Klasifikasi Multi-Label Pada Soal Berdasarkan Kategori Topik Menggunakan Metode *Support Vector Machine*

1st Alvin Renaldy Novanza
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
alvinrenaldy@student.telkomuniversity.ac.id

2nd Oktariani Nurul Pratiwi
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
onurulp@telkomuniversity.ac.id

3rd Faqih Hamami
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
faqihhamami@telkomuniversity.ac.id

***Abstrak* — Pendidikan, sebagai bagian penting dalam kehidupan manusia, senantiasa mengalami perkembangan seiring dengan adanya kemajuan ilmu pengetahuan dan teknologi (IPTEK). Salah satu inovasi penting dalam pendidikan adalah e-learning, yang memungkinkan siswa belajar tanpa terikat ruang kelas. Namun, dengan semakin banyaknya soal kuis yang beragam topiknya, terutama dalam mata pelajaran IPA yang mencakup berbagai konsep ilmiah, pengelolaan soal secara manual menjadi tidak efisien. Oleh karena itu, diperlukan sistem klasifikasi yang dapat mengorganisasi dan mengelompokkan soal secara otomatis dan efisien, sehingga bisa meningkatkan pemahaman siswa, khususnya pada mata pelajaran IPA. Penelitian ini memiliki tujuan untuk mengimplementasikan algoritma Support Vector Machine dalam proses klasifikasi soal multi-label pada mata pelajaran IPA tingkat SMP. Proses klasifikasi mencakup pembersihan data, case folding, tokenisasi, stopword removal, stemming, dan pembobotan atau ekstraksi fitur teks menggunakan TF-IDF. Pemodelan menggunakan pendekatan problem transformation dengan metode label powerset untuk mengubah soal dengan multi-label menjadi bentuk multi-class sehingga bisa dilakukan klasifikasi biner oleh SVM. Evaluasi model dilakukan menggunakan confusion matrix untuk menganalisis performa klasifikasi dan K-Fold Cross Validation untuk memastikan keakuratan dan generalisasi model. Hasil penelitian menunjukkan bahwa SVM dapat diterapkan untuk klasifikasi soal multi-label dengan akurasi 67%, serta presisi, recall, dan F1-score masing-masing sebesar 75%. Analisis confusion matrix mengungkapkan bahwa model memiliki beberapa kesalahan klasifikasi, mengindikasikan ruang untuk perbaikan lebih lanjut. Meskipun demikian, model SVM menunjukkan potensi yang baik. Penelitian ini juga mengidentifikasi beberapa area untuk perbaikan, termasuk peningkatan kualitas data dan pemilihan parameter model yang lebih optimal. Oleh karena itu, metode SVM layak dipertimbangkan dalam sistem pendidikan untuk pengembangan bank soal dan sistem evaluasi berbasis teknologi, meskipun diperlukan perbaikan lebih lanjut pada model dan data.**

***Kata kunci*— bank soal, confusion matrix e-learning, klasifikasi multi-label, K-Fold Cross Validation, Support Vector Machine.**

I. PENDAHULUAN

Pendidikan merupakan aspek vital dalam kehidupan manusia, di mana setiap orang berhak mendapatkannya sebagai upaya untuk mengembangkan potensi diri. Secara umum, pendidikan bisa diartikan sebagai proses dalam kehidupan yang bertujuan untuk mengembangkan setiap individu agar mampu hidup dan menjalani kehidupannya. Oleh karena itu, menjadi pribadi yang berpendidikan sangatlah penting [1].

Seiring dengan berkembangnya Ilmu Pengetahuan dan Teknologi (IPTEK), metode pembelajaran berkembang menjadi *e-learning*. *E-learning* merupakan suatu konsep pada proses pembelajaran yang memungkinkan siswa untuk belajar kapan saja tanpa terikat pada keberadaan ruang kelas. Pada *E-learning*, siswa dapat meningkatkan kemampuan secara mandiri dalam memahami materi pembelajaran melalui kuis. Soal-soal kuis tersebut kemudian dapat dikumpulkan dan disimpan ke dalam bank soal [2]. Namun, dengan semakin banyaknya soal yang beragam topiknya, pengelolaan soal secara manual menjadi tidak efisien dan memakan banyak waktu. Hal ini terutama berlaku untuk mata pelajaran dengan topik yang luas dan kompleks, seperti IPA, yang mencakup topik seperti manusia, hewan, dan tumbuhan, serta istilah-istilah ilmiah yang seringkali sulih dipahami oleh siswa [3].

Klasifikasi itu merupakan proses untuk mengelompokkan objek berdasarkan karakteristiknya ataupun ciri yang sama ke dalam beberapa topik [4]. Klasifikasi soal berdasarkan topik dapat dilakukan oleh guru secara manual, tetapi karena data yang diolah banyak dan beragam, proses ini akan menghabiskan banyak waktu [5]. Oleh karena itu, diperlukan penerapan teknik klasifikasi otomatis untuk mengelompokkan soal-soal berdasarkan topiknya secara lebih efisien dan akurat. Penggunaan sistem klasifikasi soal dalam platform *e-learning* akan mempermudah pengajar dan siswa dalam mengelola dan menavigasi bank soal yang kompleks, serta membantu dalam proses pengambilan keputusan terkait topik soal [6].

Selain itu, banyak soal evaluasi yang tidak hanya berhubungan dengan satu topik, tetapi juga dapat mencakup beberapa topik sekaligus (*multi-label*). Hal ini menimbulkan tantangan dalam proses klasifikasi karena setiap soal dapat memiliki lebih dari satu label topik. Oleh karena itu, metode klasifikasi *multi-label* diperlukan untuk menangani kasus di mana satu soal harus diklasifikasikan ke dalam beberapa topik sekaligus. Klasifikasi *multi-label* ini akan memungkinkan sistem untuk mengelompokkan soal-soal tersebut secara lebih akurat dan efisien, serta membantu guru dalam mengelola bank soal yang kompleks. Tantangan utama dalam klasifikasi teks *multi-label* adalah ruang label yang sangat besar, yang meningkat secara eksponensial dengan jumlah label kandidat. Selain itu, metode klasifikasi *multi-label* sebelumnya sering

kali tidak mampu menangani peningkatan jumlah label spesifik dan contoh pelatihan dengan efisien, sehingga sulit untuk diimplementasikan pada skala besar [7].

Pada klasifikasi soal secara otomatis, tentunya terdapat banyak teknik atau algoritma yang dapat diimplementasikan yaitu *SVM,* Naïve Bayes, *Decision Tree, Neural Network*, dan lain-lain [8].

