ANALISIS PERFORMA ALGORITMA SUPERVISED LEARNING TERHADAP DATA DESKRIPSI DENGAN REPRESENTASI DAN PARAMETER TUNING

Rafael Austin[1], Alfi Syahrian[2]

[12,]Program Studi Teknik Informatika, Universital Esa Unggul

Jln. Harapan Indah Boulevard No. 2, Pusaka Rakyat, Kec. Tarumajaya, Bekasi, Jawa Barat 17214
[1] raf.austin05@gmail.com, [2] fyysyahrian10@gmail.com

Abstract

Medical texts presented in narrative form often have an unstructured nature, thus requiring solutions that can be optimally utilized for their classification. This issue forms the basis of this study, which aims to evaluate the performance of various classification algorithms in processing patient complaint narratives using several text representation approaches. The dataset used consists of medically labeled descriptions in a balanced distribution and undergoes preprocessing to clean and standardize the text before being fed into machine learning models. Four text representation methods Bag of Words, TF-IDF, Word2Vec, and Hybrid were employed to transform the text into numerical features. Five classification algorithms were tested and compared based on evaluation metrics including accuracy, precision, recall, and F1-score. The results indicate that frequency-based approaches such as Bag of Words and TF-IDF, when combined with linear algorithms, can deliver the best performance. Furthermore, parameter tuning proved to be an important step in improving classification outcomes. This study emphasizes that selecting the right combination of feature representation and algorithm greatly influences the success of narrative-based medical text classification.

Keywords : *Classification Algorithms, Medical Text Classification, Text Representation, TF-IDF, Word2Vec*

Abstrak

Teks medis yang didapatkan dalam bentuk narasi sering kali memiliki sifat yang tidak terstruktur, sehingga diperlukan solusi yang dapat dimanfaatkan secara optimal untuk klasifikasi teks medis tersebut. Permasalahan ini menjadi landasan dilakukannya penelitian yang bertujuan untuk mengevaluasi performa berbagai algoritma klasifikasi dalam mengolah narasi keluhan pasien menggunakan sejumlah pendekatan representasi teks. Dataset yang digunakan terdiri dari deskripsi medis yang telah diberi label secara seimbang dan melalui proses pra-pemrosesan untuk membersihkan serta menstandarkan teks sebelum dimasukkan ke dalam model pembelajaran mesin. Empat metode representasi teks, yaitu Bag of Words, TF-IDF, Word2Vec, dan Hybrid, digunakan untuk mengubah teks menjadi fitur numerik. Lima algoritma klasifikasi diuji dan dibandingkan berdasarkan metrik evaluasi meliputi akurasi, precision, recall, dan F1-score. Hasil penelitian menunjukkan bahwa pendekatan berbasis frekuensi seperti Bag of Words dan TF-IDF, ketika dipadukan dengan algoritma linier, mampu memberikan performa terbaik. Selain itu, proses tuning parameter terbukti penting dalam meningkatkan hasil klasifikasi. Penelitian ini menegaskan bahwa pemilihan kombinasi representasi fitur dan algoritma yang tepat sangat mempengaruhi keberhasilan klasifikasi teks medis berbasis narasi.

Kata kunci : Algoritma Klasifikasi, Klasifikasi Teks Medis, Representasi Teks, TF-IDF, Word2Vec

1. PENDAHULUAN

Natural Language Processing yang memberikan dampak besar pada data berbasis teks telah memberikan pengaruh pada bidang medis. Lebih tepatnya pada diagnosa penyakit pasien berdasarkan keluhan naratif pasien. Namun data ini sering kali sulit dan dianalisis dan

harus ada pendekatan pemrosesan yang sesuai untuk dimanfaatkan secara optimal untuk klasifikasi dan decision making [1].

Salah satu aspek penting dalam mengatasi tantangan tersebut adalah bagaimana teks naratif direpresentasikan secara efektif untuk diproses oleh algoritma machine learning. Representasi tersebut mengubah teks naratif menjadi fitur numerik yang dapat diproses oleh algoritma machine learning. Salah satu metode klasik adalah Term Frequency-Inverse Document Frequency (TF-IDF), yang memberikan bobot pada kata berdasarkan frekuensinya dalam dokumen dan korpus secara keseluruhan. TF-IDF telah terbukti efektif untuk menyaring kata-kata yang lebih relevan dalam klasifikasi emosi . Selain itu, pendekatan Word2Vec menghasilkan representasi kata dalam bentuk vektor berdimensi tetap, di mana kedekatan semantik antar kata dapat dipertahankan [2]. Dalam konteks medis, pendekatan ini dinilai lebih memahami konteks linguistik dari istilah yang sering muncul dalam keluhan pasien [3]. Kombinasi dari kedua pendekatan tersebut dikenal sebagai representasi hybrid, yang bertujuan menggabungkan keunggulan statistik dari TF-IDF dan semantik dari Word2Vec [4].

Selain representasi teks, kinerja sistem klasifikasi juga sangat dipengaruhi oleh pemilihan algoritma yang tepat. Beberapa algoritma klasik yang banyak digunakan adalah Naive Bayes, yang bekerja berdasarkan probabilitas kata; Support Vector Machine (SVM), yang memisahkan kelas menggunakan hyperplane dengan margin maksimum; Logistic Regression, yang memodelkan peluang dari variabel biner; Random Forest, yang menggabungkan banyak decision tree; serta K-Nearest Neighbor (KNN), yang mengklasifikasikan berdasarkan kedekatan jarak antar vektor [5]. Penelitian sebelumnya menunjukkan bahwa tidak ada satu algoritma pun yang unggul secara mutlak, karena performa sangat bergantung pada jenis data dan representasi teks yang digunakan [6], [7].

Namun, penelitian yang mengombinasikan kelima algoritma tersebut dengan empat representasi teks sekaligus, khususnya untuk data keluhan pasien, masih terbatas. Sudah terdapat Berbagai studi yang telah membandingkan algoritma-algoritma tersebut dalam berbagai domain, namun belum banyak penelitian yang secara eksplisit menguji kelima algoritma tersebut dengan empat representasi teks sekaligus, khususnya dalam konteks klasifikasi narasi keluhan pasien [8], [9]. Oleh karena itu, penelitian ini hadir untuk mengisi celah tersebut, dengan fokus pada analisis performa klasifikasi menggunakan kombinasi lima algoritma dengan empat representasi teks (Bag of Words, TF-IDF, Word2Vec, dan Hybrid TF-IDF+Word2Vec), terhadap data deskripsi medis yang seimbang dari Kaggle.

