ITERA

PEMODELAN KLASIFIKASI PRODUK PADA E-KATALOG LEMBAGA KEBIJAKAN PENGADAAN BARANG/JASA PEMERINTAH KOTA BANDAR LAMPUNG MENGGUNAKAN METODE K-NEAREST NEIGHBORS

NASKAH SKRIPSI

Robby Hidayah Ramadhan
120450033

PROGRAM STUDI SAINS DATA
FAKULTAS SAINS
INSTITUT TEKNOLOGI SUMATERA
LAMPUNG SELATAN

2024

PEMODELAN KLASIFIKASI PRODUK PADA E-KATALOG LEMBAGA KEBIJAKAN PENGADAAN BARANG/JASA PEMERINTAH KOTA BANDAR LAMPUNG MENGGUNAKAN METODE K-NEAREST NEIGHBORS

NASKAH SKRIPSI
Diajukan sebagai syarat maju sidang tugas akhir

Robby Hidayah Ramadhan
120450033

PROGRAM STUDI SAINS DATA
FAKULTAS SAINS
INSTITUT TEKNOLOGI SUMATERA
LAMPUNG SELATAN

2024

HALAMAN PENGESAHAN

Naskah Skripsi untuk Sidang Akhir dengan judul "**Pemodelan Klasifikasi Produk pada E-Katalog Lembaga Kebijakan Pengadaan Barang/Jasa Pemerintah Kota Bandar Lampung Menggunakan Metode K-Nearest Neighbors**" adalah benar dibuat oleh saya sendiri dan belum pernah dibuat dan diserahkan sebelumnya, baik sebagian ataupun seluruhnya, baik oleh saya ataupun orang lain, baik di Institut Teknologi Sumatera maupun di institusi pendidikan lainnya.

Lampung Selatan, 26 November 2024

Penulis,

Robby Hidayah Ramadhan
NIM. 120450033

Diperiksa dan disetujui oleh,

Pembimbing I	Pembimbing II
Adhi Rahmadian, M.Si	**Luluk Muthoharoh, M.Si**
	NIP. 199504112022032014

Disahkan oleh,

Koordinator Program Studi Sains Data
Fakultas Sains
Institut Teknologi Sumatera

Tirta Setiawan, S.Pd., M.Si
NIP. 199008222022031003

Sidang Tugas Akhir :

Penguji I : Mika Alvionita S, M.Si
Penguji II : Christyan Tamaro Nadeak, M.Si

HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah karya saya sendiri dan semua sumber baik yang dikutip maupun yang dirujuk telah saya nyatakan benar.

Nama	:	**Robby Hidayah Ramadhan**
NIM	:	**120450033**
Tanda tangan	:	
Tanggal	:	**26 November 2024**

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI UNTUK KEPENTINGAN AKADEMIS

Sebagai civitas akademik Institut Teknologi Sumatera, saya yang bertanda tangan di bawah ini:

Nama	:	Robby Hidayah Ramadhan
NIM	:	120450033
Program Studi	:	Sains Data
Fakultas	:	Sains
Jenis karya	:	Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan Hak Bebas Royalti Noneksklusif (*Non-Exclusive Royalty Free Right*) kepada Institut Teknologi Sumatera atas karya ilmiah saya yang berjudul:

Pemodelan Klasifikasi Produk pada E-Katalog Lembaga Kebijakan Pengadaan Barang/Jasa Pemerintah Kota Bandar Lampung Menggunakan Metode K-Nearest Neighbors

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Institut Teknologi Sumatera berhak menyimpan, mengalihmedia/format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di	:	Lampung Selatan
Pada tanggal	:	26 November 2024

Yang menyatakan

Robby Hidayah Ramadhan

ABSTRAK

Pemodelan Klasifikasi Produk pada E-Katalog Lembaga Kebijakan Pengadaan Barang/Jasa Pemerintah Kota Bandar Lampung Menggunakan Metode K-Nearest Neighbors

Robby Hidayah Ramadhan (120450033)
Pembimbing I: Adhi Rahmadian, M.Si
Pembimbing II: Luluk Muthoharoh, M.Si

Pengadaan barang dan jasa oleh pemerintah telah mengalami modernisasi melalui platform e-katalog yang dikelola oleh Lembaga Kebijakan Pengadaan Barang/Jasa Pemerintah (LKPP). Namun, salah satu tantangan utama adalah penyedia masih mengklasifikasikan etalase produk secara manual, yang dapat menyebabkan kesalahan dalam pengelompokan produk. Penelitian ini mengembangkan model klasifikasi otomatis untuk etalase produk pada e-katalog LKPP di Kota Bandar Lampung dengan menggunakan metode *K-Nearest Neighbors* (*KNN*). Peneliti mengumpulkan data dari *e-katalog* LKPP selama periode 15 Juli hingga 15 Agustus 2024, dengan total 98.225 produk yang tersebar dalam 18 etalase. Penelitian ini melakukan tahapan praproses data, yaitu *lowercasing*, *text cleansing*, *tokenization*, dan *custom word removal*, serta mengekstraksi fitur menggunakan *Term Frequency-Inverse Document Frequency* (*TF-IDF*). Peneliti menggunakan *Grid Search* dan *Stratified K-Fold Cross Validation* (*SKCV*) untuk menentukan *hyperparameter* optimal. Model *KNN* dengan *hyperparameter* $k = 4$ dan pembobotan berbasis jarak (*distance weighting*) mencapai akurasi sebesar 90,54%, dengan *macro average precision*, *recall*, dan *F1-score* yang konsisten di atas 90%. Hasil ini membuktikan bahwa metode *KNN* mampu mengklasifikasikan produk secara efektif dalam *e-katalog* LKPP. Temuan ini diharapkan dapat mendukung percepatan realisasi Instruksi Presiden Nomor 2 Tahun 2022 dan mendorong pemberdayaan Usaha Mikro, Kecil, dan Menengah (UMKM) dalam sistem pengadaan elektronik.

Kata kunci: E-katalog, Klasifikasi Etalase Produk, Klasifikasi, Pengadaan Barang/Jasa Pemerintah

ABSTRACT

Product Classification Modeling on the E-Catalog of the Procurement Policy Agency for Goods/Services of Bandar Lampung City Government Using the K-Nearest Neighbors Method

Robby Hidayah Ramadhan (120450033)
Advisor I : Adhi Rahmadian, M.Si
Advisor II: Luluk Muthoharoh, M.Si

The procurement of goods and services by the government has been modernized through the e-catalog platform managed by the National Public Procurement Agency (LKPP). However, one of the main challenges is that suppliers still classify product catalogs manually, which can lead to errors in product categorization. This study developed an automatic classification model for product catalogs in the LKPP e-catalog in Bandar Lampung City using the K-Nearest Neighbors (KNN) method. Researchers collected data from the LKPP e-catalog during the period of July 15 to August 15, 2024, comprising a total of 98,225 products distributed across 18 catalogs. This study performed data preprocessing steps, including lowercasing, text cleansing, tokenization, and custom word removal, and extracted features using Term Frequency-Inverse Document Frequency (TF-IDF). The researchers employed Grid Search and Stratified K-Fold Cross Validation (SKCV) to determine the optimal hyperparameters. The KNN model with a hyperparameter value of k = 4 and distance-based weighting achieved an accuracy of 90.54%, with macro average precision, recall, and F1-score consistently above 90%. These results demonstrate that the KNN method can effectively classify products in the LKPP e-catalog. This finding is expected to support the acceleration of the implementation of Presidential Instruction Number 2 of 2022 and promote the empowerment of Micro, Small, and Medium Enterprises (UMKM) within the electronic procurement system.

Keywords : *E-Catalog, Product Storefront Classification, Classification, Government Procurement.*

HALAMAN PERSEMBAHAN

"This work is dedicated to those who have made this journey meaningful"

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT atas rahmat dan berkah-Nya sehingga skripsi ini dapat diselesaikan. Penyusunan skripsi ini tidak lepas dari dukungan berbagai pihak. Untuk itu, penulis mengucapkan terima kasih kepada:

1. Mama Atin, Papa Wawan, dan Kakak Galih atas doa, dukungan tanpa henti, kasih sayang yang tulus, dan pengorbanan yang tiada tara. Dukungan dari keluarga telah menjadi kekuatan terbesar bagi saya dalam menyelesaikan pendidikan ini.
2. Bapak Adhi Rahmadian, M.Si dan Ibu Luluk Muthoharoh, M.Si, selaku dosen pembimbing, yang dengan sabar telah memberikan bimbingan, arahan, serta masukan yang membangun. Terima kasih atas kesabaran, ilmu, dan motivasi yang diberikan dalam setiap tahap penyusunan skripsi ini.
3. Ibu Mika Alvionita, S., M.Si dan Bapak Christyan Tamaro Nadek, M.Si, selaku dosen penguji, atas kritik, saran, dan masukan yang sangat berharga untuk penyempurnaan skripsi ini. Terima kasih atas kesediaan waktu dan perhatian yang telah diberikan selama proses ujian.
4. Naomi Natasya, atas kebersamaan, semangat, dan dukungan yang selalu hadir di saat-saat penting. Terima kasih telah menjadi sesorang yang selalu ada di setiap langkah perjalanan ini.
5. Hafiz, Andika, Alif, dan Rizky, atas persahabatan yang telah terjalin selama delapan tahun. Terima kasih atas tawa, kebersamaan, dan dukungan yang tak pernah putus hingga saya menyelesaikan pendidikan ini.
6. Teman-teman Sains Data 2020 dan semua pihak yang tidak dapat disebutkan satu per satu.

Semoga skripsi ini bermanfaat, dan penulis memohon maaf atas segala kekurangan.

Lampung Selatan, 26 November 2024

Robby Hidayah Ramadhan

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1 Latar Belakang

Peraturan Presiden (Perpres) Nomor 12 Tahun 2021 menyebutkan bahwa Pengadaan Barang/Jasa Pemerintah (PBJP) merupakan prosedur penting yang dilaksanakan oleh lembaga dan instansi pemerintah untuk memperoleh barang atau jasa yang dibutuhkan [1]. Pada tahun 2024, Pemerintah Indonesia mengalokasikan anggaran untuk kegiatan PBJP mencapai Rp1.226 triliun, dengan memprioritaskan pembelanjaan kepada produk dalam negeri [2]. Hal ini sesuai dengan tujuan dari PBJP yang tertuang dalam Perpres Nomor 12 Tahun 2021, yaitu PBJP dilakukan untuk meningkatkan peran serta Usaha Mikro, Usaha Kecil, dan Koperasi (UMK) dalam ekosistem pengadaan pemerintah sehingga dapat mendorong pertumbuhan ekonomi Indonesia dan keberlanjutan UMKM [1].

Kebutuhan modernisasi dan digitalisasi dalam kegiatan PBJP mendorong pemerintah untuk mengeluarkan Instruksi Presiden (Inpres) Nomor 2 Tahun 2022 [3]. Inpres ini menginstruksikan peningkatan jumlah produk dalam e-katalog menuju 1.000.000 produk, terutama produk dalam negeri. E-katalog adalah solusi berbasis teknologi yang dirancang untuk mengoptimalkan proses pengadaan barang dan jasa melalui *platform* elektronik yang dikelola oleh Lembaga Kebijakan Pengadaan Barang/Jasa Pemerintah (LKPP) [4]. Kota-kota dengan jumlah UMKM yang besar, seperti Kota Bandar Lampung, memiliki potensi signifikan untuk berkontribusi dalam pengadaan barang/jasa pemerintah melalui platform ini. Pada tahun 2021, Kota Bandar Lampung mencatat terdapat 118.533 unit UMKM di wilayahnya [5]. Jumlah UMKM yang besar ini menunjukkan potensi signifikan untuk meningkatkan kontribusi UMKM dalam pengadaan barang/jasa pemerintah, yang pada gilirannya dapat mendorong pertumbuhan ekonomi lokal dan nasional.

Keputusan Deputi Bidang Monitoring, Evaluasi, dan Pengembangan Sistem Informasi Nomor 7 Tahun 2020 tentang Panduan Penggunaan Aplikasi Katalog Elektronik Versi 5.0 menunjukkan bahwa tidak adanya sistem rekomendasi etalase produk pada proses pencantuman produk penyedia menyebabkan penjual atau mitra e-katalog harus menentukan etalase produk mereka secara mandiri. Kondisi ini berpotensi menyebabkan kesulitan bagi penyedia atau mitra e-katalog dalam menentukan etalase produknya. Oleh karena itu, muncul kebutuhan untuk

pengembangan model *machine learning* yang dapat memberikan rekomendasi etalase produk e-katalog berdasarkan nama produk.

