



Perbandingan Algoritma Naïve Bayes, K-Nearest Neighbors dan Random Forest untuk Klasifikasi Sentimen Terhadap BPJS Kesehatan pada Media Twitter

Tamrizal A.M^{1*}, Ainul Yaqin²

¹Magister Teknik Informatika, Universitas Amikom Yogyakarta,
Jl. Ringroad Utara, Condongcatur, Depok, Sleman, Yogyakarta, Indonesia 55283

²Fakultas Ilmu Komputer., Universitas Amikom Yogyakarta,
Jl. Ringroad Utara, Condongcatur, Depok, Sleman, Yogyakarta, Indonesia 55283

*Email Penulis Koresponden: tamrizal@gmail.com

Abstrak:

Dari sejak didirikan, BPJS terus berusaha meningkatkan kualitas pelayanan termasuk menyediakan berbagai layanan pengaduan. Selain fasilitas pengaduan yang telah disediakan oleh BPJS, media sosial seperti Twitter sebenarnya dapat dijadikan sebagai tempat untuk mengumpulkan informasi yang berkaitan dengan BPJS. Berbagai keluhan maupun apresiasi terhadap pelayanan BPJS sering disuarakan melalui media Twitter. Pada penelitian ini, dilakukan pengujian tiga algoritma machine learning yaitu Naïve Bayes, K-Nearest Neighbors dan Random Forest, untuk mengetahui dan membandingkan tingkat akurasi dari masing-masing algoritma tersebut dalam melakukan klasifikasi terhadap sentimen masyarakat terhadap BPJS Kesehatan melalui media Twitter. Pada penelitian ini *dataset* diperoleh dengan melakukan *scrapping* menggunakan Twitter API. Data yang diperoleh kemudian diseleksi dan dilakukan *labeling*. Dari hasil seleksi dan *labeling* didapatkan *dataset* sebanyak 150 *tweet* yang terdiri atas 50 *tweet* positif, 50 *tweet* negatif dan 50 *tweet* netral yang akan digunakan dalam percobaan. Pada percobaan dengan menggunakan 90% data untuk *training* dan 10% data untuk *testing*, didapatkan tingkat akurasi sebesar 80% Naive Bayes, 67% K-Nearest Neighbors dan 87% Random Forest.

This is an open access article under the [CC BY-NC](https://creativecommons.org/licenses/by-nc/4.0/) license



Kata Kunci:

Analisis Sentimen;
Naïve Bayes;
KNN;
Random Forest;
Twitter;
BPJS

Riwayat Artikel:

Diserahkan 27 Oktober 2021
Direvisi 17 Februari 2022
Diterima 4 Maret 2022
Dipublikasi 29 April 2022

DOI:

10.22441/incomtech.v12i1.13642

1. PENDAHULUAN

Badan Penyelenggara Jaminan Sosial (BPJS) telah resmi beroperasi sejak 1 Januari 2014 sebagai tindak lanjut dari ditetapkannya undang-undang nomor 24

tahun 2011 tentang Badan Penyelenggara Jaminan Sosial (BPJS) [1-4]. Kehadiran BPJS dimaksudkan untuk memastikan seluruh penduduk Indonesia terlindungi oleh jaminan kesehatan yang komprehensif, adil, dan merata.

Sejak berdirinya sampai dengan sekarang BPJS terus berusaha meningkatkan kualitas pelayanan termasuk menyediakan berbagai layanan pengaduan melalui unit penanganan pengaduan peserta yang ada di kantor BPJS, petugas informasi dan pengaduan peserta (PIPP) yang ada di rumah sakit, BPJS Kesehatan *Care Center* di nomor 1500400 serta menu lapor pada aplikasi *mobile* JKN. Melalui berbagai fasilitas layanan pengaduan tersebut diharapkan dapat memberikan masukan untuk perbaikan layanan BPJS Kesehatan dan *stakeholder*.

Selain fasilitas pengaduan yang telah disediakan oleh BPJS tersebut, media sosial seperti Twitter sebenarnya dapat dijadikan sebagai tempat untuk mengumpulkan informasi yang berkaitan dengan BPJS. Berbagai keluhan maupun apresiasi terhadap pelayanan BPJS sering disuarakan melalui media Twitter. Hal tersebut sesungguhnya dapat dijadikan peluang untuk mengumpulkan informasi sebanyak-banyaknya untuk menjadi bahan evaluasi bagi BPJS Kesehatan.