Dalam penelitian ini, proses klasifikasi soal berdasarkan topik akan difokuskan pada mata pelajaran IPA dengan menggunakan algoritma *Support Vector Machine* (SVM). SVM merupakan algoritma *machine learning* yang dikembangkan oleh Vapnik, Guyon, dan Boser. Algoritma SVM pertama kali diperkenalkan pada *Annual Workshop on Computational Learning Theory* sekitar tahun 1992. Algoritma SVM memiliki tujuan untuk mencari *hyperplane* terbaik yang dapat membagi dua kelas pada *input space* [9]. Walaupun demikian, SVM juga dinilai dapat menjadi *classifier* karena dapat mengklasifikasikan soal dari area yang beragam dengan akurasi yang tinggi fitur BOW yang menjadi salah satu indikator dalam klasifikasi teks seperti soal-soal [10].

Penelitian sebelumnya dalam jurnal berjudul "*Question Classification using Machine Learning Approaches*" [11] telah menggunakan metode mesin pembelajaran. Hasil penelitian tersebut mengungkap bahwa algoritma *Support Vector Machine* (SVM) secara konsisten unggul dibandingkan dengan algoritma Naïve Bayes *Classifier* dalam konteks klasifikasi soal, menunjukkan bahwa SVM adalah alat klasifikasi yang efektif dalam konteks *e-learning*.

Oleh karena itu, pada penelitian ini memiliki tujuan untuk melakukan klasifikasi multi-label pada teks soal mata pelajaran IPA tingkat SMP dengan menggunakan metode *Support Vector Machine* (SVM). Penelitian ini akan mengimplementasikan SVM untuk mengelompokkan soal-soal berdasarkan topik-topik yang relevan, serta mengevaluasi kinerja yang dihasilkan oleh algoritma SVM. Dengan optimasi yang tepat, diharapkan metode ini dapat menjadi solusi yang efisien dalam membantu pengelolaan bank soal secara otomatis, khususnya dalam platform *e-learning*, serta mendukung proses pembelajaran yang lebih efektif.

II. KAJIAN TEORI

A. *E-Learning*

E-learning merupakan metode belajar jarak jauh yang menggunakan berbagai media seperti internet, CD, atau ponsel untuk menyampaikan materi pembelajaran [12]. Penggunaan *e-learning* dalam pendidikan dapat menjadikan kegiatan belajar mengajar menjadi lebih sistematis dan siswa dapat belajar dengan lebih antusias. *E-learning* dapat membantu pengajar dalam menyampaikan materi pembelajaran kepada siswa mereka dengan lebih efektif. Kemampuan dasar, kemampuan perencanaan pedagogis, dan penggunaan TIK harus menjadi fokus utama *e-learning*. *E-learning* memiliki kemampuan untuk merubah budaya pembelajaran menjadi lebih menarik [13].

B. *Data Mining*

Data Mining merupakan sebuah ilmu terapan yang memiliki tujuan untuk menemukan sebuah pola pada sekumpulan data yang besar dan hasilnya digunakan dalam pengambilan keputusan [14].

C. *Text Mining*

Penambangan data merupakan proses ekstraksi data berbentuk teks. Tujuannya adalah untuk menemukan kata-kata (*term*) yang dapat mewakili isi dokumen sehingga memungkinkan analisis hubungan antar dokumen [15]. Melalui penerapan proses-proses *text mining*, pola data, tren, serta pengetahuan yang berpotensi dari data teks dapat diperoleh [16].

D. Klasifikasi

Klasifikasi merupakan proses pengelompokan objek, gagasan, atau peristiwa berdasarkan kemiripan fitur tertentu ke dalam kelompok yang berbeda. Klasifikasi juga merupakan bagian dari metode pembelajaran yang disebut supervised learning [17]. Algoritma yang umum digunakan dalam proses klasifikasi meliputi K-Nearest Neighbor, Neural Networks, *Decision/Classification Trees*, Naïve Bayes Classifiers, Analisis Statistik, Algoritma Genetika, Rough Sets, Memory Based Reasoning, serta Support Vector Machines (SVM) [18].

E. *Support Vector Machine*

Algoritma SVM menganalisis data untuk menemukan pola yang dapat digunakan dalam proses klasifikasi data. Sistemnya bekerja dengan memasukkan data tertentu ke dalam sistem dan kemudian memprediksinya. Data kemudian dikelompokkan ke dalam dua kelas yang berbeda sebelum dimasukkan ke dalam kelas baru. Dengan demikian, data selanjutnya dikelompokkan ke dalam kelas yang sesuai [19].

Konsep *Support Vector Machine* (SVM) dimulai dari permasalahan klasifikasi dua kelas, yang memerlukan training set positif dan negatif. SVM berupaya untuk menemukan pemisah terbaik, atau disebut *hyperplane*, yang memisahkan kedua kelas dengan margin maksimum. Pada kasus tertentu, metode linier pada Support Vector Machine tidak dapat melakukan klasifikasi data sehingga dikembangkan fungsi kernel dalam melakukan proses klasifikasi data dalam bentuk non-linier [20]. Berikut Persamaan (1-2) yang digunakan untuk menentukan klasifikasi soal.

$$x_i w + b \geq +1, y_1 = +1 \quad (1)$$

$$x_i w + b \leq +1, y_1 = -1 \quad (2)$$

Keterangan:
w = normal bidang
b = posisi bidang alternatif terhadap pusat kordinat.
Persamaan di atas digunakan dalam menentukan klasifikasi berdasarkan data yang akan dikelompokkan ke dalam kelas 1 atau kelas -1.

F. *Text Preprocessing*

Praproses teks merupakan langkah awal dalam *data mining* yang mengubah data tidak terstruktur menjadi data yang terstruktur dengan cara menghilangkan *noise*, menyamakan bentuk kata, dan mengurangi volume kata.

Empat tahapan utama dalam *text preprocessing* meliputi *Case Folding, Tokenizing, Stopword Removal*, dan *Stemming* [21].

