INFORMATIKA
Jurnal Informatika, Manajemen dan Komputer, Vol. 17, No. 1, Mei 2025
eISSN : 2580-3042
pISSN : 1979-0694

IMPLEMENTASI METODE RANDOM FOREST PADA TEXT MINING UNTUK KLASIFIKASI SMS SPAM MENGGUNAKAN PYTHON

Idir Fitriyanto[1], Teuku Radillah [2*], Leonard Tambunan[3], Atika Fauziyyah[4]
Institut Teknologi Mitra Gama
Jl. Khayangan No 99 Kota Duri kode pos 28784
e-mail : idirfitriyanto45@gmail.com[1], t.radillah@gmail.com[2*]
tambunan.leonard81@gmail.com[3], atikafauziyyah224@gmail.com[4]

ABSTRAK

tPenggunaan metode *Random Forest* dalam bidang *text mining* telah menjadi fokus utama penelitian dalam upaya memerangi spam melalui pesan singkat (*SMS*). Penelitian ini mengusulkan sebuah pendekatanenggunakan metode Random Forest untuk klasifikasi *SMS* spam menggunakan bahasa pemrograman Python. Untuk proses pelatihan pada daset yang telah mendapatkan pelabelan untuk jenis SMS, yaitu 0 = SMS normal, 1 = SMS penipuan (spam), 2 = SMS promo dijadikan dataset pelatihan dan pengujian untuk mengukur kiinerja metode *Random Forest.* Adapun langkah awal untuk mengidentifikasi *SMS* spam dengan melibatkan pengumpulan data *SMS* dari berbagai sumber untuk membangun dataset yang representatif. Model *Random Forest* diimplementasikan menggunakan *library scikit-learn* dalam lingkungan *Python*. Proses pelatihan model melibatkan pembagian data menjadi set pelatihan dan set validasi untuk mengukur kinerja metode. Berbagai metrik evaluasi seperti akurasi, presisi, *recall*, dan *F1-score* digunakan untuk mengevaluasi kinerja metode tersebut. Hasil penelitian menunjukkan bahwa metode *Random Forest* mampu memberikan klasifikasi yang baik dalam membedakan antara SMS spam, SMS normal dan SMS promo dengan tingkat akurasi 89%, dengan hasil kalsifikasi presisi = 0,97%, recall = 0,83 %, dan *F1-score*= 0,89%. Implementasi metode *random forest* menunjukkan tingkat akurasi yang memuaskan dan mampu mengidentifikasi sebagian besar SMS spam dengan tingkat kesalahan yang rendah

.

***Kata Kunci**:* Klasifikasi SMS Spam, *Text Mining*, Metode *Random Forest*

ABSTRACT

The use of the Random Forest method in the field of text mining has become the focus of research in efforts to combat Short Message Service (SMS) spam. This research proposes a Random Forest approach to SMS spam classification using the Python programming language. For the training process, datasets that have been labelled for SMS types, namely 0 = normal SMS, 1 = fraudulent SMS (spam), 2 = promo SMS, are used as training and test datasets to measure the performance of the Random Forest method. The first step to identify SMS spam is to collect SMS data from different sources to build a representative dataset. The Random Forest model was implemented using the scikit-learn library in the Python environment. The model training process involves dividing the data into a training set and a validation set to measure the performance of the method. Various evaluation metrics such as accuracy, precision, recall and F1 score are used to evaluate the performance of the method. The research results show that the Random Forest method is able to provide a good classification in

INFORMATIKA
Jurnal Informatika, Manajemen dan Komputer, Vol. 17, No. 1, Mei 2025
eISSN : 2580-3042
pISSN : 1979-0694

distinguishing between spam SMS, normal SMS and promo SMS with an accuracy rate of 89%, with precision calculation results = 0.97%, recall = 0.83% and F1 score = 0.89%. The implementation of the random forest method shows a satisfactory level of accuracy and is able to identify most spam SMS with a low error rate.