Dalam mengelola informasi yang didapatkan dari Twitter, salah satu kendala yang ditemui adalah melakukan klasifikasi terhadap informasi yang didapatkan, karena tidak semua informasi yang didapatkan dari Twitter dapat digunakan sebagai masukan. Terkadang informasi yang didapatkan hanya berupa candaan yang kebetulan menggunakan kata BPJS.

Salah satu metode yang dapat digunakan untuk melakukan klasifikasi adalah menggunakan algoritma *machine learning*, berbagai algoritma *machine learning* dapat digunakan dalam membantu melakukan klasifikasi tetapi dengan tingkat akurasi yang berbeda-beda.

Penelitian mengenai analisis sentimen tentang BPJS yang pernah dilakukan diantaranya oleh M. Nurmalasari dkk [3] yang melakukan penelitian analisis sentimen terhadap opini masyarakat dalam penggunaan *mobile* JKN, penelitian ini dilakukan dengan melakukan *scrapping* data pada media Twitter kemudian dilakukan analisis untuk mengklasifikasikan sentimen, proses klasifikasi dilakukan oleh peneliti tanpa menggunakan algoritma *machine learning*.

Selanjutnya penelitian lain dilakukan oleh A. Rosyadi dkk [1] yang juga melakukan penelitian analisis sentimen opini pengguna Twitter terhadap BPJS. Tetapi pada penelitian yang dilakukan digunakan algoritma Naïve Bayes untuk melakukan klasifikasi. Hasil dari penelitian yang dilakukan didapatkan tingkat akurasi dari algoritma Naïve Bayes hanya sebesar 71%.

Kemudian penelitian lain dilakukan oleh R. Yana Yanis dkk [4] yang melakukan penelitian analisis sentimen terhadap pelayanan BPJS Kesehatan pada guru SMK Eklesia dan Bina Insani Jailolo. Pada penelitian ini juga digunakan algoritma Naïve Bayes untuk melakukan klasifikasi. Hasil dari penelitian yang dilakukan didapatkan tingkat akurasi dari algoritma Naïve Bayes sebesar 84,5%.

Dari beberapa penelitian analisis sentimen terkait BPJS Kesehatan yang pernah dilakukan, penelitian hanya dengan menggunakan satu algoritma khususnya Naïve Bayes. Dalam penelitian ini, penulis ingin melakukan pengujian tiga algoritma *machine learning* yaitu Naïve Bayes, K-Nearest Neighbors dan Random Forest, untuk membandingkan tingkat akurasi dari masing-masing algoritma tersebut dalam melakukan klasifikasi terhadap sentimen masyarakat terhadap BPJS Kesehatan melalui media Twitter. Sehingga dapat dijadikan sebagai pertimbangan

dalam memilih algoritma yang akan digunakan dalam melakukan klasifikasi sentimen analisis.

2. METODE

Naïve Bayes

Algoritma Naïve Bayes merupakan metode klasifikasi yang mengacu pada teorema bayes yang pertama kali dikemukakan oleh Thomas Bayes seorang ilmuwan yang berasal dari inggris. Metode ini menggunakan metode klasifikasi statistik. Ciri utama dari metode ini adalah asumsi akan independensi dari masing-masing kondisi [5-9]. Teorema Bayes dapat digambarkan dalam model matematika [10] sebagai berikut:

$$P(X|H) = \frac{P(X|H)}{\sum_{i=1}^n P(H_i|X)} \cdot P(H) \quad (1)$$

Keterangan:

X : Data dengan class yang belum diketahui

H : Hipotesis data merupakan suatu class spesifik

P(H|X) : Probabilitas hipotesis H berdasar kondisi X (posteriori probabilitas)

P(H) : Probabilitas hipotesis H (prior probabilitas)

P(X|H) : Probabilitas X berdasarkan kondisi pada hipotesis H

P(X) : Probabilitas X

i : 1,2,3, ... , n jumlah data Hipotesis (prior probabilitas)

S : Probabilitas total H

K-Nearest Neighbors (KNN)

Cara kerja Algoritma KNN adalah melakukan klasifikasi data dengan mengambil sejumlah data terdekat sebagai acuan untuk menentukan kelas data baru. Algoritma ini mengklasifikasikan data berdasarkan kedekatannya terhadap data lainnya.