G. TF-IDF

Metode TF-IDF merupakan teknik untuk menilai pentingnya sebuah kata pada dokumen dengan memberikan bobot berdasarkan hubungan kata tersebut. Teknik ini menggabungkan dua konsep yaitu frekuensi kemunculan kata pada dokumen (TF) dan seberapa umum kata tersebut di seluruh dokumen (IDF). Frekuensi kata dalam dokumen menunjukkan signifikansinya dalam dokumen tersebut, sementara frekuensi kemunculan kata di seluruh koleksi dokumen menggambarkan seberapa umum kata itu. Dengan demikian, bobot akan tinggi saat kata tersebut sering muncul pada dokumen tertentu namun jarang muncul di dokumen lain dalam koleksi. [22]. Berikut merupakan Persamaan (3-4) yang digunakan untuk perhitungan bobot TF-IDF.

$$w_{ij} = tf_{ij} \times idf \tag{3}$$

$$idf = \ln\left(\frac{N+1}{df_j+1}\right) + 1 \tag{4}$$

Keterangan:
i = jumlah variabel
j = jumlah data
w = bobot
N = total dokumen

H. Klasifikasi *Multi-Label*

Klasifikasi *multi-label* melibatkan pengelompokkan data ke dalam beberapa kategori berdasarkan karakteristik yang dimiliki. Dalam pendekatan *problem transformation*, masalah klasifikasi *multi-label* diubah menjadi *multi-class* [23]. Pada *problem transformation*, salah satu metode yang dapat digunakan dalam menangani kasus klasifikasi *multi-label* yaitu *label powerset*.

X	Y1	Y2	Y3
X1	1	0	0
X2	0	1	1
X3	0	0	1
X4	0	1	1
X5	1	0	0

➡

X	Class
X1	1
X2	2
X3	3
X4	2
X5	1

GAMBAR 1
Klasifikasi *Multi-Label*

Berdasarkan Gambar I, metode *label powerset* mengubah dataset *multilabel* menjadi dataset *multiclass* kemudian setiap kombinasi label yang berbeda dijadikan satu kategori baru dalam dataset *multiclass.*

I. *Confusion Matrix*

Confusion Matrix merupakan metode yang memberikan informasi tentang hasil kinerja dari klasifikasi yang telah dilakukan oleh sistem dengan cara mengukur seberapa baik akurasi dan efektivitas dari model klasifikasi [24]. Dalam contoh dengan dua kelas, biasanya disebut sebagai kelas positif (+) dan kelas negatif (-).Setelah mengetahui nilai TP, FN, FP, dan TN pada *confusion matrix*, maka dapat dihitung pula nilai pengukuran lainnya seperti *accuracy, precision, recall,* dan *f1-score.*

.

$$accuracy = \frac{TP + TN}{Total} \tag{5}$$

$$precission = \frac{TP}{TP + FP} \tag{6}$$

$$recall = \frac{TP}{TP + FN} \tag{7}$$

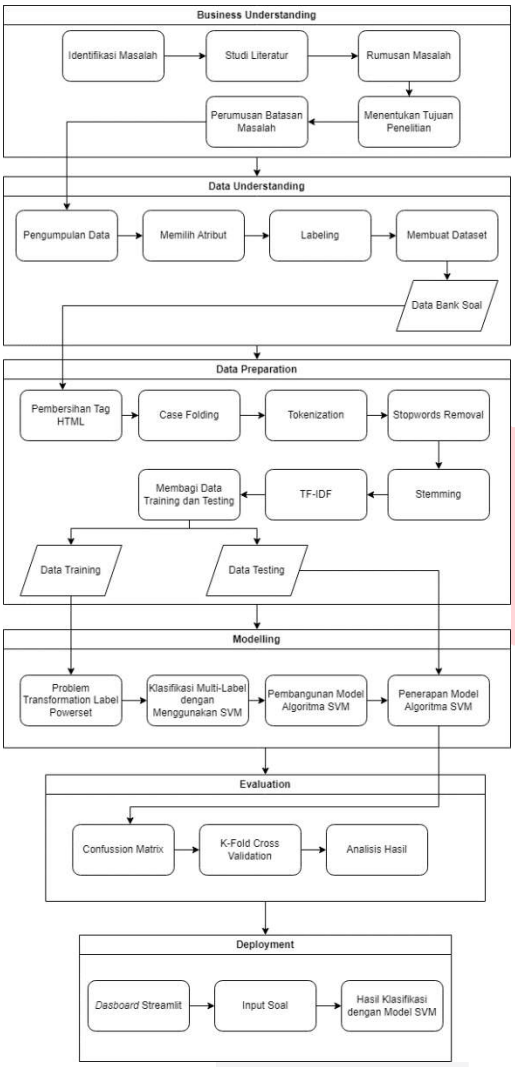
$$\text{F1 } score = 2 \times \frac{precission \times recall}{precission + recall} \tag{8}$$

J. *K-Fold Cross Validation*

K-fold cross validation merupakan metode validasi yang digunakan untuk mengevaluasi performa atau kinerja suatu algoritma dan meminimalkan waktu dalam proses klasifikasi sambil tetap menjaga tingkat keakuratan. Teknik ini melibatkan pembagian dataset ke dalam sejumlah "K" kelompok yang berbeda secara random. Tujuannya adalah untuk memastikan bahwa tidak ada tumpang tindih antara data yang digunakan untuk pengujian. Dengan *K-folds cross validation*, data dibagi menjadi "K" bagian yang sama besar, kemudian digunakan secara bergantian sebagai data uji dan data pelatihan. Validasi ini membantu dalam menguji model dengan menggunakan setiap bagian data sebagai pengujian dan pelatihan, sehingga memastikan pengujian yang lebih luas tanpa ada tumpang tindih di antara setiap bagian data [25].

III. METODE

Sistematika penelitian dalam studi ini mengikuti framework CRISP-DM (*Cross-Industry Standard Process for Data Mining*). CRISP-DM merupakan kerangka kerja dalam *data mining* yang dikembangkan pada tahun 1996 oleh lima perusahaan Integral Solutions Ltd (ISL), Teradata, Daimler AG, NCR Corporation, dan OHRA. Seiring berjalannya waktu, *framework* ini telah diperluas oleh banyak organisasi dan perusahaan di Eropa, menjadikannya metodologi standar *non-proprietary* dalam *data mining* [26]. CRISP-DM terdiri dari enam tahapan yang dijelaskan dalam Gambar 2.



GAMBAR 2
Sistematika Metode Penelitian Crisp-Dm

A. *Business Understanding*

Pada tahap *Business Understanding*, penelitian dimulai dengan identifikasi masalah untuk memahami permasalahan yang ingin diselesaikan. Selanjutnya dilakukan studi literatur guna memperoleh pemahaman mendalam mengenai konteks masalah. Setelah itu, dilakukan perumusan masalah untuk mendefinisikan secara spesifik permasalahan yang akan diteliti, dan dilanjutkan dengan menentukan tujuan penelitian yang akan dicapai. Selain itu, batasan masalah juga ditetapkan untuk menjaga penelitian tetap fokus dan terarah.

