KLASIFIKASI JUDUL VIDEO YOUTUBE MENGANDUNG CLICKBAIT MENGGUNAKAN METODE NAÏVE BAYES

Widyadhana Abidi Ahmad[1], Septiyawan Rosetya Wardhana[2]*

[1,2]Prodi Teknik Informatika, Institut Teknologi Adhi Tama Surabaya

**Penulis korespondensi : rossywardhana@itats.ac.id*

ABSTRACT

Digital content is increasingly dominating the media world, and platforms like YouTube have become a primary vehicle for content creators to share information, entertainment, and knowledge. Unfortunately, clickbait strategies have become a troubling problem on these platforms. Clickbait refers to attracting users' attention through catchy, explicit, controversial, or misleading titles to increase the number of clicks without considering the quality or relevance of the content. As a result, the user experience can be disrupted, and trust in the platform and content creators can be eroded. The problem in this study is developing a solution to identify YouTube video titles that contain clickbait elements using the Naive Bayes classification method. The dataset used in this study consists of 200 training data on YouTube video titles labeled clickbait or non-clickbait by a group of expert assessors. The research process with 50 testing data involves several stages, including data pre-processing to clean and process the title text, feature extraction using a numerical approach from the text, and training and evaluating the Naive Bayes model. The test results that have been carried out produce an accuracy value of 64%, precision of 61%, recall of 88%, and F1-score (F.Measure) of 71%. This experiment also considers using model optimization techniques to improve classification performance. Although there are some limitations, such as variations in the definition of clickbait and the influence of dynamic changes in content trends, this method positively contributes to dealing with clickbait problems on the YouTube platform.

Article History

Received : 12-03-2025
Revised : 25-03-2025
Accepted : 30-03-2025

Keywords

text classification
Clickbait
YouTube
Naive Bayes method
Accuracy
user experience
digital content

ABSTRAK

Konten digital semakin mendominasi dunia media, dan platform seperti YouTube telah menjadi wahana utama bagi pembuat konten untuk berbagi informasi, hiburan, dan pengetahuan. Sayangnya, strategi clickbait telah menjadi masalah yang meresahkan di platform ini. Clickbait merujuk pada praktik yang menarik perhatian pengguna melalui judul yang menarik, eksplisit, kontrovesi atau menyesatkan, dengan tujuan meningkatkan jumlah klik tanpa mempertimbangkan kualitas atau relevansi konten yang sebenarnya. Akibatnya, pengalaman pengguna dapat terganggu dan kepercayaan terhadap platform serta pembuat konten dapat terkikis. Permasalahan dalam penelitian ini adalah pengembangan solusi untuk mengidentifikasi judul video Youtube yang mengandung unsur clickbait dengan menggunakan metode klasifikasi Naive Bayes. Dataset yang digunakan dalam penelitian ini terdiri dari 200 data training judul-judul video Youtube yang telah diberi label clickbait atau non-clickbait oleh sekelompok penilai pakar. Proses penelitian dengan 50 data testing melibatkan beberapa tahap, termasuk pra-pemrosesan data untuk membersihkan dan mengolah teks judul, ekstraksi fitur menggunakan pendekatan numerik dari teks, serta pelatihan dan evaluasi model Naive Bayes. Hasil pengujian yang telah dilakukan menghasilkan nilai akurasi 64%, presisi 61%, recall 88%, dan F1-score (F.Measure) 71%. Eksperimen ini juga mempertimbangkan penggunaan teknik pengoptimalan model untuk meningkatkan kinerja klasifikasi. Meskipun terdapat beberapa batasan, seperti variasi dalam definisi clickbait dan pengaruh dinamika perubahan tren konten, metode ini memberikan kontribusi positif dalam menangani permasalahan clickbait di platform Youtube.

PENDAHULUAN

Internet merupakan salah satu teknologi informasi yang diminati banyak orang karena internet telah menyentuh berbagai aspek budaya manusia mulai dari gaya hidup, pendidikan, penelitian hingga bisnis. Salah satu kelebihan internet adalah menyediakan informasi dengan cepat dan fleksibel, namun terkadang informasi yang diberikan tidak sesuai dengan judul artikel di media elektronik [1].

