SKRIPSI

KLASIFIKASI VIDEO *CLICKBAIT* BERDASARKAN *SUBTITLE* KONTEN BERITA YOUTUBE BERBAHASA INDONESIA MENGGUNAKAN METODE SUPPORT VECTOR MACHINE DAN PARTICLE SWARM OPTIMIZATION

Disusun dan diajukan oleh:

NUR ANNISA YUSRAH PUTRA DJAYA
D121 19 1056

PROGRAM STUDI SARJANA TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
GOWA
2024

LEMBAR PENGESAHAN SKRIPSI

KLASIFIKASI VIDEO *CLICKBAIT* BERDASARKAN *SUBTITLE* KONTEN BERITA YOUTUBE BERBAHASA INDONESIA MENGGUNAKAN METODE SUPPORT VECTOR MACHINE DAN PARTICLE SWARM OPTIMIZATION

Disusun dan diajukan oleh

Nur Annisa Yusrah Putra Djaya
D121 19 1056

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka
Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika
Fakultas Teknik Universitas Hasanuddin
Pada tanggal 19 Juni 2024
dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama,	Pembimbing Pendamping,
Dr. Ir. Ingrid Nurtanio, M.T.	Ais Prayogi Alimuddin, S.T., M.Eng.
NIP. 196108131988112001	NIP. 198305102014041001

Ketua Program Studi,

Prof. Dr. Ir. Indrabayu, S.T., M.T., M.Bus.Sys., IPM. ASEAN. Eng.
NIP. 197507162002121004

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama	: Nur Annisa Yusrah Putra Djaya
NIM	: D121191056
Program Studi	: Teknik Informatika
Jenjang	: S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

Klasifikasi Video *Clickbait* Berdasarkan *Subtitle* Konten Berita Youtube Berbahasa Indonesia Menggunakan Metode Support Vector Machine dan Particle Swarm Optimization

Adalah karya tulisan saya sendiri dan bukan merupakan pengambilan alihan tulisan orang lain dan bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Semua informasi yang ditulis dalam skripsi yang berasal dari penulis lain telah diberi penghargaan, yakni dengan mengutip sumber dan tahun penerbitannya. Oleh karena itu, semua tulisan dalam skripsi ini sepenuhnya menjadi tanggung jawab penulis. Apabila ada pihak manapun yang merasa ada kesamaan judul dan atau hasil temuan dalam skripsi ini, maka penulis siap untuk diklarifikasi dan mempertanggungjawabkan segala risiko.

Segala data dan informasi yang diperoleh selama proses pembuatan skripsi, yang akan dipublikasi oleh penulis di masa depan harus mendapat persetujuan dari Dosen Pembimbing.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan isi skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Gowa, 21 Juni 2024

Yang menyatakan,

Nur Annisa Yusrah Putra Djaya

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Allah SWT karena atas berkat dan rahmat-Nya sehingga dapat menyelesaikan tugas akhir dengan judul **"Klasifikasi Video *Clickbait* Berdasarkan *Subtitle* Konten Berita Youtube Berbahasa Indonesia Menggunakan Metode Support Vector Machine dan Particle Swarm Optimization"** sebagai salah satu persyaratan akademik untuk menyelesaikan program Strata-1 (S1) pada Departemen Teknik Informatika, Fakultas Teknik, Universitas Hasanuddin.

Penulis menyadari banyak kesulitan dan kendala yang dihadapi saat penyusunan tugas akhir ini. Dalam prosesnya, penulis memperoleh banyak bantuan, dukungan dan bimbingan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan terima kasih kepada:

1. Allah SWT atas berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan tugas akhir ini.
2. Kedua orang tua penulis, Alm. Bapak Putra Djaya dan Ibu Irmatiah yang senantiasa mendoakan dan memberikan dukungan yang tiada hentinya, serta selalu sabar dalam membesarkan dan mendidik penulis selama ini.
3. Ibu Dr. Ir. Ingrid Nurtanio, M.T. selaku pembimbing I dan Bapak A. Ais Prayogi Alimuddin, S.T., M.Eng. selaku pembimbing II, yang senantiasa menyediakan waktu, tenaga, pikiran dan perhatian yang luar biasa dalam mengarahkan penulis untuk menyelesaikan tugas akhir.
4. Segenap dosen dan staf Departemen Teknik Informatika, Fakultas Teknik, Universitas Hasanuddin yang telah banyak membantu penulis selama perkuliahan.
5. Kak Rahma dan Kak Syamriati yang senantiasa membantu dan mendukung penulis dalam pelabelan dan validasi data untuk penelitian ini.
6. Andi Rusmiati, Pahrul, Deby Rizky Ramadana, Annisa Fitri dan teman-teman S19NIFIER lainnya yang selalu mendengar keluh kesah dan selalu membantu
   t penulis membutuhkan bantuan.
   nan-teman Ambition (Arfandy, Reinhart, Rajab, Echa, Marcel, Fadhil dan
   qa) yang telah menjadi teman yang baik dan menyenangkan bagi penulis

selama masa perkuliahan yang juga senantiasa membantu dan mendengarkan curhatan penulis.

8. Teman-teman Infor B-cek (Agil, Artia, Brillianita, Dita, Sila dan Rahma) yang telah menjadi teman baik dan teman curhat penulis dalam mengeluhkan tugas-tugas selama masa perkuliahan.
9. Ikki dan Fita yang telah menjadi teman *healing* dan teman curhat penulis dalam berkeluh kesah selama menyusun tugas akhir ini.
10. Teman-teman Kars Girl (Ayu, Nand, Kak Tiwi dan Tamara) yang telah menjadi teman yang baik dan selalu mengingatkan penulis untuk menyelesaikan tugas akhir ini sesegera mungkin.
11. Teman-teman HREV (Edo, Vicky, Hendra, Rayyan dan Agres) yang telah menjadi teman yang baik dan membantu serta menghibur penulis selama penyusunan tugas akhir ini.
12. Teman-teman KKN Posko Desa Patani Gel.108 (Mimi, Ayuni, Risda, Alfi, Albani, Yusuf, Will dan Kak Wira) yang telah memberi pengalaman berkesan kepada penulis selama KKN.
13. Serta berbagai pihak atas segala dukungan dan bantuannya yang tidak dapat penulis tuliskan satu persatu.

Penulis berharap semoga Tuhan membalas segala kebaikan yang telah diterima oleh penulis dari berbagai pihak yang telah membantu mempermudah penulis dalam mengerjakan tugas akhir ini. Penulis menyadari bahwa tugas akhir ini masih jauh dari kata sempurna, oleh karena itu penulis mengharapkan segala bentuk saran serta masukan yang membangun dari berbagai pihak. Semoga tugas akhir ini dapat memberikan pengetahuan dan manfaat bagi penulis dan pembaca.

Gowa, 30 Mei 2024

Penulis

ABSTRAK

NUR ANNISA YUSRAH PUTRA DJAYA. *Klasifikasi Video Clickbait Berdasarkan Subtitle Konten Berita Youtube Berbahasa Indonesia Menggunakan Metode Support Vector Machine dan Particle Swarm Optimization* (dibimbing oleh Ingrid Nurtanio dan A. Ais Prayogi Alimuddin).

Youtube merupakan salah satu media sosial yang memiliki pengguna aktif terbanyak di dunia. Indonesia berada pada peringkat ke 4 sebagai pengguna aktif Youtube terbanyak di dunia dengan kisaran pengguna aktif yang mencapai 139 juta pengguna. Banyaknya pengguna aktif Youtube di Indonesia menjadikan banyak masyarakat yang memanfaatkan kondisi ini untuk mendapatkan keuntungan melalui pembuatan konten. Salah satu strategi untuk meraup keuntungan dari konten video Youtube adalah dengan menggunakan judul konten yang menjebak atau judul *clickbait* sehingga dapat meningkatkan jumlah penonton dari konten tersebut. Namun, penggunaan judul *clickbait* juga menimbulkan beberapa dampak negatif yang salah satunya adalah memungkinkan penyebaran informasi hoaks. Maka dari itu, penelitian ini bertujuan mengklasifikasikan berita-berita *clickbait* dan *non-clickbait* berdasarkan kemiripan antara judul dan isi dari video berita tersebut.

Dalam penelitian ini, tahap *data preprocessing* akan dibagi menjadi tiga skenario yakni *data preprocessing* menggunakan *stopword removal* pada judul dan isi berita, *data preprocessing* menggunakan *stopword removal* hanya pada isi berita dan *data preprocessing* tanpa menggunakan *stopword removal* pada judul dan isi berita.

Selain itu, tahap pengklasifikasian berita *clickbait* dan *non-clickbait* dalam penelitian ini dibagi menjadi tiga skenario yakni klasifikasi berita menggunakan Cosine Similarity, klasifikasi berita menggunakan Support Vector Machine (SVM) dan klasifikasi berita menggunakan Support Vector Machine (SVM) yang kemudian dioptimasi dengan Particle Swarm Optimization (PSO).