B. *Data Understanding*

Tahap *Data Understanding* melibatkan pengumpulan dan pemilihan data yang relevan untuk dianalisis. Dalam penelitian ini, peneliti mengumpulkan soal-soal ujian IPA untuk tingkat SMP dari *database* Pinterran. Data yang dikumpulkan disaring untuk memastikan hanya soal yang berbentuk teks pilihan ganda, berbahasa Indonesia, dan tidak mengandung persamaan rumus yang digunakan. Langkah selanjutnya adalah memilih atribut atau kolom yang relevan dari data yang telah dikumpulkan, melakukan pelabelan pada data, dan akhirnya membuat dataset yang siap untuk dianalisis lebih lanjut. Proses ini berfungsi validasi data yang digunakan dalam penelitian ini memiliki kualitas dan relevansi yang tinggi.

C. *Data Preparation*

Data yang telah dipersiapkan kemudian melalui tahap *Data Preparation* untuk memastikan bahwa data dalam kondisi siap untuk dianalisis. Tahap ini dimulai dengan pembersihan tag HTML. Kemudian diikuti dengan serangkaian proses *text preprocessing* untuk membersihkan dan menyaring data.

1. *Case Folding*

Case Folding merupakan proses mengubah semua huruf dalam dokumen menjadi *lowercase*. Pada tahap ini, hanya huruf dari "a" hingga "z" yang dikenali, sementara karakter selain huruf akan diabaikan dan dianggap sebagai pengindeks [27].

2. *Tokenization*

Tokenisasi merupakan tahapan membagi teks ke dalam beberapa bagian yang disebut token untuk analisis lebih lanjut. Hal ini melibatkan pemisahan angka, kata, simbol, tanda baca, serta elemen penting lainnya sebagai token [27].

3. *Stopword Removal*

Stopword Removal merupakan tahapan membuang kata-kata kurang penting berdasarkan *stoplist* (daftar kata yang dihapus) atau *wordlist* (daftar kata yang disimpan) [28].

4. *Stemming*

Stemming merupakan proses pengubahan kata menjadi bentuk baku. Contohnya, kata *'introducing'* akan disederhanakan menjadi *'introduc'* atau *'introduce'*. Terdapat beberapa algoritma yang bisa melakukan *stemming*, salah satunya adalah *porter stemmer* yang tersedia dalam *Natural Language Processing Toolkit* (NLTK) [29].

Selain itu, pada tahap persiapan data ini dilakukan proses *feature extraction*, fitur-fitur penting dari data yang telah diproses diekstraksi untuk digunakan dalam proses klasifikasi. Metode yang digunakan adalah TF-IDF yang bertujuan untuk menghitung pentingnya setiap kata dalam dokumen berdasarkan frekuensinya dan keunikannya dalam dataset. TF-IDF ini membantu dalam menentukan kata-kata mana yang paling relevan dan signifikan untuk dilakukan analisis. Dalam konteks klasifikasi soal ujian, fitur-fitur yang diekstraksi dengan TF-IDF akan digunakan untuk membedakan antara berbagai kategori soal. Setelah fitur diekstraksi, tahap berikutnya adalah *splitting data*. Pada tahap ini, data dibagi menjadi dua set yaitu *data training* dan *data testing*.

D. *Modelling*

Sebelum menggunakan algoritma klasifikasi, diterapkan metode *problem transformation* dengan menggunakan *label powerset*. Metode ini mengubah masalah klasifikasi *multi-label* menjadi *single-label* yang mencakup semua kombinasi label yang mungkin. Dalam penelitian ini, digunakan SVM untuk mengklasifikasikan soal-soal ujian ke dalam kategori yang sesuai berdasarkan fitur yang telah diekstraksi. Dengan membagi data menjadi *data training* dan *data testing*, peneliti dapat melatih model pada sebagian data dan menguji kinerjanya pada data yang belum pernah dilihat oleh model, sehingga dapat mengevaluasi keefektifan dan keandalan model dalam melakukan klasifikasi.

E. *Evaluation*

Tahap evaluasi dalam penelitian ini mengukur keandalan

dan efektivitas sistem klasifikasi menggunakan dua metode yaitu *Confusion Matrix* dan *K-Fold Cross Validation*. *Confusion Matrix* memberikan gambaran performa klasifikasi dengan menampilkan jumlah data yang diklasifikasikan dengan benar maupun salah untuk setiap kelas, membantu dalam memahami kinerja model terhadap dataset uji. Sementara itu, *K-Fold Cross Validation* membagi dataset menjadi k subset, di mana model dilatih pada k-1 subset dan diuji pada subset yang tersisa secara bergantian, untuk mengevaluasi stabilitas kinerja model secara keseluruhan. Evaluasi ini memberikan informasi tentang akurasi, *presisi, recall, dan f1-score*, serta mengukur efektivitas dan keandalan algoritma SVM dalam klasifikasi *multi-label* soal pilihan ganda berdasarkan topik..

F. *Deployment*

Tahap *Deployment* adalah implementasi model SVM yang telah dibangun ke dalam sebuah dasboard Streamlit. Pengguna dapat memasukkan input soal melalui *dashboard* ini. Soal yang diinput akan melalui proses *preprocessing* dan ekstraksi fitur menggunakan TF-IDF, kemudian diklasifikasikan oleh model SVM berdasarkan kategori yang relevan. Hasil klasifikasi ditampilkan di dasbor, memungkinkan pengguna untuk melihat prediksi kategori soal dengan mudah.

IV. HASIL DAN PEMBAHASAN

A. *Data Understanding*

Pengumpulan data dilakukan melalui database Pinterran, sebuah platform pembelajaran. Data yang diambil merupakan kumpulan soal kelas 8 SMP dengan fokus mata pelajaran IPA, yang tersedia dalam format pilihan ganda. Dari kumpulan soal tersebut, peneliti memilih untuk mengambil kolom yang relevan, yaitu 'konten soal', 'opsi pilihan ganda', dan 'tags'. *'tags*' ini merupakan label atau kategori yang mengidentifikasi topik dari setiap soal.

Data yang dikumpulkan kemudian dianalisis lebih lanjut dengan melakukan klasifikasi berdasarkan topik-topik yang terdapat dalam *'tags'*. Proses pengumpulan dan pemilihan data ini merupakan langkah awal dalam mempersiapkan dataset yang akan digunakan untuk kegiatan klasifikasi topik dalam konteks pembelajaran. Dengan demikian, data yang telah disiapkan ini akan membentuk dasar untuk analisis dan pemodelan lebih lanjut dalam penelitian ini.