Semakin populernya media online kini menjadi persaingan untuk mendapatkan penghasilan tambahan. Adapun *clickbait* yang sering kita jumpai. *Clickbait* adalah konten media online yang ditujukan untuk menarik perhatian dan memikat pengunjung ke konten mereka. Clickbait adalah jenis konten yang bertujuan untuk menarik perhatian dan mendorong pengguna untuk mengeklik tautan melalui penggunaan judul atau gambar mini yang sensasional, dilebih-lebihkan, atau menyesatkan [1]. Pertumbuhan industri clickbait telah memberikan dampak yang signifikan pada ekosistem media, dengan banyak organisasi media tradisional mengadopsi teknik clickbait untuk meningkatkan lalu lintas dan pendapatan [2].

Clickbait memiliki salah satu fitur yang membuat headline eye-catching yang berguna untuk menarik perhatian agar pengunjung mengkliknya, sehingga menghasilkan pendapatan dari pengunjung yang mengkliknya [3]. Namun, penggunaan clickbait telah dikritik karena potensinya untuk merusak kepercayaan pada sumber berita dan menguras nilai merek [2].

Sorotan *clickbait* adalah penggunaan frase hiperbolik untuk menarik pembaca tertarik mengunjungi website. *Clickbait* bisa menjadi alat yang sangat berguna bagi penyedia situs web untuk menarik lebih banyak pembaca dengan tetap memperhatikan dan memberikan konten yang berkualitas. Namun masih banyak orang yang menyalahgunakan *clickbait* dengan memberikan konten topik yang tidak sesuai dengan judul walaupun itu palsu atau rekayasa. Dapat dikatakan bahwa dengan *clickbait*, pembaca dapat dimanipulasi, bahkan ditipu sehingga menyebabkan pembaca tidak membagikan (share) website karena merasa dirugikan. [4].

Berdasarkan masalah penyalahgunaan *clickbait* di atas, diperlukan klasifikasi untuk menentukan item mana yang merupakan ajakan dan mana yang bukan. Model berbasis pembelajaran mesin, seperti Support Vector Machines, Logistic Regression, dan Naive Bayes, telah digunakan untuk mengklasifikasikan konten clickbait dengan berbagai tingkat keberhasilan. Algoritma Naive Bayes adalah teknik pembelajaran mesin populer yang telah digunakan dalam berbagai tugas klasifikasi, termasuk analisis sentimen dan deteksi spam. Algoritma ini didasarkan pada teorema Bayes, yang menghitung probabilitas suatu kelas berdasarkan fitur input. Penelitian sebelumnya telah menunjukkan bahwa Naive Bayes dapat menjadi pengklasifikasi yang kuat dan efisien, bahkan dalam situasi di mana asumsi independensi dilanggar [5].

Kajian sebelumnya tentang klasifikasi ini adalah artikel Indonesia yang mengeksplorasi *clickbait* berdasarkan frekuensi kata yang biasa muncul dalam sebuah artikel menggunakan metode Naïve Bayes [6]. Sedangkan penelitian ini menggunakan metode Naïve Bayes untuk mengklasifikasikan judul video Yo uTube mengandung Clickbait berbahasa Indonesia.

Algoritma Naïve Bayes Classifier (NBC) merupakan algoritma yang digunakan untuk mencari nilai probabilitas tertinggi untuk mengklasifikasikan data eksperimen ke dalam kategori yang paling sesuai. Keuntungan dari NBC adalah klasifikasi dokumen yang sederhana dan kecepatan komputasi yang tinggi. Metode NBC juga memiliki kinerja yang baik untuk mengklasifikasikan data dokumen yang berisi angka dan teks. Sebelum langkah aplikasi, dokumen harus disajikan dalam format vektor [6].