Keywords: SMS Spam Classification, Text Mining, Random Forest Method

1. PENDAHULUAN

Text mining adalah proses ekstraksi informasi yang bermakna dan salah satu cabang dari data mining yang berkaitan dengan berguna dari teks(Virra et al., 2019). Text mining sering juga disebut bidang multidisiplin, yang melibatkan pencarian informasi, pemrosesan bahasa alami, mesin pembelajaran dan statistik(Elagamy et al., 2018). Tujuan utama dari text mining adalah untuk mengekstraksi pengetahuan yang berharga, pola-pola tersembunyi, atau informasi yang dapat diolah dari berbagai jenis teks, dokumen, artikel, *tweet*, *Short Message Service* (SMS), email, dan lain sebagainya. Dalam era digital yang semakin berkembang, pertumbuhan jumlah pesan teks atau SMS tidak terhindarkan, namun demikian, hal ini juga meningkatkan risiko munculnya SMS spam yang mengganggu. Dalam konteks ini, penggunaan teknik *text mining* dan *machine learning* menjadi semakin penting untuk mengidentifikasi dan mengklasifikasikan SMS spam dengan akurat.

Penelitian ini bertujuan untuk menerapkan metode *Random Forest* dalam kerangka *text mining* menggunakan bahasa pemrograman *Python*, dengan fokus pada klasifikasi SMS spam. Diharapkan bahwa penggabungan teknik *Random Forest* dengan algoritma *text mining* akan menghasilkan sistem yang efisien dan andal dalam memfilter pesan spam, memberikan kontribusi positif terhadap keamanan dan pengalaman pengguna dalam komunikasi melalui pesan teks SMS. Beberapa peneliti sebelumnya yang telah melakukan riset tentang Spam, yaitu penelitian yang berkaitan dengan jaringan syaraf tiruan graf semantik (Pan et al., 2022): yang mengkonversi dari email spam ke klasifikasi *graf*, selanjutnya penelitian tentang kerangka cerdas dalam mendeteksi SMS dan email spam(Maqsood et al., 2023), dan penelitian yang tentang Strategi dalam deteksi spam email yang Efisien menggunakan keputusan genetik pada pemrosesan pohon dengan fitur NLP(Ismail et al., 2022). SMS Spam merupakan pesan teks yang tidak diinginkan atau tidak diminta yang dikirim kepada pengguna melalui pesan singkat (SMS). Pesan-pesan ini seringkali berisi iklan, penipuan, atau tautan yang mencurigakan yang bertujuan untuk mempromosikan produk atau layanan, menipu pengguna untuk memberikan informasi pribadi, atau mengarahkan mereka ke situs web berbahaya(Karyawati et al., 2023). SMS spam sering dianggap mengganggu, dan dapat menghabiskan waktu dan sumber daya pengguna ketika mereka harus mengelola dan membersihkan kotak masuk mereka dari pesan-pesan yang tidak diinginkan tersebut(Sisodia et al., 2020). Penelitian klasifikasi SMS spam adalah penelitian yang bertujuan untuk mengembangkan model atau sistem yang dapat membedakan antara pesan SMS yang merupakan spam (tidak diinginkan) dan yang bukan spam (pesan yang sah atau diinginkan)(Amani Alzahrani, 2019). Selanjutnya, teknik *machine learning*, seperti algoritma klasifikasi, digunakan untuk melatih model dengan menggunakan fitur-fitur yang diekstraksi tersebut, sehingga model dapat mempelajari pola-pola dari data latih dan dapat mengklasifikasikan pesan-pesan baru ke dalam kategori spam atau bukan spam(A. B. Singh et al., 2022).

