanyak 2.897 data dihapus sebanyak 75 data sehingga menjadi 2.912 data. Berikut adalah contoh data yang dihapus.

Tabel 1 Contoh Data Komentar yang Dihapus

authorDisplayName	Komentar
Eka Aji Saputro	bang bahas enrique maluku bang, sejarah maritim indonesia
Fajar Utomo	Bang bahas diogo alves

2. Case Folding

Merupakan langkah mengubah huruf kapital menjadi huruf kecil pada data, karena ketidak konsistenan penggunaan huruf kapital dan huruf kecil dalam data tersebut, sehingga perlu dilakukan penyamarataan. Berikut contoh data sebelum dan sesudah dilakukan *case folding*.

Tabel 2 Contoh Sebelum dan Sesudah Case Folding

Sebelum dilakukan case folding	setelah dilakukan case folding
iQbal nggak suka sama dengan main lato-latto tambah berisik	iqbal nggak suka sama dengan main lato-latto tambah berisik
Saya tersanjung Dgn dosen saya yg sudah disebutkan sepulang sekolah	saya tersanjung dgn dosen saya yg sudah disebutkan sepulang sekolah
Korbannya bukan hanya manusia, TV pun jadi korban	korbannya bukan hanya manusia, tv pun jadi korban

3. Cleansing

a. Hapus Emotikon untuk menghapus emotikon di dalam sebuah komentar. Proses ini hanya menghapus emotikonnya saja.

Tabel 3 Contoh Sebelum dan Sesudah Hapus Emotikon

Sebelum Dihapus	Setelah Dihapus
semakin dilarang semakin nafsu main nya... 🤔🤔🤔	semakin dilarang semakin nafsu main nya...
mending main lato lato drpada maen game online yg berdampak buruk 😊😊😊	mending main lato lato drpada maen game online yg berdampak buruk
mndgan main lato2 drpd anak2 main hp terus 🙌🙌	mndgan main lato2 drpd anak2 main hp terus
aturan yg menghalangi kreatifitas anak 🧑🏫	aturan yg menghalangi kreatifitas anak

b. Hapus *Hashtag* untuk menghapus *hashtag* atau kampanye di dalam komentar. Proses ini hanya menghapus *hashtag*-nya saja.

Tabel 4 Contoh Hashtag yang Dihapus

Hashtag yang dihapus
#latto2aman
#latto2amankok
#bannedlattelatto
#usuttuntas

c. Hapus URL untuk menghapus tautan yang terdapat di dalam komentar.

Tabel 5 Contoh Sebelum dan Sesudah Hapus URL

Sebelum dihapus	Setelah dihapus
iya main latto2 aj biar kyak anak SD dekat rumah ku matanya pecah <a href="http://www.youtube.com/results?search_query=%23latto2aman">#latto2aman</a>	iya main latto2 aj biar kyak anak SD dekat rumah ku matanya pecah

d. Stopwords Removal untuk menghapus kata hubung yang terdapat di dalam komentar.

Tabel 6 Contoh Sebelum dan Sesudah Stopwords Removal

Sebelum	Setelah
Lim itu kan dulu.saya bahas sekarang yg pakai pelastik...mnding dah anak main lato walaupun berisik ga jadi masalah dari pada mulai PDA main HP jadi pada males belajar,udah gtu takut buka situs yg ngga	lim dulu saya bahas yg pakai pelastik mnding dah anak main lato berisik ga pda main hp males belajar,udah gtu takut buka situs yg ngga
Apaan.. lato lato sumber polusi suara. Soal anak kecanduan gadget itu tanggungjawab ortu untuk ngontrol dan mendisiplinkan anak dalam main gadget. tp caranya bukan dengan main lato-lato, berisik	lato lato sumber polusi suara anak candu gadget tanggungjawab ortu ngontrol disiplin anak main gadget tp main lato-lato berisik

C. Tokenizing

Setelah proses itu dilakukan, kalimat tersebut akan dipisah menjadi per kata dengan pembatas,

(koma) setiap katanya. Berikut adalah contoh tokenizing yang dilakukan.