2. METODOLOGI PENELITIAN

2.1. Dataset

Dataset yang digunakan dalam penelitian ini diambil dari platform Kaggle, yang menyediakan kumpulan data keluhan pasien dalam bentuk teks naratif dan telah dilabeli. Jumlah total data adalah 1.200 entri, masing-masing terbagi rata ke dalam 24 kelas label, sehingga setiap label memiliki 50 deskripsi. Format data terdiri atas dua kolom utama: kolom deskripsi teks keluhan pasien dan kolom label penyakit. Karena jumlah data tiap kelas setara, maka dataset ini dianggap seimbang, sehingga cocok untuk dilakukan evaluasi dengan metrik seperti F1-score dan precision tanpa memerlukan teknik balancing tambahan [10].

Berikut adalah link pengunduhan dataset di kaggle https://www.kaggle.com/datasets/niyarrbarman/symptom2disease/data

2.2. Pra-pemrosesan Teks

Proses pra-pemrosesan dilakukan untuk membersihkan data dari elemen-elemen yang tidak relevan dan menyiapkannya dalam bentuk yang dapat diproses oleh mesin:

1. **Case folding** : Semua huruf dalam teks diubah menjadi huruf kecil untuk menghindari duplikasi kata akibat perbedaan kapitalisasi.
2. **Penghapusan tanda baca dan angka** : Simbol, angka, dan tanda baca dihapus karena tidak memberikan informasi semantik yang signifikan.
3. **Stopwords removal** : Kata-kata umum seperti “yang”, “adalah”, atau “dengan” dihapus karena tidak memiliki kontribusi informasi khusus [11].
4. **Tokenisasi** : Teks dipisahkan menjadi kata-kata individu (token) agar dapat direpresentasikan sebagai fitur.
5. **Stemming** : Kata-kata diubah menjadi bentuk dasarnya (akar kata), misalnya “menyakitkan” menjadi “sakit”, untuk mengurangi kompleksitas kosakata [12].

Langkah-langkah ini penting agar representasi fitur tidak terganggu oleh variasi linguistik yang tidak relevan

2.3. Algoritma Klasifikasi

Lima algoritma digunakan untuk membandingkan performa klasifikasi:

- **Naive Bayes** : Model probabilistik yang mengasumsikan independensi antar fitur. Cocok untuk data teks yang bersifat diskrit dan bekerja cepat meskipun asumsi sering dilanggar [6].Algoritma Naive Bayes telah banyak digunakan untuk klasifikasi berbasis gejala penyakit karena kesederhanaan dan efektivitasnya, seperti ditunjukkan oleh Pratama dan Ramadhani dalam penelitian klasifikasi penyakit berbasis gejala [13].
- **Support Vector Machine (SVM)** : Algoritma diskriminatif yang memaksimalkan margin antar kelas. SVM terkenal efektif untuk data berdimensi tinggi seperti teks [5].
- **Logistic Regression** : Model linier yang memprediksi probabilitas keanggotaan kelas. Meskipun sederhana, model ini mampu bersaing dengan model yang lebih kompleks dalam beberapa kasus [6].
- **Random Forest** : Ensemble dari banyak decision tree yang melakukan voting untuk klasifikasi akhir. Keunggulannya terletak pada kemampuan mengurangi overfitting [7].
- **K-Nearest Neighbor (KNN)** : Metode non-parametrik yang menentukan label berdasarkan mayoritas tetangga terdekat. Kelebihannya terletak pada implementasi yang mudah dan hasil yang cukup baik untuk dataset kecil [8].

2.4. Representasi Teks

Penelitian ini menggunakan empat metode representasi teks yang masing-masing memiliki kelebihan:

1. **Bag of Words (BoW)** : Mewakili dokumen sebagai frekuensi kemunculan kata tanpa memperhatikan konteks. Meskipun sederhana dan efektif dalam tugas dasar, BoW tidak mempertimbangkan urutan kata atau semantik [14].
2. **TF-IDF** : Meningkatkan BoW dengan memberikan bobot pada kata berdasarkan tingkat kelaziman dan keberartian dalam seluruh korpus. TF-IDF telah terbukti mampu meningkatkan kinerja klasifikasi, khususnya dalam teks pendek [15].
3. **Word2Vec** : Menggunakan teknik neural network untuk membangun vektor kata yang mempertahankan informasi semantik dan konteks. Teknik ini populer dalam NLP karena mampu menangkap hubungan antar kata yang lebih kompleks.
4. **Hybrid (TF-IDF + Word2Vec)** : Menggabungkan keunggulan TF-IDF dalam menilai pentingnya kata dengan kemampuan Word2Vec dalam memahami makna. Pendekatan ini sering menunjukkan performa lebih baik daripada masing-masing teknik secara terpisah [4].

2.5. Evaluasi Algoritma

Pada tahap awal, seluruh kombinasi algoritma dan representasi teks diuji menggunakan parameter default dari library scikit-learn. Dataset dibagi dengan skema 80% data latih dan 20% data uji. Evaluasi dilakukan menggunakan empat metrik utama: F1-Score, Precision, Recall, dan Accuracy. Metrik ini digunakan untuk menangkap keseimbangan antara hasil klasifikasi benar dan salah, serta sensitivitas terhadap kelas minoritas [9].

2.6. Parameter Tuning

Setelah evaluasi awal, dilakukan proses tuning parameter menggunakan metode GridSearchCV dari scikit-learn untuk menemukan konfigurasi hyperparameter terbaik dari masing-masing algoritma . Parameter seperti jumlah tetangga (K) untuk KNN, jumlah pohon (n_estimators) untuk Random Forest, dan nilai C serta kernel untuk SVM diuji dalam grid parameter. Proses tuning ini bertujuan untuk mengoptimalkan performa F1-score secara keseluruhan. Secara teori, proses tuning termasuk dalam strategi hyperparameter optimization, yang merupakan pendekatan sistematis untuk menemukan parameter model terbaik berdasarkan data pelatihan. Pada salah satu tuning yaitu SVM, algoritma Support Vector Machine dapat mencapai akurasi tinggi dalam klasifikasi penyakit, tetapi tetap memerlukan perhatian khusus terhadap pemilihan parameter agar tidak terjadi overfitting[16]. Hal ini berlaku juga ke semua algoritma.

