

Analisis Komentar *Toxic* Terhadap Informasi COVID-19 pada *YouTube* Kementerian Kesehatan Menggunakan Metode *Naïve Bayes Classifier*

Vira Nindya Romadina*, Oktalia Juwita**, Priza Pandunata***

* ** Sistem Informasi, Fakultas Ilmu Komputer, Universitas Jember

*** Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Jember

*nindyavira@gmail.com, **oktalia@unej.ac.id, *** priza@unej.ac.id

ABSTRACT (10 PT)

Countries around the world were shocked by the outbreak of a new virus in 2020, which quickly transmitted and attacks humans of all ages. The virus is COVID-19. The government has advised through social media to stay at home and get vaccinated. YouTube has become a platform for the government, especially the Ministry of Health, to share public information in the COVID-19's pandemic. Public can put their comments on video uploaded by the Ministry of Health. An analysis of comments is needed so that the information in comments can be useful for those who read and evaluated by the government so that they can provide information that the public can understand. In analyzing toxic comments, it can used text mining. And one of that is the Naïve Bayes Classifier. This study uses the Naïve Bayes Classifier method to determine the results of the analysis. Measuring the value of accuracy, this study using the Confusion Matrix evaluation. From the final result, the highest accuracy value is in the comparison of 90%:10% with an accuracy value of 80%. And from the results of the analysis, the most toxic words used are the words 'dead', 'business', 'public' and 'fool'. From the results show that there are still many people who do not believe in the existence of COVID-19 and think that vaccines can cause death in people who are vaccinated.

Keyword: Toxic Comment, COVID-19, Naïve Bayes Classifier, YouTube

1. Pendahuluan

Negara di seluruh dunia dikagetkan oleh adanya wabah virus baru pada tahun 2020. Virus ini disebut dengan virus *corona* atau disebut juga dengan COVID-19. Virus ini menjangkit negara-negara yang ada di dunia [1]. Virus *corona* adalah virus menyebar dan dapat dengan mudah menginfeksi siapa saja tanpa memandang usia [2]. Pada awal kemunculannya, virus ini diperkirakan merupakan penyakit pneumonia karena gejalanya yang persis seperti flu biasa. Gejala-gejala yang tampak di antaranya demam, batuk, sesak nafas, kelelahan dan nafsu makan hilang. Di Indonesia sudah terdapat hampir 2,8 juta kasus dengan sekitar 73.000 kematian akibat COVID-19 [3]. Kekhawatiran pemerintah terhadap COVID-19 sehingga menganjurkan beberapa tindakan dengan tidak keluar rumah, melakukan karantina, mengikuti vaksinasi serta himbauan melalui media sosial untuk memberikan informasi demi menghambat dan menghindari penyebaran virus lebih lanjut [4].

Media sosial adalah media *online* yang mana para *user* di dalamnya dapat saling berinteraksi [5]. *YouTube* adalah salah satu sosial media dan merupakan platform berbagi video paling populer di dunia [6]. Pemerintah dalam kasus ini melalui kanal *YouTube* Kementerian Kesehatan membagikan banyak informasi medis dan edukasi terkait COVID-19 [7]. Pengguna dapat menuliskan pendapatnya pada setiap video yang diunggah ke *YouTube* melalui fitur yaitu kolom komentar [8]. Tetapi, beberapa orang cenderung berkomentar negatif dan mengandung *toxic* terhadap video yang ada di *YouTube*.

Definisi *toxic comment* sendiri adalah komentar yang mengandung bahasa kasar, kalimat membenci, tidak sopan, hingga mempermalukan orang lain yang ada di media sosial. Oleh sebab itu, dalam pemanfaatan fitur komentar dibutuhkan metode analisis yang tepat agar segala informasi dalam penyampaian pendapat melalui komentar dapat berguna bagi orang yang membaca [9].