Hasil penelitian ini menunjukkan bahwa model SVM yang telah dioptimasi dengan PSO menggunakan data yang melewati tahap *data preprocessing* tanpa menggunakan *stopword removal* pada judul dan isi berita, menghasilkan akurasi tertinggi dengan perolehan akurasi mencapai 0,75 atau 75%.

Kata kunci: Berita, Youtube, *Clickbait*, Cosine Similarity, Support Vector Machine, Particle Swarm Optimization

ABSTRACT

NUR ANNISA YUSRAH PUTRA DJAYA. *Classification of Clickbait Videos Based on Subtitles of Indonesian Youtube News Content Using Support Vector Machine and Particle Swarm Optimization Methods* (supervised Ingrid Nurtanio and A. Ais Prayogi Alimuddin).

YouTube is one of the social media that has the most active users in the world. Indonesia is ranked 4th as the most active YouTube users in the world with a range of active users reaching 139 million users. The large number of active YouTube users in Indonesia means that many people take advantage of this condition to gain profits through content creation. One strategy for reaping profits from YouTube video content is to use misleading content titles or clickbait titles so that you can increase the number of viewers of the content. However, the use of clickbait titles also has several negative impacts, one of which is that it allows the spread of hoax information. Therefore, this research aims to classify clickbait and non-clickbait news based on the similarity between the title and content of the news video.

In this research, the data preprocessing stage will be divided into three scenarios, namely data preprocessing using stopword removal on the title and content of the news, data preprocessing using stopword removal only on the content of the news and data preprocessing without using stopword removal on the title and content of the news.

Apart from that, the stage of classifying clickbait and non-clickbait news in this research is divided into three scenarios, namely news classification using Cosine Similarity, news classification using Support Vector Machine (SVM) and news classification using Support Vector Machine (SVM) which is then optimized with Particle Swarm Optimization (PSO).

The results of this research show that the SVM model that has been optimized with PSO using data that has passed the data preprocessing stage without using stopword removal in the title and content of the news, produces the highest accuracy with accuracy reaching 0.75 or 75%.

Keywords: News, Youtube, Clickbait, Cosine Similarity, Support Vector Machine, Particle Swarm Optimization

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR LAMPIRAN

DAFTAR SINGKATAN DAN ARTI SIMBOL

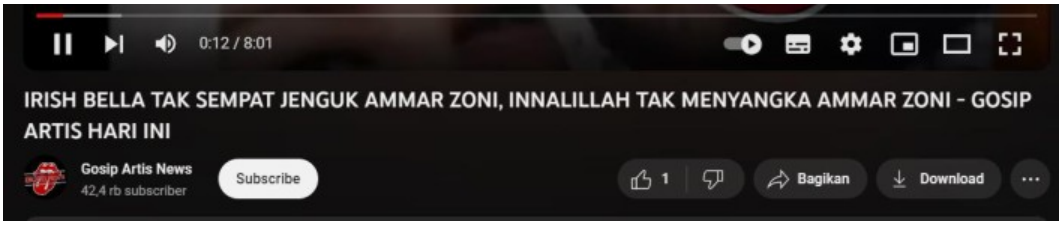
Lambang/Singkatan	Arti dan Keterangan
SVM	*Support Vector Machine*
PSO	*Particle Swarm Optimization*
RBF	*Radial Basis Function*
TF	*Term Frequency*
DF	*Document Frequency*
IDF	*Invers Document Frequency*
TF-IDF	*Term Frequency-Inverse Document Frequency*
TP	*True Positive*
TN	*True Negative*
FP	*False Positive*
FN	*False Negative*
Typo	*Typographical Error*
CC	*Closed Caption*
URL	*Uniform Resource Locator*
SUM	Fungsi Penjumlahan
η	*Learning Rate*
λ	*Lambda*

BAB I
PENDAHULUAN

1.1 Latar Belakang

Internet telah menawarkan berbagai kemudahan bagi manusia, salah satunya adalah kemudahan dalam mengakses media sosial. Kemudahan dalam mengakses media sosial terlihat dari banyaknya pengguna aktif media sosial di Indonesia. Youtube menjadi salah satu *platform* media sosial yang memiliki pengguna aktif terbanyak di Indonesia yakni mencapai sekitar 139 juta pengguna yang menjadikan Indonesia sebagai negara keempat dengan pengguna aktif Youtube terbanyak di dunia (Data Reportal, 2023). Tak heran, jika saat ini banyak masyarakat Indonesia yang memanfaatkan kesempatan ini untuk meraih keuntungan melalui pembuatan konten di Youtube. Hal ini terlihat dari merebaknya berbagai macam konten telah tersedia di Youtube. Mulai dari konten pendidikan, kecantikan, olahraga, musik, makanan, berita dan berbagai variasi konten lainnya. Namun, tidak sedikit juga pembuat konten yang menggunakan strategi *clickbait* untuk meningkatkan jumlah penonton guna meraup keuntungan yang lebih besar (Hadiyat, 2019).

Clickbait merupakan judul konten yang menjebak dan umumnya digunakan oleh pembuat konten untuk menarik pengunjung agar mengunjungi konten tersebut (Pramesti, 2020). Youtube menjadi salah satu *platform* yang banyak dimanfaatkan oleh pembuat konten untuk membuat konten yang mengandung *clickbait*.



Gambar 1. Konten Berita *Clickbait* Pada Salah Satu Kanal Youtube Nasional (Gosip Artis News, 2023)

Konten yang mengandung *clickbait* berpotensi dalam menimbulkan berbagai dampak negatif terutama bagi para pengunjung konten tersebut. Adapun dampak pat ditimbulkan dari konten *clickbait* yang pertama adalah tidak ıskan, hal ini akan sangat berpengaruh terhadap konten-konten berita jika judul beritanya tidak disampaikan secara utuh. Kedua, konten yang

mengandung *clickbait* dapat memicu terjadinya penyebaran berita hoaks dan akumulasi hoaks dalam masyarakat. Ketiga, efek jauh yang dapat ditimbulkan adalah memicu terjadinya keresahan dalam masyarakat yang berpotensi dalam menimbulkan disintegrasi horizontal di masyarakat (Hadiyat, 2019).

Padahal, berdasarkan hasil survei yang dilakukan oleh Reuters Institute pada tahun 2022, menunjukkan bahwa sebanyak 46% masyarakat Indonesia menggunakan Youtube sebagai sumber berita (Newman et al., 2022). Sebagai bentuk antisipasi terhadap menjamurnya konten berita yang mengandung *clickbait*, berbagai penelitian pun telah dilakukan. Salah satunya adalah penelitian yang dilakukan oleh Wira Satya Pratama Biantong pada tahun 2019 yang telah mengusulkan suatu sistem untuk mendeteksi *clickbait* pada artikel berita nasional dengan menggunakan metode Cosine similarity untuk mencari kesamaan antara judul dan isi artikel berita yang kemudian diklasifikasikan menjadi berita *clickbait* dan *non-clickbait*. Hasilnya, akurasi yang didapatkan mencapai 78% pada ambang batas (*threshold*) 0,4.

Maka dari itu, penelitian ini bertujuan untuk mengklasifikasikan konten berita *clickbait* berbahasa Indonesia di Youtube menggunakan metode Cosine Similarity dan Support Vector Machine (SVM) yang kemudian akan dioptimasi menggunakan Particle Swarm Optimization (PSO) guna mendapatkan akurasi yang lebih optimal.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dijelaskan di atas maka, rumusan masalah dalam penelitian ini adalah sebagai berikut:

a. Bagaimana implementasi algoritma SVM dan PSO dalam mengklasifikasikan konten berita *clickbait* dan *non-clickbait* di Youtube?
b. Bagaimana hasil akurasi dari algoritma yang digunakan untuk mengklasifikasikan konten berita *clickbait* dan *non-clickbait* di Youtube?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah sebagai berikut:

a. Mengetahui implementasi algoritma SVM dan PSO dalam mengklasifikasikan konten berita *clickbait* dan *non-clickbait* di Youtube.
b. Mengetahui hasil akurasi dari algoritma yang digunakan untuk mengklasifikasikan konten berita *clickbait* dan *non-clickbait* di Youtube.

1.4 Manfaat Penelitian

Melalui penelitian ini, diharapkan dapat membantu:

a. Bagi peneliti, dapat digunakan untuk menambah wawasan serta mengimplementasikan algoritma SVM dan PSO dalam mengklasifikasikan konten berita *clickbait* dan *non-clickbait* pada *platform* media *online* lainnya.
b. Bagi perusahaan media *online*, dapat digunakan sebagai acuan dalam peningkatan kualitas penulisan konten berita yang relevan berdasarkan hasil yang diperoleh.