Untuk menghitung jarak antara dua titik, algoritma KNN menggunakan metode *Euclidean Distance* [11-13]. Metode *Euclidean distance* [14] menggunakan formula berikut.

$$d(x, y) = \sqrt{\sum_{i=0}^n (x_{1i} - y_{1i})^2 + (x_{2i} - y_{2i})^2 + \dots} \quad (2)$$

Keterangan:

d(x,y) : Jarak antara x dan y

x : data yang akan diklasifikasi

y : data sekitar

i : jumlah fitur

Random Forest

Random Forest adalah salah satu algoritma yang dipakai untuk melakukan klasifikasi data. Klasifikasi Random Forest dilakukan dengan menggabungkan

pohon (*tree*) melalui proses *training* pada sampel data yang dimiliki. Jumlah pohon (*tree*) yang digunakan akan mempengaruhi tingkat akurasi yang akan didapatkan. Penentuan klasifikasi dengan Random Forest diambil berdasarkan hasil *voting* dari *tree* yang terbentuk [15-17].

Langkah-langkah yang dilakukan dalam melakukan penelitian ini adalah sebagai berikut:

1. Pengumpulan Data

Pengumpulan data dilakukan dengan teknik *scrapping* data pada Twitter dengan menggunakan Twitter API[18]. *Scrapping* dilakukan dengan menggunakan kata kunci “BPJS”.

2. Seleksi dan Pelabelan Data

Data hasil *scrapping* kemudian dianalisis untuk melakukan seleksi agar data yang digunakan adalah *tweet* yang menggunakan bahasa Indonesia dan bukan merupakan data berulang dari hasil *retweet*.

Selanjutnya penulis melakukan pelabelan data secara manual yang terdiri dari Positif, Negatif dan Netral. Berikut beberapa contoh *tweet* yang telah diberi label.

Contoh *tweet* positif:

- “b'Wah ada kabar baik nih gaes, jadi Peserta BPJS Kesehatan bakal dapat Vaksin Booster Di Awal Tahun 2022”
- Mobile JKN punya beragam fitur yang bikin akses layanan kesehatan jadi lebih mudah dilakukan gantre di kantor BPJS. Mantap kan? yuk instal aplikasi Mobile JKN di gawai pintar kamu.
- Malam itu gue pakai BPJS serasa pakai asuransi pribadi. Mantap!
- Mantap. BPJS Kesehatan dan Ikatan Dokter Indonesia bahu membahu demi kesuksesan program JKN. Biar pelayanan makin keren. Kabar gembira nih
- BPJS mantap zaman skrg, ibu sy pengguna BPJS mendapatkan pelayanan maksimal

Contoh *tweet* netral:

- Langsung pergi sendiri aja bebs. Kalau misalnya ada bpjs/uang sendiri langsung cus :(umur kita udh boleh ko tanggung jawab buat diri sendiri semogaaa g parah ya
- kalo wayv konser dijakarta terus perform nectar, pasien bpjs membludak
- b'@Rereunika Omo kak rere up ashdjdk siapkan bpjs
- b'-dips! Di Semarang berlaku ga KIS/BPJS digital ??
- Harga tes alergi tergolong mahal. Banyak orang berharap tes alergi ditanggung BPJS. Benarkah?

Contoh *tweet* negatif:

- Speaking of bpjs yg faskes nya ngasi slot surat rujukan yg keluar, klo sehari cuma 3 slot dan rebutan bgt terus yg sakitnya parah gmn ya? Keburu mati
- *Ga minta komut. Cukup BPJS yg ga pernah dipake ini diwajarkan aja. Gila aja, bong. Lebih parah dari asuransi swasta. Ga pake ttp discharge dan diakumulasi*
- Nunggu sdh 5 jam disini cuma utk cek up. Yg ada kambuh lagi typus gw! #bpjskesehatan #bpjssparah
- Naikin Tarif Iuran BPJS..koq hrs pakai Ngancam-Mengancam.
- Saya peserta BPJS Saya tidak merasakan pelayanan BPJS #BPJSRibet #BPJSKacau #BubarkanBPJS

Setelah proses seleksi dan pelabelan data dilakukan, didapatkan *dataset* sebanyak 150 data *tweet* yang terdiri dari 50 data positif, 50 data negatif dan 50 data netral.