Tabel 7 Contoh Sebelum dan Sesudah Tokenizing

Sebelum tokenizing	Setelah tokenizing
bahaya main hp anak main hp jam ngerusak mata udah hapal main lato klo lg ngetrend doang mh bosen jg	['bahaya', 'main', 'hp', 'anak', 'main', 'hp', 'jam', 'ngerusak', 'mata', 'udah', 'hapal', 'main', 'lato', 'klo', 'lg', 'ngetrend', 'doang', 'mh', 'bosen', 'jg']
berisik gak manfaat yg ambil gak nambah skill ganggu orang tidur	['berisik', 'gak', 'manfaat', 'yg', 'ambil', 'gak', 'nambah', 'skill', 'ganggu', 'orang', 'tidur']

D. Normalisasi

Ini merupakan proses yang cukup penting, karena pada proses ini data yang sudah dilakukan tokenizing akan diubah menggunakan kata baku. Untuk kamus yang digunakan adalah *colloquial-indonesian-lexicon*, kamus alay, kbba, serta library python kbbsi.

```

1% | 4/453 [00:00<00:14, 30.00it/s]enggak
baru
aduh
juga
main
aduh
saya
waktu
kecil
sekali
saja
lo
dari
2% | 10/453 [00:00<00:23, 19.05it/s]karena
sudah
yang
terima
saya
enggak
3% | 15/453 [00:00<00:25, 17.43it/s]lah
saya
enggak
sama
saja
enggak
baru
baru
memang
baru
    
```

Gambar 3 Proses Normalisasi

Gambar di atas merupakan proses mengubah kata-kata non-baku menjadi kata-kata baku sehingga nanti akan mempermudah dalam proses pembobotan yang dilakukan. Berikut adalah contoh proses normalisasi yang dilakukan, kata yang di warnai merah adalah kata yang tidak baku, untuk kata yang berwarna biru adalah kata yang telah diubah menjadi kata baku.



**Tabel 8** Sebelum dan Sesudah Normalisasi

Sebelum normalisasi	Setelah normalisasi
bahaya main hp anak main hp jam ngerusak mata udah hapal main lato klo lg ngetrend doang mh bosen jg	bahaya main hp anak main hp jam ngerusak mata sudah hapal main lato kalo lagi ngetrend doang mah bosen juga
berisik gak manfaat yg ambil gak nambah skill ganggu orang tidur	berisik enggak manfaat yang ambil enggak bertambah skill ganggu orang tidur

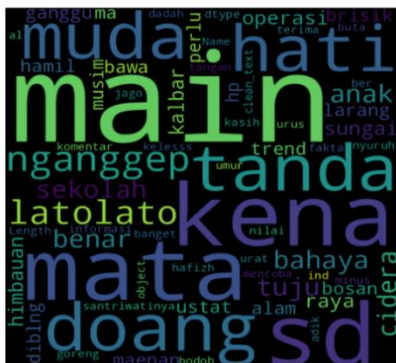
E. Pelabelan *Lexicon*

Proses pemberian label dilakukan secara otomatis dengan memanfaatkan kamus *lexicon-based* yang menggunakan bahasa Indonesia. *Lexicon* merupakan sebuah kamus yang mengevaluasi komentar atau sentimen dengan menghitung kata-kata yang diberi skor polaritas, sehingga dapat menentukan apakah tanggapan atau komentar dari masyarakat bersifat positif atau negatif. Berikut adalah beberapa contoh kata yang ada dalam kamus *lexicon-based*.

**Tabel 9** Contoh Bobot Kamus *Lexicon*

word	weight	number_of_words
menakjubkan	3	1
menolong	4	1
alhamdulillah	5	1
harum	5	1
membocorkan	-4	1
meleleh	-5	1

Langkah yang dilakukan adalah menghitung kata yang paling sering keluar berdasarkan dataset untuk melihat kata-kata yang paling sering muncul di dalam data. Langkah tersebut di visualisasikan dengan *word cloud* seperti gambar di bawah ini.