Oleh karena itu, dalam penelitian ini diusulkan suatu sistem untuk mengklasifikasi judul video Youtube mengandung Clickbait berbahasa Indonesia dengan menggunakan metode Naïve Bayes. Dengan dukungan metode ini diharapkan dapat membantu pengguna Youtube dalam memilih judul video yang mengandung *clickbait* dan yang tidak mengandung *clickbait* agar bisa lebih cermat untuk memilih judul video yang kontrovesi dan sensasional dari isi video yang di publikasikan di media youtube.

TINJAUAN PUSTAKA

Text Mining Preprocessing

Text mining merupakan teknik yang digunakan untuk menangani klasifikasi, clustering, information extraction, dan information retrival. Perbedaan text mining dan data mining adalah pola yang digunakan text mining diambil dari sekumpulan bahasa alami yang tidak terstruktur, sedangkan data mining pola yang diambil dari database yang terstruktur [7]

ISSN: 2774-4345, DOI: 10.31284/j.kernel.2023.v4i2.7520

Text mining dikerjakanan setelah teks dalam dokumen yang digunakan sudah dipersiapkan terlebih dahulu dan masuk ke dalam proses utama. Tahapan proses teks dalam dokumen atau data mentah dipersiapkan dinamakan text processing.

Data mentah memiliki beberapa karakteristik seperti gangguan (*noise*) dan teks yang tidak terstruktur. Data dalam teks yang tidak terstruktur tersebut diubah menjadi data teks yang terstruktur. Berikut merupakan tahap *preprocessing.* [8]

a. Case Folding

Penggunaan huruf kapital teks dalam text *processing* harus konsisten. Fungsi dari *case folding* adalah untuk penyamaan *case* dalam teks. Oleh karena itu, keseluruhan teks dalam dokumen akan diproses dan dikonversi menjadi bentuk standar. Bentuk standar tersebut berupa huruf kecil.

b. Tokenizing

Tahapan *tokenizing* merupakan proses pemotongan dokumen *string input* pada tiap penyusun kata yang berasal dari hasil *case folding*.

c. Filtering

Filtering merupakan proses seleksi atau pengambilan kata yang dihasilkan oleh proses *tokenizing* dan dilakukan dengan menggunakan cara *stopword* maupun *wordlist.*

d. Stemming

Stemming merupakan proses penemuan akar kata dari hasil *filtering* dengan menggunakan aturan tertentu. [8]

Naïve Bayes Classifier

Naïve bayes classifier merupakan sebuah metode klasifikasi yang berakar pada teorema *bayes*. Metode pengklasifikasian dengan menggunakan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai *Teorema Bayes.* Ciri utama dari naive bayes classifier ini adalah asumsi yang sangat kuat (*naif*) akan independensi dari masing-masing kondisi atau kejadian[9]. Dasar dari teorema naive bayes yang digunakan dalam pemrograman adalah rumus Bayes persamaan (1) berikut ini [10][11] :

$$P\ (A|B) = \left(P(A|B)\ *P(A)\right)\ /\ P(B) \quad \text{.....(1)}$$

Peluang kejadian A sebagai B ditentukan dari peluang B saat A, peluang A, dan peluang B. Pada pengaplikasiannya rumus ini berubah menjadi persamaan (2) :

$$P(C_i|D) = \ (P(D|C_i)\ *\ P(C_i))\ /\ P(D) \quad \text{.....(2)}$$

Naive Bayes Classifier merupakan model penyederhanaan dari algoritma *bayes* yang cocok dalam pengklasifikasian text atau dokumen. Persamaannya menggunakan persamaan (3)

$$V_{MAP} = \arg\max P\left(v_j \middle| a_1, a_2, \dots, a_n\right) \quad \text{.....(3)}$$

Berdasarkan persamaan ini, maka rumus *bayes* dapat ditulis seperti persamaan (4) :

$$V_{MAP} = \underset{v_j \in V}{\arg max} \frac{P\left(a_1, a_2, \dots, a_n \middle| v_j\right) P\ (v_j)}{P\ (a_1, a_2, \dots, a_n)} \quad \text{.....(4)}$$