3. *Preprocessing*

Preprocessing merupakan sebuah proses transformasi data agar dapat lebih mudah dipahami [1], data yang telah melalui proses *preprocessing* akan lebih mudah untuk diolah dalam pengujian. Dalam penelitian ini *preprocessing* yang dilakukan adalah *stopword removal* untuk melakukan filter pada *dataset* untuk mengurangi penggunaan kata yang mempunyai frekuensi kemunculan yang tinggi, *punctuation filter* untuk menghilangkan tanda baca ataupun simbol yang terdapat pada *dataset* dan *stemming* untuk mengembalikan kata yang digunakan ke kata dasarnya.

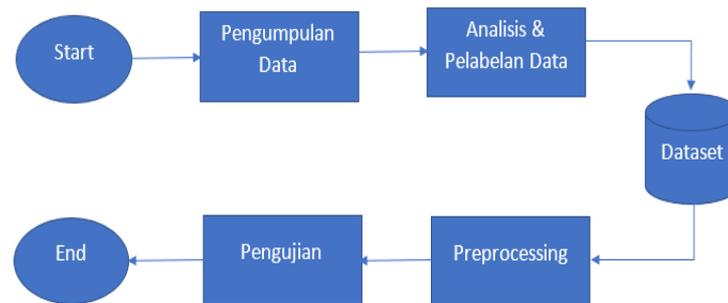
4. *Term Weighting*

Term weighting atau pembobotan kata merupakan proses menghitung bobot setiap kata pada sebuah dokumen agar diketahui kemiripan antar kata pada dokumen tersebut [19]. Pada penelitian ini, *term weighting* dilakukan dengan menghitung *Term Frequency-Inverse Document Frequency* (TF-IDF)[20].

5. Pengujian

Pengujian dilakukan dengan menggunakan algoritma *Naive Bayes*, *K-Nearest Neighbors* dan *Random Forest*. Masing-masing algoritma akan dilakukan pengujian sebanyak 3 kali dengan ketentuan pengujian pertama menggunakan 50% sebagai *data training* dan 50% sebagai *data testing*, pengujian kedua menggunakan 75% sebagai *data training* dan 25% sebagai *data testing* serta pengujian ketiga menggunakan 90% sebagai *data training* dan 10% sebagai *data testing*.

Langkah-langkah penelitian yang dilakukan dapat digambarkan dalam Gambar 1.



Gambar 1. Alur Penelitian

Penelitian dilakukan menggunakan bahasa pemrograman Python 3 pada platform Anaconda.

3. HASIL DAN PEMBAHASAN

Dataset yang digunakan merupakan data hasil *scrapping* yang telah diseleksi dan dilakukan pelabelan data. Kemudian dilakukan *preprocessing* dengan langkah-langkah.

1. *Stopword Removal*

Merupakan langkah untuk melakukan filter terhadap *dataset* untuk mengurangi penggunaan kata yang mempunyai frekuensi kemunculan yang tinggi.

2. *Punctuation Filter*

Punctuation filter merupakan proses untuk menghilangkan tanda baca ataupun simbol yang terdapat pada dataset. Tanda baca atau simbol dihapus karena tidak mempengaruhi hasil sentimen analisis.

3. *Stemming*

Merupakan proses untuk mengembalikan kata yang digunakan ke kata dasarnya [4][5]. Proses *stemming* pada penelitian ini menggunakan *library* sastrawi.

Setelah melakukan *preprocessing* dengan melakukan *stopword removal*, *punctuation filter* serta *stemming*, dilakukan pembobotan kata dengan melakukan perhitungan TF-IDF. Secara umum model matematika yang digunakan untuk melakukan perhitungan TF-IDF [21] adalah sebagai berikut:

$$tf_i = \frac{freq_i(d_j)}{\sum_{i=1}^k freq_i(d_j)} \quad (3)$$

$$idf_i = \log\left(\frac{|D|}{|\{d:t_i \in d\}|}\right) \quad (4)$$

$$(tf - idf)_{ij} = tf_i(d_j) \times idf_i \quad (5)$$

Keterangan:

$freq_i(d_j)$: frekuensi term (i) pada dokumen (d)

$|D|$: jumlah dokumen

$d: t_i$: dokumen yang mengandung term (i)

Pada penelitian ini perhitungan TF-IDF dilakukan dengan memanfaatkan fungsi *Tfidfvectorizer* yang telah disediakan oleh *scikit learn library*.