Berbagai studi sebelumnya telah mengemukakan pendekatan dan metodologi untuk klasifikasi kategori (etalase) produk secara otomatis menggunakan model *machine learning*. Penelitian oleh Mathivanan, Ghani, dan Janor (2019) menerapkan algoritma seperti Naive Bayes, *K-Nearest Neighbor* (KNN), *Decision Tree*, *Support Vector Machine* (SVM), dan *Random Forest* pada data judul produk *e-commerce*, menunjukkan bahwa KNN mencapai akurasi tertinggi dengan 94,66% untuk set data *Household* dan 82,96% untuk set data *Fresh Food* [6]. Penelitian oleh Sebastian (2019) menggunakan KNN pada 150 data produk dari e-marketplace Tokopedia dan Bukalapak, menghasilkan akurasi tertinggi sebesar 97,33% [7]. Studi oleh Hassan, Ahamed, dan Ahmad (2022) membandingkan berbagai algoritma *machine learning* termasuk SVM, KNN, *Logistic Regression*, *Multinomial Naive Bayes*, dan *Random Forest* pada set data IMDB dan SPAM, dengan KNN menunjukkan akurasi 98,5% pada set data SPAM [8].

Berdasarkan temuan permasalahan yang telah diuraikan, penelitian ini dilakukan sebagai upaya untuk mendukung Inpres Nomor 2 Tahun 2022 tentang percepatan peningkatan penggunaan produk dalam negeri serta pemberdayaan UMKM. Hasil penelitian terdahulu yang diuraikan pada paragraf sebelumnya menunjukkan bahwa algoritma *K-Nearest Neighbors* (KNN) mampu memprediksi kategori (etalase) produk dengan tingkat kesalahan yang rendah. Oleh karena itu, fokus penelitian ini adalah pada pemodelan *machine learning* untuk klasifikasi etalase e-katalog LKPP di Kota Bandar Lampung dengan menggunakan algoritma KNN. Algoritma KNN menentukan label objek berdasarkan mayoritas kelas di antara k objek terdekat dalam set data pelatihan, di mana "k objek" merujuk pada k tetangga terdekat yang dipertimbangkan untuk menentukan kelas objek. Dalam penelitian ini, jarak antara objek dihitung menggunakan jarak *euclidean*. Penentuan nilai k pada KNN menggunakan *Grid Search Stratified K-Fold Cross-Validation*, yaitu menguji berbagai nilai K untuk algoritma KNN dan menentukan nilai K yang optimal berdasarkan kinerja rata-rata model pada setiap *fold*.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka peneliti merumuskan pertanyaan penelitian sebagai berikut:

1. Bagaimana pemodelan klasifikasi etalase produk menggunakan metode K-Nearest Neighbor pada e-katalog LKPP di Kota Bandar Lampung?

2. Bagaimana hasil evaluasi model klasifikasi etalase produk menggunakan metode K-Nearest Neighbor pada e-katalog LKPP di Kota Bandar Lampung?
3. Bagaimana distribusi kesalahan prediksi model?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah, maka penelitian ini bertujuan untuk:

1. Pembuatan model klasifikasi menggunakan metode *K-Nearest Neighbor* (KNN) yang dapat secara efektif mengklasifikasikan etalase produk pada e-katalog LKPP khususnya di Kota Bandar Lampung.
2. Menganalisis evaluasi model dari *K-Nearest Neighbor* (KNN) pada klasifikasi etalase produk e-katalog LKPP di Kota Bandar Lampung.
3. Identifikasi penyebab kesalahan klasifikasi pada etalase dengan kesalahan klasifikasi tertinggi.

1.4 Batasan Masalah

Batasan Masalah dari penelitian ini diantaranya:

1. Objek studi kasus penelitian ini adalah e-katalog LKPP Kota Bandar Lampung.
2. Penelitian ini hanya menghasilkan model klasifikasi yang performanya baik, sehingga tidak termasuk proses integrasi model tersebut ke dalam sistem input produk LKPP.
3. Data yang diambil melalui *scraping* dari platform e-katalog LKPP hanya mencakup informasi yang diperlukan untuk penelitian ini, yaitu nama produk dan etalasenya. Informasi yang menyangkut identitas individu tidak akan dikumpulkan, memastikan privasi pengguna tetap terlindungi.

1.5 Manfaat Penelitian

1. Sebagai bentuk mendukung realisasi Instruksi Presiden Nomor 2 Tahun 2022 tentang percepatan peningkatan penggunaan produk dalam negeri serta pemberdayaan Usaha Mikro, Kecil, dan Menengah (UMKM).
2. Ditinjau dari perspektif akademis, penelitian ini diharapkan menjadi referensi untuk pengembangan model klasifikasi yang lebih efisien, efektif, dan adaptif dalam pengadaan barang/jasa pemerintah di masa depan.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian oleh Norsyela Muhammad Noor Mathivanan, Nor Azura Md. Ghani, dan Roziah Mohd Janor (2019) menggunakan dua set data, *Fresh Food* dan *Household*, dengan berbagai metode klasifikasi seperti *Naive Bayes*, *K-Nearest Neighbors* (KNN), *Decision Tree*, *Support Vector Machine*, dan *Random Forest*. Hasil menunjukkan bahwa KNN memiliki akurasi tertinggi dengan 94,66% untuk *Household* dan 82,96% untuk *Fresh Food*, serta tercepat dalam waktu komputasi, menjadikannya metode yang efektif untuk klasifikasi produk di *e-commerce* [6].

Danny Sebastian (2019) melakukan penelitian menggunakan algoritma *K-Nearest Neighbor* (KNN) untuk klasifikasi produk handphone dari *e-marketplace* Tokopedia dan Bukalapak. Penelitian ini menggunakan metode TF-IDF untuk pembobotan dan *cosine similarity* untuk menghitung jarak kesamaan antar produk. Sebanyak 450 data produk digunakan, dengan 150 data untuk pelatihan dan 300 data untuk pengujian. Dalam pengujian pertama, nilai $K = 5$ memberikan akurasi terbaik sebesar 97,33%, sementara pada pengujian kedua dengan data produk campuran dari lima merek, akurasi mencapai 96,67%. Kesalahan klasifikasi yang terjadi sebagian besar disebabkan oleh ketidakkonsistenan penulisan judul produk, seperti penggunaan simbol yang tidak dikenali oleh sistem. [7].

Penelitian oleh Sayar Ul Hassan, Jameel Ahamed, dan Khaleel Ahmad (2022) membandingkan efisiensi algoritma pembelajaran mesin seperti SVM, k-NN, LR, MNB, dan RF untuk klasifikasi teks menggunakan set data IMDB dan SPAM. Pada set data SPAM yang terdiri dari 50.572 pesan SMS dengan klasifikasi biner (ham dan spam), yang memiliki karakteristik data sampel yang tidak seimbang, (KNN) mencapai akurasi 98,5% dengan *precision* dan *recall* masing-masing 99%. Hasil menunjukkan LR dan SVM unggul pada set data IMDB [8].

Dari ketiga penelitian tersebut, hasil menunjukkan bahwa KNN mampu memprediksi kategori produk dengan tingkat kesalahan yang rendah. Untuk memberikan konteks dan pemahaman yang lebih baik mengenai uraian teori diatas, Tabel 2.1 menampilkan ringkasan berikut.

Tabel 2.1 Penelitian Terdahulu

No	Judul Penelitian	Nama Peneliti, Tahun	Data	Metode	Hasil Penelitian	Future Works
1	Performance Analysis of Supervised Learning Models for Product Title Classification	Norsyela Muhammad Noor Mathivanan, Nor Azura Md. Ghani, Rozaiah Mohd Janor, 2019	Fresh Food & Household Products	Naïve Bayes, K-Nearest Neighbor (KNN), Decision Tree, Support Vector Machine (SVM), Random Forest	KNN memiliki akurasi tertinggi dan waktu komputasi tercepat untuk kedua set data, yaitu sebesar 94.66% pada household set data and 82.96% pada fresh food set data. Naïve Bayes memiliki kinerja terburuk.	Pencarian lebih lanjut mengenai nilai optimal hyperparameter K untuk KNN dan eksplorasi fitur tambahan seperti harga atau deskripsi produk yang lebih lengkap serta eksperimen pada jenis set data yang berbeda
2	Implementasi Algoritma K-Nearest Neighbor untuk Melakukan Klasifikasi Produk dari beberapa E-marketplace	Danny Sebastian, 2019	Judul produk dari Tokopedia dan Bukalapak, kategori handphone	TF-IDF, Cosine Similarity, K-Nearest Neighbor (KNN)	K=5 menghasilkan akurasi terbaik, yaitu 97,33% pada pengujian pertama dan 96,6% pada pengujian kedua. Kesalahan klasifikasi disebabkan oleh penulisan nama produk yang tidak konsisten, terutama penggunaan simbol yang tidak dikenali sistem.	Lakukan pengujian ke data produk selain kategori handphone atau pengujian ke data produk yang berasal dari e-marketplace selain Tokopedia dan Bukalapak
3	Analytics of machine learning-based algorithms for text classification	Sayar Ul Hassan, Jameel Ahamed, Khaleel Ahmad, 2022	IMDB dan SPAM set data (imbalanced)	Support Vector Machine (SVM), k-Nearest Neighbor (k-NN), Logistic Regression (LR), Multinomial Naïve Bayes (MNB), Random Forest (RF)	LR dan SVM unggul pada set data IMDB dengan akurasi masing-masing 85.8% dan 85.5%. k-NN unggul pada set data SPAM dengan akurasi 98.5%.	Penelitian lebih lanjut dapat mencakup lebih banyak algoritma dengan penyesuaian hyperparameter dan pendekatan ensemble untuk pengoptimalan parameter

2.2 E-Katalog

Pengadaan barang dan jasa oleh pemerintah, yang diatur dalam Peraturan Presiden Nomor 12 Tahun 2021, merupakan proses strategis untuk memenuhi kebutuhan operasional dan melayani masyarakat, serta berperan dalam manajemen keuangan negara yang akuntabel dan transparan. Untuk meningkatkan efisiensi dan transparansi dalam pengadaan, pemerintah menciptakan platform e-katalog, yang mencantumkan daftar dan harga barang/jasa dari berbagai penyedia, memungkinkan proses pengadaan yang lebih cepat dan terbuka. Selain itu, e-katalog juga mendukung keterlibatan UMKM dalam kegiatan pengadaan, sejalan dengan kebijakan Percepatan Peningkatan Penggunaan Produk Dalam Negeri (P3DN) yang diatur dalam Instruksi Presiden Nomor 2 Tahun 2022.

E-katalog adalah sistem informasi elektronik yang memuat daftar, spesifikasi teknis, dan harga barang dan jasa dari penyedia yang diakui oleh pemerintah, dikelola oleh LKPP [9]. Sistem ini meningkatkan transparansi dan efisiensi dalam pengadaan barang dan jasa dengan menyediakan informasi secara terbuka dan online. Prosedurnya mencakup pengumpulan informasi produk, pendaftaran penyedia, permintaan dan evaluasi penawaran, serta pengadaan secara elektronik. Regulasi yang ada menekankan keterbukaan dan keamanan data, dengan evaluasi rutin untuk menjaga integritas sistem sesuai perkembangan teknologi [4].

Penelitian dalam studi *"Penerapan Win-Win Solution Dalam Sengketa Pengadaan Barang/Jasa Pemerintah Berdasarkan Kontrak Secara Elektronik"* menunjukkan bahwa sistem pengadaan konvensional memerlukan interaksi langsung dan berulang antara Pejabat Pembuat Komitmen (PPK) dan penyedia barang/jasa di setiap tahap. Sistem ini menimbulkan berbagai kelemahan, termasuk keterbatasan penyebaran informasi dan kurangnya transparansi, yang dapat mengakibatkan inefisiensi dan korupsi [10]. Selain itu, disebutkan bahwa sistem pengadaan pemerintah melalui e-katalog terbukti efektif dalam mengurangi kemungkinan manipulasi harga yang tidak rasional dalam proses pengadaan [9].

Penelitian yang dilakukan oleh Dani Ariza pada tahun 2024 mengemukakan beberapa tantangan dan rekomendasi untuk pengembangan e-katalog, seperti potensi celah keamanan dan kurangnya kesadaran teknologi di kalangan penyedia barang dan jasa. Sejumlah rekomendasi telah diajukan untuk menghadapi tantangan tersebut. LKPP disarankan untuk memperketat pengawasan dan evaluasi terhadap proses pengadaan melalui e-katalog serta mengembangkan sistem keamanan yang lebih maju guna mencegah penyalahgunaan [4].

2.3 Praproses Teks

Praproses teks adalah langkah awal yang krusial dalam proses analisis data teks. Langkah ini difokuskan pada eliminasi unsur-unsur yang tidak diperlukan (*noise*) seperti tanda baca, tag HTML, URL, emoji, dll. [11]. Inti dari praproses teks adalah untuk menyederhanakan data teks menjadi format yang lebih sederhana dan siap analisis dan memastikan bahwa data yang akan dianalisis terbebas dari segala elemen yang bisa mengganggu analisis yang akan dilakukan. Tahapan praproses teks pada umumnya terdiri dari empat tahapan, yaitu *lowercasing*, *data cleansing*, tokenisasi, dan *custom word removal* [12].