**Gambar 5** Visualisasi *Word Cloud*

Lalu langkah selanjutnya adalah dengan menghitung nilai sentiment dengan matriks berdasarkan kamus *lexicon* dengan cara menjumlahkan skor dari sentiment tersebut.

```

sencol = []
senrow = np.array([])
nsen = 0
factory = StemmerFactory()
stemmer = factory.create_stemmer()
sentiment_list = []
# function to write the word's sentiment if it is founded
def found_word(ind, words, word, sen, sencol, sentiment, add):
    # if it is already included in the bag of words matrix, then just increase the value
    if word in sencol:
        sen[sencol.index(word)] += 1
    else:
        #if not, than add new word
        sencol.append(word)
        sen.append(1)
        add += 1
    #if there is a negation word before it, the sentiment would be the negation of it's sentiment
    if (words[ind-1] in negasi):
        sentiment += -lexicon['weight'][lexicon_word.index(word)]
    else:
        sentiment += lexicon['weight'][lexicon_word.index(word)]
    return sen, sencol, sentiment, add
    
```

**Gambar 4** Proses Pelabelan

Setelah dilakukan proses perhitungan, maka didapat hasil dari perhitungan sentimennya adalah sebagai berikut.

**Tabel 10** Contoh Hasil Pelabelan

Teks Sentimen	sentiment
bahaya bola lato lato nya leparakan kepala	-5
bagus dibolehin main lato lato pulang sekolah dirumah	7
larang sih berisik banget	-2
dukung main viralkan biar anak remaja perlu tuanya main sibuk mainin gadget	4

Dari hasil pelabelan, didapat data sebanyak 2.297 data yang terdiri dari 1.213 sentimen positif dan 1.066 sentimen negatif.

```

df['label'].value_counts()

positif    1213
negatif    1066
Name: label, dtype: int64
    
```

**Gambar 6** Jumlah Sentimen Positif dan Negatif

Dari hasil tersebut didapat bahwa sentimen positif sebesar 53,2% dan sentimen negatif sebanyak 46,8%. Hasil tersebut menunjukkan bahwa sentimen masyarakat Indonesia terhadap mainan lato-lato lebih banyak yang berkomentar positif.

```
#hitung persentase sentiment
total = 2279
positif = 1213
negatif = 1066

print('persentase positif = ', positif/total * 100, '%')
print('persentase negatif = ', negatif/total * 100, '%')

persentase positif = 53.22509872751206 %
persentase negatif = 46.77490127248794 %
```

**Gambar 7** Persentase Positif dan Negatif

#### F. Pembobotan TF-IDF

Dilakukan perhitungan bobot dengan melakukan pembobotan *term-frequency inverse document frequency* (TF-IDF). Ada tiga proses untuk melakukan pembobotan, yaitu menghitung banyaknya kata yang keluar (*term frequency*), proses *inverse document frequency* dan yang terakhir menghitung bobot dengan mengkali TF dengan IDF.

Proses TF akan menghitung *term* dari banyaknya kata yang keluar. Hasil perhitungannya akan di tokenisasi agar dapat dilakukan proses selanjutnya yaitu perhitungan IDF.

```
#Menghitung bobot
def calc_TF(document):
    # Counts the number of times the word appears in review
    TF_dict = {}
    for term in document:
        if term in TF_dict:
            TF_dict[term] += 1
        else:
            TF_dict[term] = 1
    # Computes tf for each word
    for term in TF_dict:
        TF_dict[term] = TF_dict[term] / len(document)
    return TF_dict

df["TF_dict"] = df['tokenized'].apply(calc_TF)
```

**Gambar 8** Proses Perhitungan TF

Proses Selanjutnya adalah menghitung IDF berdasarkan data yang akan digunakan. Setelah itu baru dilakukan perhitungan TF-IDF dengan mengkali hasil TF dan hasil IDF.

```
# code untuk menghitung IDF
def calc_IDF(n_document, DF):
    IDF_Dict = {}
    for term in DF:
        IDF_Dict[term] = np.log(n_document / (DF[term] + 1))
    return IDF_Dict

# Hitung IDF
IDF = calc_IDF(len(df), DF)

# code untuk menghitung TF-IDF
def calc_TF_IDF(TF):
    TF_IDF_Dict = {}
    for key in TF:
        TF_IDF_Dict[key] = TF[key] * IDF[key]
    return TF_IDF_Dict

# Set perhitungan TF-IDF untuk membuat kolom "TF-IDF_dict"
df["TF-IDF_dict"] = df["TF_dict"].apply(calc_TF_IDF)

results = []

for key, value in df["TF-IDF_dict"].items():
    row = {"term": key, "TF-IDF": value}
    results.append(row)

# buat hasil dalam dataframe
result_df = pd.DataFrame(results)
```