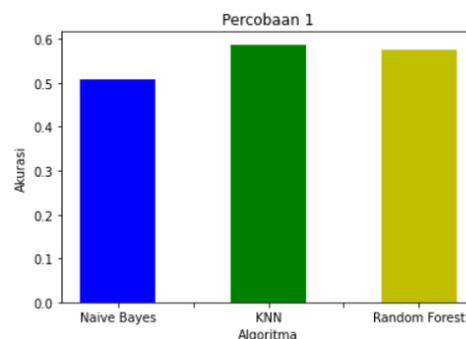
Setelah nilai TF-IDF didapatkan, dilakukan percobaan untuk mengetahui tingkat akurasi dari 3 algoritma yang digunakan, serta membandingkannya. Proses uji coba dilakukan dengan menggunakan modul *Naïve Bayes*, *KNN* dan *Random Forest* yang telah disediakan oleh *scikit learn library*.

1. Percobaan Pertama

Pada percobaan pertama *dataset* yang digunakan dibagi atas 50% sebagai *data training* dan 50% sebagai *data testing*. Hasil dari ujicoba pertama dapat dilihat pada [Tabel 1](#) dan [Gambar 2](#) di bawah ini.

Tabel 1. Hasil Percobaan Pertama

Algoritma	Akurasi
<i>Naïve Bayes</i>	51%
<i>K-Nearest Neighbors</i>	59%
<i>Random Forest</i>	57%



Gambar 2. Grafik Percobaan 1

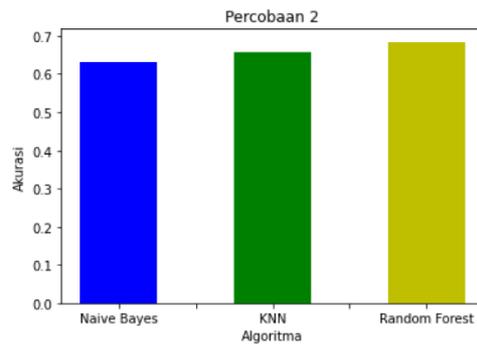
Pada percobaan pertama tingkat akurasi tertinggi didapatkan pada algoritma *KNN* yang mencapai 59% sedangkan tingkat akurasi paling rendah pada algoritma *Naïve Bayes* sebesar 51%.

2. Percobaan Kedua

Pada percobaan kedua *dataset* yang digunakan dibagi atas 75% sebagai *data training* dan 25% sebagai *data testing*. Hasil dari ujicoba kedua dapat dilihat pada [Tabel 2](#) dan [Gambar 3](#) di bawah ini.

Tabel 2. Hasil Percobaan Kedua

Algoritma	Akurasi
<i>Naïve Bayes</i>	63%
<i>K-Nearest Neighbors</i>	66%
<i>Random Forest</i>	68%



Gambar 3. Grafik Percobaan 2

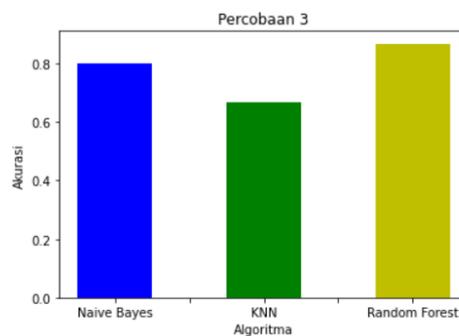
Pada percobaan kedua tingkat akurasi tertinggi didapatkan pada algoritma *Random Forest* yang mencapai 68% sedangkan tingkat akurasi paling rendah pada algoritma *Naive Bayes* sebesar 63%.

3. Percobaan Ketiga

Pada percobaan ketiga *dataset* yang digunakan dibagi atas 90% sebagai *data training* dan 10% sebagai *data testing*. Hasil dari ujicoba ketiga dapat dilihat pada [Tabel 3](#) dan [Gambar 4](#) di bawah ini.

Tabel 3. Hasil Percobaan Ketiga

Algoritma	Akurasi
<i>Naive Bayes</i>	80%
<i>K-Nearest Neighbors</i>	67%
<i>Random Forest</i>	87%



Gambar 4. Grafik Percobaan 3

Pada percobaan ketiga tingkat akurasi tertinggi didapatkan pada algoritma *Random Forest* yang mencapai 87% sedangkan tingkat akurasi paling rendah pada algoritma *Naive Bayes* sebesar 67%.