1.5 Ruang Lingkup

a. Konten berita bersumber dari media sosial Youtube.
b. Konten berita yang digunakan memiliki *subtitle.*
c. Data yang digunakan terdiri dari judul dan isi konten yang diambil dari *subtitle* video konten berita.
d. Kategori konten berita yang digunakan adalah kategori politik, kriminal, *entertainment,* bencana alam dan kecelakaan.
e. Data bersumber dari beberapa kanal berita nasional di Youtube yang diantaranya KompasTV, TvOneNews, Tribunnews dan Gosip Artis News.
f. Maksimal durasi video yang digunakan berdurasi 8 menit 15 detik.

BAB II
TINJAUAN PUSTAKA

2.1 Youtube

Youtube didirikan pertama kali pada bulan Februari tahun 2005 oleh tiga orang mantan karyawan *PayPal* yakni Chad Hurley, Steve Chen dan Jawed Karim. Nama Youtube sendiri terinspirasi dari nama sebuah kedai pizza dan restoran Jepang di San Mateo, California (Chandra, 2017). Youtube merupakan salah satu media sosial yang cukup banyak dinikmati oleh berbagai kalangan. Youtube menyediakan berbagai macam informasi yang disajikan dalam bentuk video. Video yang tersedia pada Youtube sendiri sangatlah bervariasi, mulai dari pendidikan, politik, hiburan, bisnis, olahraga dan masih banyak lagi. Bervariasinya video di Youtube juga sendiri juga dipengaruhi oleh maraknya para pembuat konten atau yang lebih dikenal dengan istilah *content creator* yang telah berpartisipasi dalam membuat video yang akan dinikmati oleh para pengguna Youtube. Sehingga tak heran, jika saat ini Youtube menjadi salah satu media sosial yang paling populer dan digemari di dunia.

bar 2. *Website* yang Paling Banyak Dikunjungi oleh Masyarakat Indonesia (Kemp, 2024)

kan gambar 2, diketahui bahwa Youtube berada di peringkat kedua sebagai sial yang paling diminati oleh masyarakat Indonesia. Adapun jumlah

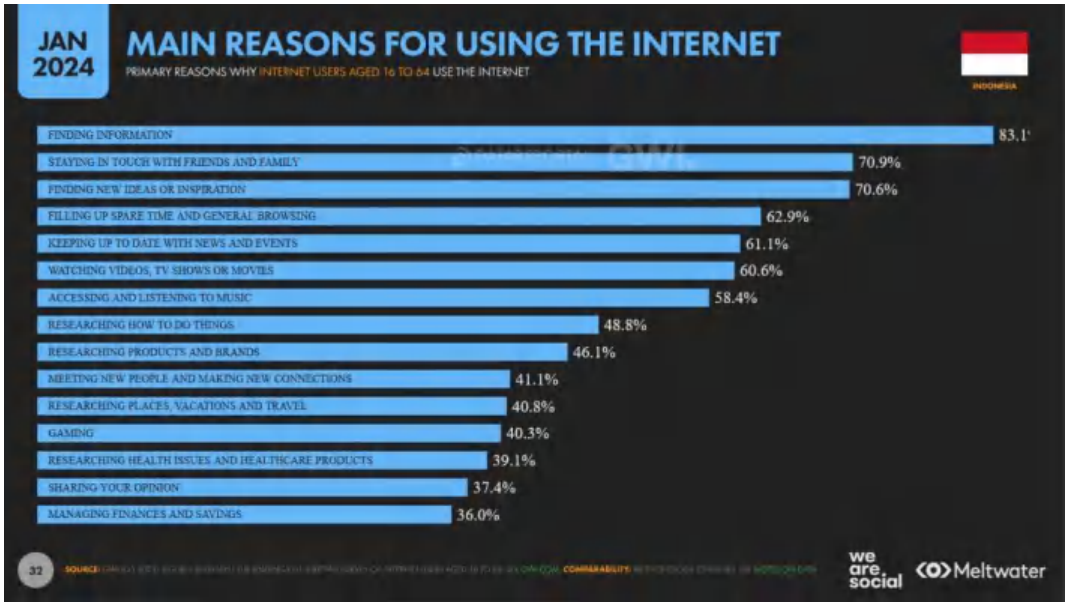
kunjungan Youtube oleh masyarakat Indonesia mencapai 814 juta kunjungan setiap bulannya.

Banyaknya pengguna Youtube juga dipengaruhi oleh adanya kesempatan yang diberikan kepada para penggunanya untuk berkolaborasi dengan membuat video sehingga mereka dapat memperoleh penghasilan melalui penanyangan video yang telah mereka buat (David et al., 2017). Karya yang diunggah ke Youtube berpotensi dalam menghasilkan keuntungan hingga jutaan bahkan miliaran rupiah. Perolehan keuntungan ini didapatkan melalui pemasangan iklan pada setiap video ataupun jumlah *subscriber* yang dimiliki sebagai pembuatan konten (*content creator*) (Wiryany & Pratami, 2019).

Selain itu, beberapa alasan utama yang membuat Youtube menjadi salah satu media sosial yang paling digemari adalah tidak adanya batasan durasi untuk mengunggah video, sistem keamanan yang akurat melalui pembatasan pengamanan dengan tidak mengizinkan video yang mengandung sara dan *illegal*, adanya honorarium (bayaran) pada video yang mencapai 1.000 *views*, serta adanya fitur editor yang sederhana sehingga mudah untuk digunakan oleh kalangan awam sekalipun (Faiqah et al., 2016)

2.2 Berita

Berita merupakan laporan yang memuat informasi suatu peristiwa yang telah atau sedang terjadi. Tujuan dari adanya berita sendiri adalah sebagai wadah bagi masyarakat untuk memperoleh informasi dari suatu peristiwa yang bersifat aktual. Umumnya, berita dapat tersampaikan dari mulut ke mulut, melalui media cetak, siaran, hingga melalui internet yang marak seperti sekarang ini. Melalui penggunaan internet, masyarakat mampu memperoleh berbagai informasi dengan mudah dan sangat cepat. Media sosial menjadi salah satu wadah yang marak digunakan masyarakat untuk memperoleh informasi atau berita terkini. Melalui media sosial, informasi atau berita dapat tersampaikan dengan mudah dan cepat.



Gambar 3. Beberapa Alasan Utama Masyarakat Indonesia Menggunakan Internet (Kemp, 2024)

Berdasarkan gambar 3, dapat diketahui bahwa sebanyak 83,1% masyarakat Indonesia menggunakan internet untuk menemukan informasi dan 61,1% diantaranya menggunakan internet untuk mengakses berita-berita terkini. Hal ini menunjukkan bahwa lebih dari setengah populasi masyarakat Indonesia saat ini mengakses informasi dan berita menggunakan internet. Tentunya ini menjadi sebuah tuntutan bagi para produsen konten berita untuk mampu beradaptasi terhadap dinamika penyebaran berita dan informasi melalui *platform* di media sosial (Kertanegara, 2018). Selain lebih mudah dan cepat untuk diakses, penyebaran berita secara *online* juga memiliki kelebihan secara multimedia. Portal berita *online* dapat memuat berbagai informasi dalam bentuk teks, audio, video dan foto secara bersamaan. Selain itu, melalui internet, pembaruan informasi dapat dilakukan dengan cepat sehingga memungkinkan masyarakat dapat memperoleh informasi dimana saja dan kapan saja. Melalui portal berita *online*, memungkinkan terjadinya interaksi antara produsen konten berita dan pengunjung situs berita melalui kolom komentar (Kencana et al., 2022)

Salah satu jenis berita yang banyak ditemui adalah *straight news* atau berita

. Berita langsung (*straight news*) merupakan salah satu dari empat jenis
ng umumnya ditulis secara langsung kepada inti berita (*to the point*),
an tidak berbelit-belit. Jenis berita ini biasanya memuat informasi yang

aktual dan menarik bagi masyarakat. Berita langsung (*straight news*) sendiri terdiri dari 2 macam yakni berita keras (*hard news*) dan berita lunak (*soft news*). Adapun perbedaan dari berita keras (*hard news*) dan berita lunak (*soft news*) adalah sebagai berikut:

Tabel 1. Perbedaan Berita Keras (*Hard News*) dan Berita Lunak (*Soft News*)

Berita keras (*hard news*)	**Berita lunak (*soft news*)**
Harus ada peristiwa terlebih dahulu.	Tidak perlu ada peristiwa terlebih dahulu.
Berita bersifat aktual (baru terjadi).	Berita tidak selalu bersifat aktual.
Mengutamakan informasi yang terpenting saja.	Menekankan pada detail informasi.
Tidak menekankan sisi ketertarikan manusia (*human interest*).	Sangat menekankan sisi ketertarikan manusia (*human interest*).
Contoh berita: berita perang, politik, kriminalitas dan ekonomi negara.	Contoh berita: berita seni, hiburan, gaya hidup, gosip dan fiksi.