Lower casing adalah proses mengubah seluruh huruf dalam teks menjadi huruf kecil untuk menyamakan penggunaan huruf yang tidak konsisten akibat kesalahan penulisan [13], dengan hasil proses ini disajikan pada Tabel 2.2 yang diambil dari data penelitian berjudul "Implementasi Algoritma K-Nearest Neighbor untuk Melakukan Klasifikasi Produk dari beberapa E-marketplace" [7].

Tabel 2.2 Proses *Lower Casing*

Nama Produk	*Lower Casing*
Apple iphone 6s+ 16gb rosegold	apple iphone 6s+ 16gb rosegold
iPhone 6s+Garansi Apple Internasional	iphone 6s+garansi apple internasional
[NEW] XIAOMI MI6 / MI6 PRO RAM 6GB INTERNAL 128GB	[new] xiaomi mi6 / mi6 pro ram 6gb internal 128gb

Text cleansing melibatkan pembersihan teks dari kata-kata yang tidak relevan atau tidak diinginkan, seperti simbol, angka, atau tanda baca [14], dengan hasil pembersihan ditunjukkan pada Tabel 2.3.

Tabel 2.3 Proses *Text Cleansing*

Lower Casing	*Text Cleansing*
apple iphone 6s+ 16gb rosegold	apple iphone rosegold
iphone 6s+ garansi apple internasional	iphone garansi apple internasional
[new] xiaomi mi6 / mi6 pro ram 6gb internal 128gb	new xiaomi pro ram internal

Tokenization adalah proses memecah teks menjadi unit-unit lebih kecil yang disebut

token, seperti kata, frasa, simbol, atau kalimat utuh [15], sebagaimana diperlihatkan pada Tabel 2.4.

Tabel 2.4 Proses *Tokenization*

Text Cleansing	*Tokenization*
apple iphone rosegold	[apple, iphone, rosegold]
iphone garansi apple internasional	[iphone, garansi, apple, internasional]
new xiaomi pro ram internal	[new, xiaomi, pro, ram, internal]

Custom word removal merupakan tahap menghapus kata-kata yang sering muncul tetapi tidak memberikan informasi signifikan atau dianggap tidak dibutuhkan [16], dengan contoh hasil penghapusan disajikan pada Tabel 2.5.

Tabel 2.5 Proses *Custom Word Removal*

Text Cleansing	*Custom Word Removal*
apple iphone rosegold	apple iphone
iphone garansi apple internasional	iphone apple
new xiaomi pro ram internal	xiaomi
hp vivo hp vivo second hp vivo murah hp second murah	hp vivo hp vivo hp vivo hp

Penelitian yang dilakukan oleh *Işık* dan *Dağ* (2020) menunjukkan bahwa tahapan praproses teks memiliki dampak signifikan terhadap performa model klasifikasi sentimen. Penghapusan emotikon dan tanda baca, memberikan sedikit peningkatan yang tidak signifikan dalam akurasi klasifikasi, dan bahkan dapat mengurangi performa dalam beberapa kasus. Tokenisasi merupakan tahapan penting yang mempengaruhi tahap-tahap selanjutnya dalam analisis teks. Penggunaan *lowercase* terbukti meningkatkan performa classifier dengan menyederhanakan teks dan mengurangi ambiguitas kapitalisasi, sementara penghapusan kata-kata yang tidak relevan (*stopword*), terutama dengan *custom stopword*, secara konsisten meningkatkan akurasi klasifikasi. Selain itu, urutan penerapan metode praproses seperti *data cleansing*, penggunaan tokenisasi, dan *stopword* juga mempengaruhi hasil, dengan beberapa urutan menunjukkan peningkatan akurasi hingga 2% [15].

2.4 Ekstraksi Fitur TF-IDF

Dalam klasifikasi teks, data teks perlu diubah menjadi representasi numerik agar dapat diproses oleh algoritma klasifikasi, yang biasanya membutuhkan data dalam format angka. Salah satu metode umum untuk ekstraksi fitur teks adalah *Term Frequency-Inverse Document Frequency* (TF-IDF). TF-IDF menimbang pentingnya kata dalam suatu dokumen relatif terhadap seluruh korpus, memberikan bobot lebih tinggi pada kata-kata spesifik yang unik dalam dokumen tertentu, sementara kata-kata umum yang sering muncul memiliki bobot lebih rendah [17] [18].

Secara sistematis, perhitungan TF-IDF dan normalisasi dilakukan melalui langkah-langkah berikut:

1. **Menghitung *Term Frequency* (TF)**:
   Term Frequency untuk istilah t_i dalam dokumen d_j menunjukkan seberapa sering kata tersebut muncul dalam dokumen tertentu relatif terhadap jumlah total kata dalam dokumen tersebut. Perhitungan TF ditunjukkan dalam Persamaan 2.1 .

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{2.1}$$

   Keterangan:

   $n_{i,j}$ **:** Jumlah kemunculan istilah t_i dalam dokumen d_j.

   $\sum_k n_{k,j}$ **:** Jumlah total kata dalam dokumen d_j.

2. **Menghitung *Inverse Document Frequency* (IDF)**:
   Inverse Document Frequency untuk suatu istilah t_i menghitung kepentingan kata tersebut di seluruh korpus, dengan memberikan bobot yang lebih tinggi pada kata-kata yang jarang muncul di banyak dokumen. Perhitungan IDF Perhitungan TF ditunjukkan dalam Persamaan 2.2.

$$IDF_i = \log\left(\frac{|D|}{|\{d : t_i \in d\}|}\right) \tag{2.2}$$

   Keterangan:

   $|D|$ **:** Jumlah total dokumen dalam korpus.

   $|\{d : t_i \in d\}|$ **:** Jumlah dokumen yang mengandung istilah t_i.

3. **Menghitung Bobot TF-IDF**:
   Setelah mendapatkan nilai TF dan IDF, bobot TF-IDF dihitung dengan

mengalikan kedua nilai ini. Nilai bobot akhir ini mewakili pentingnya setiap istilah dalam suatu dokumen relatif terhadap korpus. Perhitungan TF-IDF ditunjukkan dalam Persamaan 2.3.

$$TF\text{-}IDF_{i,j} = TF_{i,j} \times IDF_i \tag{2.3}$$

4. **Normalisasi Vektor TF-IDF**:
   Setelah menghitung bobot TF-IDF untuk semua istilah dalam dokumen, langkah berikutnya adalah melakukan normalisasi vektor TF-IDF. Normalisasi bertujuan untuk memastikan bahwa semua dokumen memiliki skala yang seragam dalam ruang vektor, sehingga menghindari bias yang disebabkan oleh panjang dokumen yang bervariasi [18]. Perhitungan normalisasi vektor ditunjukkan dalam Persamaan 2.4.

$$\text{TF-IDF}_{\text{norm},i,} = \frac{\text{TF-IDF}_{i,j}}{\|\mathbf{x}\|_p} \tag{2.4}$$

Keterangan:

$TF - IDF_{i,j}$ **:** Bobot TF-IDF untuk istilah t_i dalam dokumen d_j.

$\|\mathbf{x}\|_p$ **:** Norma vektor TF-IDF untuk dokumen d_j, di mana p menentukan jenis norma yang digunakan (L1 atau L2).

Terdapat dua jenis norma yang digunakan dalam normalisasi vektor TF-IDF, yaitu:

1. **L1 *Norm*** (*Manhattan Norm* atau *Taxicab Norm*):
   L1 *norm* adalah jumlah nilai absolut dari semua elemen dalam vektor. Dalam konteks TF-IDF, normalisasi L1 akan menghitung nilai TF-IDF yang dinormalisasi dengan membagi setiap elemen vektor dengan jumlah absolut seluruh bobot TF-IDF dalam dokumen [19]. Perhitungan L1 *norm* untuk vektor $\mathbf{x} = [x_1, x_2, \ldots, x_n]$ ditunjukkan pada persamaan 2.5.

$$\|\mathbf{x}\|_1 = |x_1| + |x_2| + \cdots + |x_n| \tag{2.5}$$

   Sehingga, nilai TF-IDF yang dinormalisasi dengan L1 untuk istilah t_i dalam dokumen d_j ditunjukkan pada persamaan 2.6.

$$\text{TF-IDF}_{\text{L1-norm},i,j} = \frac{\text{TF-IDF}_{i,j}}{\|\mathbf{x}\|_1} \tag{2.6}$$

2. **L2 *Norm*** (*Euclidean Norm*):

L2 *norm* adalah akar kuadrat dari jumlah kuadrat semua elemen dalam vektor. Normalisasi L2 digunakan untuk menghasilkan vektor dengan panjang satuan (unit length), yang berarti jarak antara vektor lebih representatif dalam ruang vektor [19]. Perhitungan L2 *norm* untuk vektor $\mathbf{x} = [x_1, x_2, \ldots, x_n]$ ditunjukkan pada persamaan 2.7.

$$\|\mathbf{x}\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} \tag{2.7}$$

Sehingga, nilai TF-IDF yang dinormalisasi dengan L2 untuk istilah t_i dalam dokumen d_j ditunjukkan pada persamaan 2.8.

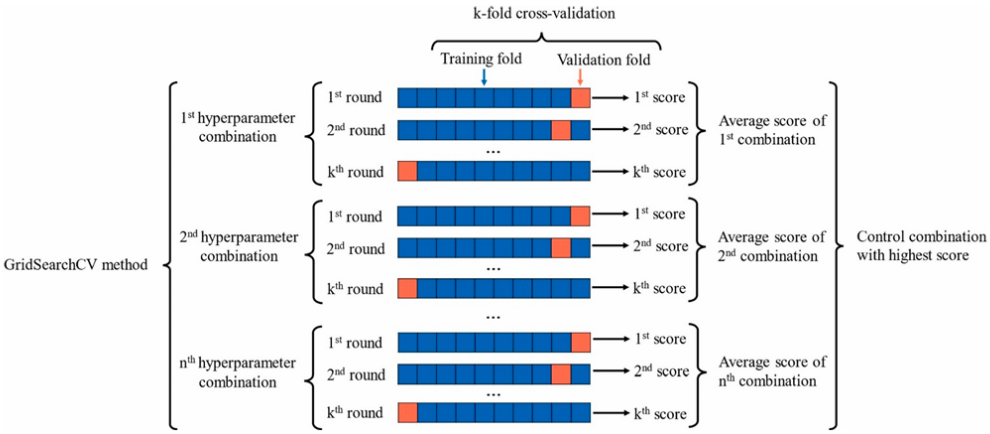
$$\text{TF-IDF}_{\text{L2-norm},i,j} = \frac{\text{TF-IDF}_{i,j}}{\|\mathbf{x}\|_2} \tag{2.8}$$

Setelah normalisasi, vektor TF-IDF yang dinormalisasi ini digunakan sebagai representasi fitur untuk setiap dokumen. Normalisasi ini membantu meningkatkan akurasi algoritma klasifikasi teks dengan mengurangi bias pada dokumen yang memiliki jumlah kata lebih banyak dan memungkinkan perbandingan yang lebih konsisten antar dokumen [18].

2.5 Grid Search CV

Grid search merupakan metode pencarian *hyperparameter* optimal model dengan mencoba berbagai kombinasi yang telah ditetapkan sebelumnya untuk menemukan konfigurasi terbaik yang meminimalkan kesalahan model [20]. Penggunaan *grid search* bersama *cross-validation* (CV) bertujuan untuk mengevaluasi performa kombinasi *hyperparameter* secara akurat dengan menguji setiap kombinasi pada beberapa subset (*fold*) data latih. Hal ini dilakukan untuk memastikan kestabilan performa model dan mencegah bias yang bergantung pada satu subset tertentu [21]. Salah satu variasi CV yang umum digunakan adalah *k-fold cross-validation*, yaitu pembagian data ke dalam beberapa subset yang secara bergantian berperan sebagai data uji dan data latih.

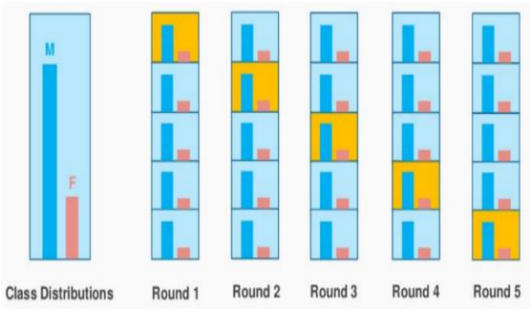
Gambar 2.1 menunjukkan cara kerja *grid search CV* dengan menggunakan *k-fold cross-validation* untuk menguji berbagai kombinasi *hyperparameter*. Setiap kombinasi diuji beberapa kali dengan membagi data menjadi beberapa subset, sehingga setiap subset bergantian menjadi data validasi sementara subset lainnya menjadi data latih. Hasil dari tiap subset kemudian dirata-rata untuk mendapatkan skor keseluruhan dari kombinasi tersebut. Setelah semua kombinasi diuji, konfigurasi dengan skor rata-rata tertinggi dipilih sebagai konfigurasi terbaik yang akan digunakan pada model.