Setelah data soal dikumpulkan, langkah selanjutnya adalah memprosesnya untuk mengidentifikasi topik-topik yang relevan. Konten soal yang telah terkumpul sebelumnya telah diberi label atau *tags* oleh ahli (guru, dosen, dan sebagainya) untuk mengidentifikasi topik-topik yang tercakup dalam setiap soal. Namun, untuk memastikan keberagaman dan representativitas dataset, hanya topik-topik yang memiliki lebih dari 40 konten soal yang dipilih untuk dilanjutkan dalam analisis. Setelah seleksi ini dilakukan, terdapat 10 topik teratas yang dipilih, yang secara total memiliki 488 konten soal. Topik-topik tersebut kemudian akan dijadikan sebagai label untuk proses klasifikasi selanjutnya. Dengan demikian, dataset telah disiapkan dengan baik untuk analisis, dengan mempertimbangkan keberagaman dan representasi dari topik-topik yang relevan dalam pembelajaran IPA untuk kelas 8 SMP. Berikut merupakan Tabel 1 yang memaparkan persebaran pada setiap label.

TABEL 1
Distribusi Label

Topik	Jumlah
listrik_statis	71
zat_adiktif_narkotika	61
objek_ipa_dan_pengamatannya	60
Gaya	59
Gelombang	57
Getaran	50
zat_adiktif_psikotropika	49
muatan_listrik	45
benda_langit	44
darah	42

B. *Data Preparation*

Pada tahap selanjutnya adalah tahap *data preparation*. Proses ini mencakup beberapa tahapan, mulai dari pembersihan data, *case folding*, *tokenization stopwords removal*, dan *stemming*. Tujuan dari *preprocessing* data adalah untuk mempersiapkan struktur data teks agar lebih terstruktur dan memudahkan dalam analisis klasifikasi, dengan harapan hasil analisis menjadi lebih akurat dan informatif.pada 2 merupakan sampel teks sebelum praproses pembersihan data.

TABEL 2
Sampel Teks Soal Sebelum *Preprocessing*

Soal	opsi1	opsi2	opsi3	opsi4
Mistar yang telah digosok dengan rambut kering akan bermuatan negatif karena....	Kehilangan proton	Kehilangan elektron	Penambahan proton	Penambahan elektron

Pada tahap ini, peneliti melakukan pembersihan teks HTML pada soal dan opsi pilihan gandanya, kemudian kolom “soal”, “opsi1”, “opsi2”, “opsi3”, “opsi4” akan dilakukan penggabungan. Tahap selanjutnya yaitu dilakukan *text preprocessing* yakni tahapan *case folding, tokenizing, stopword removal*, dan *stemming* yang dijabarkan pada Tabel 3

TABEL 3
Sampel Teks Soal Setelah *Preprocessing*

Penghapusan tag HTML	*Case Folding*	*Tokenizing*	*Stopword Removal*	*Stemming*
Mistar yang telah digosok dengan rambut kering akan bermuatan negatif karena Kehilangan proton Kehilangan elektron Penambahan proton Penambahan elektron	mistar yang telah digosok dengan rambut kering akan bermuatan negatif karena kehilangan proton kehilangan elektron penambahan proton penambahan elektron	['mistar', 'yang', 'telah', 'digosok', 'dengan', 'rambut', 'kering', 'akan', 'bermuatan', 'negatif', 'karena', 'kehilangan', 'proton', 'kehilangan', 'elektron', 'penambahan', 'proton', 'penambahan', 'elektron']	['mistar', 'digosok', 'rambut', 'kering', 'bermuatan', 'negatif', 'kehilangan', 'proton', 'kehilangan', 'elektron', 'penambahan', 'proton', 'penambahan', 'elektron']	['mistar', 'gosok', 'rambut', 'kering', 'muat', 'negatif', 'hilang', 'proton', 'hilang', 'elektron', 'tambah', 'proton', 'tambah', 'elektron']

Setelah dilakukan *text preprocessing*, selanjutnya peneliti melakukan *feature extraction* menggunakan TF-IDF. Tabel 4 merupakan sampel dokumen untuk perhitungan manual TF-IDF.

TABEL 4
Sampel Dokumen Sebelum Tf-Idf

Nomor Dokumen	Teks
1	kelaianan tulang sebab biasa duduk condong kifosis lordosis osteoporosis scoliosis

Dokumen 1 pada Tabel 4 akan dihitung nilai TF-IDF pada setiap atribut kata, pada Tabel 5 dijabarkan nilai akhir TF-IDF setelah dilakukan normalisasi.

TABEL 5
Hasil Perhitungan Tf-Idf

Preprocessing	TF-IDF
kelaianan", "tulang", "sebab" , "biasa", "duduk", "condong", "kifosis", "londosis", "osteoporosis", "scoliosis"	"biasa: 0.3359", "condong: 0.3502", "duduk: 0.2774", "kelaianan: 0.3502", "kifosis: 0.3158", "lordosis: 0.3502", "osteoporosis: 0.3502", "sebab: 0.2450", "skoliosis: 0.3248", "tulang: 0.2327"

Setelah fitur-fitur yang relevan diekstraksi menggunakan metode TF-IDF, langkah berikutnya adalah memisahkan data menjadi dua set yaitu data *training* dan data *testing*. Dalam penelitian ini, data dipisahkan dengan perbandingan 80% untuk data pelatihan dan 20% untuk data pengujian. Pada Tabel 6, dijabarkan distribusi label ada proses *splitting data* ini.

TABEL 6
Detail Jumlah Soal Dan Label Pada Tiap Dataset

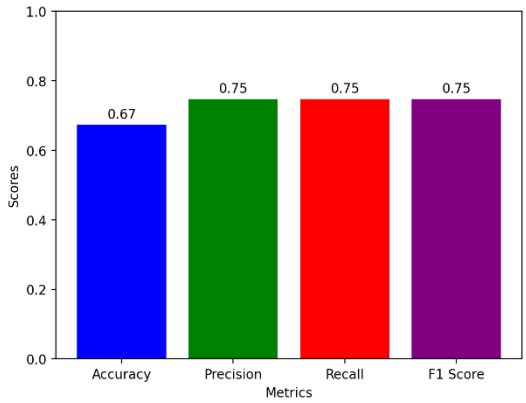
		Data		Total
		Training	Testing	
Jumlah Soal		431	107	538
Jumlah Penggunaan Label	listrik_statis	57	14	71
	zat_adiktif_narkotika	47	14	61
	objek_ipa_dan_pengamatannya	45	15	60
	gaya	50	9	59
	gelombang	46	11	57
	getaran	41	9	50
	zat_adiktif_psikotropika	38	11	49
	muatan_listrik	35	10	45
	benda_langit	38	6	44
	darah	34	8	42