Machine Learning adalah istilah yang digunakan untuk konsep *software* yang bekerja secara otomatis dalam memecahkan suatu masalah [10]. Dalam menganalisis komentar pada video *YouTube* dengan *machine learning* diperlukan adanya algoritma. Dalam menganalisis komentar pada penelitian ini algoritma yang digunakan adalah algoritma klasifikasi yang digolongkan ke dalam kategori *supervised learning* yaitu algoritma *Naïve Bayes* [11]. Berdasarkan penelitian yang dilakukan oleh Renaldy Permana Sidiq, Budi Arif Dermawan dan Yuyun Umaidah dalam jurnalnya menjelaskan bahwa metode *Naïve Bayes* sederhana dalam

implementasinya tetapi memiliki performa yang cepat dan efektif. Kemudian Rian Ardianto dalam penelitiannya mendapatkan hasil bahwa *Naïve Bayes* memperoleh nilai akurasi lebih baik daripada *Support Vector Machine*. Oleh karena itu, penelitian ini menggunakan metode *Naïve Bayes Classifier* dengan tujuan untuk menganalisis komentar *toxic* terhadap video informasi COVID-19 dan objek yang digunakan adalah kanal *YouTube* milik pemerintah yaitu Kementerian Kesehatan. Selain itu, dalam penelitian ini digunakan evaluasi *confusion matrix* untuk mengukur seberapa baik tingkat akurasi dalam pengimplementasian metode *Naïve Bayes* serta dilakukan visualisasi dengan menggunakan *wordcloud* agar dapat diketahui kata dengan makna *toxic* apa saja yang paling banyak muncul dan merupakan kata dominan.

2. Metode Penelitian

Penelitian ini merupakan penelitian kuantitatif dengan menggunakan pendekatan eksperimen [12]. Tahap awal yang dilakukan adalah pengumpulan data dengan melakukan *crawling* untuk mendapatkan *dataset*. Kemudian dilanjutkan dengan proses *pre-processing*, TF-IDF atau pembobotan kata, penerapan *Naïve Bayes* dengan melakukan pembagian data latih dan data uji dan tahap akhir yaitu uji performansi untuk mengetahui nilai akurasi, *precision*, *recall* dan *F-1 Score*

2.1. Pengumpulan Data

Pengumpulan data dilakukan dengan cara *crawling/scraping* komentar pada video akun *YouTube* Kementerian Kesehatan. Proses ini menggunakan bantuan *website* dengan cara melakukan *copy paste* terhadap *link* video yang dikehendaki.

2.2. Pre-Processing

Pre-processing adalah tahap mempersiapkan data mentah sebelum masuk dan diproses ke tahap selanjutnya. Tahapan *pre-processing* ini menggunakan modul *ReGex (Regular Expression)*. *Regular expression* adalah sebuah metode yang dipakai dalam kodifikasi pencarian. Tahapan *preprocessing* tersebut diantaranya:

1. *Cleansing*
Tahapan ini merupakan proses pembersihan data. Tujuan dari *cleansing* yaitu agar data bersih dari *noise*. Karakter yang dibersihkan berupa *link url*, *symbol*, *username*, tanda baca dan *e-mail* [13].
2. *Case Folding*
Tahapan mengubah bentuk teks menjadi sama yaitu *lower case* (huruf kecil).
3. *Tokenizing*
Tahapan ini mengubah atau memisah teks kalimat menjadi kata [14].
4. *Filtering/Stopword Removal*
Stopword Removal merupakan tahapan pengambilan kata-kata penting dari hasil proses *tokenizing* dan menghilangkan karakter ataupun kata yang tak bermakna dan dianggap tidak berperan penting dalam proses analisis ke tahap selanjutnya [15].
5. *Stemming*
Tahapan mengubah kata yang terdapat imbuhan kembali ke kata semula yaitu kata tak berimbuhan atau kata dasar.

2.3. Pembobotan Kata TF-IDF

Proses ini memberikan bobot nilai pada masing-masing kata dalam sebuah kalimat dengan cara mengalikan *Term Frequency* dan *Inverse Document Frequency*.

1. *Weighting Term Frequency*

$$W_{tf_{t,d}} = \begin{cases} 1 + \log_{10} tf_{t,d}, & \text{if } tf_{t,d} > 0 \\ 0, & \text{lainnya} \end{cases} \quad (1)$$

2. *Inverse Document Frequency*

$$Idf_t = \log_{10} \frac{N}{df_t} \quad (2)$$

3. *Term Frequency-Inverse Document Frequency (TF-IDF)*

$$W_{t,d} = W_{tf_{t,d}} \times Idf_t \quad (3)$$

2.4. Penerapan Model Naïve Bayes

Model *Naïve Bayes* merupakan algoritma yang didalamnya menyiratkan independensi prediksi guna memecahkan permasalahan klasifikasi sebuah variabel tertentu dimana proses klasifikanya menggunakan probabilistik *method* dan juga statistik [16]. Secara umum bentuk teorema *Bayes* adalah sebagai berikut:

$$P(A|X) = \frac{P(X|A)}{P(X)} \cdot P(A)$$

(4)

Pada penelitian ini, digunakan model *Naïve Bayes* untuk proses pengujian model. Dalam tahap ini perlu dilakukan pembagian rasio data antara data *training* dan juga data *testing*. Perbandingan rasio yang akan digunakan yaitu 60:40, 70:30, 80:20, 90:10.