Gambar 2.1 Ilustrasi *Hyperparameter Tuning* dengan *Grid Search* CV [22]

Stratified k-fold cross-validation (SKCV) adalah pengembangan dari metode *k-fold* yang bertujuan untuk memastikan distribusi kelas yang seimbang di setiap *fold*, menjadikannya lebih cocok untuk kasus klasifikasi dengan distribusi kelas yang tidak seimbang (*imbalanced*) [23]. Metode ini dirancang untuk mengatasi kelemahan pada *k-fold cross-validation* konvensional, di mana distribusi kelas pada setiap *fold* dapat berbeda secara signifikan dari data awal, terutama dalam set data dengan ketidakseimbangan kelas yang tinggi.

Pada SKCV, data dibagi menjadi n subset (*n-fold*), di mana satu subset digunakan sebagai set validasi dan $n - 1$ subset lainnya digunakan sebagai set latih. Proses stratifikasi memastikan bahwa proporsi sampel dari setiap kelas dipertahankan dalam setiap *fold*, sehingga setiap subset mencerminkan distribusi kelas yang serupa dengan keseluruhan dataset. Gambar 2.2 memberikan ilustrasi bagaimana setiap kelas, baik mayoritas maupun minoritas, didistribusikan secara proporsional ke dalam beberapa *fold*. Proses ini memastikan evaluasi model yang lebih representatif dan membantu meningkatkan keakuratan pada data yang tidak seimbang.



Gambar 2.2 Ilustrasi *Stratified K-Fold* [24]

Penentuan nilai k optimal dalam algoritma K-Nearest Neighbors (KNN) dapat dilakukan melalui *cross-validation*, pengetahuan domain, atau *trial and error*, tergantung pada masalah spesifik dan karakteristik data [25]. Penggunaan *cross-validation* juga terbukti meningkatkan performa model klasifikasi pada algoritma seperti *Random Forest*, *Naive Bayes*, *Support Vector Machines*, *Decision Tree*, dan KNN [26]. Penelitian merekomendasikan nilai *n-fold* dalam *cross-validation* disesuaikan dengan ukuran data, yaitu 5-6 untuk set data dengan 5.000 hingga 100.000 baris, 3-5 untuk 100.000 hingga 1.000.000 baris, dan 2-3 untuk data lebih besar dari 1.000.000 baris, agar mengurangi beban komputasi tanpa mengorbankan performa [27]. Namun, perubahan nilai *n-fold* tidak selalu meningkatkan performa dan kadang hanya menambah kompleksitas komputasi tanpa perbaikan signifikan [28].

2.6 K-Nearest Neighbors (KNN)

K-Nearest Neighbor (KNN) adalah algoritma yang memprediksi data baru berdasarkan kemiripan dengan titik data terdekat, yaitu dengan menghitung jarak antara titik data baru dengan sejumlah titik data terdekat dari set data latih [29].

Algoritma *K-Nearest Neighbor* (KNN) bekerja melalui beberapa tahapan utama untuk menentukan kelas atau nilai prediksi, sebagai berikut:

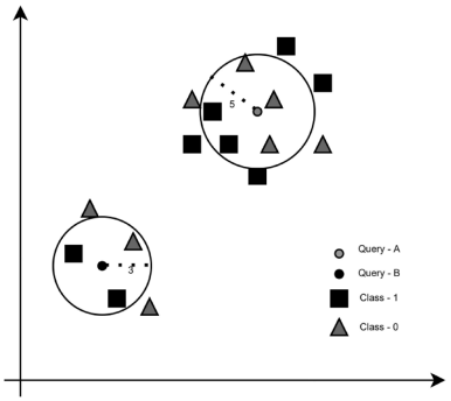
1. **Menghitung Jarak antara Data Uji dan Data Latih:**
   Pada tahap ini, KNN menghitung kedekatan antara data uji dengan semua data dalam set data latih menggunakan metrik jarak tertentu. Pemilihan metrik jarak mempengaruhi hasil klasifikasi pada KNN, salah satu metrik yang paling umum digunakan adalah jarak *Euclidean*, yang dihitung sebagai akar kuadrat dari jumlah kuadrat perbedaan antara komponen-komponen dalam vektor data uji dan data latih. perhitungan jarak *Euclidean* ditunjukkan dalam Persamaan 2.9.

$$d(a,b) = \sqrt{\sum_{k=1}^{d}(a_k - b_k)^2} \tag{2.9}$$

Keterangan:

$d(a,b)$ **:** Hasil jarak *Euclidean* antara data uji a dan data latih b.
a_k **:** Komponen ke-k dari vektor titik a (data uji).
b_k **:** Komponen ke-k dari vektor titik b (data latih).
d **:** Jumlah dimensi dari ruang vektor.
$\sum$ **:** Simbol untuk penjumlahan.

2. **Menentukan Tetangga Terdekat (K):**
   Setelah menghitung jarak antara data uji dan setiap data latih, algoritma KNN memilih sejumlah k tetangga terdekat, yaitu data latih dengan jarak terkecil. Gambar 2.3 menunjukkan ilustrasi visual dari proses penentuan k tetangga terdekat algoritma KNN:
   - Titik data uji(*query*) ditandai dengan simbol lingkaran putih, yang dikelilingi oleh data latih dari dua kelas berbeda, ditunjukkan dengan simbol persegi (kelas 1) dan segitiga (kelas 0).
   - Dua lingkaran di sekitar titik kueri menggambarkan konsep jarak dalam KNN, dengan masing-masing lingkaran mencakup sejumlah tetangga terdekat (data latih) yang berbeda. Lingkaran pertama mencakup tiga tetangga terdekat, sementara lingkaran kedua mencakup lima tetangga terdekat.
   - Dalam konteks KNN, nilai k dapat diambil sebagai 3 atau 5 sesuai lingkup tetangga yang ingin diperhitungkan. Jika $k = 3$, maka tiga tetangga terdekat dalam lingkaran pertama akan digunakan untuk menentukan kelas prediksi. Jika $k = 5$, maka lima tetangga dalam lingkaran kedua akan digunakan.
   - Proses ini menunjukkan bahwa pilihan nilai K akan memengaruhi hasil klasifikasi. Jika $k = 3$, maka kelas yang dominan di antara tiga tetangga tersebut akan menjadi prediksi untuk titik kueri. Sebaliknya, jika $k = 5$, maka kelas yang dominan di antara lima tetangga akan menjadi prediksi.



Gambar 2.3 Ilustrasi *K-Nearest Neighbors* [30]

3. **Mengklasifikasikan atau Memprediksi Nilai Data Uji:**
   Berdasarkan k tetangga terdekat yang telah dipilih, KNN akan menentukan kelas atau nilai prediksi untuk data uji. Terdapat dua pendekatan utama dalam menentukan kelas atau nilai prediksi:

- ***Uniform Weight*:**
  pada *uniform weight*, semua tetangga terdekat memiliki kontribusi yang sama terhadap prediksi, tanpa memperhatikan jaraknya dari data uji. Algoritma KNN akan menghitung jumlah tetangga dari masing-masing kelas dan memilih kelas yang paling dominan di antara k tetangga terdekat (*voting* mayoritas). Jika terjadi situasi seimbang dalam *voting*, hasil prediksi akan berdasarkan urutan data dalam set pelatihan. Dengan demikian, data latih yang muncul lebih awal dalam daftar akan menjadi penentu hasil prediksi [31].
- ***Distance Weight*:**
  pada *distance weight*, kontribusi setiap tetangga terhadap prediksi bergantung pada jaraknya dari data uji. Tetangga yang lebih dekat dengan data uji akan diberi bobot lebih besar, sehingga pengaruhnya lebih signifikan terhadap hasil akhir. Perhitungan bobot ditunjukkan pada persamaan 2.10.

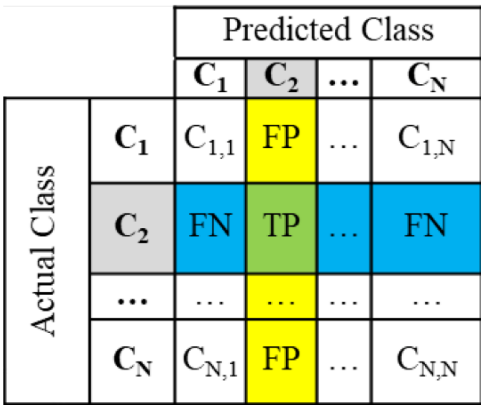
$$w_i = \frac{1}{d_i} \tag{2.10}$$

2.7 Evaluasi Model

Evaluasi model bertujuan untuk mengukur kemampuan model dalam memprediksi data uji [32]. Salah satu metode dalam evaluasi model klasifikasi adalah penggunaan *multi-class confusion matrix*.

Multi-class confusion matrix adalah alat evaluasi model yang menunjukkan jumlah prediksi benar dan salah untuk setiap kelas dalam masalah klasifikasi yang memiliki lebih dari dua kelas [33]. Matriks ini memberikan representasi yang mendetail mengenai kinerja model, di mana setiap sel dalam matriks menunjukkan jumlah prediksi untuk setiap kombinasi antara kelas aktual dan kelas prediksi. Untuk setiap kelas, konsep seperti *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) dapat diterapkan dalam pendekatan *one-vs-all*, dimana kelas yang sedang dievaluasi diperlakukan sebagai kelas positif dan semua kelas lainnya sebagai kelas negatif [34]. Ilustrasi mengenai *multi-class confusion matrix* disajikan pada Gambar 2.4.

Accuracy menunjukkan seberapa sering model klasifikasi membuat prediksi yang tepat, baik sebagai prediksi positif yang benar (TP) maupun negatif yang benar (TN), dibandingkan dengan semua prediksi yang dibuat [36]. Perhitungan *accuracy* dijelaskan dalam persamaan 2.11.



Gambar 2.4 Ilustrasi Multi Kelas *Confusion Matrix* [35]

$$Accuracy_i = \frac{\mathrm{TP}_i + \mathrm{TN}_i}{\mathrm{TP}_i + \mathrm{FP}_i + \mathrm{FN}_i + \mathrm{TN}_i} \tag{2.11}$$

Precision mengukur proporsi prediksi kelas positif yang benar-benar positif. *Precision* menilai ketepatan model dalam memprediksi kelas positif [36]. Perhitungan *precision* dijelaskan dalam Persamaan 2.12.

$$Precision_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FP}_i} \tag{2.12}$$

Recall mengukur proporsi kelas positif yang sebenarnya berhasil diidentifikasi dengan benar oleh model. *Recall* untuk memastikan bahwa semua kasus positif terdeteksi dengan baik. Perhitungan *recall* ditunjukkan dalam Persamaan 2.13.

$$Recall_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i} \tag{2.13}$$

F1-score diperoleh dari perhitungan yang menggabungkan *precision* dan *recall* menjadi satu nilai yang memberikan keseimbangan antara kedua metrik, terutama ketika kelas dalam set data tidak seimbang [36]. Perhitungan *f1-score* ditunjukkan dalam Persamaan 2.14.

$$F1\text{-}Score_i = 2 \times \frac{\mathrm{Precision}_i \times \mathrm{Recall}_i}{\mathrm{Precision}_i + \mathrm{Recall}_i} \tag{2.14}$$

Evaluasi model klasifikasi multi kelas membutuhkan pendekatan khusus dalam mengukur performa keseluruhan model. Metode agregasi evaluasi *macro average*, *micro average*, dan *weighted average* digunakan untuk mengukur kinerja model

dalam memprediksi keseluruhan kelas.

Macro Average menghitung rata-rata metrik untuk setiap kelas secara setara, tanpa memperhitungkan frekuensi kemunculan. Pendekatan ini cocok untuk set data dengan distribusi kelas tidak seimbang karena memberikan bobot yang sama pada kinerja semua kelas, termasuk kelas minoritas [37]. Perhitungan *macro average* ditunjukkan dalam persamaan 2.15.

$$Macro\ Average = \frac{1}{n} \sum_{i=1}^{n} \text{Score}_i \tag{2.15}$$

Keterangan:

n **:** Jumlah label.
$Score_i$ **:** Nilai metrik untuk label ke-i.