Sumber: (Morissan, 2010)

2.3 Umpan Klik (*Clickbait*)

Umpan klik atau yang lebih dikenal dengan istilah *clickbait* merupakan salah satu kiat untuk menarik perhatian pengunjung untuk menekan atau mengeklik suatu konten. Umumnya, *clickbait* digunakan dengan membuat judul atau tampilan konten yang melebih-lebihkan dan cenderung terlihat kontroversial. Selain untuk meningkatkan pengunjung, penggunaan judul *clickbait* juga bertujuan untuk meningkatkan *traffic* dan *pageviews* sehingga menghasilkan keuntungan yang lebih besar (Habibie, 2018). Adapun ciri-ciri dari konten yang mengandung *clickbait* adalah sebagai berikut:

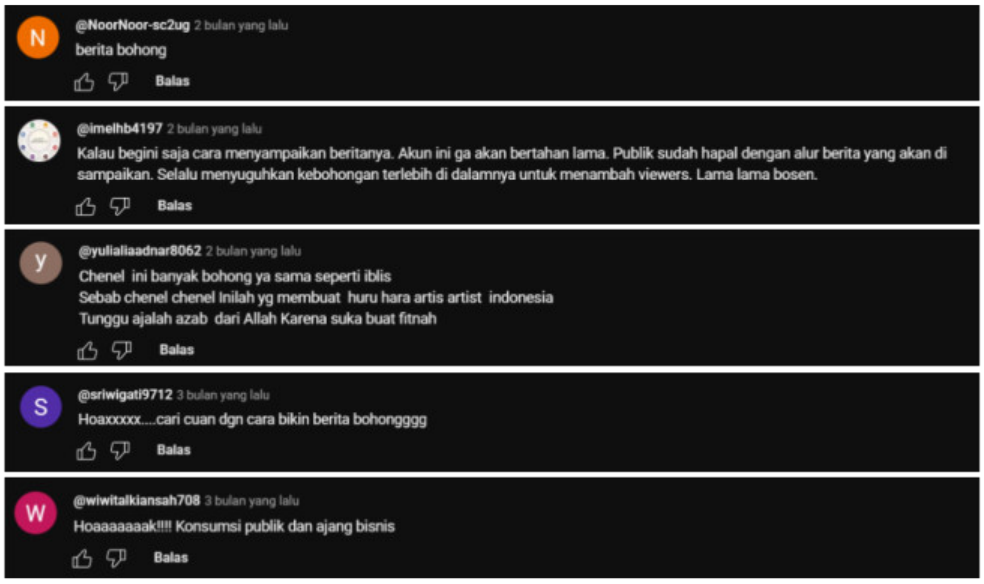
Tabel 2. Ciri-ciri Konten yang Mengandung *Clickbait*

Kategorisasi	**Ciri-ciri konten *clickbait***
Panjang judul (*headline*)	Panjang judul rata-rata 10 kata
ıaan kata	Menggunakan kata-kata yang mencolok dan sensasional
ıaan bahasa	Menggunakan bahasa tidak resmi (*slang*)

Kategorisasi	Ciri-ciri konten *clickbait*
Pola tanda baca dalam judul	Menggunakan tanda baca yang bersifat tidak formal seperti !?, …, ***, !!!
Kata penghubung dalam judul	Menggunakan kata penghubung dalam judul, seperti “dan”, “maupun”, “bila”, “hingga”, “ketika”, “karena” dan beberapa kata penghubung lainnya.
Topik dalam judul berita	Topik dalam satu judul berita bisa berbeda dengan isi berita tersebut
Penekanan angka di awal judul	Terdapat angka di awal judul yang membuat penikmat berita akan merasa lebih ingin tahu
Judul bersifat narasi	Judul menceritakan sesuatu dengan deskripsi yang panjang

Sumber: (Chakraborty et al., 2016)

Walaupun penggunaan judul konten *clickbait* ini menguntungkan bagi sebagian pembuat konten atau *content creator*, tentunya tidak demikian dengan para pengunjung dari konten tersebut.

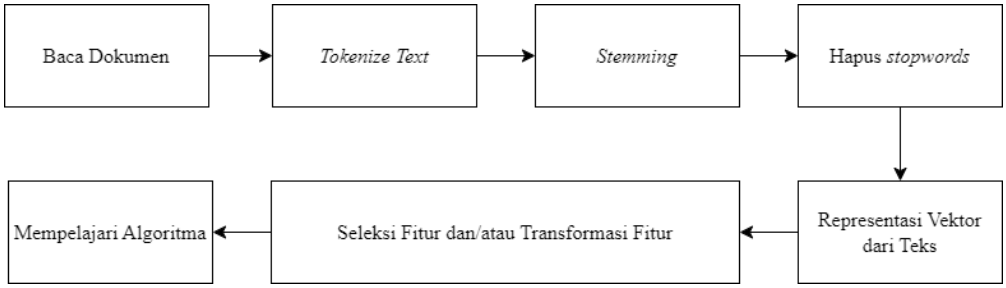


Gambar 4. Hasil Survei Komentar Penonton Berita yang Mengandung *Clickbait* pada Salah Satu Kanal Youtube

Penggunaan judul yang mengandung *clickbait* tentunya akan berdampak pada ng konten tersebut. Hal ini terlihat dari komentar-komentar pengunjung ı kanal berita di Youtube yang merasa telah dirugikan dengan adanya suaian antar judul dan isi dari konten tersebut.

2.4 Klasifikasi Teks (*Text Classification*)

Klasifikasi teks merupakan salah satu metode pengelompokkan data teks ke dalam kelas-kelas tertentu. Umumnya, proses ini dilakukan dengan mengelompokkan data teks sesuai dengan karakteristiknya ke dalam satu atau lebih kelas yang telah didefinisikan sebelumnya (Bagaskoro et al., 2018). Adapun proses dari klasifikasi teks dapat direpresentasikan seperti berikut:



Gambar 5. Proses Klasifikasi Teks (*Text Classification*)
(Ikonomakis et al., 2005)

Berdasarkan gambar 3, dapat diketahu bahwa dalam proses klasifikasi teks umumnya terdiri dari tiga komponen yaitu praproses data (*data preprocessing*), konstruksi pengklasifikasi dan pengkategorian dokumen. Tahap praproses data ini terdiri dari *case folding, tokenizing, filtering,* dan *stemming*. Terdapat juga beberapa tahap lain seperti pembentukan kamus, pemilihan fitur (*feature selection*) dan pembobotan. Untuk tahap konstruksi atau pembentukan pengklasifikasi merupakan tahap pembentukan model klasifikasi melalui proses pembelajaran terhadap data latih (*training data*). Sedangkan untuk tahap pengkategorian dokumen dapat didefinisikan sebagai proses pengujian (*testing*) dari data uji (*testing data*) atau data yang akan ditentukan kategorinya menggunakan model klasifikasi yang telah dibentuk pada tahap sebelumnya (Ridok & Latifah, 2015).

2.5 Pra-proses Teks (*Text Preprocessing*)

Pra-proses Teks atau *text preprocessing* merupakan tahapan pemrosesan data teks yang diolah sedemikian rupa untuk menghilangkan data yang tidak relevan ...gunakan pada proses selanjutnya. Tujuan dari tahapan ini adalah untuk ...h bentuk data menjadi bentuk yang lebih mudah untuk diproses oleh ...Syakuro, 2017). Terdapat beberapa tahapan *text preprocessing* yakni ...erikut:

2.5.1 *Case Folding*

Tahap ini bertujuan untuk mengubah seluruh huruf kapital (*upper case*) menjadi huruf kecil (*lower case*) yang terdapat dalam data yang telah diperoleh sebelumnya. Pada tahap ini, huruf kapital seperti 'A'-'Z' akan diubah menjadi 'a'-'z'. Berikut contoh penerapan *case folding* yang dapat dilihat pada Tabel 3.

Tabel 3. Contoh Penerapan *Case Folding*

Sebelum *Case Folding*	**Setelah *Case Folding***
Kejam! Jenazah Lansia Korban Tabrak Lari di Kulon Progo Ditolak Keluarganya, Ogah Urusi Pemakaman	**k**ejam! **j**enazah **l**ansia **k**orban **t**abrak **l**ari di **k**ulon **p**rogo **d**itolak **k**eluarganya, **o**gah **u**rusi **p**emakaman

2.5.2 *Tokenizing*

Tahap ini merupakan tahap pemotongan *string input* dengan tujuan untuk memecah kalimat menjadi kata, frasa dan entitas penting lainnya atau disebut sebagai *token* dari sebuah teks. Berikut contoh penerapan *Tokenizing* yang dapat dilihat pada Tabel 4.