Metode *Random Forest* adalah salah satu teknik dalam *machine learning* yang digunakan untuk klasifikasi, regresi, dan berbagai tugas lainnya(Tambunan et al., 2021). Metode *Random forest* ini termasuk dalam kategori ensemble learning(Savargiv et al., 2021), di mana beberapa model pembelajaran mesin (disebut pohon keputusan) digabungkan

INFORMATIKA
Jurnal Informatika, Manajemen dan Komputer, Vol. 17, No. 1, Mei 2025
eISSN : 2580-3042
pISSN : 1979-0694

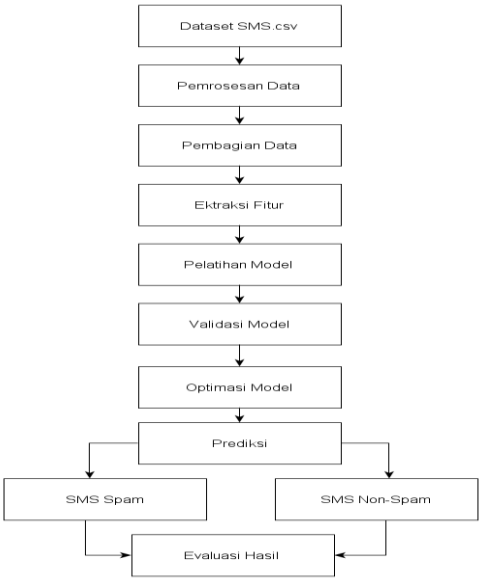
untuk membuat prediksi yang lebih akurat(Sejati et al., 2022). *Random Forest* dapat meningkatkan prediksi banyak metode supervised, terutama pohon keputusan(Kurniawan et al., 2023), selain itu juga memiliki beberapa keunggulan, seperti kemampuannya dalam meningkatkan akurasi prediksi dan efektif menangani dataset besar yang berisi parameter yang rumit(Khalik & Arifin, 2023).

Berikut ini adalah cara kerja metode *Random forest* :

1. Pohon Keputusan
   Random Forest terdiri dari banyak pohon keputusan yang dibangun secara acak. Setiap pohon keputusan pada dasarnya adalah serangkaian keputusan yang diambil berdasarkan fitur-fitur input untuk memprediksi output(Trishna et al., 2019).
2. *Bootsrap Sampling*
   Untuk setiap sampel bootstrap, sebuah pohon keputusan dibangun menggunakan algoritma pembangunan pohon keputusan, seperti CART *(Classification and Regression Trees*). Pada setiap simpul, subset fitur acak dipilih untuk mengevaluasi pemisahan data. Proses ini terus berlanjut hingga kriteria pemberhentian tertentu terpenuhi, misalnya, saat mencapai kedalaman maksimum atau ketika tidak ada peningkatan yang signifikan dalam pemisahan data
3. Pembangunan Pohon
   Setelah semua pohon keputusan dibangun, mereka digunakan untuk membuat prediksi untuk setiap data uji atau data baru(Parida et al., 2021). Pada tahap ini, setiap pohon menghasilkan prediksi berdasarkan fitur-fitur input
4. Voting
   Prediksi dari setiap pohon digabungkan menggunakan proses voting. Dalam klasifikasi, kelas yang paling sering muncul di antara prediksi-prediksi tersebut diambil sebagai prediksi akhir. Dalam regresi, nilai rata-rata dari prediksi semua pohon diambil sebagai prediksi akhir
5. Evaluasi Kinerja
   Akhirnya, kinerja model Random Forest dievaluasi menggunakan metrik-metrik seperti akurasi, presisi, *recall*, atau F1-*score*. Ini membantu memahami seberapa baik model telah melakukan dalam membuat prediksi dan membedakan antara kelas-kelas yang berbeda.

2. METODOLOGI PENELITIAN

Untuk mengklasifikasi SMS Spam diperlukan suatu metode yang dapat dijadikan acuan dalam menentukan SMS Spam. Implementasi metode *Random Forest* pada klasifikasi SMS spam menggunakan bahasa pemrograman *python* memiliki beberapa tahapan(S. N. Singh & Sarraf, 2020) yang dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Klasifikasi *SMS Spam*

Adapun penjelasan yang berkaitan dengan tahapan klasifikas SMS Spam sebagai berikut :

1. Dataset SMS.csv
   Dataset untuk kalsifikasi SMS spam berupa kumpulan SMS yang disimpan dalm format data csv.
2. Pemrosesan Data