Micro Average menghitung metrik dengan menjumlahkan *true positives* (TP), *false positives* (FP), dan *false negatives* (FN) di seluruh kelas. Pendekatan ini merepresentasikan kinerja keseluruhan model, tetapi dapat dipengaruhi oleh kelas mayoritas pada data tidak seimbang [37]. Perhitungan *micro average* ditunjukkan pada persamaan 2.16, 2.17, 2.18.

$$Micro\ Precision = \frac{\sum_{i=1}^{n} \text{TP}_i}{\sum_{i=1}^{n} (\text{TP}_i + \text{FP}_i)} \tag{2.16}$$

$$Micro\ Recall = \frac{\sum_{i=1}^{n} \text{TP}_i}{\sum_{i=1}^{n} (\text{TP}_i + \text{FN}_i)} \tag{2.17}$$

$$Micro\ F1\text{-}Score = \frac{2 \times \text{Micro Precision} \times \text{Micro Recall}}{\text{Micro Precision} + \text{Micro Recall}} \tag{2.18}$$

Weighted Average menghitung rata-rata dengan mempertimbangkan frekuensi setiap kelas, memberikan bobot sesuai proporsi kemunculannya [37]. Perhitungan ditunjukan pada persamaan 2.19.

$$Weighted\ Average = \frac{\sum_{i=1}^{n} (\text{Score}_i \times \text{Weight}_i)}{\sum_{i=1}^{n} \text{Weight}_i} \tag{2.19}$$

Keterangan:

$Weight_i$ **:** Proporsi kemunculan kelas ke-i.

BAB III

METODE PENELITIAN

3.1 Jenis Penelitian

Penelitian ini menggunakan pendekatan kuantitatif, di mana tujuan utamanya adalah mengembangkan model untuk mengklasifikasikan etalase (kategori) produk berdasarkan nama produk pada e-katalog LKPP di Kota Bandar Lampung menggunakan metode K-Nearest Neighbors (KNN). Pendekatan kuantitatif dipilih mengingat penelitian ini melibatkan pengolahan data numerik dan analisis statistik.

3.2 Pengumpulan Data

Data dikumpulkan melalui teknik *web scraping* dari situs resmi e-katalog LKPP. Proses pengumpulan dilakukan selama periode penelitian dalam rentang waktu 15 Juli 2024 hingga 15 Agustus 2024. Pengumpulan data dilakukan menggunakan laptop dengan spesifikasi Intel(R) Core(TM) i3-1005G1 CPU @ 1.20GHz dan RAM 8GB. Data yang diperoleh kemudian disimpan dalam format excel untuk pengolahan lebih lanjut. Tahap pengumpulan data pada penelitian ini tidak melanggar ketentuan Undang-Undang No. 27 Tahun 2022 tentang Perlindungan Data Pribadi karena hanya mengakses informasi yang bersifat publik.

3.3 Deskripsi Data

Data yang digunakan dalam penelitian ini adalah data sekunder berbasis teks yang diperoleh dari e-katalog LKPP, berisi informasi terkait "Nama Produk" dan "Etalase Produk", dengan total 27 etalase produk yang tersedia di e-katalog nasional, terfokus pada wilayah Kota Bandar Lampung. Namun, dari 27 etalase tersebut, hanya 18 etalase yang memiliki data produk, sementara 9 etalase sisanya tidak memiliki produk sama sekali.

3.4 Teknis Analisis Data

Tahap analisis data dimulai dengan praproses data, yang meliputi beberapa langkah untuk membersihkan dan memformat data teks berupa "Nama Produk" yang diperoleh dari e-katalog LKPP Kota Bandar Lampung melalui *scraping*. Tahapan praproses ini diikuti oleh ekstraksi fitur untuk mengubah teks "Nama Produk" tersebut menjadi representasi numerik. Selanjutnya, pemodelan klasifikasi menggunakan algoritma *K-Nearest Neighbors* (KNN).

3.4.1 Praproses Data

Tahap praproses melibatkan beberapa langkah. Pertama, penulisan huruf kecil (*lower casing*). Kedua, pembersihan teks (*text cleansing*). Ketiga, *tokenization*, dan terakhir, penghapusan kata khusus (*custom word removal*). Hasil dari tahap praproses ini adalah data teks yang telah dibersihkan dari gangguan tekstual *noise*, sehingga teks yang tersisa lebih fokus pada informasi yang relevan dan penting.

3.4.1.1 *Lower Casing*

Lower casing merupakan proses yang dilakukan untuk menyamakan format teks dengan mengubah semua huruf menjadi huruf non kapital, yang bertujuan untuk mengurangi keberagaman dari teks yang akan menjadi fitur menjadi lebih sedikit. Tahap ini menggunakan fungsi *lower* di Python.

3.4.1.2 *Text Cleansing*

Proses selanjutnya adalah *text cleansing*, yaitu penyaringan karakter yang tidak relevan dengan informasi, seperti tanda baca, simbol, emoji, angka, dan spasi berlebih, baik di awal, tengah, maupun akhir kalimat. Tahap ini menggunakan *library* TextPrettifier.

3.4.1.3 *Tokenization*

tokenization adalah proses memecah teks menjadi unit-unit yang lebih kecil,yang disebut token. Pada penelitian ini, proses *tokenization* akan memecah kalimat menjadi unit yang lebih kecil, yaitu kata per kata, yang bertujuan untuk memudahkan penghapusan kata yang tidak relevan pada tahap *custom word removal*. Dengan memecah teks produk menjadi token terlebih dahulu, penghapusan kata menjadi lebih efektif, karena setiap kata dapat diidentifikasi dan dihapus secara langsung tanpa perlu memproses seluruh teks sekaligus. Tahap ini peneliti menggunakan *library nltk.tokenize*

3.4.1.4 *Custom Word Removal*

Custom word removal merupakan tahap yang bertujuan untuk mengeliminasi kata-kata yang tidak memiliki nilai diskriminatif antar etalase produk. Dalam penelitian ini, digunakan daftar stopword yang disediakan oleh library NLTK dan juga pendefinisan manual kata-kata tambahan.

3.4.2 Label Encoding

Variabel "Etalase Produk" sebagai variabel kategorikal *Y* (target), masih memiliki bentuk dasar sebuah teks, sehingga dilakukan tahap *label encoding* untuk mengubah etalase produk (variabel target) dari bentuk teks menjadi representasi numerik. Pada penelitian ini menggunakan *library Label Encoder*.

3.4.3 Ekstraksi Fitur

Masukan dari tahapan ini merupakan teks "Nama Produk" yang bersih dari *noise*, selanjutnya teks akan diubah menjadi representasi numerik. Pada penelitian ini akan menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF), yang didapatkan dengan menggunakan instrumen Python dan library *scikit-learn* versi 1.3.0. Vektor hasil TF-IDF ini yang akan menjadi representasi fitur dari teks yang akan menjadi masukan dari model klasifikasi. Hasil dari tahapan ini berupa vektor representasi dari setiap produk.

3.4.4 Pemodelan

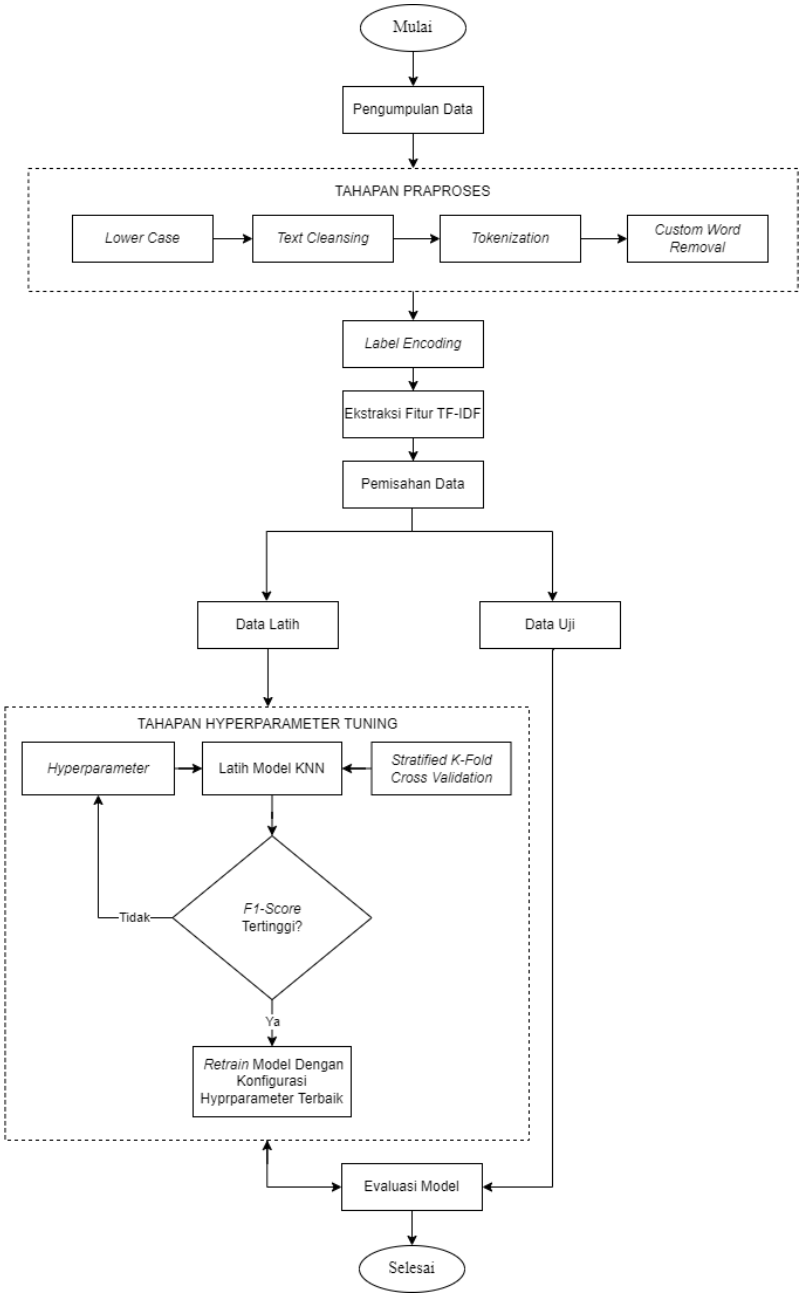
Algoritma *K-Nearest Neighbors* (KNN) digunakan untuk membangun model klasifikasi etalase produk. Penentuan *hyperparameter* optimal (*hyperparameter tuning*), yaitu jumlah tetangga terdekat (k) dan metode pembobotan (*uniform* atau *distance weighting*), dilakukan menggunakan *Grid Search* yang mengevaluasi berbagai kombinasi *hyperparameter* melalui teknik *resampling Stratified K-Fold Cross Validation* (SKCV). Dalam SKCV, data latih dibagi menjadi beberapa *fold*, di mana setiap *fold* secara bergantian digunakan sebagai data validasi, sedangkan *fold* lain digunakan untuk melatih model. Berdasarkan rata-rata hasil performa dari semua iterasi SKCV, kombinasi *hyperparameter* terbaik dipilih. Setelah kombinasi optimal ditemukan, model dilatih ulang menggunakan seluruh data latih dengan konfigurasi tersebut.

3.5 Evaluasi Model

Pada penelitian ini, *confusion matrix* dan *classification report* digunakan untuk mengevaluasi performa model *K-Nearest Neighbors* (KNN) pada set pengujian. Prediksi etalase produk dibandingkan dengan label sebenarnya, dan hasilnya disajikan dalam *confusion matrix*, yang menampilkan prediksi benar dan salah untuk setiap kelas. Terakhir dihitung dengan metrik evaluasi *accuracy*, *precision*, *recall*, dan *F1-score*, sesuai dengan persamaan 2.11, 2.12, 2.13, 2.14.

3.6 Desain Penelitian

Dalam pelaksanaan penelitian, gambar 3.1 menyajikan diagram alir pada setiap tahap yang dilakukan pada penelitian ini.



Gambar 3.1 Diagram Alir Penelitian

BAB IV

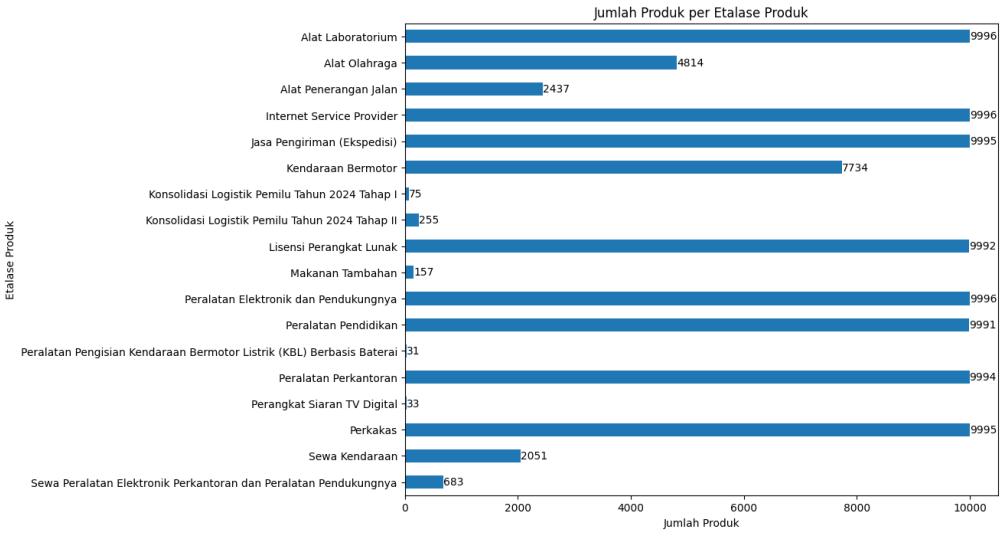
HASIL DAN PEMBAHASAN

4.1 Pemodelan Klasifikasi KNN

4.1.1 Pengumpulan Data

Data penelitian ini dikumpulkan dari situs e-katalog LKPP dengan fokus pada produk e-katalog nasional yang tersedia di Kota Bandar Lampung. Pengumpulan data dilakukan dalam rentang waktu 15 Juli 2024 hingga 15 Agustus 2024 menggunakan metode *web scraping*, menghasilkan total 98.225 produk yang tersebar dalam 18 etalase.