Tabel 4. Contoh Penerapan *Tokenizing*

Sebelum *Tokenizing*	**Setelah *Tokenizing***
kejam! jenazah lansia korban tabrak lari di kulon progo ditolak keluarganya, ogah urusi pemakaman	"kejam", "!", "jenazah", "lansia", "korban", "tabrak", "lari", "di", "kulon", "progo", "ditolak", "keluarganya", ",", "ogah", "urusi", "pemakaman"

2.5.3 Normalisasi

Tahap ini bertujuan untuk memproses kata-kata yang singkat atau disingkat, kata-kata tidak baku (*slang*), kata-kata yang disensor dan kata-kata yang mengalami kesalahan penulisan atau dikenal dengan isitlah *typo* (*typographical error*) yang kemudian akan dikonversi menjadi bentuk kata-kata yang lengkap atau kata-kata yang sebenarnya sesuai dengan kamus normalisasi yang telah dibuat. Adapun contoh dari kamus normalisasi yang dimaksud dapat dilihat pada Tabel 5 dan contoh

nnya dapat dilihat pada Tabel 6 sebagai berikut:

Tabel 5. Contoh Kamus Normalisasi Kata

Kata Singkatan, *Slang*, *Typo,* Sensor	**Normalisasi**
lansia	lanjut usia
ngamuk	mengamuk
ususl	usul
cer4i	cerai

Tabel 6. Contoh Penerapan Normalisasi

Sebelum Normalisasi	**Setelah Normalisasi**
“kejam”, “!”, “jenazah”, **“lansia”**, “korban”, “tabrak”, “lari”, “di”, “kulon”, “progo”, “ditolak”, “keluarganya”, “,”, “ogah”, “urusi”, “pemakaman”	“kejam”, “!”, “jenazah”, **“lanjut”, “usia”**, “korban”, “tabrak”, “lari”, “di”, “kulon”, “progo”, “ditolak”, “keluarganya”, “,”, “ogah”, “urusi”, “pemakaman”

2.5.4 *Remove Punctuation*

Tahap ini bertujuan untuk menghapus karakter tanda baca atau simbol yang terdapat dalam *dataset*. Karakter yang akan dihapus seperti (?!,/=+-><;”(){}[].:| dan beberapa karakter lainnya). Hal ini dilakukan untuk menyederhanakan proses *training* nantinya. Berikut contoh penerapan *remove punctuation* yang dapat dilihat pada Tabel 7.

Tabel 7. Contoh Penerapan *Remove Punctuation*

Sebelum *Remove Punctuation*	**Setelah *Remove Punctuation***
“kejam”, **“!”**, “jenazah”, “lanjut”, “usia”, “korban”, “tabrak”, “lari”, “di”, “kulon”, “progo”, “ditolak”, “keluarganya”, **“,”**, “ogah”, “urusi”, “pemakaman”	“kejam”, “jenazah”, “lanjut”, “usia”, “korban”, “tabrak”, “lari”, “di”, “kulon”, “progo”, “ditolak”, “keluarganya”, “ogah”, “urusi”, “pemakaman”

2.5.5 *Stopword Removal*

Tahap selanjutnya adalah penghapusan kata-kata yang akan diabaikan selama an. Tahap ini dilakukan untuk menghapus atau mengurangi kata-kata ng dianggap tidak memiliki makna. Berikut contoh penerapan *Stopword* yang dapat dilihat pada Tabel 8.

Tabel 8. Contoh Penerapan *Stopword Removal*

Sebelum *Stopword Removal*	Setelah *Stopword Removal*
“kejam”, “jenazah”, **“lanjut”**, “usia”, “korban”, “tabrak”, “lari”, **“di”**, “kulon”, “progo”, “ditolak”, “keluarganya”, “ogah”, “urusi”, “pemakaman”	“kejam”, “jenazah”, “usia”, “korban”, “tabrak”, “lari”, “kulon”, “progo”, “ditolak”, “keluarganya”, “ogah”, “urusi”, “pemakaman”

2.5.6 *Stemming*

Tahap ini bertujuan untuk melakukan penghapusan imbuhan dari sebuah kata yang kemudian diubah ke dalam bentuk kata dasarnya. Seperti melakukan stemming pada kata “memakan” yang kemudian akan diubah menjadi “makan”. Berikut contoh penerapan *stemming* yang dapat dilihat pada Tabel 9.

Tabel 9. Contoh Penerapan *Stemming*

Sebelum *Stemming*	Setelah *Stemming*
“kejam”, “jenazah”, “usia”, “korban”, “tabrak”, “lari”, “kulon”, “progo”, **“ditolak”**, **“keluarganya”**, “ogah”, **“urusi”**, **“pemakaman”**	“kejam”, “jenazah”, “usia”, “korban”, “tabrak”, “lari”, “kulon”, “progo”, **“tolak”**, **“keluarga”**, “ogah”, **“urus”**, **“makam”**

2.6 Pembobotan Kata

Tahap selanjutnya adalah memberikan bobot pada setiap kata berdasarkan frekuensi kata dalam dokumen dengan menggunakan *Term Frequency-Inverse Document Frequency* atau yang lebih dikenal dengan istilah TF-IDF. TF-IDF merupakan salah satu tahap pemrosesan data teks yang bertujuan untuk menghitung bobot setiap kata berdasarkan frekuensi kemunculan kata tersebut. Bobot tersebut diperoleh dari hasil perhitungan antara frekuensi kemunculan kata (*term*) di suatu dokumen yang dikenal dengan istilah *term frequency* (TF) dan bobot kebalikan dari banyaknya frekuensi atau jumlah dalam dokumen yang mengandung suatu kata yang lebih dikenal dengan istlilah *invers document frequency* (IDF) (Purnamasari stuti, 2017). Tujuan dari tahap ini adalah untuk mengetahui dan luasi seberapa penting dan seberapa umum sebuah kata di dalam sebuah atau dalam sebuah kalimat. Frekuensi kemunculan kata (TF) di dalam

dokumen menunjukkan seberapa penting kata tersebut pada sebuah dokumen. Sedangkan frekuensi dokumen (DF) yang mengandung kata tersebut menunjukkan seberapa umum kata tersebut digunakan. (Biantong, 2019).

Proses pembobotan kata memerlukan beberapa nilai yakni nilai *term frequency* (TF), *document frequency* (DF), *inverse document frequency* (IDF) dan juga hasil perkalian antara TF dan IDF (Mursianto et al., 2022).

2.6.1 *Term Frequency* (TF)

Term Frequency yang umum disingkat dengan TF merupakan jumlah kemunculan kata atau *term* pada sebuah dokumen. TF sendiri memiliki beberapa jenis (Sierra, 2019) yang diantaranya sebagai berikut:

a. *Binary TF* (TF Biner)

Binary TF atau TF biner melihat apakah suatu kata ada atau tidak dalam sebuah dokumen. Jika ada, TF yang diberikan bernilai satu (1) dan jika tidak ada, TF yang diberikan bernilai nol (0).

b. *Raw TF* (TF Murni)

Raw TF atau TF murni memberikan nilai TF berdasarkan jumlah kemunculan kata pada sebuah dokumen. Sehingga jika kata muncul sebanyak lima (5) kali pada sebuah dokumen, maka nilai TF yang diberikan bernilai lima (5) pula.

c. *Normalized TF* (TF Normalisasi)

Normalized TF atau TF normalisasi memberikan nilai TF berdasarkan hasil perbandingan antara frekuensi kemunculan kata pada sebuah dokumen dengan kumpulan frekuensi kata yang ada pada sebuah dokumen yang sama.

d. TF Logaritmik

TF logaritmik memberikan nilai TF berdasarkan variasi dari perhitungan TF dalam analisis teks yang menggantikan frekuensi kata dengan logaritma dari frekuensi tersebut. Hal ini dilakukan untuk menghindari dominansi dokumen yang mengandung sedikit kata dalam *query* sehingga dengan menggunakan logaritma semakin tinggi frekuensi kemunculan kata maka ıi TF atau bobotnya akan semakin rendah dan sebaliknya, semakin rendah ɛuensi kemunculan kata maka nilai TF atau bobotnya akan semakin tinggi.