2.7. Perbandingan Akhir

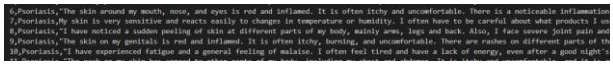
Pada tahap akhir akan dibandingkan 2 algoritma dengan pasangan representasi dan parameter yang terbaik. Perbandingan akan dilakukan dengan metrik Akurasi, Precision,

Recall, F1-Score. Akan ditampilkan juga confusion matriks untuk menganalisis kesulitan algoritma terdapat dimana. Lalu akan dilakukan analisis apa kelebihan dan kekurangan algoritma tersebut.

3. HASIL DAN PEMBAHASAN

3.1 Data Set dan Pra-pemrosesan

Berikut adalah potongan data dari dataset yang dipakai :



Gambar 1. Potongan Dataset

Setelah itu dataset yang di dapatkan tersebut dilakukan Tahap pra-pemrosesan yang dilakukan untuk membersihkan dan menyiapkan data teks agar dapat digunakan secara optimal dalam proses pembelajaran mesin. Proses ini meliputi beberapa tahapan yaitu Case folding, Penghapusan tanda baca dan angka, Penghapusan stopwords secara semantik, serta stemming.

TABEL I. PERBANDINGAN SEBELUM DAN SETELAH PRA-PEMROSESAN

Data deskripsi sebelum dilakukan pra pemrosesan	Data deskripsi sesudah dilakukan pra pemrosesan
I have been experiencing a skin rash on my arms, legs, and torso for the past few weeks. It is red, itchy, and covered in dry, scaly patches.	experienc skin rash arm leg torso past week red itchi cover dri scali patch

Dapat dilihat bahwa data mengalami perubahan seperti hilangnya huruf kapital, penghapusan tanda baca, serta pengubahan kata menjadi bentuk dasar melalui proses stemming. Transformasi ini dilakukan untuk menyederhanakan struktur kalimat dan menstandarkan bentuk kata sehingga memudahkan proses pengolahan data oleh algoritma

3.2 Evaluasi Awal

Sesudah data dilakukan pra-pemrosesan akan dilakukan evaluasi awal dimana setiap algoritma akan dilakukan Evaluasi dengan metrik akurasi, recall, precision, dan f1-score. Pada evaluasi awal ini setiap algoritma akan di tes pada setiap representasi teks yang akan dipakai tanpa perubahan parameter umum untuk menunjukkan perbedaan berikut ini adalah hasil dari evaluasi awal algoritma terhadap representasi teks

TABEL II. HASIL EVALUASI PARAMETER UMUM

Algoritma	Akurasi	Precision	Recall	F1-score
Support Vector Machine (TF-IDF)	0.971	0.98	0.97	0.97
Support Vector Machine (Word2Vec)	0.925	0.94	0.92	0.92
Support Vector Machine (BoW)	0.975	0.98	0.98	0.97
Support Vector Machine (Hybrid)	0.975	0.98	0.98	0.97
Logistic Regression (TF-IDF)	0.958	0.96	0.96	0.96
Logistic Regression (Word2Vec)	0.858	0.87	0.86	0.85
Logistic Regression (BoW)	0.983	0.99	0.98	0.98
Logistic Regression (Hybrid)	0.954	0.96	0.95	0.95
Naive Bayes (TF-	0.95	0.96	0.95	0.95

IDF)				
Naive Bayes (Word2Vec)	0.854	0.87	0.85	0.86
Naive Bayes (BoW)	0.95	0.96	0.95	0.95
Naive Bayes (Hybrid)	0.945	0.95	0.95	0.95
K-nearest Neighbor (TF-IDF)	0.95	0.96	0.95	0.95
K-nearest Neighbor (Word2Vec)	0.883	0.90	0.88	0.88
K-nearest Neighbor (BoW)	0.916	0.94	0.92	0.92
K-nearest Neighbor (Hybrid)	0.954	0.96	0.95	0.95
Random Forest (TF-IDF)	0.958	0.97	0.96	0.96
Random Forest (Word2Vec)	0.887	0.90	0.89	0.89
Random Forest (BoW)	0.966	0.97	0.97	0.97
Random Forest (Hybrid)	0.92	0.92	0.92	0.92

Berdasarkan hasil evaluasi terhadap 20 kombinasi model dan representasi fitur, diperoleh beberapa poin :

1. Model terbaik secara umum adalah *Support Vector Machine (SVM)* dengan representasi *BoW* dan *Hybrid (TF-IDF + Word2Vec)*, yang mencapai akurasi dan F1-score sebesar 0.975 dan 0.97. Hal ini menunjukkan bahwa SVM sangat efektif untuk menangani data teks berdimensi tinggi seperti hasil dari BoW maupun gabungan fitur hybrid.
2. Logistic Regression juga menunjukkan performa sangat baik, khususnya dengan representasi *BoW*, dengan akurasi 0.983 dan F1-score 0.98, bahkan sedikit lebih tinggi dari SVM pada kombinasi tertentu.
3. Naive Bayes menunjukkan performa stabil di seluruh representasi, dengan F1-score berkisar antara 0.85–0.95. Model ini sangat cocok untuk representasi TF-IDF dan BoW, namun performanya menurun saat digunakan dengan Word2Vec.
4. K-Nearest Neighbor (KNN) memiliki performa kompetitif pada representasi TF-IDF, BoW, dan Hybrid dengan F1-score hingga 0.95, tetapi kurang optimal ketika digunakan dengan Word2Vec (F1-score 0.88), yang kemungkinan disebabkan oleh sensitivitas KNN terhadap dimensi dan skala fitur.
5. Random Forest bekerja sangat baik dengan representasi BoW dan TF-IDF (F1-score 0.96–0.97), tetapi kinerjanya menurun saat menggunakan Word2Vec dan Hybrid. Hal ini menunjukkan bahwa model ensemble seperti RF lebih cocok digunakan dengan representasi fitur berbasis frekuensi.
6. Representasi fitur terbaik secara umum adalah *Bag-of-Words (BoW)*, yang secara konsisten menghasilkan nilai akurasi dan F1-score tinggi pada hampir semua algoritma. TF-IDF juga memberikan hasil baik, sementara Word2Vec cenderung memberikan performa lebih rendah, terutama pada model yang tidak berbasis neural.