Distribusi data menunjukkan adanya ketidakseimbangan etalase yang signifikan (*class imbalance*), yaitu beberapa etalase memiliki jumlah produk yang jauh lebih banyak dibandingkan dengan etalase lainnya. Sebagai contoh, etalase dengan jumlah produk terbanyak memiliki 9.996 data, sedangkan etalase dengan jumlah produk paling sedikit hanya terdiri dari 19 data. Distribusi jumlah produk per etalase disajikan dalam Gambar 4.1.



Gambar 4.1 Distribusi Data Produk per Etalase (Kategori)

4.1.2 Tahap Praproses Data

4.1.2.1 *Lower Casing*

Gambar 4.1 menampilkan hasil *lowercasing* pada tiga sampel nama produk dari total 98.227, masing-masing berasal dari tiga kategori berbeda dari total 18

kategori produk. Terlihat bahwa semua huruf telah diubah menjadi huruf non kapital untuk memastikan konsistensi dalam pengolahan data teks, sehingga mendukung perbedaan kapitalisasi tidak memengaruhi hasil klasifikasi.

Tabel 4.1 Hasil Tahap *Lower Casing*

Nama Produk	**Hasil *Lower Casing***
GREENLEAF TEMPAT SAMPAH FANTASIA 5 LITER - 02043	greenleaf tempat sampah fantasia 5 liter - 02043
Fiber Optik Domestik 1 Gbps Dedicated	fiber optik domestik 1 gbps dedicated
AYURI Kursi Siswa KE LEOMORD 755L	ayuri kursi siswa ke leomord 755l

4.1.2.2 *Text Cleansing*

Tabel 4.2 menampilkan hasil *text cleansing* pada tiga sampel nama produk dari total 98.227, masing-masing berasal dari tiga kategori berbeda dari total 18 kategori produk. Pada tahap *text cleansing*, tanda baca, karakter khusus, dan angka telah dihapus dari nama produk, sehingga hanya tersisa huruf alfabet. Dengan demikian, tahap ini membersihkan karakter-karakter yang tidak relevan dalam klasifikasi etalase produk sehingga dapat meningkatkan kinerja model pada tahap klasifikasi [15]. Hasil dari tahap ini menunjukkan bahwa nama produk menjadi lebih sederhana dan hanya berisi informasi penting.

Tabel 4.2 Hasil Tahap *Text Cleansing*

Hasil *Lower Casing*	**Hasil *Text Cleansing***
greenleaf tempat sampah fantasia 5 liter - 02043	greenleaf tempat sampah fantasia liter
fiber optik domestik 1 gbps dedicated	fiber optik domestik gbps dedicated
ayuri kursi siswa ke leomord 755l	ayuri kursi siswa ke leomord

4.1.2.3 *Tokenization*

Tahap *tokenization* dilakukan pada nama produk setelah *text cleansing*, di mana teks dipisahkan berdasarkan spasi dan tanda baca. Tabel 4.3 menampilkan hasil tokenisasi pada tiga sampel nama produk dari total 98.227, masing-masing berasal

dari tiga kategori berbeda dari total 18 kategori produk. Pada tabel tersebut, terlihat bahwa setiap nama produk telah dipecah menjadi kata-kata individu.

Tabel 4.3 Hasil Tahap *Tokenization*

Hasil *Text Cleansing*	Hasil *Tokenization*
greenleaf tempat sampah fantasia liter	[greenleaf, tempat, sampah, fantasia, liter]
fiber optik domestik gbps dedicated	[fiber, optik, dosmetik, gbps, dedicated]
ayuri kursi siswa ke leomord	[ayuri, kursi, siswa, ke, leomord]

4.1.2.4 *Custom Word Removal*

Pada tahap *custom word removal*, penelitian ini menggunakan daftar *stopwords* dari pustaka NLTK yang dirancang khusus untuk bahasa Indonesia. Daftar ini kemudian dimodifikasi dengan menambahkan kata-kata tambahan untuk daftar *stopwords*, yaitu kata-kata yang berhubungan dengan satuan ukuran. Penambahan kata-kata tersebut bertujuan untuk menghilangkan istilah yang tidak relevan bagi klasifikasi produk, karena kata-kata tersebut lebih berkaitan dengan satuan ukuran dan tidak memberikan kontribusi signifikan dalam membedakan kategori produk.

Tabel 4.4 menunjukkan hasil *custom word removal* pada tiga sampel nama produk dari total 98.225 data, masing-masing berasal dari tiga kategori berbeda dari total 18 kategori. Terlihat bahwa kata-kata seperti 'tempat', 'gbps', dan 'ke' telah dihapus, menghasilkan nama produk yang lebih ringkas dan fokus pada kata-kata kunci yang relevan untuk klasifikasi etalase produk.

Tabel 4.4 Hasil Tahap *Custom Word Removal*

Hasil *Tokenization*	Hasil *Custom Word Removal*
[greenleaf, tempat, sampah, fantasia, liter]	[greenleaf tempat sampah fantasia
[fiber, optik, dosmetik, gbps, dedicated]	fiber optik dosmetik dedicated
[ayuri, kursi, siswa, ke, leomord]	ayuri kursi siswa leomord

Tahap *custom word removal* merupakan tahap terakhir dalam tahapan praproses teks pada penelitian ini. Pada tahap ini, dilakukan penghapusan baris data duplikat. Dari total 98.225 baris data, ditemukan 48.518 baris duplikat. tahap *custom word removal* menyisakan 49.707 baris data nama produk yang unik.

4.1.3 Ekstraksi Fitur TF-IDF

Pada tahap ini, nama-nama produk telah melalui praproses teks, menyisakan kata-kata relevan untuk etalase produk. Penggunaan *hyperparameter n-gram range* senilai (1,2) memungkinkan model menangkap konteks dari *unigram* dan *bigram* [38], yaitu pola dari kata individual dan pasangan kata berurutan. Dengan demikian, penggunaan *n-gram* ini. Representasi TF-IDF dinormalisasi menggunakan L2 untuk memastikan setiap vektor fitur memiliki panjang yang sama, mengurangi bias panjang vektor [39] pada perhitungan jarak antar vektor di KNN menggunakan jarak *euclidean* [40].

Tabel 4.2 menunjukkan contoh hasil vektorisasi *TF-IDF* pada nama produk "meja pingpong donic". Terlihat bahwa Bigram seperti "meja pingpong" dan "pingpong donic" memiliki nilai TF-IDF lebih tinggi dibandingkan unigram "meja" atau "pingpong", menunjukkan bahwa kombinasi dua kata tersebut membawa informasi lebih spesifik dan relevan dan bahwa kata "meja" dan "pingpong" pada penelitian ini merupakan kata yang cukup umum dan sering muncul dalam dokumen lainnya, sehingga dianggap kurang informatif untuk membedakan produk satu dengan lainnya. Selain itu, kata-kata umum seperti "meja" diberi bobot lebih rendah (IDF 3.84) dibandingkan kata-kata spesifik seperti "donic" (IDF 10.42), ini menegaskan bahwa kata "donic" lebih spesifik dan lebih jarang muncul dalam keseluruhan dokumen, menjadikannya lebih diskriminatif dalam konteks klasifikasi produk. Hal ini mencerminkan kemampuan TF-IDF dalam membedakan kata-kata umum dan diskriminatif antar kategori. Hasil ini menunjukkan bahwa konfigurasi *hyperparameter* yang digunakan pada penelitian ini berhasil mengidentifikasi istilah-istilah yang lebih spesifik dan informatif. Dengan demikian, ekstraksi fitur TF-IDF menghasilkan vektor dengan bobot numerik yang merepresentasikan seberapa unik suatu kata dalam keseluruhan dokumen (*corpus*).

Indeks dokumen: 8644
Nama produk: Meja Pingpong Donic

word	df	tf	idf	tf-idf (no norm)	tf-idf (L2 norm)
donic	3	1	10.4276	10.4276	0.525368
meja	2891	1	3.84422	3.84422	0.193681
meja pingpong	15	1	9.04133	9.04133	0.455524
pingpong	24	1	8.59505	8.59505	0.433039
pingpong donic	2	1	10.7153	10.7153	0.539863

Gambar 4.2 Sampel Hasil TF-IDF

4.1.4 *Label Encoding*

Variabel etalase produk sebagai variabel *Y* (target), masih memiliki bentuk dasar sebuah teks, sehingga dilakukan tahap *label encoding* untuk mengubah etalase produk (variabel target) dari bentuk teks menjadi representasi numerik.

Tabel 4.5 menunjukkan hasil tahap label encoding pada penelitian ini. Dimana setiap etalase produk diberi nilai integer yang mewakili satu etalase produk tertentu, tanpa menunjukkan adanya hubungan hierarkis atau prioritas antar etalase.

Tabel 4.5 Hasil *Label Encoding*

Etalase Produk	Hasil Label Encoding
Alat Laboratorium	0
Alat Olahraga	1
Alat Penerangan Jalan	2
Internet Service Provider	3
Jasa Pengiriman (Ekspedisi)	4
Kendaraan Bermotor	5
Konsolidasi Logistik Pemilu Tahun 2024 Tahap I	6
Konsolidasi Logistik Pemilu Tahun 2024 Tahap II	7
Lisensi Perangkat Lunak	8
Makanan Tambahan	9
Peralatan Elektronik dan Pendukungnya	10
Peralatan Pendidikan	11
Peralatan Pengisian Kendaraan Bermotor Listrik (KBL) Berbasis Baterai	12
Peralatan Perkantoran	13
Perangkat Siaran TV Digital	14
Perkakas	15
Sewa Kendaraan	16
Sewa Peralatan Elektronik Perkantoran dan Peralatan Pendukungnya	17

4.1.5 Pemisahan Data

Tahap terakhir sebelum pemodelan adalah memisahkan data menjadi latih dan uji dengan proporsi 80:20, yang memberikan keseimbangan antara kompleksitas model dan generalisasi performa [41] [42]. Karena data produk pada penelitian ini memiliki distribusi yang tidak seimbang (*imbalanced*), digunakan *stratified split* untuk menjaga proporsi kelas sesuai distribusi asli [43]. Selain itu, *shuffle* diterapkan untuk memastikan setiap etalase tersebar merata dalam set latih dan uji. Seperti terlihat pada Tabel 4.6, proporsi etalase 4 (Jasa Pengiriman) sebesar 0,19 tetap terjaga baik di data keseluruhan, data latih, dan data uji, memastikan data yang representatif untuk pelatihan model.

Tabel 4.6 Distribusi Data Latih dan Data Uji

Etalase Produk	Jumlah Data Keseluruhan	Jumlah Data Latih	Jumlah Data Uji	Proporsi Data Keseluruhan, Latih & Uji
0	6387	5110	1277	0.1285
1	2995	2396	599	0.0603
2	844	675	169	0.0170
3	728	583	145	0.0146
4	9598	7678	1920	0.1931
5	5361	4289	1072	0.1079
6	75	60	15	0.0015
7	207	166	41	0.0042
8	5548	4438	1110	0.1116
9	121	97	24	0.0024
10	2920	2336	584	0.0587
11	4943	3954	989	0.0994
12	19	15	4	0.0004
13	3463	2770	693	0.0697
14	31	25	6	0.0006
15	4908	3926	982	0.0987
16	1285	1028	257	0.0259
17	274	219	55	0.0055
Total	**49707**	**39765**	**9942**	**1**

4.1.6 *Hyperparameter Tuning*

Penelitian ini menggunakan *Grid Search* dengan *Stratified K-Fold Cross Validation* (SKCV) untuk menemukan kombinasi *hyperparameter* optimal (*hyperparameter tuning* [25]. Tabel 4.7 menampilkan *Hyperparameter* yang diuji berupa:

1. ***k* Tetangga Terdekat (*n-neighbors*)**:
   Jumlah *k* tetangga terdekat (*n-neighbors*) yang digunakan dalam algoritma KNN, diuji dalam rentang nilai 1 hingga 10. Penentuan rentang nilai *k* pada penelitian ini dikarenakan nilai *k* yang lebih rendah, dengan rentang 1 hingga 5 sering memberikan hasil optimal pada dataset multi kelas, terutama untuk dataset besar [44].
2. **Pembobotan (*weights*)**:
   Dua skema pembobotan diuji, yaitu pembobotan seragam (*uniform weighting*) dan pembobotan berbasis jarak (*distance weighting*). Alasan pengujian dua skema bobot pada penelitian ini didasarkan penelitian oleh Xingyang yang menyatakan bahwa pembobotan seragam cenderung kurang efektif pada set data dengan distribusi kelas yang tidak merata. Sebaliknya, pembobotan berbasis jarak lebih responsif terhadap data yang tidak seimbang [45]. Namun hasil penelitian oleh Teng dan Lee menunjukkan bahwa pembobotan seragam dan pembobotan berbasis jarak menghasilkan evaluasi model yang relatif serupa pada penelitian tersebut, meskipun dengan distribusi data yang tidak seimbang [46].