Adapun rumus untuk menghitung TF dapat dituliskan seperti berikut (El Emary & Atwan, 2005):

$$TF_{(t,d)} = \frac{Frekuensi\ kata\ t\ dalam\ dokumen\ d}{Jumlah\ total\ kata\ dalam\ dokumen\ d} \quad (1)$$

Berikut contoh perhitungan *term frequency* (TF) dapat dilihat pada Tabel 10 di bawah ini:

Tabel 10. Contoh Perhitungan *Term Frequency* (TF)

Dokumen (D)	
D1	berita tentang kebakaran hutan di kalimantan
D2	pemerintah provinsi kalimantan barat akan melakukan upaya pemadaman kebakaran hutan kalimantan

Kata (*Term*)	**TF (D1)**	**TF (D2)**
berita	1/6	0
tentang	1/6	0
kebakaran	1/6	1/11
hutan	1/6	1/11
di	1/6	0
kalimantan	1/6	2/11
pemerintah	0	1/11
provinsi	0	1/11
barat	0	1/11
akan	0	1/11
melakukan	0	1/11
upaya	0	1/11
pemadaman	0	1/11

2.6.2 *Invers Document Frequency* (IDF)

Document Frequency (DF) merupakan jumlah dokumen yang mengandung suatu kata. Sedangkan *Invers Document Frequency* (IDF) merupakan bobot kebalikan dari nilai DF yang berarti semakin jarang sebuah kata muncul dalam dokumen, maka semakin besar bobot atau nilai IDFnya sehingga dapat diketahui ıgkat keumuman kata tersebut dalam seluruh dokumen juga akan semakin \dapun rumus untuk menghitung IDF dapat dituliskan seperti berikut & Saputra, 2018):

$$idf_t = log\left(\frac{D}{df_t}\right) + 1 \quad (2)$$

Keterangan:

idf_t = Bobot *inverse* dari nilai DF

df_t = Jumlah dokumen yang mengandung suatu kata (*term*)

D = Jumlah dokumen

Berikut contoh perhitungan nilai DF dapat dilihat pada Tabel 11 di bawah ini:

Tabel 11. Contoh Perhitungan *Document Frequency* (DF)

Dokumen (D)			
D1	berita tentang kebakaran hutan di kalimantan		
D2	pemerintah provinsi kalimantan barat akan melakukan upaya pemadaman kebakaran hutan kalimantan		
Kata (*Term*)	**TF (D1)**	**TF (D2)**	**DF**
berita	1/6	0	1
tentang	1/6	0	1
kebakaran	1/6	1/11	2
hutan	1/6	1/11	2
di	1/6	0	1
kalimantan	1/6	2/11	2
pemerintah	0	1/11	1
provinsi	0	1/11	1
barat	0	1/11	1
akan	0	1/11	1
melakukan	0	1/11	1
upaya	0	1/11	1
pemadaman	0	1/11	1

Setelah mendapatkan nilai DF, tahap selanjutnya adalah melakukan perhitungan untuk mendapatkan nilai IDF. Berikut contoh perhitungan nilai IDF dapat dilihat pada Tabel 12:

Tabel 12. Contoh Perhitungan *Inverse-Document Frequency* (IDF)

	Dokumen (D)			
D1	berita tentang kebakaran hutan di kalimantan			
D2	pemerintah provinsi kalimantan barat akan melakukan upaya pemadaman kebakaran hutan kalimantan			
Kata (*Term*)	**TF (D1)**	**TF (D2)**	**DF**	**IDF (log(D/DF) + 1)**
berita	1/6	0	1	1,301
tentang	1/6	0	1	1,301
kebakaran	1/6	1/11	2	1
hutan	1/6	1/11	2	1
di	1/6	0	1	1,301
kalimantan	1/6	2/11	2	1
pemerintah	0	1/11	1	1,301
provinsi	0	1/11	1	1,301
barat	0	1/11	1	1,301
akan	0	1/11	1	1,301
melakukan	0	1/11	1	1,301
upaya	0	1/11	1	1,301
pemadaman	0	1/11	1	1,301

2.6.3 *Term Frequency – Invers Document Frequency* (TF-IDF)

Term Frequency – Inverse Document Frequency atau TF-IDF merupakan hasil perkalian antara nilai *Term Frequency* (TF) dan *Inverse Document frequency* (IDF). Untuk melakukan perhitungan bobot TF-IDF (w_{dt}) dapat menggunakan rumus sebagai berikut:

$$w_{dt} = tf_{dt} \times idf_t \quad (3)$$

Keterangan:

d = Dokumen ke-d

t = Kata ke-t dari kata kunci

tf_{dt} = Jumlah kemunculan kata pada dokumen ke-d terhadap kata ke-t

: = Bobot dokumen ke-d terhadap kata ke-t

t = Bobot *inverse* dari nilai DF

= Jumlah dokumen yang mengandung suatu kata (*term*)

Berikut contoh perhitungan nilai TF-IDF yang dapat dilihat pada Tabel 13 di bawah ini:

Tabel 13. Contoh Perhitungan TF-IDF

	Dokumen (D)
D1	berita tentang kebakaran hutan di kalimantan
D2	pemerintah provinsi kalimantan barat akan melakukan upaya pemadaman kebakaran hutan kalimantan

Kata (*Term*)	**TF (D1)**	**TF (D2)**	**IDF**	**TF-IDF** (TF x IDF)	
				D1	**D2**
berita	1/6	0	1,301	0,217	0
tentang	1/6	0	1,301	0,217	0
kebakaran	1/6	1/11	1	0,167	0,091
hutan	1/6	1/11	1	0,167	0,091
di	1/6	0	1,301	0,217	0
kalimantan	1/6	2/11	1	0,167	0,181
pemerintah	0	1/11	1,301	0	0,118
provinsi	0	1/11	1,301	0	0,118
barat	0	1/11	1,301	0	0,118
akan	0	1/11	1,301	0	0,118
melakukan	0	1/11	1,301	0	0,118
upaya	0	1/11	1,301	0	0,118
pemadaman	0	1/11	1,301	0	0,118
Nilai Bobot (D1 dan D2)				**1,152**	**1,189**

2.7 Cosine Similarity

Cosine Similarity merupakan sebuah metode untuk menghitung tingkat similaritas antar dokumen. Metode ini bekerja dengan cara mencari kesamaan antara dua buah objek yang direpresentasikan ke dalam bentuk vektor menggunakan metode *Term Frequency – Inverse Document Frequency* (TF-IDF) (Biantong, 2019). Cosine Similarity akan menghasilkan nilai similaritas yang ıntara nilai 0 sampai dengan nilai 1. Hal ini disebabkan oleh pengukuran ara dua vektor dalam ruang dimensi, yang dimana nilai 1 menunjukkan ıa vektor memiliki arah yang sama dan nilai 0 menunjukkan bahwa dua

vektor saling tegak lurus. Sehingga dapat diketahui bahwa semakin tinggi nilai Cosine Similarity yang dihasilkan, maka semakin mirip pula arah kedua vektor yang dihasilkan dan semakin mirip pula dokumen tersebut. Adapun untuk menghitung nilai Cosine Similarity dapat menggunakan rumus seperti berikut (Alhaq et al., 2021):

$$Cosine\ Similarity = \frac{TF\_IDF_{term\ d1} \cdot TF\_IDF_{term\ d2}}{||TF_{IDF_{term\ d1}}||\ ||TF_{IDF_{term\ d2}}||} \quad (4)$$

jika dijabarkan menjadi:

$$Cosine\ Similarity = \frac{\sum(TF\_IDF_{term\ d1} * TF\_IDF_{term\ d2})}{\sqrt{\sum(TF\_IDF_{term\ d1}{}^2)} \times \sqrt{\sum(TF\_IDF_{term\ d2}{}^2)}} \quad (5)$$

Pada rumus diatas, diketahui bahwa pada pembilang menerapkan *dot product* yang berfungsi untuk menghitung kesamaan arah antara vektor TF-IDF pada setiap *term* yang ada di dokumen 1 dan vektor TF-IDF pada setiap *term* yang ada di dokumen 2. Jika nilai yang dihasilkan bernilai positif, maka kedua vektor tersebut cenderung memiliki arah yang sama. Sebaliknya, jika nilai yang dihasilkan bernilai negatif, maka kedua vektor tersebut memiliki arah yang berlawanan. Sedangkan pada penyebutnya, rumus ini menerapkan norm L2 (*Euclidean Norm*) yang berperan dalam menghitung jarak antara vektor TF-IDF pada setiap *term* yang ada di dokumen 1 dan vektor TF-IDF pada setiap *term* yang ada di dokumen 2. Selain itu, penggunaan norm L2 dalam pengukuran jarak vektor ini memastikan bahwa nilai jarak yang dihasilkan akan selalu bernilai positif.