Secara umum jumlah *data training* yang lebih banyak akan menyebabkan meningkatnya tingkat akurasi dari model algoritma yang digunakan, hal ini dikarenakan algoritma akan memiliki semakin banyak variasi pengetahuan yang berasal dari dataset. Dengan semakin banyak *data training* yang digunakan akan dapat meningkatkan kemampuan dari algoritma dalam melakukan klasifikasi.

Dari ketiga ujicoba yang dilakukan, secara umum peningkatan jumlah *data training* dapat meningkatkan akurasi dari algoritma yang diujicoba. Pengecualian

pada algoritma KNN, percobaan kedua dan ketiga tingkat akurasi relatif tidak mengalami peningkatan.

Hasil percobaan menunjukkan algoritma KNN mempunyai tingkat akurasi tertinggi pada percobaan pertama sedangkan *Random Forest* mempunyai tingkat akurasi tertinggi pada percobaan kedua dan ketiga.

Dari tiga kali percobaan yang dilakukan, perbedaan tingkat akurasi dari ketiga algoritma relatif tidak terlalu jauh, kecuali algoritma KNN pada percobaan ketiga memiliki selisih tingkat akurasi mencapai 13% dibandingkan algoritma *Naïve Bayes* dan selisih akurasi mencapai 20% dibandingkan dengan algoritma *Random Forest*.

4. KESIMPULAN

Dari tiga kali percobaan yang dilakukan algoritma *Random Forest* menunjukkan tingkat akurasi terbaik pada percobaan kedua sebesar 68% dengan *data training* 75% dari *dataset* dan percobaan ketiga sebesar 87% dengan *data training* 90% dari *dataset*. Sehingga algoritma tersebut dapat dipertimbangkan untuk digunakan dalam melakukan klasifikasi sentimen masyarakat terhadap BPJS Kesehatan pada media Twitter.

Selanjutnya pengujian dapat dilakukan dengan menggunakan *dataset* yang lebih besar untuk menilai efektifitas algoritma tersebut atau dengan menggunakan algoritma lain sebagai pembandingan tingkat akurasi.

REFERENSI

- [1] A. Rosadi *et al.*, “Analisis Sentimen Berdasarkan Opini Pengguna pada Media Twitter Terhadap BPJS Menggunakan Metode Lexicon Based dan *Naïve Bayes* Classifier,” *J. Ilm. Komputasi*, vol. 20, no. 1, pp. 39–52, 2021, doi: 10.32409/jikstik.20.1.401.
- [2] F. Faturohman, B. Irawan, and C. Setianingsih, “Analisis Sentimen Pada Bpjs Kesehatan Menggunakan Recurrent Neural Network,” *e-Proceeding Eng.*, vol. 7, no. 2, pp. 4545–4552, 2020.
- [3] M. Nurmalasari, N. Anggita Temesvari, S. Ni, and matul Maula, “Analisis Sentimen terhadap Opini Masyarakat dalam Penggunaan Mobile-JKN untuk Pelayanan BPJS Kesehatan Tahun 2019,” *Heal. Inf. Manag. J. ISSN*, vol. 8, no. 1, pp. 2655–9129, 2020.
- [4] R. yana Yanis and A. Iriani, “Sentiment Analysis of Bpjs Kesehatan Services To Smk Eklesia and Bina Insani Jailolo Teachers,” *J. Terap. Teknol. Inf.*, vol. 2, no. 2, pp. 113–122, 2018, doi: 10.21460/jutei.2018.22.105.
- [5] P. D. Wulandari, U. Enri, and A. Primajaya, “Analisis Sentimen Terhadap Kenaikan Iuran Bpjs Kesehatan Pada Twitter Menggunakan *Naïve Bayes* Classifier,” *J. Ilmu Komput. dan ...*, vol. 1, no. 2, pp. 18–22, 2020, [Online]. Available: <http://ejournal.uhb.ac.id/index.php/IKOMTI/article/download/575/483>.
- [6] T. W. Putra, A. Triayudi, and Andrianingsih, “Analisis Sentimen Pembelajaran Daring menggunakan Metode *Naïve Bayes*, KNN, dan Decision Tree,” *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 6, no. 1, 2021.
- [7] E. E. Pratama, “Klasifikasi Kategori Permasalahan Akademik Mahasiswa Comdev Universitas Tanjungpura Menggunakan Algoritma *Naive Bayes*,” *J. Edukasi dan Penelit. Inform.*, vol. 7, no. 2, p. 106, 2021, doi: 10.26418/jp.v7i2.47199.
- [8] S. Rahmawati and M. Habibi, “Public Sentiments Analysis about Indonesian Social Insurance Administration Organization on Twitter,” *IJID (International J. Informatics Dev.)*, vol. 9, no. 2, pp. 87–93, 2020, doi: 10.14421/ijid.2020.09205.
- [9] S. P. Kristanto, J. A. Prasetyo, and E. Pramana, “*Naive Bayes* Classifier on Twitter Sentiment Analysis BPJS of HEALTH,” *Proc. - 2019 2nd Int. Conf. Comput. Informatics*