Konfigurasi *hyperparameter* KNN yang menghasilkan evaluasi model dengan metrik *f1-score* tertinggi akan dipilih sebagai konfigurasi *hyperparameter* KNN optimal untuk penelitian ini. Metrik *f1-score* dipilih karena mempertimbangkan *precision* dan *recall* secara bersamaan, dikarenakan kondisi set data yang tidak seimbang pada penelitian ini, metrik *accuracy* tidak digunakan karena hanya menghitung proporsi prediksi benar secara keseluruhan tanpa memperhatikan kemampuan model dalam memprediksi setiap kelas secara individual [46] [33].

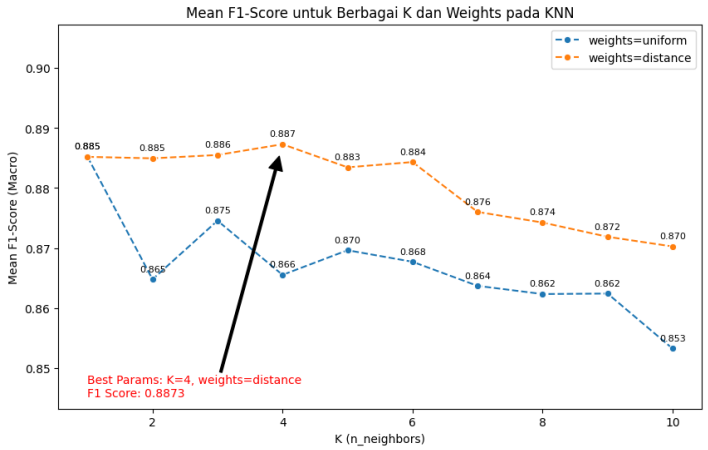
Tabel 4.7 *Hyperparamter Grid* pada KNN

Hyperparameter	**Rentang Nilai**
k tetangga terdekat (*n-neighbors*)	1,2,3,4,5,6,7,8,9,10
pembobotan (*weights*)	*distance*, *uniform*

Selama proses *Grid Search*, data latih di-*resampling* menggunakan *Stratified K-Fold Cross Validation* (SKCV) untuk mengevaluasi kinerja setiap kombinasi

hyperparameter dan mengukur kemampuan prediksi model pada berbagai set data. Pemilihan 5-*fold* dalam SKCV didasarkan pada ukuran dataset yang berkisar antara 5.000 hingga 100.000 baris, sesuai rekomendasi dari hasil penelitian sebelumnya [27]. Selain itu, penggunaan 5-*fold* memastikan keseimbangan antara efisiensi komputasi dan kualitas evaluasi, karena peningkatan jumlah *fold* tidak selalu meningkatkan performa model, namun dapat menambah kompleksitas komputasi [28].

Gambar 4.3 menunjukkan hasil *hyperparameter tuning* pada penelitian ini, dengan membandingkan kinerja *f1-score* menggunakan pembobotan berbasis jarak (*distance weighting*) dan pembobotan seragam (*uniform weighting*) pada rentang nilai k tetangga terdekat dari 1 hingga 10. Secara keseluruhan, pembobotan berbasis jarak secara konsisten menghasilkan *f1-score* yang lebih tinggi dibandingkan pembobotan seragam pada seluruh rentang nilai k, menunjukkan bahwa pembobotan seragam memberikan pengaruh signifikan terhadap peningkatan performa klasifikasi pada penelitian ini. Skor tertinggi dicapai pada $k = 4$ dan pembobotan berbasis jarak dengan *f1-score* sebesar 0,887, yang ditetapkan sebagai konfigurasi optimal KNN. Setelah mencapai puncaknya, *f1-score* untuk kedua pembobotan mengalami penurunan seiring dengan bertambahnya nilai k. Penurunan ini mengindikasikan bahwa pemilihan jumlah tetangga yang terlalu besar cenderung memperkenalkan informasi yang kurang relevan dari sampel-sampel yang lebih jauh, yang pada akhirnya menurunkan performa model. Temuan ini konsisten dengan literatur yang menunjukkan bahwa nilai k rentang 1 hingga 5 lebih optimal dalam meningkatkan performa model pada set data besar [44], serta penggunaan pembobotan berbasis jarak (*distance weighting*) untuk menangani distribusi data yang tidak seimbang [45].



Gambar 4.3 Hasil *Hyperparameter Tuning*

Tabel 4.8 menunjukkan nilai *f1-score* untuk setiap *fold* pada konfigurasi hyperparameter optimal KNN pada penelitian ini, yaitu $k = 4$ dan pembobotan berbasis jarak (*distance weighting*). *F1-score* untuk *fold* 1 higga *fold* 5 memiliki rentang skor antara 87,26 hingga 89,98.

Tabel 4.8 *F1-Score* Di Setiap *Fold* untuk *Hyperparameter* Optimal

Fold	F1-Score
1	87,21
2	91,33
3	90,17
4	86,29
5	88,62

4.1.7 Pelatihan Akhir Model KNN

Tahap pelatihan akhir model *K-Nearest Neighbors* (KNN) merupakan tahap yang dilakukan setelah dilakukannya pencarian nilai kombinasi *hyperparameter* terbaik. Sebelumnya menggunakan 5 *fold* yang artinya pada tahap sebelumnya hanya menggunakan 80% dari keseluruhan data latih, sehingga pada tahap pelatihan akhir model KNN menggunakan keseluruhan data latih berjumlah 39760 data produk. Pemodelan KNN dilatih menggunakan nilai jumlah tetangga terdekat(k) = 4 dengan pembobotan berbasis jarak (*weighting distance)*, yang dipilih berdasarkan hasil tahap *grid search* SKCV sebelumnya. Metrik yang digunakan untuk mengukur jarak antar data adalah *euclidean distance*, yang dipilih berdasarkan literatur yang menunjukkan toleransinya yang baik terhadap *noise* dalam data [29]. Toleransi ini sangat penting dalam konteks klasifikasi etalase produk, di mana variasi kecil dalam nama produk bisa mengakibatkan perbedaan signifikan dalam klasifikasi.

4.2 Evaluasi Model

4.2.1 Evaluasi Metrik Klasifikasi

Sebanyak 9.941 produk dalam data uji diprediksi menggunakan model *K-Nearest Neighbor* (KNN) yang telah dilatih, kemudian hasil prediksi dibandingkan dengan label sebenarnya untuk evaluasi. Penelitian ini menggunakan metrik *macro average* sebagai acuan utama dalam evaluasi model karena metrik ini memberikan penilaian yang lebih adil terhadap kinerja model pada setiap kelas, terutama dalam konteks data yang tidak seimbang [33] [46].

Evaluasi model dilakukan menggunakan metrik klasifikasi, yaitu *accuracy*, *precision*, *recall*, dan *f1-score*. Tabel 4.9 menyajikan ringkasan hasil evaluasi dan menunjukkan bahwa model mencapai akurasi sebesar 0,9062 atau 90,62% dalam mengklasifikasikan sampel dengan benar.

Hasil evaluasi lebih lanjut mengindikasikan bahwa *macro precision* sebesar 0,9103 menunjukkan 91,03% dari prediksi model sesuai dengan etalase sebenarnya saat memprediksi produk ke dalam etalase tertentu. Sementara itu, *macro recall* sebesar 0,9119 mengindikasikan bahwa model mampu mengenali 91,19% dari seluruh produk yang benar-benar termasuk dalam etalase yang tepat, menyoroti kemampuan model dalam mendeteksi produk pada etalase yang memiliki jumlah sampel lebih sedikit.

Terakhir, *macro f1-score* sebesar 90,96% mencerminkan keseimbangan antara ketepatan prediksi (*precision*) dan kemampuan deteksi model untuk semua produk pada etalase masing-masing (*recall*). Dengan demikian, model KNN yang digunakan dalam penelitian ini menunjukkan kinerja yang tinggi dan konsisten dalam mengklasifikasikan produk dengan *accuracy* di atas 90% dan mempertahankan keseimbangan antara *precision* dan *recall*.

Tabel 4.9 Ringkasan Hasil Evaluasi Klasifikasi pada Data Uji: *Average*

Accuracy	0,9062		
	Precision	**Recall**	**F1-Score**
Macro Average	0,9103	0,9119	0,9096

4.2.2 Evaluasi Distribusi Kesalahan Klasifikasi

Lampiran B.1 menunjukkan laporan evaluasi klasifikasi model KNN pada setiap etalase produk dalam bentuk metrik *precision*, *recall*, dan *f1-score*. Evaluasi klasifikasi per etalase mencerminkan kemampuan model dalam mengenali produk di berbagai etalase, yang membantu mengidentifikasi etalase dengan kesalahan klasifikasi tinggi maupun rendah.

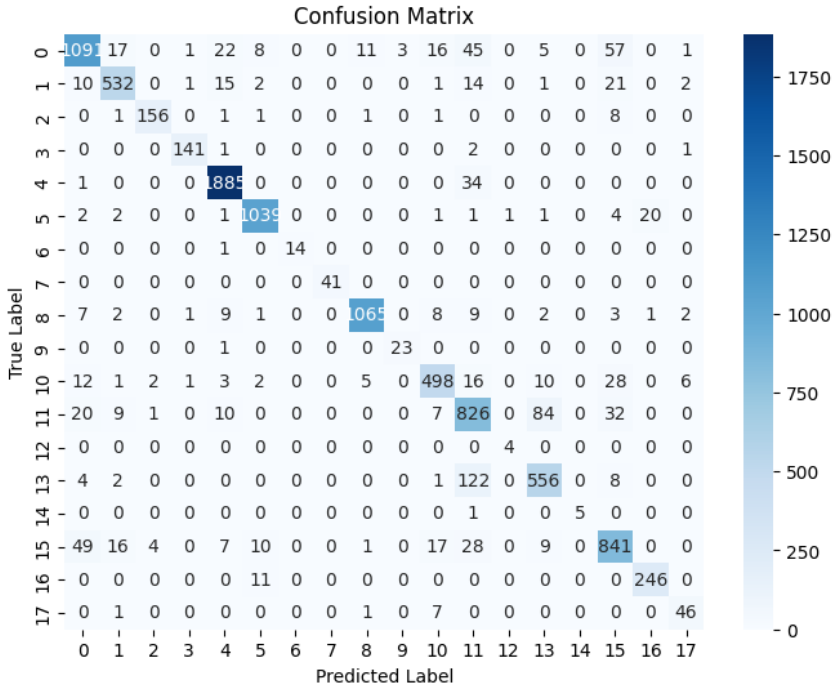
Model KNN menunjukkan performa yang sangat baik pada beberapa etalase berdasarkan metrik evaluasi *f1-score*. Pada etalase 7 (Konsolidasi Logistik Pemilu Tahap II), model mencapai *f1-score* tertinggi sebesar 1,0 yang menandakan tidak adanya kesalahan dalam prediksi produk pada etalase ini. Selain itu, etalase 4 (Jasa Pengiriman) dan etalase 5 (Kendaraan Bermotor) juga mencapai *f1-score* sebesar

0,97. Tingginya nilai *f1-score* pada kedua etalase tersebut didukung oleh *precision* dan *recall* yang juga sangat tinggi, keduanya di atas 0,97. Dengan demikian, hasil ini menunjukkan bahwa model KNN memiliki kemampuan untuk mengklasifikasikan produk dengan akurasi yang tinggi, terutama pada etalase yang terdefinisi secara jelas dan spesifik.

Model KNN juga mampu menunjukkan ketahanan terhadap ketidakseimbangan data. Pada etalase dengan *support* (jumlah data uji) yang kecil, seperti etalase 6 (Konsolidasi Logistik Pemilu Tahap I) dengan hanya 15 sampel dan etalase 14 (Perangkat Siaran TV Digital) dengan hanya 6 sampel, model tetap memiliki *f1-score* yang cukup tinggi, yaitu 0,97 dan 0,91. Hal ini menunjukkan bahwa model tidak mengalami penurunan performa yang signifikan meskipun jumlah sampel pada etalase tersebut sedikit.