2.5. Evaluation / Uji Performansi

Uji performansi dilakukan dengan menggunakan *confusion matrix*. Tahapan ini dilakukan untuk menghitung nilai akurasi, *precision*, *recall*, dan juga *F1-Score*.

3. Hasil dan Pembahasan

Bab ini membahas hasil dari implementasi metode yang digunakan. Tahap awal yang dilakukan adalah proses *crawling* untuk mendapatkan *dataset*.

3.1. Hasil Crawling Data

Dalam memperoleh *dataset*, perlu dilakukan pengambilan data melalui proses *crawling*. Pada penelitian ini, proses *crawling* dilakukan dengan menggunakan bantuan *website* yaitu “*app.coberry.com*”. Link *YouTube* yang dikehendaki untuk di *crawling* komentarnya di *copy paste*. Setelah itu melakukan centang terhadap kolom apa saja yang ingin ditampilkan pada hasil *crawling* yang berupa *file CSV*. Berikut adalah contoh dari hasil *crawling* data komentar yang masih berbentuk data kasar ditampilkan pada tabel di bawah ini:

Tabel 1. Hasil Crawling Dataset

<i>Comment ID</i>	<i>Author Name</i>	<i>Comment Content</i>	<i>Comment URL</i>
UgxzgpSpCC Pu4JihiWB4A aABAg	Nona Craft	Terimakasih dok atas penjelasannya ,semoga bermanfaat bagi kita semua ,untuk mengusir dan mencegah Covid 19	https://www.youtube.com/watch?v=4WvAsJtQPbQ&lc=UgxzgpSpCCPu4JihiWB4AaABAg
UgyA4bHPU m2XS5gWTz 54AaABAg	Salman Suck	gak banyak yg nonton sekarang wkww banyak yg mati setelah di faksin masih aja kasih doktrin 🙄	https://www.youtube.com/watch?v=4WvAsJtQPbQ&lc=UgyA4bHPUm2XS5gWTz54AaABAg

3.2. Hasil implementasi pre-processing

Proses ini dibutuhkan supaya *dataset* bersih dari *noise* dan siap di olah pada saat masuk ke tahap selanjutnya. Dibawah ini adalah tahapan-tahapan *pre-processing* yang telah dilakukan antara lain:

a. Hasil *Cleansing*

Proses ini membutuhkan impor sebuah modul yang bernama *ReGex* atau *Regular Expression* dengan kode “import re”. Modul *ReGex* digunakan untuk mengecek apakah sebuah *string* berisikan pola pencarian tertentu yang telah ditentukan sebelumnya. Berikut hasil *pre-processing* pada tahap *cleansing* disajikan dalam tabel 2.

Tabel 2. Hasil Cleansing

Input	Output
Terimakasih dok atas penjelasannya ,semoga bermanfaat bagi kita semua ,untuk mengusir dan mencegah Covid 19	Terimakasih dok atas penjelasannya semoga bermanfaat bagi kita semua untuk mengusir dan mencegah Covid

b. Hasil Case Folding

Pada penelitian ini, *case folding* yang dilakukan adalah mengubah semua huruf menjadi *lower case* (huruf kecil). Berikut hasil tahapan *case folding* yang disajikan dalam tabel di bawah ini.

Tabel 3. Hasil Case Folding

Input	Output
Terimakasih dok atas penjelasannya semoga bermanfaat bagi kita semua untuk mengusir dan mencegah Covid	terimakasih dok atas penjelasannya semoga bermanfaat bagi kita semua untuk mengusir dan mencegah covid

c. Hasil Tokenizing

Tahapan ini menggunakan data yang merupakan hasil dari tahap *pre-processing* sebelumnya yaitu *case folding*. Berikut hasil tahapan *tokenizing* yang disajikan dalam tabel di bawah ini.