C. *Modelling*

Modelling adalah langkah penting yang terjadi setelah seluruh proses data *preprocessing* dan *feature extraction* selesai dilakukan. Pada tahap ini, klasifikasi berdasarkan topik pada soal IPA kelas 8 dilakukan dengan algoritma *Support Vector Machine* (SVM). Dalam klasifikasi *multi-label*, setiap soal bisa dikategorikan ke dalam lebih dari satu label. Untuk mengatasi kasus ini, *problem transformation* dengan menggunakan metode *label powerset* diterapkan. *label powerset* adalah salah satu metode yang mengubah masalah klasifikasi *multi-label* menjadi klasifikasi *single-label* dengan memperlakukan setiap kombinasi label yang unik sebagai satu label tersendiri. Dengan demikian, model dapat dilatih untuk mengenali setiap kombinasi label tersebut.

D. Evaluasi Klasifikasi *Support Vector Machine*

Evaluasi kinerja algoritma SVM dilakukan menggunakan akurasi, presisi, *recall*, dan *F1-Score*. Akurasi menunjukkan proporsi prediksi yang benar dari total prediksi. Presisi mengukur ketepatan prediksi positif, sedangkan *recall* mengukur kemampuan model dalam mendeteksi data positif yang sebenarnya. *F1-Score* adalah rata-rata harmonis presisi dan recall, menggambarkan keseimbangan antara keduanya. Evaluasi ini memberikan pandangan komprehensif tentang efektivitas model dalam klasifikasi data.



GAMBAR 3
Evaluasi Klasifikasi Svm

Pada Gambar 3, hasil evaluasi model *Support Vector Machine* (SVM) ditunjukkan melalui empat metrik utama. Model SVM memiliki akurasi sebesar 0.67, yang berarti 67% dari prediksi model adalah benar. Presisi model mencapai 0.75, menandakan bahwa 75% dari prediksi positif adalah benar-benar positif. *Recall* juga berada di angka 0.75, menunjukkan bahwa 75% dari data yang sebenarnya positif berhasil diidentifikasi sebagai positif oleh model. *F1-Score*, yang menggabungkan presisi dan *recall*, juga sebesar 0.75, mencerminkan keseimbangan yang baik antara kedua metrik tersebut. Secara keseluruhan, hasil ini menunjukkan bahwa model SVM menunjukkan performa yang konsisten dalam klasifikasi, meskipun ada ruang untuk meningkatkan akurasi.

Untuk mengevaluasi performa model *Support Vector Machine* (SVM), digunakan *confusion matrix*. Matriks ini memetakan hasil prediksi model terhadap nilai sebenarnya dari data uji dalam beberapa kategori yang berbeda. Empat metrik utama yang diukur adalah *True Positive*, *False Positive*, *False Negative*, dan *True Negative* yang dijabarkan pada Tabel 7.

TABEL 7
Evaluasi Hasil *Confusion Matrix*

Label	*TP*	*FP*	*FN*	*TN*
benda_langit	4	1	2	91
darah	7	7	1	83
gaya	6	4	3	85
gelombang	7	0	4	87
getaran	7	0	2	89
listrik_statis	14	3	0	81
muatan_listrik	6	3	4	85
objek_ipa_dan_pengamatannya	13	1	2	82
zat_adiktif_narkotika	13	8	1	76

zat_adiktif_psikotropika	3	0	8	87
Total	**80**	**27**	**27**	**846**

Dari hasil *confusion matrix* pada Tabel 7, terlihat bahwa performa model SVM dalam mengklasifikasikan berbagai topik cukup bervariasi. Label "listrik_statis" memiliki jumlah *true positive* tertinggi (14), menunjukkan bahwa model sangat efektif dalam mengidentifikasi topik ini dengan benar. Namun, label "zat_adiktif_narkotika" memiliki jumlah *false positive* tertinggi (8), yang berarti model sering salah mengklasifikasikan soal dari topik lain sebagai zat adiktif narkotika. Selain itu, label "zat_adiktif_psikotropika" memiliki jumlah *false negative* tertinggi (8), menunjukkan bahwa model sering gagal mengenali data soal yang benar-benar termasuk dalam kategori ini sebagai positif. Secara keseluruhan, total prediksi yang benar (*true positive* + *true negative*) adalah 926, sedangkan total prediksi yang salah (*false positive* + *false negative*) adalah 54, menandakan bahwa meskipun ada beberapa kesalahan, model memiliki performa yang cukup baik.

E. Evaluasi *K-Fold Cross Validation*

Pada penelitian ini, digunakan K=5 , di mana data dibagi menjadi 5 *fold* dan model dilatih serta diuji sebanyak 5 kali.

TABEL 8
Evaluasi *K-Fold Cross Validation*

K	*Accuracy*	*Precision*	*Recall*	*F1 - Score*
1	0.6735	0.7477	0.7477	0.7477
2	0.6327	0.6729	0.6857	0.6792
3	0.6224	0.6827	0.6827	0.6827
4	0.5876	0.6981	0.6549	0.6758
5	0.6082	0.7087	0.6697	0.6887
Average	**0.6249**	**0.7015**	**0.6872**	**0.6947**

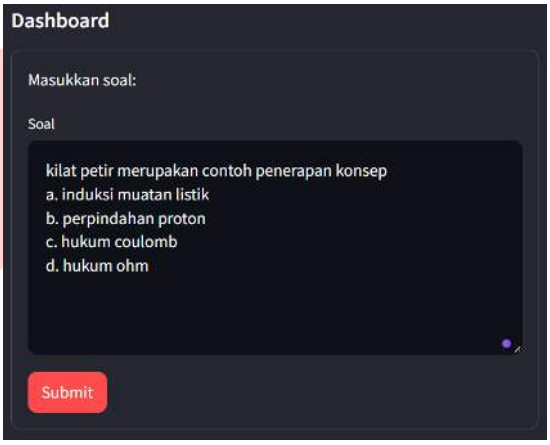
Tabel 8 menunjukkan nilai akurasi, presisi, *recall,* dan *F1-Score* untuk setiap *fold*. Pada *fold* pertama, akurasi adalah 0.6735, dengan presisi, *recall*, dan *F1-Score* masing-masing 0.7477. Pada *fold* kedua, akurasi turun menjadi 0.6327, dengan presisi 0.6729, *recall* 0.6857, dan *F1-Score* 0.6792. *Fold* ketiga memiliki akurasi 0.6224, dengan presisi, *recall*, dan *F1-Score* masing-masing 0.6827. *Fold* keempat menunjukkan akurasi 0.5876, presisi 0.6981, *recall* 0.6549, dan *F1-Score* 0.6758. Fold kelima memiliki akurasi 0.6082, presisi 0.7087, *recall* 0.6697, dan *F1-Score* 0.6887.