Salah satu kelebihan dari Cosine Similarity adalah mengukur tingkat similaritas dokumen berdasarkan pengukuran kesamaan arah sudut antara dua vektor. Sehingga panjang pendeknya suatu dokumen tidak akan mempengaruhi hasil yang diperoleh (Melita, 2018). Berikut contoh perhitungan Cosine Similarity yang dapat dilihat pada Tabel 14:

Tabel 14. Contoh Perhitungan Similaritas dengan *Cosine Similarity*

	Dokumen (D)				
D1	berita tentang kebakaran hutan di kalimantan				
D2	pemerintah provinsi kalimantan barat akan melakukan upaya pemadaman kebakaran hutan kalimantan				
Kata (*Term*)	**TF-IDF (D1)**	**TF-IDF (D2)**	**TF-IDF (D1) x TF-IDF(D2)**	**TF-IDF** (TF x IDF)	
				$(D1)^2$	**$(D2)^2$**
berita	0,217	0	0	0,047	0
tentang	0,217	0	0	0,047	0
kebakaran	0,167	0,091	0,015	0,027	0,008
hutan	0,167	0,091	0,015	0,027	0,008
di	0,217	0	0	0,047	0
kalimantan	0,167	0,181	0,030	0,027	0,033
pemerintah	0	0,118	0	0	0,014
provinsi	0	0,118	0	0	0,014
barat	0	0,118	0	0	0,014
akan	0	0,118	0	0	0,014
melakukan	0	0,118	0	0	0,014
upaya	0	0,118	0	0	0,014
pemadaman	0	0,118	0	0	0,014
	SUM		**0,060**	**0,222**	**0,147**

Menghitung Cosine Similarity antara D1 dan D2:

$$Cosine\ Similarity = \frac{0{,}060}{\sqrt{0{,}222}x\ \sqrt{0{,}147}}$$

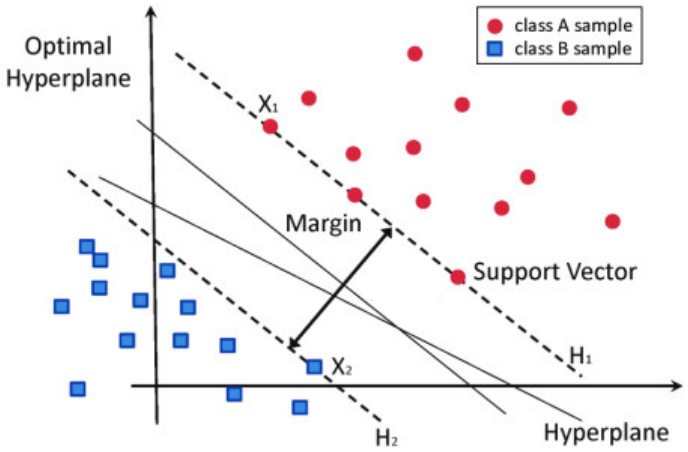
$$Cosine\ Similarity = \frac{0{,}060}{0{,}471\ x\ 0{,}383}$$

$$Cosine\ Similarity = 0{,}333$$

Berdasarkan hasil perhitungan diatas, dapat diketahui tingkat similaritas atau kemiripan antara dokumen 1 (D1) dan dokumen 2 (D2) memperoleh nilai similaritas 0,333.

2.8 Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan salah satu algoritma sering digunakan sebagai algoritma klasifikasi. SVM pertama kali ditemukan dan diusulkan oleh Boser, Guyon dan Vapnik pada tahun 1992 sebagai teknik klasifikasi yang efisien dalam mengatasi masalah data *non-linear*. Secara sederhana, konsep kerja dari algoritma SVM ini adalah menemukan *hyperplane* terbaik yang berfungsi sebagai pemisah antara dua buah kelas pada input *space* dengan memaksimalkan jarak antar kelas atau *margin* (Alhaq et al., 2021).



Gambar 6. Ilustrasi Mekanisme Kerja Algoritma SVM
(Gonzalo et al., 2016)

Dalam proses kerjanya, hasil klasifikasi SVM dipengaruhi oleh beberapa hal berikut (Samsudiney, 2019):

a. *Support Vector*

Support vector merupakan objek data yang paling sulit untuk diklasifikasikan. Hal ini disebabkan oleh letak objek data tersebut yang berada paling dekat dengan *hyperplane*. Letaknya yang begitu dekat dengan *hyperplane*, membuat *support vector* ini menjadi objek data yang hampir tumpang tindih (*overlap*) dengan kelas lain. Maka dari itu, *support vector* inilah yang nantinya akan diperhitungkan untuk menemukan *hyperplane* yang paling optimal oleh algoritma SVM ini.

b. *Hyperplane*

Hyperplane merupakan suatu fungsi dalam SVM yang digunakan sebagai pemisah antar kelas. *Hyperplane* sendiri terbagi menjadi 2 jenis yakni *linear*

hyperplane dan *non-linear hyperplane*. Berikut perbedaan dari kedua jenis *hyperplane* SVM:

Tabel 15. Perbedaan *Linear Hyperplane* dan *Non-linear Hyperplane*

Linear Hyperplane	***Non-linear Hyperplane***
Cocok untuk data yang dapat dipisahkan secara linier (*linearly separable*).	Cocok untuk data yang tidak dapat dipisahkan secara linier (*not linearly separable*).
Memisahkan dua kelas dengan garis lurus (*line whereas*) untuk ruang dua dimensi (2D) atau dengan bidang datar (*plane similarly*) untuk ruang tiga dimensi (3D). Pemilihan metode pemisahan kelas tergantung dari kompleksitas data yang dikelola.	Menggunakan pendekatan *kernel trick* untuk memetakan data ke dimensi yang lebih tinggi dengan menggunakan kernel seperti *polynomial, radial basis function* (RBF) serta beberapa jenis kernel lainnya untuk menciptakan *hyperplane non-linear*.

c. *Margin*

Margin merupakan jarak antara *support vector* dan *hyperplane*. *Margin* sendiri terbagi menjadi dua jenis yakni *margin* positif yang merupakan jarak antara *support vector* dan *hyperplane* dari kelas positif dan *margin* negatif yang merupakan jarak antara *support vector* dan *hyperplane* dari kelas negatif. Semakin besar *margin,* maka dianggap jika proses klasifikasi memiliki generalisasi yang lebih baik dan hal ini juga berpotensi dalam mengurangi risiko terjadinya *overfitting*.

Proses untuk melakukan klasifikasi menggunakan SVM, dapat diawali dengan mencari *hyperplane* sebagai pemisah antara dua buah kelas. Pencarian dilakukan dengan melakukan perhitungan *margin* antara *hyperplane* dan *support vector* menggunakan persamaan berikut (Mursianto et al., 2022):

$$optimal\_hyperplane = (w.x) + b = 0 \quad (6)$$

Dalam data x_i, termasuk ke dalam kelas negatif yang diinisialisasikan dengan nilai -1 dan [illegible]finisikan sebagai bidang pembatas pertama serta dalam data x_i lainnya, ke dalam kelas positif yang diinisalisasikan dengan nilai 1 sebagai bidang kedua. Kedua pembatas tersebut dapat dituliskan ke dalam persamaan

$$margin\_negative = (w.x) + b = -1 \quad (7)$$

$$margin\_positive = (w.x) + b = 1 \quad (8)$$

Maka pencarian *hyperplane* terbaik dengan memaksimalkan kedua bidang pembatas pada (6) dan (7) dapat diimplementasikan menggunakan persamaan berikut:

$$margin = y_i((x_i w) + b) \geq 1 \quad (9)$$

Persamaan diatas umumnya digunakan sebagai kondisi untuk melakukan *update* pada bobot vektor (w) dan nilai bias (b) selama proses iterasi berlangsung. Proses *updating* dilakukan dengan melihat kondisi, kondisi ini terbagi menjadi dua yakni: Jika kondisi bernilai *true* atau hasil klasifikasi bernilai benar dan *margin* yang digunakan cukup, maka proses *updating* hanya dilakukan pada bobot vektor saja. Sedangkan jika kondisi yang diperoleh bernilai *false* atau hasil klasifikasi bernilai salah atau *margin* yang digunakan tidak cukup, maka proses *updating* dilakukan pada bobot vektor dan nilai bias. Proses ini dapat dilakukan dengan menggunakan persamaan berikut:

$$w_{update} = w + \eta((x_i . y_i) - 2\lambda w) \quad (10)$$

$$b_{update} = b + (\eta y_i) \quad (11)$$

Keterangan:

w = Bobot vektor

y_i = Kelas atau label data

x_i = Nilai atribut

b = *Scalar* yang digunakan sebagai nilai bias atau konstanta

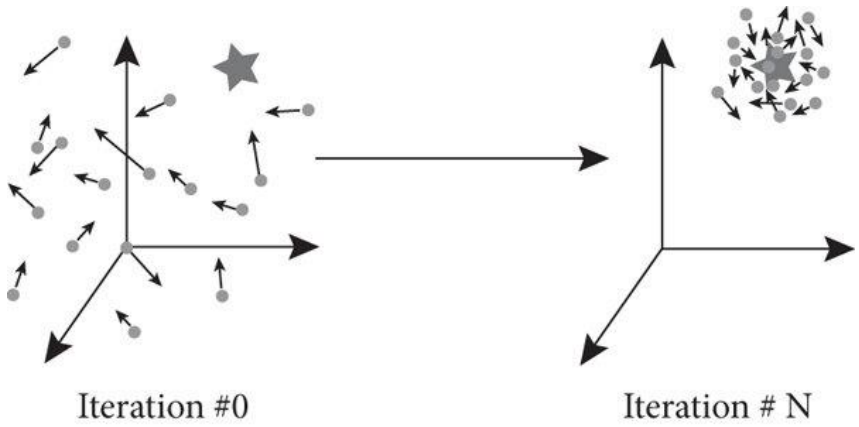
η = *Learning rate* atau besar langkah untuk proses *updating*

λ = *Lambda* digunakan untuk regulasi model

2.9 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) merupakan salah satu teknik optimasi yang termasuk sebagai metode metaheuristik yang bertujuan untuk mencari solusi optimal dalam berbagai masalah optimalisasi seperti optimalisasi fungsi matematis, ın parameter, penjadwalan tugas, hingga pemodelan kecerdasan buatan *intelligence*). Dalam proses pencarian solusi, mekanisme kerja dari PSO rinspirasi oleh populasi dari kawanan burung atau ikan, dimana setiap

populasi tersebut memiliki individu yang dapat mempengaruhi individu lainnya. Kawanan tersebut diasumsikan sebagai serangkaian partikel yang memiliki ukuran tertentu serta posisi awalnya terletak di suatu lokasi yang acak dalam ruang multidimensi. Partikel-partikel inilah yang nantinya akan bergerak dalam ruang pencarian untuk melakukan eksplorasi dan menemukan solusi terbaik melalui interaksi satu sama lain (Pahrul, 2023).