Secara keseluruhan, kombinasi Logistic Regression atau SVM dengan BoW atau Hybrid memberikan hasil terbaik, dan dapat direkomendasikan untuk implementasi lebih lanjut dalam klasifikasi penyakit berdasarkan deskripsi gejala. Logistic Regression dan Support Vector Machine mampu memberikan hasil yang terbaik karena kesesuaian nya dalam penanganan data berdimensi tinggi seperti pada dataset yang dipakai yaitu teks. Logistic Regression efektif dalam klasifikasi fitur yang

independen dan SVM dapat memisahkan kelas secara optimal dengan hyperplane terutama pada BoW dan TF-IDF

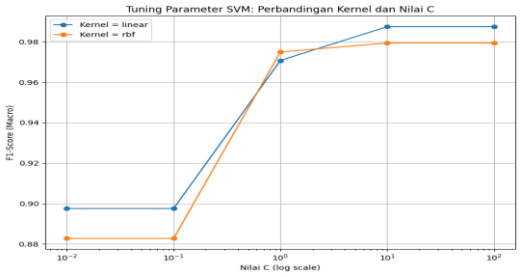
3.3. Evaluasi dan Perbandingan Parameter Tuning

Setelah evaluasi awal yang menggunakan parameter *default*, dilakukan evaluasi tahap kedua. Pada tahap ini, parameter setiap algoritma disesuaikan (*tuning*) untuk mendapatkan performa yang optimal.

1. **Algoritma Support Vector**

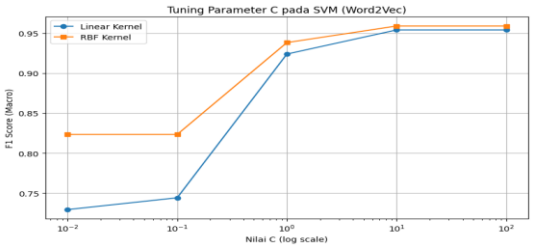
   Pada algoritma SVM akan dilakukan parameter tuning pada bagian nilai C serta tipe kernel nilai C yang akan dilakukan tes akan menggunakan 0.01, 0.1, 1, 10, 100 serta kernel yang akan dipakai adalah RBF dan Linear. Berikut adalah grafik perbandingannya.

Pada SVM + TF-IDF



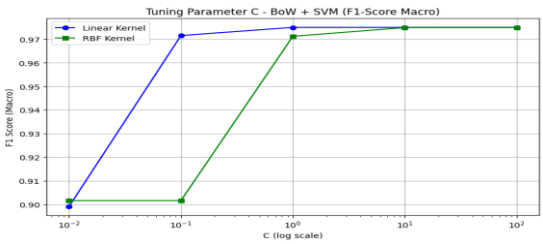
Gambar 2. SVM + TF-IDF

Pada SVM + Word2Vec



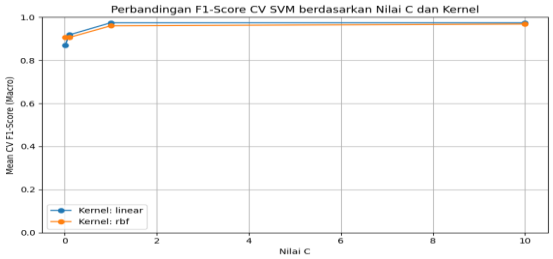
Gambar 3. SVM + Word2Vec

Pada SVM + BoW



Gambar 4. SVM + BoW

Pada SVM + Hybrid



Gambar 5. SVM + Hybrid

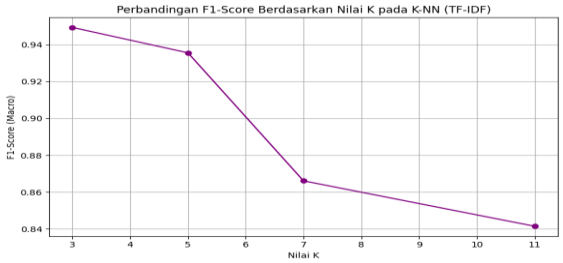
Dari grafik-grafik perbandingan di atas, parameter paling optimal adalah C = 10, sedangkan C = 100 juga memberikan hasil tinggi namun memiliki kemungkinan overfitting yang lebih besar. Hal ini terjadi karena nilai C yang terlalu besar membuat SVM berusaha memisahkan data dengan margin sekecil mungkin, sehingga model menjadi terlalu sensitif terhadap noise pada data. Kernel yang paling efektif adalah linear, karena data teks yang dihasilkan dari TF-IDF dan BoW bersifat berdimensi tinggi dan jarang (sparse), sehingga pemisahan kelas dapat dilakukan secara efisien di ruang fitur linear tanpa memerlukan transformasi non-linear yang kompleks. Kombinasi ini memungkinkan SVM membentuk batas pemisah yang sederhana namun tetap optimal, sehingga mengurangi risiko overfitting dan meningkatkan kemampuan generalisasi model. Temuan ini sejalan dengan Cahyani & Patasik [1], yang menunjukkan efektivitas TF-IDF dalam meningkatkan kinerja model linear pada data teks berdimensi tinggi, serta konsisten dengan studi [6] yang menegaskan stabilitas TF-IDF dibandingkan Word2Vec untuk teks medis. Hasil ini juga didukung oleh Ramadhan et al. [16], yang menyoroti bahwa tuning parameter pada SVM berperan penting dalam mencegah overfitting sekaligus memaksimalkan akurasi pada klasifikasi penyakit.

2. **Algoritma K-Nearest Neighbour**

   Pada algoritma KNN parameter yang akan dilakukan tuning untuk mendapatkan hasil terbaik adalah nilai K yang menentukan tetangga terdekat yang akan dimasukan kedalam perbandingan klasifikasi. Untuk

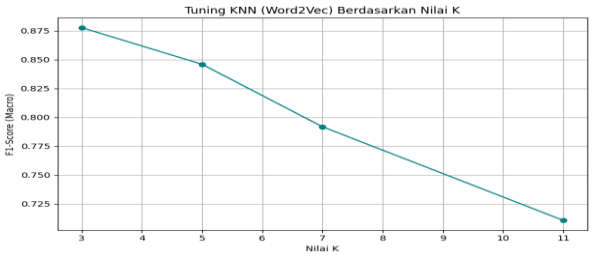
nilai yang akan dipakai adalah 3, 5, 7, dan 11. Berikut adalah hasilnya