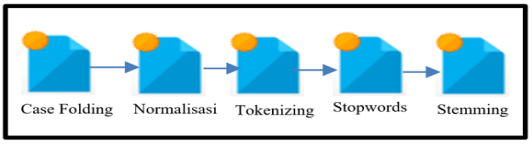
INFORMATIKA
Jurnal Informatika, Manajemen dan Komputer, Vol. 17, No. 1, Mei 2025
eISSN : 2580-3042
pISSN : 1979-0694

Data SMS yang sudah dikumpulkan dan di-label (spam atau bukan spam).

3. Pembagian Data
   Lakukan pembagian data menjadi dua subset: data pelatihan (*train*) dan data pengujian (*test*).
4. Ektraksi Fitur
   Ekstraksi fitur dari teks SMS, untuk mengukur pentingnya kata-kata dalam setiap SMS).
5. Pelatihan model *Random forest*
   Latih model Random Forest menggunakan data pelatihan.
6. Validasi model *Random forest*
   Validasi model menggunakan data pengujian untuk mengukur kinerjanya.
7. Optimasi model *Random forest*
   lakukan optimasi model dengan menyesuaikan parameter atau menggunakan teknik seperti validasi silang untuk meningkatkan kinerja model.
8. Prediksi
   Gunakan model yang telah dilatih untuk melakukan prediksi kelas (spam atau bukan spam) dari SMS yang belum terlihat sebelumnya.
9. Evaluasi Hasil
   Evaluasi performa model terakhir menggunakan metrik evaluasi yang sesuai, dan Simpan model yang dilatih untuk digunakan di masa depan pada data baru

2.1 Pre-Processing Data

Sebelum proses klasifikasi SMS spam menggunakan metode *Random Forest* dalam menentukan SMS spam, SMS normal dan SMS Promo, dilakukan, dataset yang digunakan terlebih dahulu dibersihkan(Tyasnurita & Hapsari, 2020). Adapun proses pre-processing data text SMS(Utami et al., 2020) tersebut dapat dilihat pada Gambar 2.Text-prosessing(Harjito et al., 2019) sebagai berikut :



Gambar 2. Text-prosessing

a. Case folding
Case folding adalah proses mengonversi teks menjadi format yang konsisten dalam hal huruf besar dan huruf kecil. Biasanya, ini dilakukan dengan mengubah semua huruf dalam teks menjadi huruf kecil.

b. Normalisasi
Normalisasi adalah proses untuk menghasilkan representasi teks yang seragam dan konsisten.

c. Tokenizing
Tokenizing adalah proses memecah teks menjadi unit-unit yang lebih kecil, yang disebut token. Token bisa berupa kata-kata, frasa, tanda baca, atau entitas lainnya yang memiliki makna dalam bahasa atau konteks tertentu

d. Stopwords
Stopword adalah kata-kata umum yang sering muncul dalam teks tetapi cenderung tidak memberikan banyak informasi atau makna khusus dalam konteks tertentu. Kata-kata ini biasanya dianggap tidak relevan dalam analisis teks karena keberadaan mereka yang umum dan sering muncul dalam berbagai jenis teks

e. Stemming
Stemming adalah proses memotong akhiran atau awalan dari kata-kata untuk menghasilkan bentuk dasar atau akar kata, yang disebut "stem". Tujuan utama dari stemming adalah untuk mengurangi variasi morfologis dalam teks, sehingga kata-kata dengan akar yang sama akan dianggap identik, meskipun mereka mungkin memiliki bentuk yang berbeda

2.2 Implementasi Random Forest Pada Python

Implementasi metode *Random Forest* Pada *Python* menggunakan *Pycharm Community edition 2023.1.2.* Adapun tahapan penerapaannya sebagai berikut :

INFORMATIKA
Jurnal Informatika, Manajemen dan Komputer, Vol. 17, No. 1, Mei 2025
eISSN : 2580-3042
pISSN : 1979-0694