- Eng. Artif. Intell. Roles Ind. Revolut. 4.0, IC2IE 2019*, pp. 24–28, 2019, doi: 10.1109/IC2IE47452.2019.8940900.
- [10] N. L. W. S. R. Ginantra *et al.*, *Data Mining dan Penerapan Algoritma*. Yayasan Kita Menulis, 2021.
- [11] L. S. Riza, A. D. Pertiwi, E. F. Rahman, Munir, and C. U. Abdullah, “Question generator system of sentence completion in TOEFL using NLP and K-nearest Neighbor,” *Indones. J. Sci. Technol.*, vol. 4, no. 2, pp. 294–311, 2019, doi: 10.17509/ijost.v4i2.18202.
- [12] Z. Chen, L. J. Zhou, X. Da Li, J. N. Zhang, and W. J. Huo, “The Lao text classification method based on KNN,” *Procedia Comput. Sci.*, vol. 166, pp. 523–528, 2020, doi: 10.1016/j.procs.2020.02.053.
- [13] E. H. Muktafin and P. Kusriani, “Sentiments analysis of customer satisfaction in public services using *K-Nearest Neighbors* algorithm and natural language processing approach,” *Telkonnika (Telecommunication Comput. Electron. Control.*, vol. 19, no. 1, pp. 146–154, 2021, doi: 10.12928/TELKOMNIKA.V19I1.17417.
- [14] L. E. Peterson, “K-nearest neighbor,” *Scholarpedia*, 2009. http://scholarpedia.org/article/K-nearest_neighbor (accessed Oct. 25, 2021).
- [15] P. Karthika, R. Murugeswari, and R. Manoranjithem, “Sentiment Analysis of Social Media Network Using *Random Forest* Algorithm,” *IEEE Int. Conf. Intell. Tech. Control. Optim. Signal Process. INCOS 2019*, pp. 1–5, 2019, doi: 10.1109/INCOS45849.2019.8951367.
- [16] M. Yasen and S. Tedmori, “Movies reviews sentiment analysis and classification,” *2019 IEEE Jordan Int. Jt. Conf. Electr. Eng. Inf. Technol. JEEIT 2019 - Proc.*, pp. 860–865, 2019, doi: 10.1109/JEEIT.2019.8717422.
- [17] D. Alita and A. R. Isnain, “Pendeteksian Sarkasme pada Proses Analisis Sentimen Menggunakan *Random Forest* Classifier,” *J. Komputasi*, vol. 8, no. 2, pp. 50–58, 2020, doi: 10.23960/komputasi.v8i2.2615.
- [18] “Twitter API,” *Twitter, Inc*, 2021. <https://developer.twitter.com/en/docs/twitter-api> (accessed Oct. 25, 2021).
- [19] T. N. Wijaya, Rini Indriati, and M. N. Muzaki, “Analisis Sentimen Opini Publik Tentang Undang- Undang Cipta Kerja Pada Twitter,” *Jambura J. Electr. Electron. Eng.*, vol. 3, pp. 78–83, 2021.
- [20] B. Stecanella, “Understanding TF-ID: A Simple Introduction,” *MonkeyLearn*, 2019. [https://monkeylearn.com/blog/what-is-tf-idf/#:~:text=TF-IDF \(term frequency-,across a set of documents](https://monkeylearn.com/blog/what-is-tf-idf/#:~:text=TF-IDF%20(term%20frequency%20across%20a%20set%20of%20documents).
- [21] N. K. Widyasanti, I. K. G. Darma Putra, and N. K. Dwi Rusjyanthi, “Seleksi Fitur Bobot Kata dengan Metode TFIDF untuk Ringkasan Bahasa Indonesia,” *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, vol. 6, no. 2, p. 119, 2018, doi: 10.24843/jim.2018.v06.i02.p06.