Pada KNN + TF-IDF

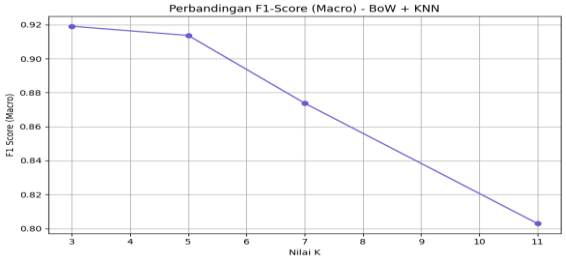


Gambar 6. KNN + TF-IDF

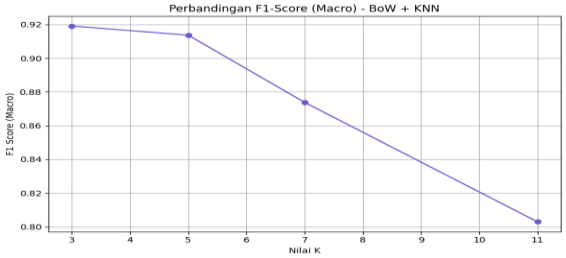
Pada KNN + Word2Vec



Gambar 7. KNN + Word2Vec

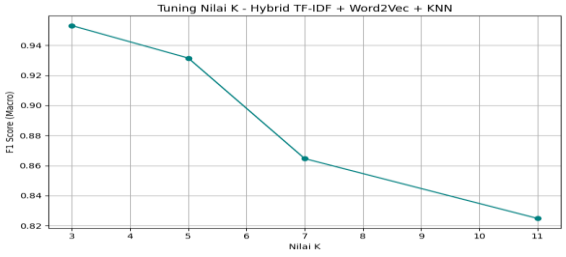


Pada KNN + BoW



Gambar 8. KNN + BoW

Pada KNN + Hybrid



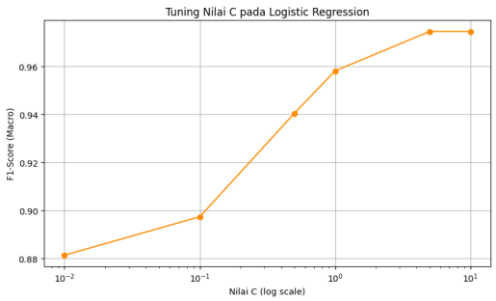
Gambar 9. KNN + hybrid

dari Perbandingan grafik diatas dapat disimpulkan K yang paling efektif adalah 3 karena pada setiap representasi semakin tinggi K yang keluar dari nilai 3 semakin menurun F1-Score hal ini dapat disebabkan karena menyebabkan karena menyebabkan underfitting, yaitu ketika model terlalu "melonggarkan" batas klasifikasi sehingga kehilangan sensitivitas terhadap pola lokal dalam data. Dari grafik diatas hasil paling optimal dapat dilihat pada Gambar 6 dan Gambar 9 adalah KNN dengan K=3 dengan representasi teks TF-IDF dan Hybrid.

3. **Algoritma Logistic Regression**

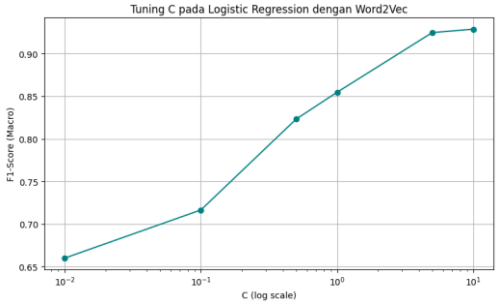
Pada algoritma ini parameter yang akan dilakukan tuning adalah pada nilai regularisasi C. Parameter C mengontrol kekuatan dari regularisasi dalam model. Nilai C yang lebih kecil akan membuat model lebih sederhana dan tinggi akan membuat model lebih kompleks. Pemilihan C ini penting dalam menghindari overfitting dan underfitting. Berikut adalah hasilnya.

Pada Logistic Regression + TF-IDF



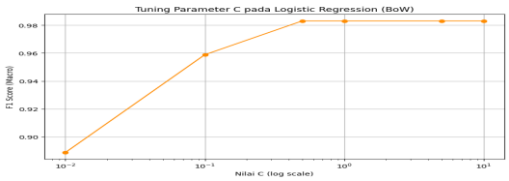
Gambar 10. Logistic Regression + TF-IDF

Pada Logistic Regression + Word2Vec



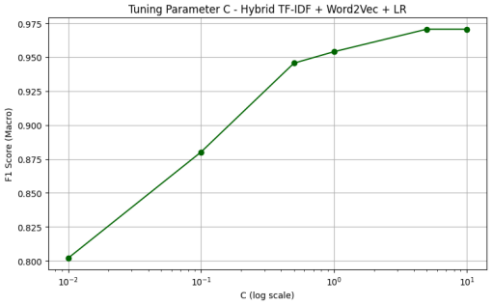
Gambar 11. Logistic Regression + Word2Vec

Pada Logistic Regression + BoW



Gambar 12. Logistic Regression + BoW

Pada Logistic Regression + Hybrid



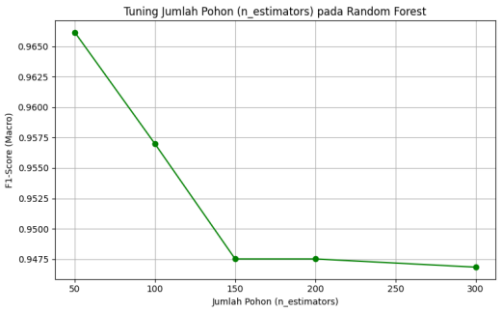
Gambar 13. Logistic Regression + Hybrid

dari hasil tuning yang telah dilakukan tergantung C performa dapat menurun atau meningkat dengan semakin banyak C semakin naik performanya dan semakin turun akan semakin turun juga performanya. Secara konseptual, parameter C berperan sebagai pengatur kekuatan regularisasi dalam Logistic Regression. Nilai C yang terlalu kecil menyebabkan regularisasi yang kuat sehingga model menjadi terlalu sederhana dan rentan mengalami underfitting, sementara nilai C yang terlalu besar justru melemahkan regularisasi. Hasil tertinggi dicapai dengan C = 10 dengan representasi BoW dan TF-IDF. Hasil tinggi Logistic Regression, terutama pada representasi BoW, sejalan dengan Das et al. [9] yang menemukan performa kompetitif Logistic Regression pada klasifikasi teks kebencian dengan TF-IDF. Kannan & Menaga [14] juga mendukung bahwa model ini efektif untuk data berdimensi tinggi, seperti fitur BoW.