1. Import Library pendukung *sklearn*.

```
# Evaluasi model
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
```

Gambar 3. *Pseudocode Import Library Sklearn*

2. Membaca Dataset

```
# Membaca dataset
data = pd.read_csv('dataset_sms_spam_v1.csv')
```

Gambar 4. *Pseudocode* Baca Dataset

3. Fitur dan Target

```
# Pemilihan fitur ('text') dan target ('label')
X = data['Teks']
y = data['label']
```

Gambar 5. *Pseudocode* Fitur dan Target

4. Data Latih dan data Uji

```
# Pembagian data menjadi data pelatihan dan data uji
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Gambar 6. *Pseudocode* Data Latih dan Data Uji

5. Ektraksi Fitur Menggunakan *Countvectorizer*

```
# Ekstraksi fitur menggunakan CountVectorizer
vectorizer = CountVectorizer()
X_train_vectorized = vectorizer.fit_transform(X_train)
X_test_vectorized = vectorizer.transform(X_test)
```

Gambar 7. *Pseudocode* Ektraksi Fitur *Countvectorizer*

6. Insialisasi dan Pelatihan

```
22  # Inisialisasi dan pelatihan model Random Forest
23  rf_model = RandomForestClassifier(random_state=42)
24  rf_model.fit(X_train_vectorized, y_train)
```

Gambar 8. *Pseudocode* Pelatihan Dataset

7. Prediksi dan Pengujian

```
25
26  # Melakukan prediksi pada data uji
27  y_pred = rf_model.predict(X_test_vectorized)
28
```

Gambar 9. *Pseudocode* Prediksi dan Pengujian Dataset

INFORMATIKA
Jurnal Informatika, Manajemen dan Komputer, Vol. 17, No. 1, Mei 2025
eISSN : 2580-3042
pISSN : 1979-0694

8. Evaluasi *Random forest*

```
main.py
1  import pandas as pd
2  from sklearn.model_selection import train_test_split
3  from sklearn.feature_extraction.text import CountVectorizer
4  from sklearn.ensemble import RandomForestClassifier
5  from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
```

Gambar 10. *Pseudocode* Evaluasi Model

9. Hasil Evaluasi