ISSN: 2774-4345, DOI: 10.31284/j.kernel.2023.v4i2.7520

$P\ (a_1, a_2, \dots, a_n)$ merupakan bilangan konstan, sehingga dapat dihilangkan menjadi persamaan (5),

$$V_{MAP} = \underset{v_j \in V}{\arg max}\, P(a_1, a_2, \dots, a_n | v_j)\ P\ (v_j) \quad (5)$$

Karena $P(a_1, a_2, \dots, a_n | v_j)\ P\ (v_j)$ sulit untuk dihitung, maka akan diasumsikan bahwa setiap kata pada dokumen tidak mempunyai keterkaitan,

$$V_{MAP} = \underset{v_j \in V}{\arg max}\, P(v_j)\ \prod_i P\ (a_i | v_j) \quad (6)$$

Dengan :

$$P\ (v_j) = \frac{|docs_j|}{|Cont\ \ |} \quad (7)$$

$$P\ (w_k | v_j) = \frac{n_k + 1}{n + |kosakata|} \quad (8)$$

Dimana untuk :

$P\ (v_j)$ adalah probabilitas setiap dokumen terhadap sekumpulan dokumen.

$P\ (w_k | v_j)$ adalah probabilitas kemunculan kata w_k pada suatu dokumen dengan kategori kelas v_j.

$|docs|$ adalah frekuensi dokumen pada setiap kategori. $|Contoh|$ adalah jumlah dokumen yang ada.

n_k adalah frekuensi kata ke-k pada setiap kategori kosakata adalah jumlah kata pada dokumen test [9].

Evaluasi Kualitas Klasifikasi

Evaluasi kualitas klasifikasi Naive Bayes dilakukan untuk mengukur sejauh mana model tersebut efektif dalam memprediksi dan mengklasifikasikan data dengan benar. Beberapa metrik evaluasi umum yang digunakan untuk mengevaluasi kualitas klasifikasi, yaitu [12]:

1. *Precision*

Precision adalah perhitungan yang digunakan untuk mengetahui seberapa banyak persentase prediksi klasifikasi yang di prediksi true dan terbukti true (TP) dibandingkan dengan jumlah keseluruhan prediksi bernilai true. Persamaan (9) merupakan persamaan untuk menghitung precision:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (9)$$

2. *Recall*

Recall adalah perhitungan yang digunakan untuk mengetahui seberapa banyak persentase prediksi klasifikasi yang di prediksi true dan terbukti true (TP) dibandingkan dengan keseluruhan jumlah data aktual bernilai true. Persamaan (10) merupakan persamaan untuk menghitung *recall* :

$$\text{Recall} = \frac{TP}{TP+F} \quad (10)$$

3. *F-Measure* (F1)

F-Measure adalah perhitungan terhadap penggabungan nilai precision dan recall. Nilai yang dihasilkan pada perhitungan F-Measure ini merupakan nilai yang mewakili keseluruhan kinerja sistem. Semakin besar nilai F-Measure yang dihasilkan maka semakin baik performansinya. Persamaan (11) merupakan persamaan untuk menghitung nilai F-Measure :

$$\text{F.Measure} = 2 * \frac{recall*precision}{recall+precisio} \quad (11)$$

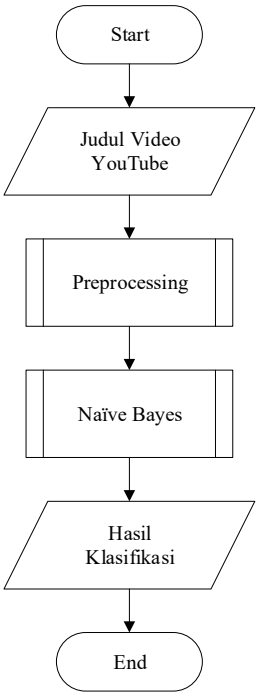
4. *Accuracy*

Accuracy adalah jumlah dokumen yang diklasifikasikan dengan benar, baik itu true positive maupun true negative. Persamaan (12) merupakan deskripsi dari perhitungan Accuracy.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \quad (12)$$

METODE

Perancangan sistem adalah proses membuat sebuah laporan perencanaan sistem yang mendukung tujuan yang ingin dicapai. Perencanaan sistem bertujuan untuk melihat peluang dan meminimalisir kesalahan. Dalam hal ini, harus dapat dipastikan bahwa semua prasyarat kebutuhan sistem harus sudah terpenuhi.