Penurunan nilai pada empat metrik yang terlihat dari *fold* pertama hingga *fold* terakhir dalam Tabel 8 menunjukkan bahwa model sensitif terhadap variasi dalam data. Performa model menurun seiring dengan perubahan subset data yang digunakan untuk validasi. Sensitivitas ini mengindikasikan bahwa model mungkin kurang mampu menangani variasi dalam pola data, yang bisa disebabkan oleh ketidakseimbangan data atau kompleksitas model yang tidak cukup memadai untuk menyesuaikan diri dengan perubahan subset data.

Performa terbaik terlihat pada fold pertama (k=1), namun seiring bertambahnya jumlah fold, nilai metrik mengalami penurunan yang konsisten. Hal ini memperlihatkan bahwa meskipun fold pertama menunjukkan hasil yang baik, model mungkin tidak memiliki generalisasi yang kuat.

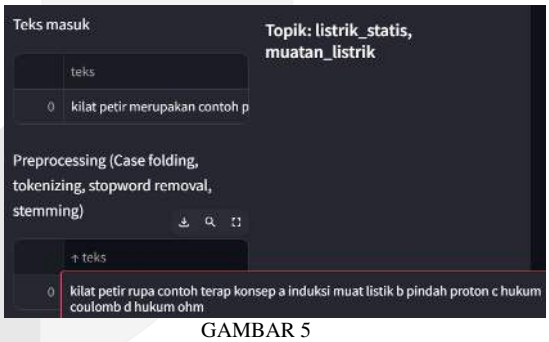
F. *Deployment* Model Klasifikasi

Pada tahap ini, peneliti menggunakan *dashboard* yang dibuat dengan *tools* Streamlit untuk melakukan *deployment* model klasifikasi *multilabel* pada soal IPA. Model yang digunakan pada penelitian ini adalah *Support Vector Machine* (SVM) yang telah dilatih sebelumnya. *Dashboard* ini dirancang untuk memprediksi kategori topik dari soal IPA berdasarkan input yang diberikan oleh pengguna.



GAMBAR 4
Dashboard Input Soal

Pengguna dapat memasukkan soal melalui antarmuka dashboard, seperti yang ditunjukkan pada Gambar 4. Setelah soal diinputkan, pengguna cukup menekan tombol "Submit," dan model akan melakukan prediksi untuk menentukan kategori label yang sesuai dengan soal tersebut.



GAMBAR 5
Dashboard Hasil Prediksi

Gambar 5 menunjukan hasil prediksi dari input soal yang telah dimasukkan yaitu label 'listrik_statis' dan 'muatan_listrik. Proses prediksi dimulai dengan melakukan *preprocessing* teks pada soal tersebut. Tahapan praproses teks mencakup *case folding*, tokenisasi, *stopword removal*, dan *stemming*. Setelah itu, teks yang telah diproses akan dinilai bobot kepentingannya pada setiap kata dengan menggunakan metode TF-IDF sebelum akhirnya dimasukkan ke model SVM untuk prediksi kategori label yang sesuai.

V. KESIMPULAN

Berdasarkan penelitian ini, implementasi algoritma SVM untuk klasifikasi soal *multi-label* pada mata pelajaran IPA

tingkat SMP dimulai dengan tahap *data preparation*, yang mencakup pembersihan data, *case folding,* tokenisasi, *stopword removal*, dan *stemming*. Selanjutnya, ekstraksi fitur dilakukan menggunakan TF-IDF untuk mengubah teks menjadi representasi numerik yang dapat diproses oleh model. Pemodelan data dilakukan dengan metode *label powerset*, yang mengubah masalah *multi-label* menjadi beberapa masalah klasifikasi biner. Evaluasi performansi model menunjukkan bahwa metode ini dapat diterapkan untuk klasifikasi soal *multi-label*.

Namun, tingkat akurasi yang dihasilkan dari penerapan metode SVM dalam klasifikasi soal *multi-label* pada dataset yang digunakan menunjukkan hasil yang kurang memadai dalam konteks pendidikan. Evaluasi menunjukkan bahwa akurasi model hanya mencapai 67%, sedangkan nilai presisi, *recall,* dan *F1-score* masing-masing sebesar 75%. Meskipun *precision, recall*, dan *F1-score* menunjukkan kinerja yang lebih konsisten, akurasi 67% mengindikasikan bahwa model SVM masih belum cukup efektif dalam mencapai tingkat ketepatan yang tinggi. Hasil ini menunjukkan bahwa, meskipun model SVM memiliki potensi, masih diperlukan upaya perbaikan yang signifikan untuk meningkatkan performa dan keandalannya dalam klasifikasi soal *multi-label*.

REFERENSI

[1] Yayan Alpian, Sri Wulan Anggraeni, Unika Wiharti, and Nizmah Maratos Soleha, "PENTINGNYA PENDIDIKAN BAGI MANUSIA," JURNAL BUANA PENGABDIAN, vol. 1, no. 1, pp. 66–72, Aug. 2019, doi: 10.36805/jurnalbuanapengabdian.v1i1.581.

[2] S. Dan and A. B. L. Mailangkay, "PENERAPAN E-LEARNING SEBAGAI ALAT BANTU MENGAJAR DALAM DUNIA PENDIDIKAN," Jurnal Ilmiah Widya, vol. 3, 2016.

[3] N. A. Yulistiawati, "Pentingnya Motivasi Peserta Didik terhadap Hasil Belajar Biologi.," Seminar Nasioal Biologi VI, 2019, pp. 1–4.

[4] K. E. Widiastuti, N. I., Rainarli, E., & Dewi, "Peringkasan dan support vector machine pada klasifikasi dokumen," Jurnal Infotel, vol. 9(4), pp. 416–421, 2017, doi: https://doi.org/10.20895/infotel.v9i4.312.

[5] R. Ariandi, O. N. Pratiwi, and R. Y. Fa'rifah, "Klasifikasi Soal Sejarah Tingkat SMA Berdasarkan Level Kognitif Revised Bloom's Taxonomy Menggunakan Algoritma K-Nearest Neighbour Manhattan," 2023.