```
34  # Menampilkan hasil evaluasi
35  print(f'Accuracy: {accuracy:.4f}\n')
36  print('Confusion Matrix:')
37  print(conf_matrix)
38  print('\nClassification Report:')
39  print(classification_rep)
```

Gambar 11. *Pseudocode* Hasil Evaluasi

3. HASIL DAN PEMBAHASAN

Pada bagian ini membahas penerapan metode *Random forest* dalam melakukan kalsifikasi SMS penipuan, SMS normal, dan SMS promo. Adapun hasil penerapan tersebut dapat dilihat pada table matriks prediksi pada Tabel 1.

TABEL 1
PREDIKSI CONFUSION MATRIX

Label	Kelas 0	Kelas 1	Kelas 2
0	95 (TP)	0 (FN)	4 (FP)
1	8 (FP)	68 (TP)	6 (FP)
2	3 (FP)	2 (FP)	43 (TP)

Dari hasil pengujian tersebut diperoleh tingkat akurasi **89%** dengan *Confusin Matrix* sebagai berikut :

[[95 0 4]

[8 68 6]

[3 2 43]]

Adapun keterangan dari *Confusin Matrix* sebagai berikut :

Terdapat 95 prediksi yang benar/ *True Positive* (TP) untuk kelas 0, 68 prediksi yang benar untuk kelas 1, dan 43 prediksi yang benar untuk kelas 2., selanjutnya Terdapat 4 prediksi yang salah/*False Positive* (FP) untuk kelas 0 (diprediksi sebagai kelas 2), 8 prediksi yang salah untuk kelas 1 (diprediksi sebagai kelas 0), dan 6 prediksi yang salah untuk kelas 2 (diprediksi sebagai kelas 1). Untuk lebih jelas dapat dilihat pada Tabel 2. Prediksi *Confusin Matrix.*

4. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan dapat disimpulkan bahwa :

1. Implementasi metode *Random Forest* menggunakan *library scikit-learn* pada *Python* untuk proses pelatihan pada daset yang telah mendapatkan pelabelan untuk jenis SMS, yaitu 0 = SMS normal, 1 = SMS penipuan (spam), 2 = SMS promo dijadikan dataset pelatihan dan pengujian untuk mengukur kiinerja metode *Random Forest.* Evaluasi kinerja model dilakukan dengan menggunakan berbagai metrik, seperti akurasi, presisi, *recall, dan F1-score.*
2. Hasil penelitian menunjukkan bahwa metode *Random Forest* mampu memberikan klasifikasi yang baik dalam membedakan antara SMS normal, SMS spam (penipuan), dan SMS promo dengan

tingkat akurasi sebesar 89%. Dengan nilai presisi sebesar 0,97%, recall sebesar 0,83%, dan F1-score sebesar 0,89%, metode ini mampu mengidentifikasi sebagian besar SMS spam dengan tingkat kesalahan yang rendah. Dengan implementasi metode *Random Forest* menggunakan *Python*, penelitian ini memberikan kontribusi penting dalam pengembangan solusi untuk masalah spam SMS. Pendekatan yang diusulkan terbukti efektif dan efisien dalam memerangi SMS spam, dan dapat diterapkan dalam lingkup aplikasi yang luas dalam keamanan informasi dan privasi pengguna

5. REFERENSI

Amani Alzahrani, D. B. rawat. (2019). Comparative Study of Machine Learning Algorithms SMS Spam Detection. *UTC from IEEE Xplore*.

Elagamy, M. N., Stanier, C., & Sharp, B. (2018). Stock market random forest-text mining system mining critical indicators of stock market movements. *2nd International Conference on Natural Language and Speech Processing, ICNLSP 2018*, 1–8. https://doi.org/10.1109/ICNLSP.2018.8374370

Harjito, B., Wijayanto, A., Aini, K. N., & Murtiyas, B. (2019). Comparison of Multinomial Naïve Bayes with K-Nearest Neighbors, Support Vector Machine and Random Forest for Classification of "Network Attacks" Document. *Proceedings of 2019 4th International Conference on Informatics and Computing, ICIC 2019*. https://doi.org/10.1109/ICIC47613.2019.8985919

Ismail, S. S. I., Mansour, R. F., Abd El-Aziz, R. M., & Taloba, A. I. (2022). Efficient E-Mail Spam Detection Strategy Using Genetic Decision Tree Processing with NLP Features. *Computational Intelligence and Neuroscience*, *2022*. https://doi.org/10.1155/2022/7710005

Karyawati, A. E., Wijaya, K. D. Y., Supriana, I. W., & Supriana, I. W. (2023). a Comparison of Different Kernel Functions of Svm Classification Method for Spam Detection. *JITK (Jurnal Ilmu Pengetahuan Dan Teknologi Komputer)*, *8*(2), 91–97. https://doi.org/10.33480/jitk.v8i2.2463

Khalik, M. F. M., & Arifin, F. (2023). Klasifikasi Indeks Kedalaman Kemiskinan Provinsi Sulawesi Selatan Berbasis Decision Tree, K-Nearest Neighbor, Naive Bayes, Neural Network, dan Random Forest. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, *9*(2), 282. https://doi.org/10.26418/jp.v9i2.67492