Gambar 1 Flowchart Perancangan Klasifikasi Judul Video YouTube

Gambar 1 adalah perancangan diagram yang menjelaskan alur proses program klasifikasi judul video YouTube berdasarkan *clickbait* dan *non-clickbait*. Proses dimulai dari, dilakukan pencarian judul video YouTube. Kedua, melakukan *preprocessing* data dengan menggunakan, yaitu melakukan proses pemrosesan awal data dan menyiapkan data teks untuk mengerjakan proses klasifikasi. Selanjutnya, dilakukan proses pencarian nilai probabilitas terbaik untuk proses klasifikasi pada judul video YouTube menggunakan metode Naïve Bayes.

Preprocessing

Proses pertama yang dilakukan adalah menginputkan judul video YouTube kemudian melakukan proses *case folding*, yaitu proses untuk mengubah semua huruf kapital menjadi huruf kecil. Kedua, dilakukan proses tokenizing yang merupakan proses pemotongan dokumen *string input* pada tiap penyusun kata. Ketiga, proses *filtering* yang adalah proses pengambilan kata yang didapatkan proses *tokenizing* dengan menerapkan *stopword*. Keempat, dilakukan proses *stemming* untuk menemukan akar kata dari hasil *filtering*. Terakhir didapatkan hasil proses *preprocessing* judul video YouTube.

1. Case Folding
   Pada proses pertama *preprocessing* data dilakukan proses *case folding*. Dimana *case folding* merupakan proses untuk mengubah huruf kapital menjadi huruf kecil

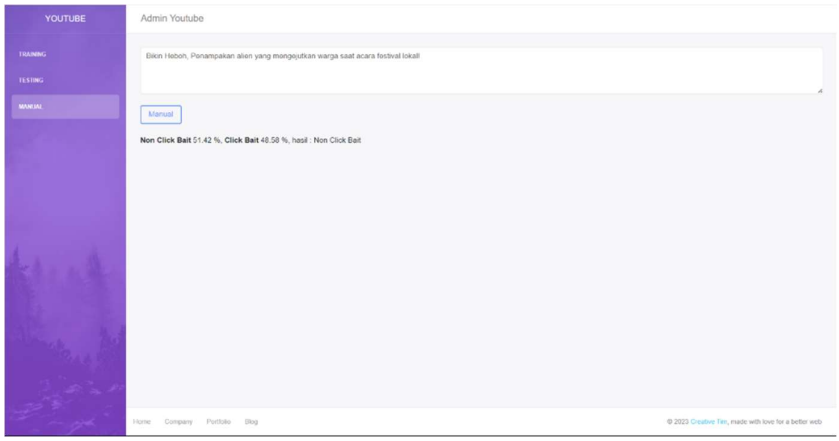
2. Tokenizing
   Pada proses kedua *preprocessing* data dilakukan proses *tokenizing*. Dimana *tokenizing* merupakan proses penghapusan dokumen *string input* pada tiap penyusun kata

3. Filtering
   Pada proses ketiga *preprocessing* data dilakukan proses *Filtering*. Dimana *filtering* merupakan proses pengambilan kata yang dihasilkan oleh proses *tokenizing* dengan memakai cara *stopword*.
4. Stemming
   Pada proses keempat *preprocessing* data dilakukan proses *stemming*. Dimana *stemming* merupakan proses penemuan akar kata dari hasil *filtering*.

HASIL DAN PEMBAHASAN

Implementasi program dibuat menggunakan bahasa pemrograman *php*. Selain bahasa pemrograman php, sistem ini juga menggunakan MySQL sebagai basis datanya.