**Gambar 9** Proses Perhitungan IDF

Berikut ini adalah hasil TF-IDF yang telah dilakukan perhitungan:

term	TF	TF-IDF
0	{"luju": 1.0}	{"luju": 4.364196199259211}
1	{"benar": 0.05555555555555555, "bahaya": 0.0}	{"benar": 0.2529687888276521, "bahaya": 0.0}
2	{"maenan": 0.1, "musim": 0.1, "dibling": ...}	{"maenan": 0.4686969591522261, "musim": 0.0}
3	{"tidak": 0.2, "perlu": 0.2, "trend": 0.0}	{"tidak": 0.3123762593508457, "perlu": 1.1}
4	{"tanda": 0.6666666666666666, "alam": 0.33}	{"tanda": 3.3949564664202834, "alam": 1.63}
...	...	...
2274	{"ustat": 0.16666666666666666, "hamil": 0.0}	{"ustat": 1.1730574747809563, "hamil": 0.0}
2275	{"keless": 0.14285714285714285, "kena": 0.0}	{"keless": 1.0054778355265341, "kena": 0.0}
2276	{"terima": 0.3333333333333333, "kasih": 0.0}	{"terima": 1.6327595617298227, "kasih": 1.0}
2277	{"al": 0.027777777777777776, "hafizh": 0.0}	{"al": 0.1649925711116008, "hafizh": 0.176}
2278	{"sd": 0.09090909090909091, "jago": 0.0909}	{"sd": 0.3630747646329378, "jago": 0.52596}

**Gambar 10** Hasil Perhitungan TF-IDF

#### G. Implementasi Klasifikasi SVM

Langkah awal adalah melakukan pemisahan data menjadi dua bagian, yakni data latih dan data uji. Untuk *split* data uji sebanyak 20% dari total data, dan otomatis sisanya sebanyak 80% merupakan *split* untuk data latih.

```
# Menghitung jumlah data yang akan digunakan sebagai data testing sebanyak 20%
test_ratio = 0.2
num_testing_samples = int(test_ratio * len(X))
```

**Gambar 11** Split Data 20% untuk Data Uji

Berikut merupakan hasil dari *split* data yang dilakukan, didapatkan data uji sebanyak 455 data, dan untuk data training sebanyak 1824 data.

```
len(X_test) #jumlah data test yg diuji/test
455

len(X_train) #jumlah data train
1824
```

**Gambar 12** Hasil Split Data

Proses selanjutnya adalah implementasi algoritma yang menggunakan *TF-IDF Vectorizer* dengan data latih dan data uji yang digunakan pada saat proses klasifikasi.

```
#TF-IDF vectorizer
vectorizer = TfidfVectorizer()
X_train_tfidf = vectorizer.fit_transform(X_train)
X_test_tfidf = vectorizer.transform(X_test)
```

**Gambar 13** Proses Vektorisasi TF-IDF

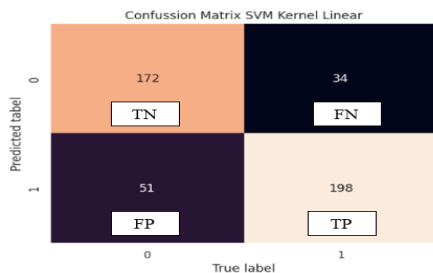


Sebelum proses data uji, dilakukan proses klasifikasi algoritma SVM dengan kernel linear.

```
# Klasifikasi SVM dengan kernel linear
svm = SVC(kernel='linear')
svm.fit(X_train_tfidf, y_train)
```

**Gambar 14** Proses SVM dengan Kernel Linear

Langkah berikutnya adalah melakukan pemrosesan data uji yang telah dibagi dengan menggunakan *confusion matrix* untuk mengetahui label data sebenarnya



**Gambar 15** Hasil Confusion Matrix

Proses terakhir dalam implemtasi algoritma adalah melakukan pengujian algoritmanya. Dari data yang diolah, didapatkan hasil pengujian pada Algoritma *Support Vector Machine* (SVM) dengan kernel *linear* mendapatkan nilai akurasi sebesar 81,3%.

```
accuracy = accuracy_score(y_test, y_pred)
print("Akurasi:", accuracy)

Akurasi: 0.8131868131868132
```

**Gambar 17** Hasil Akurasi SVM

#### H. Pengujian Evaluasi

Tahap penutup dari penelitian ini melibatkan pengujian dan evaluasi terhadap data uji. Pengujian ini penting agar mendapatkan hasil dari *precision*, *recall*, dan *f1-score* dari sentimen positif dan negatif.