Tabel 4. Hasil Tokenizing

Input	Output
terimakasih dok atas penjelasannya semoga bermanfaat bagi kita semua untuk mengusir dan mencegah covid	["terimakasih", "dok", "atas", "penjelasannya", "semoga", "bermanfaat", "bagi", "kita", "semua", "untuk", "mengusir", "dan", "mencegah", "covid"]

d. Hasil Stopwords/Filtering

Hasil implementasi *pre-processing* pada tahap *tokenizing* kemudian digunakan pada tahap *stopwords removal* atau biasa disebut juga dengan *filtering*. Berikut hasil tahapan *tokenizing* yang disajikan dalam tabel di bawah ini.

Tabel 5. Hasil Stopwords

Input	Output
["terimakasih", "dok", "atas", "penjelasannya", "semoga", "bermanfaat", "bagi", "kita", "semua", "untuk", "mengusir", "dan", "mencegah", "covid"]	["terimakasih", "dok", "penjelasannya", "semoga", "bermanfaat", "mengusir", "mencegah", "covid"]

e. Hasil Stemming

Proses ini menggunakan *library* sastrawi. Data dari hasil proses *stemming* ini kemudian disimpan dalam bentuk *csv* dan merupakan data yang sudah siap digunakan pada tahap selanjutnya.

Tabel 6. Hasil Stemming

Input	Output
[“terimakasih”, “dok”, “penjelasannya”, “semoga”, “bermanfaat”, “mengusir”, “mencegah”, “covid”]	[“terimakasih”, “dok”, “jelas”, “moga”, “manfaat”, “usir”, “cegah”, “covid”]

3.3. Hasil Penyusunan Model Naïve Bayes

Setelah melalui tahapan *pre-processing* dan juga pembobotan kata menggunakan TF-IDF, dilanjutkan dengan proses penyusunan model yang dibagi menjadi empat percobaan dengan perbandingan rasio yaitu 90:10, 80:20, 70:30 dan 60:40. Pada pembagian data *testing* untuk perbandingan 90:10, dengan data *training* yaitu 90% dari total *dataset* yang digunakan sebanyak 1516 berjumlah 1364 dan data *testing* yaitu 10% dari total 1516 *dataset* berjumlah 152 di dapatkan nilai akurasi sebesar 0.80, nilai *precision* sebesar 1.0, nilai *recall* sebesar 0.03 dan nilai *f1-score* sebesar 0.06. untuk hasil *running* terhadap kode pada *google colabs* disajikan dalam gambar 1. sebagai berikut.

	precision	recall	f1-score	support
0	0.79	1.00	0.89	120
1	1.00	0.03	0.06	32
accuracy			0.80	152
macro avg	0.90	0.52	0.47	152
weighted avg	0.84	0.80	0.71	152

Gambar 1. Hasil Penyusunan Model 10:90

Kemudian, pada pembagian data *testing* untuk perbandingan 80:20, dengan data *training* yaitu 80% dari total *dataset* yang digunakan sebanyak 1516 berjumlah 1212 dan data *testing* yaitu 20% dari total 1516 *dataset* berjumlah 304 di dapatkan nilai akurasi sebesar 0.74, nilai *precision* sebesar 1.0, nilai *recall* sebesar 0.01 dan nilai *f1-score* sebesar 0.02. untuk hasil *running* terhadap kode pada *google colabs* disajikan dalam gambar 2. sebagai berikut.

	precision	recall	f1-score	support
0	0.74	1.00	0.85	224
1	1.00	0.01	0.02	80
accuracy			0.74	304
macro avg	0.87	0.51	0.44	304
weighted avg	0.81	0.74	0.63	304

Gambar 2. Hasil Penyusunan Model 20:80

pada pembagian data *testing* untuk perbandingan 70:30, dengan data *training* yaitu 70% dari total *dataset* yang digunakan sebanyak 1516 berjumlah 1061 dan data *testing* yaitu 30% dari total 1516 *dataset* berjumlah 455 di dapatkan nilai akurasi sebesar 0.75, nilai *precision* sebesar 1.0, nilai *recall* sebesar 0.008 dan nilai *f1-score* sebesar 0.017. untuk hasil *running* terhadap kode pada *google colabs* disajikan dalam gambar 3. sebagai berikut.