Kurniawan, I., Buani, D. C. P., Abdussomad, A., Apriliah, W., & Saputra, R. A. (2023). Implementasi Algoritma Random Forest Untuk Menentukan Penerima Bantuan Raskin. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *10*(2), 421–428. https://doi.org/10.25126/jtiik.2023102625

Maqsood, U., Ur Rehman, S., Ali, T., Mahmood, K., Alsaedi, T., & Kundi, M. (2023). An Intelligent Framework Based on Deep Learning for SMS and e-mail Spam Detection. *Applied Computational Intelligence and Soft Computing*, *2023*(Dl). https://doi.org/10.1155/2023/6648970

Pan, W., Li, J., Gao, L., Yue, L., Yang, Y., Deng, L., & Deng, C. (2022). Semantic Graph Neural Network: A Conversion from Spam Email Classification to Graph Classification. *Scientific Programming*, *2022*(ii). https://doi.org/10.1155/2022/6737080

Parida, U., Nayak, M., & Nayak, A. K. (2021). News text categorization using random forest and naïve bayes. *1st Odisha International Conference on Electrical Power Engineering, Communication and Computing Technology, ODICON 2021*, 0–3. https://doi.org/10.1109/ODICON50556.2021.9428925

Savargiv, M., Masoumi, B., & Keyvanpour, M. R. (2021). A new random forest algorithm based on learning automata. *Computational Intelligence and Neuroscience*, *2021*. https://doi.org/10.1155/2021/5572781

Sejati, P., Munawar, Pilliang, M., & Akbar, H. (2022). Studi Komparasi Naive Bayes , K-Nearest Neighbor, dan Random Forest Untuk Prediksi Calon Mahasiswa Yang Diterima Atau Comparative Study Of Naive Bayes , K-Nearest Neighbor , And Random Forest For The Prediction Of Prospective Students. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, *9*(7), 1341–1348. https://doi.org/10.25126/jtiik.202296737

Singh, A. B., Singh, K. M., Chanu, Y. J., Thongam, K., & Singh, K. J. (2022). An Improved Image Spam Classification Model Based on Deep Learning Techniques. *Security and Communication Networks*, *2022*. https://doi.org/10.1155/2022/8905424

Singh, S. N., & Sarraf, T. (2020). Sentiment analysis of a product based on user reviews using random forests algorithm. *Proceedings of the Confluence 2020 - 10th International Conference on Cloud Computing, Data Science and Engineering*, 112–116. https://doi.org/10.1109/Confluence47617.2020.9058128

Sisodia, D. S., Mahapatra, S., & Sharma, A. (2020). Automated SMS classification and spam analysis using topic modeling. *2nd International Conference on Data, Engineering and Applications, IDEA 2020*. https://doi.org/10.1109/IDEA49133.2020.9170710

Tambunan, S. M., Nataliani, Y., & Lestari, E. S. (2021). Perbandingan Klasifikasi dengan Pendekatan Pembelajaran Mesin untuk Mengidentifikasi Tweet Hoaks di Media Sosial Twitter. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, *7*(2), 112. https://doi.org/10.26418/jp.v7i2.47232

Trishna, T. I., Emon, S. U., Ema, R. R., Sajal, G. I. H., Kundu, S., & Islam, T. (2019). Detection of Hepatitis (A, B, C and E) Viruses Based on Random Forest, K-nearest and Naïve Bayes Classifier. *2019 10th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2019*, 1–7. https://doi.org/10.1109/ICCCNT45670.2019.8944455

Tyasnurita, R., & Hapsari, S. W. (2020). Identification of Chronic Kidney Disease Using Naive Bayes, Adaboost, and Random Forest Learning Methods. *JITK (Jurnal Ilmu Pengetahuan Dan Teknologi Komputer)*, *6*(1), 115–120. https://doi.org/10.33480/jitk.v6i1.1403

Utami, M. P., Nurhayati, O. D., & Warsito, B. (2020). Hoax Information Detection System Using Apriori Algorithm and Random Forest Algorithm in Twitter. *6th International Conference on Interactive Digital Media, ICIDM 2020, Icidm*. https://doi.org/10.1109/ICIDM51048.2020.9339648

Virra, K., Andreswari, R., & Hasibuan, M. A. (2019). Sentiment Analysis of Social Media Users Using Naïve Bayes, Decision Tree, Random Forest Algorithm: A Case Study of Draft Law on the Elimination of Sexual Violence (RUU PKS). *ICSECC 2019 - International Conference on Sustainable Engineering and Creative Computing: New Idea, New Innovation, Proceedings*, 239–244. https://doi.org/10.1109/ICSECC.2019.8907228