Keterangan:

-1 = Negatif

1 = Positif

Pada proses evaluasi, didapatkan untuk sentiment negatif *precision* sebanyak 83%, *recall* 77%, dan *f1-score* sebanyak 80%. Sedangkan untuk sentiment positif didapatkan nilai *precision*

sebanyak 80%, *recall* 85%, dan *f1-score* sebanyak 82%.

	precision	recall	f1-score	support
-1	0.83	0.77	0.80	223
1	0.80	0.85	0.82	232
accuracy			0.81	455
macro avg	0.82	0.81	0.81	455
weighted avg	0.81	0.81	0.81	455

**Gambar 16** Hasil Evaluasi SVM Kernel Linear

## 5. SIMPULAN

Hasil yang didapatkan bahwa, dari total data yang didapatkan dari hasil *crawling youtube* sebanyak 2.279 data terdapat 1.213 data yang bersentimen positif, dan 1.066 data yang bersentimen negatif.

Dari hasil proses yang dihasilkan, nilai akurasi pada pengujian Algoritma *Support Vector Machine* (SVM) dengan menggunakan kernel *linear* mendapat nilai 81,3% dan pada proses pengujian evaluasi menghasilkan sentiment negatif *precision* sebanyak 83%, *recall* 77%, dan *f1-score* sebanyak 80%. Sedangkan untuk sentiment positif didapat nilai *precision* sebanyak 80%, *recall* 85% dan *f1-score* sebanyak 82%.

## KEPUSTAKAAN

- [1] A. Mahmudan, "Pengguna Youtube Indonesia Terbesar Ketiga di Dunia pada 2022," 2022. <https://dataindonesia.id/internet/detail/pengguna-youtube-indonesia-terbesar-ketiga-di-dunia-pada-2022> (accessed Jan. 13, 2023).
- [2] Chely Aulia 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," *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 4, no. 1, pp. 207–215, 2023, doi: 10.37859/coscitech.v4i1.4790.
- [3] L. Zain, "Sejarah Lato-Lato yang Sedang Tren, Bukan dari Indonesia," 2023. <https://www.idntimes.com/science/discovery/1-aili-zain-damaika-1/sejarah-lato-lato?page=all> (accessed May 06, 2023).
- [4] H. Syah and A. Witanti, "Analisis Sentimen Masyarakat Terhadap Vaksinasi Covid-19 Pada Media Sosial Twitter Menggunakan Algoritma Support Vector Machine (Svm)," *J. Sist. Inf. dan Inform.*, vol. 5, no. 1, pp. 59–67, 2022, doi: 10.47080/simika.v5i1.1411.
- [5] V. Kevin, S. Que, A. Iriani, and H. D. Purnomo,