Analisa Sistem



Gambar 2 Tampilan Hasil Persentase Judul Video Youtube

Pada gambar 2 menampilkan input judul video dengan percobaan hasil persentase *Non Click Bait 51,42%* dan persentase *Click Bait 48,58%* dengan klasifikasi hasil *Non Click Bait*. Setelah itu, hasil dari pencarian terakhir tadi akan muncul pada tab menu *Testing*. Disana adalah daftar hasil input pencarian hasil persentase nilai judul video youtube apakah termasuk kategori clickbait atau

non-clickbait, dari semua data tersebut merupakan hasil presentase probabilitas yang dihasilkan dari proses klasifikasi *naive bayes.*

Hasil Pengujian

Banyak nilai TN adalah 9 data. Banyak nilai FP adalah 15 data. Banyak nilai FN adalah 3 data. Selanjutnya dari nilai tersebut dapat dihitung nilai Accuracy, Precision, Recall dan F.Measure.

1. Accuracy

Akurasi adalah jumlah dokumen yang diklasifikasikan dengan benar, baik itu *true positive* maupun *true negative*. Dimana nilai TP adalah 23 dan nilai TN adalah 9.

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

$$\frac{23+9}{23+15+3+9} = 64\%$$

2. Precision

Presisi adalah perhitungan yang digunakan untuk mengetahui seberapa banyak persentase prediksi klasifikasi yang diperiksa *true* dan terbukti *true* (TP) dibandingkan dengan jumlah keseluruhan prediksi bernilai *true*. Dimana nilai TP adalah 23 dan nilai FP adalah 15.

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\frac{23}{23+15} = 61\%$$

3. Recall

Recall adalah perhitungan yang digunakan untuk mengetahui seberapa banyak persentase prediksi klasifikasi yang diprediksi *true* dan terbukti *true* (TP) dibandingkan dengan keseluruhan jumlah data aktual bernilai *true*. Dimana nilai TP adalah 23 dan nilai FN adalah 3.

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$\frac{23}{23+3} = 88\%$$

4. F.Measure

F.Measure menggambarkan perbandingan rata-rata precision dan recall yang dibobotkan. Accuracy tepat kita gunakan sebagai acuan performansi algoritma jika dataset kita memiliki jumlah data False Negatif dan False Positif yang sangat mendekati (symmetric). Namun jika jumlahnya tidak mendekati, maka sebaiknya kita menggunakan F.Measure sebagai acuan.

$$\text{F.Measure} = 2 * \frac{recall*precision}{recall+precision}$$

$$2 * \frac{88\% * 61\%}{88\% + 61\%} = 72\%$$

Metode *Naïve Bayes Classifier* dapat diimplementasikan untuk mengklasifikasikan judul video youtube berdasarkan clickbait atau non clickbait. Hasil uji coba dengan nilai persentase *Non Click Bait 51.42%* dan *Click Bait 48.58%* dengan klasifikasi *Non Click Bait. pengujian yang telah*

dilakukan pada 50 data testing maka didapatkan nilai akurasi sebesar 64%, nilai presisi sebesar 61%, nilai recall sebesar 88% dan nilai F.Measure sebesar 72%.

Hasil penelitian ini menunjukkan keefektifan algoritma Naive Bayes dalam mengklasifikasikan judul video YouTube sebagai clickbait atau non-clickbait. Naive Bayes telah terbukti menjadi pengklasifikasi yang tangguh dan efisien, bahkan dalam situasi di mana asumsi independensi dilanggar.

Namun, penting untuk dicatat bahwa kinerja pengklasifikasi Naive Bayes dapat dipengaruhi oleh kualitas dan relevansi fitur input. Dalam praktiknya, selalu disarankan untuk membandingkan berbagai model klasifikasi pada kumpulan data tertentu dan mempertimbangkan kinerja prediksi serta efisiensi komputasi.