4. **Algoritma Random Forest**

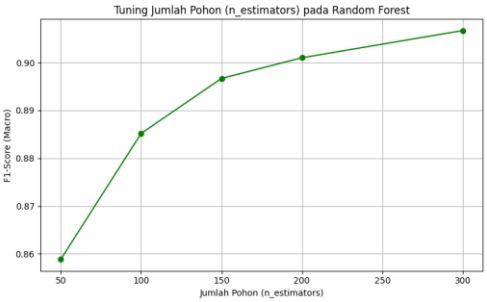
Pada algoritma ini yang akan dilakukan parameter tuning adalah nilai n_estimator yang menentukan jumlah pohon yang akan dibuat. Jika n_estimator terlalu rendah akan terjadi overfitting dan kesulitan dalam prediksi variasi.N estimator yang akan di tes adalah 50,100,150,200,300.

Pada Random Forest + TF-IDF



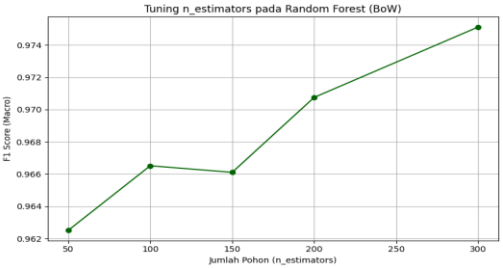
Gambar 14.Random Forest + TF-IDF

Pada Random Forest + Word2Vec



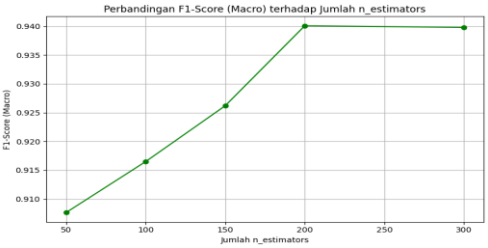
Gambar 15.Random Forest + Word2Vec

Pada Random Forest + BoW



Gambar 16.Random Forest + BoW

Pada Random Forest + Hybrid



Gambar 17.Random Forest + Hybrid

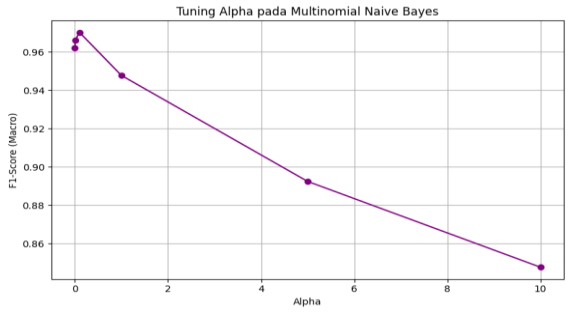
Kinerja model BoW menunjukkan peningkatan yang konsisten dengan seiring

bertambahnya jumlah pohon. Hal ini menunjukkan representasi berbasis frekuensi kata sangat efektif dalam klasifikasi dan cocok untuk algoritma Random Forest. Sementara pada TF-IDF menunjukkan sebuah variasi yaitu semakin banyak pohon semakin berkurang kinerjanya yang dapat disebabkan oleh amplifikasi noise yang membuat algoritma kesulitan. Word2Vec mempunyai performa terendah dari semuanya

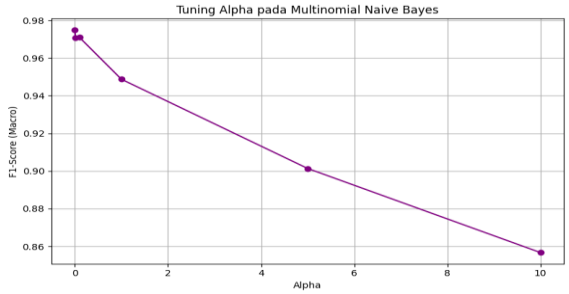
5. **Algoritma Naive Bayes**

Pada Algoritma ini hanya akan dilakukan parameter tuning pada TF-IDF dan BoW karena satu satunya yang memiliki parameter untuk dilakukan tuning yaitu Alpha untuk Smoothing. Berikut hasilnya.

Pada Naive Bayes + TF-IDF



Gambar 18. Naive Bayes + TF-IDF



Pada Naive Bayes + BoW

Gambar 19. Naive Bayes + BoW

Pada grafik dapat disimpulkan dari kedua grafik bahwa kinerja model akan mengalami penurunan seiring dengan kenaikan nilai Alpha, hal ini menunjukkan bahwa model tidak memerlukan smoothing yang berlebihan karena saat nilai alpha diperbesar bobot pada setiap kata akan lebih merata dan dapat berdampak pada bobot kata kata penting yang menjadi pembeda utama antara kelas dan membuat model mengalami penurunan performa

3. 4. Kesimpulan Akhir Parameter Tuning

Setelah dilakukan evaluasi dan pengujian yang mencakup saat sebelum dan sesudah parameter tuning dapat disimpulkan bahwa Logistic Regression dan Support Vector Machine (SVM) secara konsisten menunjukkan performa yang paling unggul dari seluruh algoritma yang diuji, terutama pada metode representasi kata Bag of Words (BoW) dan TF-IDF Serta ketidaksesuaian nya Word2Vec, Kinerja Word2Vec yang relatif lebih rendah dalam penelitian ini kemungkinan disebabkan oleh ukuran dataset yang terbatas, sehingga embedding tidak mampu menangkap hubungan semantik secara optimal. Hal ini sejalan dengan temuan Shao et al. [3] yang melaporkan BoW dapat mengungguli Word2Vec pada teks medis berskala kecil, serta didukung oleh Almazaydeh et al. [2] yang menyatakan bahwa kualitas embedding sangat bergantung pada kelengkapan dan relevansi corpus pelatihan.
Dari parameter tuning dapat di simpulkan beberapa yaitu

- SVM dengan TF-IDF dan parameter kernel linear dan C = 10 adalah parameter paling efektif pada data ini
- Naive Bayes pada TF-IDF dan BoW dengan parameter Alpha = 0.1 adalah parameter paling efektif
- N_estimator = 300 dan BoW pada Random Forest
- K = 3 untuk KNN
- Logistic Regression dengan C = 10

Pentingnya parameter tuning yang teridentifikasi dalam penelitian ini sejalan dengan Ramadhan et al. [16] yang melaporkan peningkatan akurasi signifikan pada SVM dan Random Forest setelah optimalisasi parameter, serta Untoro et al. [7] yang menegaskan bahwa konfigurasi parameter berpengaruh langsung terhadap performa model klasifikasi.