	precision	recall	f1-score	support
0	0.75	1.00	0.86	342
1	1.00	0.01	0.02	113
accuracy			0.75	455
macro avg	0.88	0.50	0.44	455
weighted avg	0.81	0.75	0.65	455

Gambar 3. Hasil Penyusunan Model 30:70

Dan terakhir, pada pembagian data *testing* untuk perbandingan 60:40, dengan data *training* yaitu 60% dari total *dataset* yang digunakan sebanyak 1516 berjumlah 909 dan data *testing* yaitu 40% dari total 1516 *dataset* berjumlah 607 di dapatkan nilai akurasi sebesar 0.76, nilai *precision* sebesar 1.0, nilai *recall* sebesar 0.006 dan nilai *f1-score* sebesar 0.013. untuk hasil *running* terhadap kode pada *google colabs* disajikan dalam gambar 4. sebagai berikut.

	precision	recall	f1-score	support
0	0.76	1.00	0.87	463
1	1.00	0.01	0.01	144
accuracy			0.76	607
macro avg	0.88	0.50	0.44	607
weighted avg	0.82	0.76	0.66	607

Gambar 4. Hasil Penyusunan Model 40:60

Pada *split* data antara data *training* dan data *testing* dilakukan pengacakan data atau *random state* sebesar bilangan bulat yaitu 42 dengan tujuan agar nilai akurasi tetap sama dan pembagian *dataset* menjadi konsisten setiap kali model dijalankan.

3.4. Evaluasi Model

Evaluasi model dilakukan dengan menggunakan *confusion matrix*. Tahap ini dilakukan untuk melihat seberapa baik model dalam melakukan klasifikasi dengan mengacu pada seberapa baik nilai akurasi. Data *training* dan data *testing* dibagi ke dalam 4 percobaan yaitu 90:10, 80:20, 70:30, 40:60. Selain nilai akurasi, dalam penelitian ini dilakukan juga penghitungan terhadap nilai *precision*, *recall*, dan juga *f-measure*. Untuk hasil dari evaluasi model dapat dilihat dalam tabel berikut.

Tabel 11. Hasil Confusion Matrix

Nilai	Perbandingan 90%:10%	Perbandingan 80%:20%	Perbandingan 70%:30%	Perbandingan 60%:40%
Akurasi	0.80	0.74	0.75	0.76
<i>Precision</i>	1.0	1.0	1.0	1.0
<i>Recall</i>	0.031	0.012	0.008	0.006
<i>F-Measure</i>	0.06	0.02	0.017	0.013

Berdasarkan empat percobaan yang dilakukan pada evaluasi model, di dapatkan hasil akhir terbaik pada klasifikasi pembagian data *training* dan data *testing* 90%:10% yaitu dengan nilai akurasi sebesar 80%. Nilai akurasi akan menghasilkan nilai terbaik apabila penggunaan data *training* lebih banyak. Metode *Naïve Bayes* bekerja dengan sangat baik pada perbandingan data *training* dan juga data *testing* 90%:10% dengan nilai akurasi sebesar 80%. Proses pada tahapan *pre-processing* sangat berpengaruh terhadap hasil dari nilai akurasi agar *dataset* lebih bersih saat memasuki tahap selanjutnya sehingga nilai akurasi menjadi lebih baik lagi dari sebelumnya. Dan hasil penerapan *confusion matrix* disajikan pada gambar 5 di bawah ini.

4. Kesimpulan

Pada analisis komentar *toxic* terhadap video informasi COVID-19 pada akun *YouTube* Kementerian Kesehatan dengan mengimplementasikan metode *Naïve Bayes Classifier*, hasil yang diperoleh yaitu cukup baik pada semua pembagian keempat percobaan berdasarkan nilai akurasi. Metode *Naïve Bayes* bekerja dengan sangat baik pada perbandingan data *training* dan juga data *testing* 90%:10% dengan nilai akurasi sebesar 80%. Pada komentar yang berlabel kan *toxic*, kata dominan atau kata yang sering muncul pada komentar yang menunjukkan *toxic* di antaranya yaitu 'mati', 'bisnis', dan 'bodoh'. Kata tersebut sering muncul pada komentar yang berlabel kan *toxic*. Pemerintah terkhusus Kementerian Kesehatan dalam hal ini dapat memberikan edukasi yang lebih meyakinkan terkait adanya virus COVID-19 agar masyarakat bisa percaya terhadap penularan virus tersebut dan mengikuti anjuran untuk melaksanakan vaksinasi untuk kekebalan tubuh.