KESIMPULAN

Dalam makalah penelitian ini, kami telah mengusulkan sebuah metode untuk mengklasifikasikan judul video YouTube sebagai clickbait atau non-clickbait menggunakan algoritma Naive Bayes. Temuan kami menunjukkan keefektifan algoritma Naive Bayes dalam tugas ini, dengan potensi untuk digunakan oleh kreator konten guna mengoptimalkan judul video mereka dan memaksimalkan pendapatan mereka di platform tersebut.Hasil uji coba dengan nilai persentase *Non Click Bait 51.42%* dan *Click Bait 48.58%* dengan klasifikasi *Non Click Bait.* Berdasarkan pengujian yang telah dilakukan pada 50 data *testing* maka didapatkan nilai akurasi sebesar 64%, nilai presisi sebesar 61%, nilai recall sebesar 88% dan nilai F.Measure sebesar 72%.

DAFTAR PUSTAKA

[1] Gothankar, R., Troia, F. D., & Stamp, M. (2021). Clickbait Detection in YouTube Videos. In arXiv (Cornell University). Cornell University. https://doi.org/10.48550/arxiv.2107.12791
[2] Rony, M. M. U., Hassan, N., & Yousuf, M. (2017). Diving Deep into Clickbaits: Who Use Them to What Extents in Which Topics with What Effects? In arXiv (Cornell University). Cornell University. https://doi.org/10.48550/arxiv.1703.09400
[3] Sagita, R., Enri, U., & Primajaya, A. (2020). Klasifikasi Berita Clickbait Menggunakan K-Nearest Neighbor (KNN). *JOINS (Journal of Information System)*, *5*(2), 230–239. https://doi.org/10.33633/joins.v5i2.3705
[4] Puji Lestari, D. W., Perdana, R. S., & Adikara, P. P. (2019). Klasifikasi Video Clickbait pada YouTube Berdasarkan Analisis Sentimen Komentar Menggunakan Learning Vector Quantization (LVQ) dan Lexicon- Based Features. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, *3*(2), 8772–8777. http://j-ptiik.ub.ac.id
[5] Raschka, S. (2014). Naive Bayes and Text Classification I - Introduction and Theory. In arXiv (Cornell University). Cornell University. https://doi.org/10.48550/arxiv.1410.5329
[6] Yavi, A. F. (2018). Klasifikasi Artikel Berbahasa Indonesia untuk Mendeteksi Clickbait menggunakan Metode Naïve Bayes. *Journal of Chemical Information and Modeling*, *53*(9), 1689–1699.
[7] Agustina, D. A., Subanti, S., & Zukhronah, E. (2021). Implementasi Text Mining Pada Analisis Sentimen Pengguna Twitter Terhadap Marketplace di Indonesia Menggunakan Algoritma Support Vector Machine. *Indonesian Journal of Applied Statistics*, *3*(2), 109. https://doi.org/10.13057/ijas.v3i2.44337
[8] Kartikasari, T. S., Setiawan, H., & Lucky Tirma Irawan, P. (2020). Implementasi Text Mining Untuk Analisis Opini Publik Terhadap Calon Presiden. *Jurnal Simantec*, *7*(1), 39–47. https://doi.org/10.21107/simantec.v7i1.6528
[9] A. A. Mahran, R. K. Hapsari, and H. Nugroho, 'Penerapan Naive Bayes Gaussian Pada Klasifikasi Jenis Jamur Berdasarkan Ciri Statistik Orde Pertama', J. Ilm. NERO, vol. 5, no. 2, 2020.

[10] Chandra, W. N. (2016). Spam Filtering Dengan Metode Pos Tagger Dan Klasifikasi Naïve Bayes. *Jurnal Ilmiah Teknologi Informasi Asia*, 47-55.

[11] M. M. Rahma and A. D. Salman, 'Heart Disease Classification-Based on the Best Machine Learning Model', Iraqi J. Sci., vol. 63, no. 9, pp. 3966–3976, 2022, doi: 10.24996/ijs.2022.63.9.28.

[12] R. K. Hapsari, A. H. Salim, B. D. Meilani, T. Indriyani, and A. Rachman, 'Comparison of the Normalization Method of Data in Classifying Brain Tumors with the k-NN Algorithm', in The 2nd International Conference on Neural Networks and Machine Learning, Atlantis Press International BV, 2023, pp. 21–29. doi: 10.2991/978-94-6463-174-6_3.