Gambar 7. Ilustrasi Mekanisme Kerja Metode PSO
(He & He, 2022)

Pada gambar 5 diatas, diketahui bahwa dalam PSO terdapat tiga variabel penting yakni partikel, *local best* (pbest) dan *global best* (gbest). Partikel merupakan representasi dari individu yang akan mencari solusi terbaik. *Local best* (pbest) merupakan posisi terbaik yang pernah dicapai oleh suatu partikel. Sedangkan *global best* (gbest) yaitu posisi terbaik dari keseluruhan partikel. Variabel pbest dan gbest berfungsi dalam melakukan perhitungan kecepatan partikel yang nantinya akan digunakan untuk menghitung posisi partikel untuk iterasi berikutnya (Nugraha & Abdulloh, 2022). Setiap partikel diasumsikan memiliki dua karakteristik yakni posisi dan kecepatan. Posisi dan kecepatan setiap partikel nantinya akan diperbarui (*update*) secara iteratif. Proses ini akan berhenti hingga solusi optimal ditemukan atau mencapai kondisi tertentu.

Adapun persamaan yang digunakan untuk melakukan pergerakan partikel pada metode PSO adalah sebagai berikut (Nugraha & Abdulloh, 2022):

$$v_i^{t+1} = w.v_i^t + c_1 r_1(pBest_i - x_i^t) + c_2 r_2(gBest - x_i^t) \quad (12)$$

Persamaan diatas merupakan persamaan untuk melakukan *updating* pada kecepatan partikel. Setelah melakukan *updating* kecepatan, selanjutnya adalah

melakukan *updating* posisi tiap partikel. Hal ini dapat dilakukan dengan menggunakan persamaan berikut:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (13)$$

Keterangan:

v_i^{t+1} = Kecepatan partikel *i* pada iterasi *t* + 1

v_i^t = Kecepatan partikel *i* pada iterasi *t*

x_i^{t+1} = Posisi partikel *i* pada iterasi *t* + 1

x_i^t = Posisi partikel *i* pada iterasi *t*

w = Bobot inersia

c_1 dan c_2 = Konstanta percepatan

r_1 dan r_2 = Bilangan acak antara 0 dan 1

$pBest_i$ = Posisi terbaik yang pernah dicapai oleh partikel *i*

$gBest$ = Posisi terbaik yang pernah dicapai oleh seluruh *swarm*

2.10 *Confusion Matrix*

Confusion Matrix merupakan suatu metode yang umumnya digunakan untuk mengukur tingkat akurasi dari hasil klasifikasi. Perhitungan akurasi dilakukan untuk mengevaluasi kinerja suatu model klasifikasi pada sebuah set data uji yang telah diketahui nilai sebenarnya.

Tabel 16. *Confusion Matrix*

Kelas Prediksi	**Kelas Sebenarnya**	
	Clickbait	***Non-clickbait***
Clickbait	*True Positive* (TP)	*False Positive* (FP)
Non-clickbait	*False Negative* (FN)	*True Negative* (TN)

Sumber: (Biantong, 2019)

Pada tabel 16 diatas, diketahui bahwa *Confusion Matrix* memiliki beberapa nilai yakni *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN). Adapun nilai-nilai dari *Confusion Matrix* ini didefinisikan sebagai [illegible] (Mursianto et al., 2022):

e Positive (TP)

Jumlah data positif suatu kelas yang memiliki hasil positif saat dilakukan diksi maupun pada hasil yang sebenarnya. Dalam kasus ini, variabel TP

akan mewakili jumlah prediksi dokumen *clickbait* yang memiliki hasil klasifikasi dengan benar (dokumen terklasifikasi sebagai *clickbait* juga).

b. *True Negative* (TN)

Jumlah data negatif suatu kelas yang memiliki hasil negatif pula saat dilakukan prediksi begitupun pada hasil yang sebenarnya. Dalam kasus ini, variabel TN akan mewakili jumlah prediksi dokumen *non-clickbait* yang memiliki hasil klasifikasi dengan benar (dokumen terklasifikasi sebagai *non-clickbait* juga).

c. *False Positive* (FP)

Jumlah data dari suatu kelas yang diprediksi memiliki hasil positif, namun hasil yang sebenarnya bernilai negatif. Dalam kasus ini, variabel FP akan mewakili jumlah prediksi dokumen *non-clickbait* yang memiliki hasil klasifikasi dengan tidak benar (dokumen terklasifikasi sebagai *clickbait*).

d. *False Negative* (FN)

Jumlah data dari suatu kelas yang diprediksi memiliki hasil negatif, namun hasil yang sebenarnya bernilai positif. Dalam kasus ini, variabel FN akan mewakili jumlah prediksi dokumen *clickbait* yang memiliki hasil klasifikasi dengan tidak benar (dokumen terklasifikasi sebagai *non-clickbait*).

Confusion Matrix juga biasanya digunakan untuk menghitung beberapa nilai, mulai dari nilai akurasi (*accuracy*), nilai presisi (*precision*), nilai *recall* dan *F1-Score*. Untuk melakukan perhitungan dari beberapa nilai tersebut, dapat dilakukan dengan menggunakan persamaan-persamaan berikut (Pahrul, 2023):

a. Akurasi (*Accuracy*)

Akurasi mendefinisikan nilai keakuratan sistem dalam melakukan proses klasifikasi dengan benar. Nilai akurasi dapat diperoleh dengan menggunakan persamaan berikut:

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (14)$$

b. Presisi (*Precision*)

Presisi atau *precision* terbagi menjadi 2 yakni presisi positif dan presisi atif. Nilai presisi kelas positif mendefinisikan nilai perbandingan atau o antara jumlah prediksi benar bernilai positif dibandingkan dengan eluruhan hasil yang diprediksi bernilai positif. Sedangkan nilai presisi

kelas negatif mendefinisikan nilai perbandingan atau rasio antara jumlah prediksi benar bernilai negatif dibandingkan dengan keseluruhan hasil yang diprediksi bernilai negatif. Nilai presisi kelas positif dan kelas negatif dapat diperoleh dengan menggunakan persamaan berikut:

$$Precision\_positive = \frac{TP}{TP+FP} \quad (15)$$

$$Precision\_negative = \frac{TN}{TN+FN} \quad (16)$$

c. *Recall*

Sama halnya dengan nilai presisi, Nilai *recall* juga terbagi menjadi 2 yakni nilai *recall* kelas positif dan nilai *recall* kelas negatif. Untuk kelas positif, nilai *recall* didefinisikan sebagai jumlah prediksi yang bernilai benar dari semua kelas yang bernilai positif. Sedangkan nilai *recall* kelas negatif dapat didefinisikan sebagai jumlah prediksi yang bernilai negatif dari semua kelas yang bernilai negatif. Nilai *recall* untuk kelas positif dan negatif dapat diperoleh dengan menggunakan persamaan berikut:

$$Recall\_positive = \frac{TP}{TP+FN} \quad (17)$$

$$Recall\_negative = \frac{TN}{TN+FP} \quad (18)$$

d. *F1-Score*

Nilai *F1-Score* terbagi menjadi 2 yang terbagi berdasarkan kelas yang diantaranya kelas positif dan kelas negatif. *F1-Score* mendefinisikan perbandingan antara nilai rata-rata presisi dan nilai *recall*. Nilai *F1-Score* untuk kelas positif dan kelas negatif dapat diperoleh dengan menggunakan persamaan berikut:

$$F1 - Score\_positive = \frac{2x\ (Precision\_positive\ x\ Recall\_positive)}{Precision\_positive+Recall\_positive} \quad (19)$$

$$F1 - Score\_negative = \frac{2x\ (Precision\_negative\ x\ Recall\_negative)}{Precision\_negative+Recall\_negative} \quad (20)$$