3.5. Perbandingan Akhir

Dari kesimpulan yang telah didapatkan dari parameter tuning dan pengujian representasi teks, didapatkan 2 algoritma terbaik yang akan dilakukan perbandingan akhir yaitu Support Vector Machine dengan representasi teks TF-IDF serta dengan C = 10 dimana akurasi memuncak

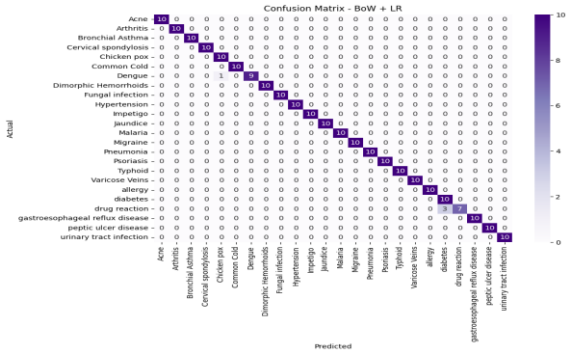
pada C = 10. Untuk Algoritma kedua diambil Logistic Regression dengan representasi Berikut adalah hasil akurasi Precision, recall, F1-score

TABEL III. PERBANDINGAN AKHIR

Metrik	SVM (TF-IDF)	LR (BoW)
Akurasi	0.9874	0.9833
Recall	0.9875	0.9833
Precision	0.9892	0.9865
F1-Score	0.9874	0.9830

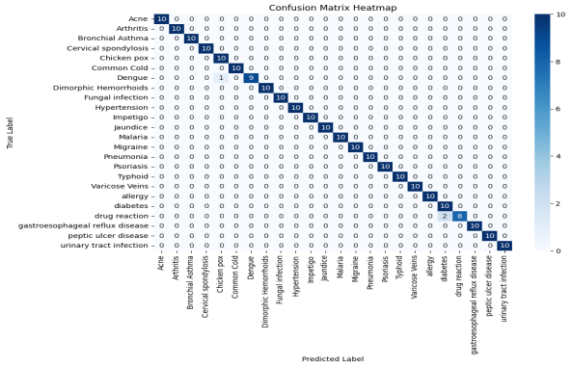
berdasarkan data table yang didapatkan dapat disimpulkan secara konsisten SVM menunjukkan keunggulan pada setiap bagian meliputi Akurasi Recall precision dan F1-score. Dan setelah ini akan ditampilkan confusion matrix untuk kedua algoritma untuk memeriksa kesalahan algoritma.

Matrix Confusion Logistic Regression



Gambar 20. Matrix Confusion Logistic Regression

Matrix Confusion SVM



Gambar 21. Matrix Confusion SVM

Dari Kedua Matrix confusion ini dapat disimpulkan kebanyakan kesalahan yang membuat perbedaan terdapat pada bagian Drug reaction dimana SVM mengalami 2 kesalahan sementara Logistic Regression mengalami 3 kesalahan.

Meskipun perbedaannya sangat tipis, dapat disimpulkan bahwa kemampuan separasi SVM lebih baik karena memiliki keunggulan dalam menangani data teks berdimensi sangat tinggi. Perbedaan satu prediksi ini menunjukkan bahwa pemisah yang dibentuk oleh SVM sedikit lebih akurat dalam membedakan beberapa gejala, terutama pada kasus yang ambigu atau sulit dibedakan. Keunggulan ini sejalan dengan Cahyani & Patasik [1], yang menegaskan efektivitas TF-IDF dalam memaksimalkan performa SVM pada data teks, serta studi [6] yang menunjukkan kestabilan SVM dengan TF-IDF untuk teks medis. Hasil ini juga diperkuat oleh Ramadhan et al. [16], yang menemukan bahwa tuning parameter yang tepat pada SVM dapat meningkatkan akurasi dan mengurangi kesalahan prediksi pada klasifikasi penyakit.

4. Kesimpulan dan Saran

Kesimpulan

Berdasarkan analisis dan evaluasi komprehensif yang telah dilakukan, dapat ditarik beberapa kesimpulan utama.

1. Algoritma dengan Performa Terbaik : Support Vector Machine (SVM) dan Logistic Regression secara konsisten menunjukkan performa paling unggul di antara lima algoritma yang diuji. Keduanya terbukti sangat efektif dalam menangani data teks berdimensi tinggi yang dihasilkan oleh representasi berbasis frekuensi.
2. Representasi Teks Paling Efektif : Bag-of-Words (BoW) dan TF-IDF merupakan dua metode representasi teks yang paling efektif dan stabil untuk klasifikasi narasi medis pada penelitian ini. Sebaliknya, representasi Word2Vec menunjukkan kinerja lebih rendah, terutama saat dipasangkan dengan model non-neural seperti K-NN dan Random Forest, yang mengindikasikan adanya ketidakcocokan dalam menginterpretasi vektor semantik yang padat.
3. Kombinasi Optimal : Kombinasi model terbaik secara keseluruhan setelah proses

tuning adalah SVM dengan representasi TF-IDF, yang menggunakan parameter kernel linear dan C = 10, berhasil mencapai F1-score tertinggi sebesar 0.9874.

4. Pentingnya *Tuning Parameter* : Proses *tuning parameter* terbukti signifikan dalam mengoptimalkan performa setiap algoritma. Penelitian ini berhasil mengidentifikasi parameter optimal untuk masing-masing model, seperti Alpha = 0.1 untuk Naive Bayes, n_estimators = 300 untuk Random Forest, dan K = 3 untuk K-NN, yang secara konsisten memberikan hasil terbaik pada data yang digunakan.

Secara keseluruhan, penelitian ini menegaskan bahwa tidak ada satu kombinasi yang sempurna untuk semua kasus, namun untuk klasifikasi teks naratif medis dengan karakteristik seperti pada dataset ini, pendekatan menggunakan model linier (SVM dan Logistic Regression) dengan representasi fitur berbasis frekuensi (BoW dan TF-IDF) adalah pilihan yang paling direkomendasikan.