- “Analisis Sentimen Transportasi Online Menggunakan Support Vector Machine Berbasis Particle Swarm Optimization ( Online Transportation Sentiment Analysis Using Support Vector Machine Based on Particle Swarm Optimization ),” vol. 9, no. 2, pp. 162–170, 2020.
- [6] I. Verawati and B. S. Audit, “Algoritma Naïve Bayes Classifier Untuk Analisis Sentiment Pengguna Twitter Terhadap Provider By.u,” *J. Media Inform. Budidarma*, vol. 6, no. 3, p. 1411, 2022, doi: 10.30865/mib.v6i3.4132.
- [7] H. Tuhuteru and A. Iriani, “Analisis Sentimen Perusahaan Listrik Negara Cabang Ambon Menggunakan Metode Support Vector Machine dan Naive Bayes Classifier,” *J. Inform. J. Pengemb. IT*, vol. 3, no. 3, pp. 394–401, 2018, doi: 10.30591/jpit.v3i3.977.
- [8] B. Laurensz and Eko Sedyono, “Analisis Sentimen Masyarakat terhadap Tindakan Vaksinasi dalam Upaya Mengatasi Pandemi Covid-19,” *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 10, no. 2, pp. 118–123, 2021, doi: 10.22146/jnteti.v10i2.1421.
- [9] R. Aditya, “Fakta Lato-lato, Begini Asal-usul hingga Manfaat Mainan Anak Viral di Media Sosial,” 2022. <https://www.suara.com/news/2022/12/24/114007/fakta-lato-lato-begini-asal-usul-hingga-manfaat-mainan-anak-viral-di-media-sosial> (accessed Jan. 14, 2022).
- [10] D. V. Putsanra, “Apa Itu Lato Lato, Berasal dari Mana, dan Siapa Penciptanya?,” 2022. <https://tirto.id/apa-itu-lato-lato-berasal-dari-mana-dan-siapa-penciptanya-gAq7> (accessed Feb. 13, 2023).
- [11] D. Mualfah, Ramadhoni, R. Gunawan, and D. Mulyadipa Suratno, “Analisis Sentimen Komentar YouTube TvOne Tentang Ustadz Abdul Somad Dideportasi Dari Singapura Menggunakan Algoritma SVM,” *J. Fasilkom*, vol. 13, no. 01, pp. 72–80, 2023, doi: 10.37859/jf.v13i01.4920.
- [12] Iin Ernawati, “Naïve Bayes Classifier Dan Support Vector Machine Sebagai Alternatif Solusi Untuk Text Mining,” *JTIP*, vol. 12, no. 2, pp. 1–7, 2019.
- [13] Z. Alhaq, A. Mustopa, S. Mulyatun, and J. D. Santoso, “Penerapan Metode Support Vector Machine Untuk Analisis Sentimen Pengguna Twitter,” *J. Inf. Syst. Manag.*, vol. 3, no. 2, pp. 44–49, 2021, doi: 10.24076/joism.2021v3i2.558.
- [14] E. R. Indriyani, P. Paradise, and M. Wibowo, “Perbandingan Metode Naïve Bayes dan Support Vector Machine Untuk Analisis Sentimen Terhadap Vaksin Astrazeneca di Twitter,” *J. Media Inform. Budidarma*, vol. 6, no. 3, p. 1545, 2022, doi: 10.30865/mib.v6i3.4220.
- [15] F. Romadoni, Y. Umidah, and B. N. Sari, “Text Mining Untuk Analisis Sentimen Pelanggan Terhadap Layanan Uang Elektronik Menggunakan Algoritma Support Vector Machine,” *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 2, pp. 247–253, 2020, doi: 10.32736/sisfokom.v9i2.903.
- [16] M. R. A. Yudianto, A. Rahim, P. Sukmasetya, and R. A. Hasani, “Perbandingan Metode Support Vector Machine Dengan Metode Lexicon Dalam Analisis Sentimen Bahasa Indonesia,” *J. Teknol. Inf.*, vol. 6, no. 1, pp. 7–13, 2022, [Online]. Available: <https://github.com/fajri91/InSet>.
- [17] D. D. Kurnianto and S. Waluyo, “Pajak Diperiksa Kpk Pada Youtube Menggunakan Metode K-Nearest Neighbor Analysis Of Public Sentiment Towards Former Tax Officials Examined By The Kpk On Youtube Using The K- Nearest Neighbor Method,” *Senafiti*, vol. 2, no. September, pp. 632–641, 2023.
- [18] A. N. Ulfah and M. K. Anam, “Analisis Sentimen Hate Speech Pada Portal Berita Online Menggunakan Support Vector Machine (SVM),” *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 7, no. 1, pp. 1–10, 2020, doi: 10.35957/jatisi.v7i1.196.
- [19] F. J. Wahidna and P. Nerisafitra, “Analisis Sentimen Pengguna Sistem Pay Later Menggunakan Support Vector Machine Metode Pembobotan Lexicon,” *J. Informatics Comput. Sci.*, vol. 04, pp. 334–343, 2023, doi: 10.26740/jinacs.v4n03.p334-343.
- [20] W. Bourequat and H. Mourad, “Sentiment Analysis Approach for Analyzing iPhone Release using Support Vector Machine,” *Int. J. Adv. Data Inf. Syst.*, vol. 2, no. 1, pp. 36–44, 2021, doi: 10.25008/ijadis.v2i1.1216.
- [21] D. Darwis, E. S. Pratiwi, and A. F. O. Pasaribu, “Penerapan Algoritma Svm Untuk Analisis Sentimen Pada Data Twitter Komisi Pemberantasan Korupsi Republik Indonesia,” *Edutic - Sci. J. Informatics Educ.*, vol. 7, no. 1, pp. 1–11, 2020, doi: 10.21107/edutic.v7i1.8779.