DAFTAR PUSTAKA

- [1] R. Sebayang, "Awat! WHO akhirnya tetapkan Corona darurat global," *CNBC Indonesia*, 2020.
- [2] D. R. Beniac, A. Andonov, E. Grudeski, and T. F. Booth, "Architecture of the SARS coronavirus prefusion spike," *Nat Struct Mol Biol*, vol. 13, no. 8, pp. 751–752, 2006, doi: 10.1038/nsmb1123.
- [3] G. Lele, "Concurrency as crisis decision-making governance: Lessons from Indonesia's response to the COVID-19 pandemic," *Regional & Federal Studies*, pp. 1–26, 2021.
- [4] M. Maulana Ibrahim and R. Edi Irawan, "Pengaruh Konten Media Sosial Terhadap Sumber Informasi Pandemi Covid-19," *Jurnal Representamen*, vol. 7, no. 02, 2021.
- [5] A. S. Cahyono, "Pengaruh media sosial terhadap perubahan sosial masyarakat di Indonesia," *Jurnal Publiciana*, vol. 9, no. 1, pp. 140–157, 2016.
- [6] M. Azak, K. Şahin, N. Korkmaz, and S. Yıldız, "YouTube as a source of information about COVID-19 for children: Content quality, reliability, and audience participation analysis," *J Pediatr Nurs*, no. xxxx, 2021, doi: 10.1016/j.pedn.2021.06.024.
- [7] H. Erdem and A. Sisik, "The reliability of bariatric surgery videos in YouTube platform," *Obes Surg*, vol. 28, no. 3, pp. 712–716, 2018.
- [8] R. Yacob, "Penggunaan Bahasa pada Kolom Komentar di Youtube: Studi Kajian Awal," *Seminar Bahasa dan Sastra Indonesia*, vol. 2, no. 1, pp. 169–174, 2019, [Online]. Available: <http://conference.unsri.ac.id/index.php/sembadra/article/view/1612>
- [9] J. Risch and R. Krestel, "Toxic Comment Detection in Online Discussions," pp. 85–109, 2020, doi: 10.1007/978-981-15-1216-2_4.
- [10] J. S. Suri *et al.*, "COVID-19 pathways for brain and heart injury in comorbidity patients: A role of medical imaging and artificial intelligence-based COVID severity classification: A review," *Comput Biol Med*, vol. 124, p. 103960, 2020.
- [11] F.-J. Yang, "An Implementation of Naive Bayes Classifier," in *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2018, pp. 301–306. doi: 10.1109/CSCI46756.2018.00065.
- [12] D. Jenderal Penguatan Riset dan Pengembangan *et al.*, "Kredo 4 (2021) KREDO: Jurnal Ilmiah Bahasa dan Sastra Terakreditasi Sinta 4 berdasarkan Keputusan EKSPERIMENTASI MODEL PEMBELAJARAN BERBASIS PROYEK UNTUK MENGOPTIMALKAN KEMAMPUAN MENULIS SISWA." [Online]. Available: <https://jurnal.umk.ac.id/index.php/kredo/index>
- [13] N. A. Sivi, *Analysis sentimen pada media sosial menggunakan metode naive bayes classifier tesis*. 2018.
- [14] P. Arsi, R. Wahyudi, and R. Waluyo, "Optimasi SVM Berbasis PSO pada Analisis Sentimen Wacana Pindah Ibu Kota Indonesia," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 2, pp. 231–237, Apr. 2021, doi: 10.29207/resti.v5i2.2698.
- [15] A. Fadjeri, K. Hidayat, and D. R. Handayani, "DETEKSI EMOSI PADA TEKS MENGGUNAKAN ALGORITMA NAÏVE BAYES," vol. 1, no. 2, pp. 1–4, 2021, doi: 10.53863/juristik.v1i02.365.
- [16] A. Jadhav, A. Pandita, A. Pawar, and V. Singh, "Classification of unstructured data using naïve bayes classifier and predictive analysis for RTI application," *An International Journal of Engineering & Technology*, vol. 3, no. 6, pp. 1–6, 2016.