[6] L. A. Muhaimin, O. N. Pratiwi, and R. Y. Fa'rifah, "Klasifikasi Soal Berdasarkan Kategori Topik Menggunakan Metode Algoritma Naïve Bayes Dan Algoritma C4.5," e-Proceeding of Engineering, vol. Vol.10, No.2, pp. 1535–1541, 2023.

[7] A. Y. Taha and S. Tiun, "Binary relevance (BR) method classifier of multi-label classification for arabic text," J Theor Appl Inf Technol, vol. 84, no. 3, 2016.

[8] A. P. Wibawa, M. G. A. Purnama, M. F. Akbar, and F. A. Dwiyanto, "Metode-metode Klasifikasi," Prosiding Seminar Ilmu Komputer dan Teknologi Informasi, vol. 3, pp. 134–138, 2018.

[9] A. Handayanto, K. Latifa, N. D. Saputro, and R. R. Waliyansyah, "Analisis dan Penerapan Algoritma Support Vector Machine (SVM) dalam Data Mining untuk Menunjang Strategi Promosi (Analysis and Application of Algorithm Support Vector Machine (SVM) in Data Mining to Support Promotional Strategies)," 2019.

[10] A. Sangodiah et al., "MINIMIZING STUDENT ATTRITION IN HIGHER LEARNING INSTITUTIONS IN MALAYSIA USING SUPPORT VECTOR MACHINE," J Theor Appl Inf Technol, vol. 31, no. 3, 2015, [Online]. Available: www.jatit.org

[11] A. D. Panicker and S. Venkitakrishnan, "Question Classification using Machine Learning Approaches," 2012.

[12] N. Wirani, "Pentingnya Penggunaan Model Pembelajaran Berbasis Web Untuk Mencegah Penyebaran Covid-19," Al'adzkiya International of Education and Sosial, vol. 1, no. 1, 2020.

[13] N. L. U. Chusna, "PEMBELAJARAN E-LEARNING," Prosiding Seminar Nasional Pendidikan KALUNI, vol. 2, Feb. 2019, doi: 10.30998/prokaluni.v2i0.36.

[14] R. Yanto, "Implementasi Data Mining Estimasi Ketersediaan Lahan Pembuangan Sampah menggunakan Algoritma Simple Linear Regression," Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), vol. 2, no. 1, pp. 361–366, Apr. 2018, doi: 10.29207/resti.v2i1.282.

[15] Husni and B. Arifin, "RANCANGAN DAN IMPLEMENTASI APLIKASI PENCARIAN TEKS TERJEMAHAN AL QUR'AN BERBASIS MODEL RUANG VEKTOR," Jurnal SimanteC, vol. Vol. 7, No.2, pp. 90–96, 2019.

[16] N. S. Wardani, A. Prahutama, and P. Kartikasari, "ANALISIS SENTIMEN PEMINDAHAN IBU KOTA NEGARA DENGAN KLASIFIKASI NAÏVE BAYES UNTUK MODEL BERNOULLI DAN MULTINOMIAL," 2020, [Online]. Available: https://ejournal3.undip.ac.id/index.php/gaussian/

[17] A. D. Wibisono, S. Dadi Rizkiono, and A. Wantoro, "FILTERING SPAM EMAIL MENGGUNAKAN METODE NAIVE BAYES," TELEFORTECH : Journal of Telematics and Information Technology, vol. 1, no. 1, 2020, doi: 10.33365/tft.v1i1.685.

[18] H. Annur, "KLASIFIKASI MASYARAKAT MISKIN MENGGUNAKAN METODE NAÏVE BAYES," 2018.

[19] S. K. Lidya, O. S. Sitompul, and S. Efendi, "Sentiment Analysis pada Teks Bahasa Indonesia Menggunakan Support Vector Machine (SVM) dan K-Nearest Neighbor (K-NN)," Seminar Nasional Teknologi Informasi dan Komunikas, 2015 (SENTIKA 2015), 2015.

[20] A. Pratama, R. Cahya Wihandika, and D. E. Ratnawati, "Implementasi Algoritme Support Vector Machine (SVM) untuk Prediksi Ketepatan Waktu Kelulusan Mahasiswa," 2018. [Online]. Available: http://j-ptiik.ub.ac.id

[21] S. Khomsah and A. S. Aribowo, "Model Text-Preprocessing Komentar Youtube Dalam Bahasa Indonesia," JURNAL RESTI, vol. 1, no. 3, 2017.

[22] Imamah and F. H. Rachman, "Twitter Sentiment Analysis of Covid-19 Using Term Weighting TF-IDF And Logistic Regresion," in 2020 6th Information Technology International Seminar (ITIS), 2020, pp. 238–242. doi: 10.1109/ITIS50118.2020.9320958.

[23] E. E. Fitriani and W. Yustanti, "Perbandingan Kinerja Metode Problem Transformation-KNN dan Algorithm Adaptation-KNN pada Klasifikasi Multi-Label Pertanyaan Kotakode," Journal of Emerging Information Systems and Business Intelligence, vol. 3, no. 3, 2022.

[24] D. Ananda and R. R. Suryono, "JURNAL MEDIA INFORMATIKA BUDIDARMA Analisis Sentimen Publik Terhadap Pengungsi Rohingya di Indonesia dengan Metode Support Vector Machine dan Naïve Bayes," Jurnal Media Informatika Budidarma, vol. 8, no. 2, pp. 748–757, 2024, doi: 10.30865/mib.v8i2.7517.

[25] O. Arifin and T. B. Sasongko, "Seminar Nasional Teknologi Informasi dan Multimedia," UNIVERSITAS AMIKOM Yogyakarta, 2018.

[26] D. T. Larose, Data Mining Methods and Models. 2006. doi: 10.1002/0471756482.

[27] D. Juang, "ANALISIS SPAM DENGAN MENGGUNAKAN NAÏVE BAYES," 2016.

[28] A. E. Budiman and A. Widjaja, "Analisis Pengaruh Teks Preprocessing Terhadap Deteksi Plagiarisme Pada Dokumen Tugas Akhir," Jurnal Teknik Informatika dan Sistem Informasi, vol. 6, no. 3, Dec. 2020, doi: 10.28932/jutisi.v6i3.2892.

[29] I. R. Vanani, "Text analytics of customers on twitter: Brand sentiments in customer support," Journal of Information Technology Management, vol. 11, no. 2, pp. 43–58, 2019, doi: 10.22059/JITM.2019.291087.2410.