Saran

Untuk pengembangan penelitian di masa mendatang, beberapa saran berikut dapat dipertimbangkan:

- Eksplorasi Model *Deep Learning*: Mengingat performa Word2Vec kurang optimal pada model klasik, disarankan untuk melakukan penelitian lanjutan menggunakan model berbasis *deep learning* seperti Long Short-Term Memory (LSTM) atau Transformer (misalnya, BERT) yang dirancang khusus untuk memahami konteks dan hubungan semantik dari *word embeddings*.
- Penggunaan Dataset yang Lebih Besar dan Beragam: Penelitian selanjutnya dapat menggunakan dataset dengan skala yang lebih besar dan variasi yang lebih luas, termasuk data yang tidak seimbang (*imbalanced*), untuk menguji ketahanan dan kemampuan generalisasi dari setiap model.
- Analisis Fitur Lanjutan: Melakukan analisis fitur yang lebih mendalam, seperti menggunakan teknik *feature selection* atau interpretasi model (misalnya dengan SHAP atau LIME), untuk memahami kata-kata atau frasa kunci yang paling berpengaruh dalam proses klasifikasi setiap penyakit.
- Implementasi dalam Sistem Nyata: Mengimplementasikan model dengan performa terbaik ke dalam sebuah prototipe sistem nyata, seperti *chatbot* diagnosis awal atau alat bantu entri data medis, untuk menguji efektivitasnya dalam skenario penggunaan praktis.

5. UCAPAN TERIMA KASIH

Penulis berterima kasih kepada semua pihak yang telah berkontribusi dan mendukung terselesaikannya penelitian ini, khususnya kepada penyedia dataset Kaggle dan para pengembang tools open-source yang digunakan.

Daftar Pustaka:

[1] D. E. Cahyani and I. Patasik, "Performance comparison of tf-idf and word2vec models for emotion text classification," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, pp. 2780–2788, Oct. 2021, doi: 10.11591/eei.v10i5.3157.

[2] L. Almazaydeh, M. Abuhelaleh, A. Al Tawil, and K. Elleithy, "Clinical Text Classification with Word Representation Features and Machine Learning Algorithms," *International journal of online and biomedical engineering*, vol. 19, no. 4, pp. 65–76, 2023, doi: 10.3991/ijoe.v19i04.36099.

[3] Y. Shao, S. Taylor, N. Marshall, C. Morioka, and Q. Zeng-Treitler, "Clinical Text Classification with Word Embedding Features vs. Bag-of-Words Features," in *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, Institute of Electrical and Electronics Engineers Inc., Jul. 2018, pp. 2874–2878. doi: 10.1109/BigData.2018.8622345.

[4] M. Kavitha and P. Prabhavathy, "A review on machine learning techniques for text classification," in *Proceedings of the 2021 4th International Conference on Computing and Communications Technologies, ICCCT 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 605–610. doi: 10.1109/ICCCT53315.2021.9711858.

[5] T.-D. Le, R. Noumeir, J. Rambaud, G. Sans, and P. Jouvet, "Machine Learning Based on Natural Language Processing to Detect Cardiac Failure in Clinical Narratives," Dec. 2021, [Online]. Available: http://arxiv.org/abs/2104.03934

[6] "TF-IDF vs Word Embeddings for Morbidity Identification in Clinical Notes: An Initial Study," 2020. [Online]. Available: https://n2c2.dbmi.hms.harvard.edu/

[7] M. C. Untoro, M. Praseptiawan, M. Widianingsih, I. F. Ashari, A. Afriansyah, and Oktafianto, "Evaluation of Decision Tree, K-NN, Naive Bayes and SVM with MWMOTE on UCI Dataset," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, 2020. doi: 10.1088/1742-6596/1477/3/032005.

[8] A. M. Aubaid, A. Mishra, and A. Mishra, "Machine learning and rule-based embedding techniques for classifying text documents," *International Journal of System Assurance Engineering and Management*, vol. 15, no. 12, pp. 5637–5652, Dec. 2024, doi: 10.1007/s13198-024-02555-w.

[9] S. Das, K. Bhattacharyya, and S. Sarkar, "Performance Analysis of Logistic Regression, Naive Bayes, KNN, Decision Tree, Random Forest and SVM on Hate Speech Detection from Twitter," *International Research Journal of Innovations in Engineering and Technology*, vol. 07, no. 03, pp. 07–03, 2023, doi: 10.47001/irjiet/2023.703004.

[10] Nikhil Sanjay Suryawanshi, "Sentiment analysis with machine learning and deep learning: A survey of techniques and applications," *International Journal of Science and Research Archive*, vol. 12, no. 2, pp. 005–015, Jul. 2024, doi: 10.30574/ijsra.2024.12.2.1205.

[11] S. Alagarsamy, V. James, and R. S. P. Raj, "An Experimental Analysis of Optimal Hybrid Word Embedding Methods for Text Classification Using a Movie Review Dataset," *Brazilian Archives of Biology and Technology*, vol. 65, 2022, doi: 10.1590/1678-4324-2022210830.

[12] I. Sariah *et al.*, "Nanotechnology Perceptions ISSN 1660-6795 www.nano-ntp," 2024. [Online]. Available: www.nano-ntp.com

[13] B. Putra Aryadi and N. Hendrastuty, "PENERAPAN ALGORITMA K-MEANS UNTUK MELAKUKAN KLASTERISASI PADA VARIETAS PADI," 2024. [Online]. Available: http://e-journal.stmiklombok.ac.id/index.php/jireISSN.2620-6900

[14] K. Kannan and A. Menaga, "Risk Factor Prediction by Naive Bayes Classifier, Logistic Regression Models, Various Classification and Regression Machine Learning Techniques," *Proceedings of the National Academy of Sciences India Section B - Biological Sciences*, vol. 92, no. 1, pp. 63–79, Mar. 2022, doi: 10.1007/s40011-021-01278-3.

[15] M. Das, S. Kamalanathan, and P. Alphonse, "A Comparative Study on TF-IDF feature Weighting Method and its Analysis using Unstructured Dataset," 2020.

[16] K. G. Ramadhan *et al.*, "KOMPARASI DETEKSI PENYAKIT GINJAL KRONIS MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE DAN RANDOM FOREST," 2025. [Online]. Available: http://e-journal.stmiklombok.ac.id/index.php/jireISSN.2620-6900

Lampiran

Program yang dipakai selama penelitian dapat di akses di https://github.com/LuckRaf/Supervised-Learning-Analysis-with-Representation-and-parameter-tuning.git