Namun, pada etalase 11 (Peralatan Pendidikan), model mencatat *f1-score* terendah dibandingkan dengan etalase lainnya, yaitu sebesar 0,79. Meskipun *recall* sebesar 0,84 menunjukkan bahwa sebagian besar produk dari etalase ini berhasil dikenali oleh model, *precision* hanya mencapai 0,75. Hal ini berarti sekitar 25% dari prediksi untuk etalase 11 salah, karena produk tersebut sebenarnya berasal dari etalase lain. Dengan demikian, hal ini mencerminkan adanya penurunan performa model dalam pengklasifikasian produk di etalase 11.

Gambar 4.4 menunjukkan *confusion matrix* yang memetakan prediksi dan etalase aktual. Fokus analisis diarahkan pada etalase 11 (Peralatan Pendidikan), yang memiliki *f1-score* terendah sebesar 0,79, menandakan bahwa model memiliki kelemahan dalam mengklasifikasikan etalase ini dengan baik. Etalase 11 mencatat total 163 *false negative* (FN). Dari total 163 FN pada etalase 11, sebanyak 84 di antaranya disebabkan oleh kesalahan klasifikasi produk ke dalam etalase 13 (Peralatan Perkantoran). Selain itu, terdapat 272 kasus *false positive* (FP), yang berarti 272 produk dari etalase lain salah diklasifikasikan sebagai etalase 11. Dari jumlah tersebut, 122 produk berasal dari etalase 13, menunjukkan bahwa model sering salah mengklasifikasikan produk dari etalase 13 sebagai produk etalase 11. Dengan demikian, distribusi kesalahan FN dan FP mengindikasikan *f1-score* rendah pada etalase 11 disebabkan oleh kesulitan model KNN dalam membedakan produk antara etalase 11 dan etalase 13.



Gambar 4.4 *Confusion Matrix* Hasil Prediksi pada Data Uji

Gambar 4.5 menampilkan visualisasi *WordCloud*, yang mendukung temuan dari *confusion matrix* dan *classification report* sebelumnya, terkait kesalahan model dalam membedakan antara etalase 11 (Peralatan Pendidikan) dan etalase 13 (Peralatan Perkantoran). Kata-kata seperti "meja", "kursi", "lemari", dan sebagainya sering muncul pada kesalahan klasifikasi, yang mengindikasikan bahwa nama-nama produk ini ditemukan di kedua etalase. Selain itu, nama merek seperti "chitose" yang sering muncul juga menandakan bahwa produk dari merek yang sama terdapat di kedua etalase. Akibatnya, model mengalami kesulitan membedakan produk antara kedua etalase tersebut, karena produk dengan nama yang sama atau dari merek yang sama sering diklasifikasikan secara keliru.

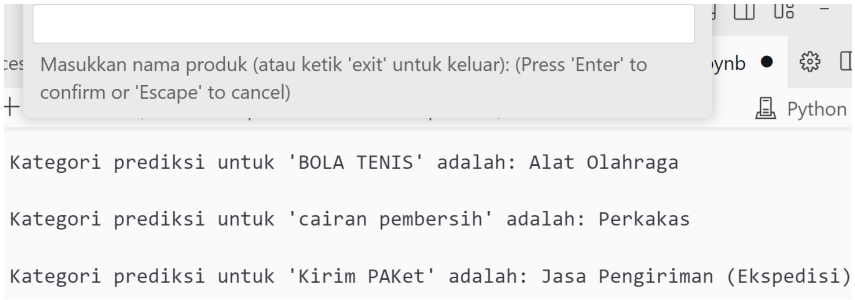
Temuan pada penelitian ini juga didukung berdasarkan penelitian "On the class overlap problem in imbalanced data classification" yang menyebutkan kondisi kesalahan klasifikasi yang diakibatkan kemiripan karakteristik antar data di etalase yang berbeda disebut dengan tumpang tindih etalase (*class overlap*), dimana ditemukan bahwa tumpang tindih etalase secara signifikan mengurangi *recall* model, dan kondisi ini diperburuk ketika ketidakseimbangan etalase juga terjadi [47].

Gambar 4.5 *Wordcloud* Kesalahan Klasifikasi Antara Etalase Pendidikan dan Perkantoran

4.3 Klasifikasi Etalase Produk

Pada tahap ini, dilakukan pengujian klasifikasi etalase produk (*test case*) untuk memastikan keakuratan model klasifikasi dalam menempatkan produk pada etalase yang paling sesuai, berdasarkan nama produk yang diinputkan oleh pengguna,dengan format teks yang berbeda-beda.

Gambar 4.6 menunjukkan bahwa Model *k-nearest neighbors* (KNN) yang digunakan dalam penelitian ini mampu memprediksi kategori produk dengan benar pada ketiga test case tersebut. Produk "BOLA TENIS" diklasifikasikan ke dalam etalase "Alat Olahraga," yang merupakan kategori paling relevan dari total 18 kategori produk yang tersedia. Demikian pula, "cairan pembersih" diprediksi dengan benar sebagai "Perkakas," dan "Kirim PAket" dikategorikan sebagai "Jasa Pengiriman (Ekspedisi)". Dengan demikian, menunjukkan bahwa model klasifikasi knn pada penelitian ini cukup andal dalam mengklasifikasikan kategori produk yang sesuai, bahkan ketika terdapat perbedaan format penulisan nama produk yang diinputkan oleh pengguna.



Gambar 4.6 Hasil Prediksi Produk

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan yang telah diuraikan di bab sebelumnya, diperoleh kesimpulan sebagai berikut:

1. Pemodelan klasifikasi dilakukan melalui beberapa tahapan, yaitu tahap praproses data, tahap ekstraksi fitur menggunakan *Term Frequency-Inverse Document Frequency* (TF-IDF), tahap pemisahan data menggunakan *stratified train-test split*, dan pemodelan menggunakan *K-Nearest Neighbors* (KNN) dengan *grid search* dan *Stratified K-Fold Cross-Validation* (SKCV) untuk *hyperparameter tuning*.
2. Kombinasi *hyperparameter* tetangga terdekat (k) = 4 dan pembobotan berbasis jarak (*distance weighting*) pada KNN menghasilkan akurasi sebesar 90,62%, dengan *macro precision* 91,03%, *macro recall* 91,19%, dan *macro F1-score* 90,96%. Penggunaan pembobotan berbasis jarak, yang memberi bobot lebih besar pada tetangga terdekat dalam *voting*, serta pemisahan data dengan mempertahankan proporsi kelas (*stratified split*) juga membantu model mengenali kelas minoritas, meskipun distribusi data tidak seimbang.
3. Kesalahan klasifikasi tertinggi terjadi pada etalase "Peralatan Pendidikan", dengan nilai *F1-score* terendah sebesar 0,79. Hal ini disebabkan oleh kemiripan nama produk pada etalase "Peralatan Perkantoran", yang menunjukkan kelemahan algoritma *K-Nearest Neighbors* (KNN) di penelitian ini dalam mengklasifikasikan etalase yang memiliki kemiripan nama produk, sehingga menyebabkan peningkatan kesalahan klasifikasi.

5.2 Saran

Berdasarkan hasil penelitian, berikut beberapa saran untuk penelitian lanjutan:

1. Penelitian ini hanya menggunakan data produk e-katalog Kota Bandar Lampung, penelitian selanjutnya disarankan mencakup data e-katalog dari berbagai wilayah lain.
2. Mengeksplorasi metode lainnya untuk penanganan ketidakseimbangan data, guna membandingkan kinerjanya dengan metode yang digunakan dalam penelitian ini.

DAFTAR PUSTAKA

[1] *PERPRES no. 12 tahun 2021*, https://peraturan.bpk.go.id/Details/161828/perpres-no-12-tahun-2021, Diakses: 2024-9-24.

[2] H. A. Al Hikam, *Anggaran pengadaan Barang/Jasa pemerintah rp 1.226 t, wajib buat produk lokal!*, https://finance.detik.com/berita-ekonomi-bisnis/d-7363086/anggaran-pengadaan-barang-jasa-pemerintah-rp-1-226-t-wajib-buat-produk-lokal, Diakses: 2024-7-28.

[3] *INPRES no. 2 tahun 2022*, https://peraturan.bpk.go.id/Details/204320/inpres-no-2-tahun-2022, Diakses: 2024-9-24.

[4] D. Ariza, "E-KATALOG: Langkah strategis pemerintah dalam memerangi fraud pengadaan barang dan jasa", *jmeb*, vol. 4, no. 1, hlmn. 20–29, Jan. 2024.

[5] D. Silvia, M. Sari, dan N. Salma, "Pengaruh sistem informasi akuntansi dan e-commerce terhadap kinerja umkm di kota bandar lampung", *Journal of Finance and Business Digital*, vol. 1, hlmn. 119–128, Okt. 2022.

[6] N. M. N. Mathivanan, N. A. M. Ghani, dan R. M. Janor, "Performance analysis of supervised learning models for product title classification", *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, hlmn. 228, Des. 2019.

[7] D. Sebastian, "Implementasi algoritma k-nearest neighbor untuk melakukan klasifikasi produk dari beberapa e-marketplace", *Jurnal Tekniki Informatika Dan Sistem Informasi*, vol. 5, no. 1, Mei 2019.

[8] S. U. Hassan, J. Ahamed, dan K. Ahmad, "Analytics of machine learning-based algorithms for text classification", *Sustainable Operations and Computers*, vol. 3, hlmn. 238–248, Apr. 2022.

[9] M. Iqbal, "PENGARUH PELAKSANAAN E KATALOG DALAM PENGADAAN BARANG/JASA PEMERINTAH TERHADAP UMKM", *JURNAL USM LAW REVIEW*, vol. 3, no. 1, hlmn. 77, Mei 2020.

[10] Karwiyah, F. F. Eprilia, dan A. P. Pertiwi, "Penerapan Win-Win solution dalam sengketa pengadaan Barang/Jasa pemerintah berdasarkan kontrak secara elektronik melalui katalog Elektronik/E-Purchasing", *Jurnal Hukum Lex Generalis*, vol. 3, hlmn. 291–313, Apr. 2022.

[11] S. Hakak, M. Alazab, S. Khan, T. R. Gadekallu, P. K. R. Maddikunta, dan W. Z. Khan, "An ensemble machine learning approach through effective feature extraction to classify fake news", *Future Generation Computer Systems*, vol. 117, hlmn. 47–58, Apr. 2021.

[12] I. Rozi, V. N. Wijayaningrum, dan N. Khozin, "Klasifikasi teks laporan masyarakat pada situs lapor! menggunakan recurrent neural network", *SISTEMASI*, vol. 9, hlmn. 633–645, Sept. 2020.

[13] M. A. Rosid, A. S. Fitrani, I. R. I. Astutik, N. I. Mulloh, dan H. A. Gozali, "Improving text preprocessing for student complaint document classification using sastrawi", *IOP Conference Series: Materials Science and Engineering*, vol. 874, no. 1, hlmn. 012 017, Juni 2020.

[14] N. Garg dan K. Sharma, "Text pre-processing of multilingual for sentiment analysis based on social network data", *Int. J. Electr. Comput. Eng. (IJECE)*, vol. 12, no. 1, hlmn. 776, Feb. 2022.

[15] M. Işık dan H. Dag, "The impact of text preprocessing on the prediction of review ratings", *TURKISH JOURNAL OF ELECTRICAL ENGINEERING COMPUTER SCIENCES*, vol. 28, hlmn. 1405–1421, Mei 2020.

[16] H.-T. Duong dan T.-A. Nguyen-Thi, "A review: Preprocessing techniques and data augmentation for sentiment analysis", *Computational Social Networks*, vol. 8, Jan. 2021.

[17] A. I. Kadhim, "Term weighting for feature extraction on twitter: A comparison between bm25 and tf-idf", *2019 International Conference on Advanced Science and Engineering (ICOASE)*, hlmn. 124–128, Apr. 2019.

[18] H. D. Abubakar dan M. Umar, "Sentiment classification: Review of text vectorization methods: Bag of words, tf-idf, word2vec and doc2vec", *SLU Journal of Science and Technology*, 2022.

[19] T. V. Maliamanis dan G. A. Papakostas, "Chapter 3 - machine learning vulnerability in medical imaging", di dalam *Machine Learning, Big Data, and IoT for Medical Informatics*, Ser. Intelligent Data-Centric Systems, P. Kumar, Y. Kumar, dan M. A. Tawhid, timed., Academic Press, 2021, hlmn. 53–70. sumber: https://www.sciencedirect.com/science/article/pii/B9780128217771000045.

[20] B. H. Shekar dan G. Dagnew, "Grid search-based hyperparameter tuning and classification of microarray cancer data", *2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP)*, hlmn. 1–8, Feb. 2019.

[21] O. Chamorro-Atalaya, J. Arévalo-Tuesta, D. Balarezo-Mares, dkk., "K-fold cross-validation through identification of the opinion classification algorithm for the satisfaction of university students", *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 19, no. 11, hlmn. 140–158, 2023.

[22] A. Shatnawi, H. M. Alkassar, N. M. Al-Abdaly, E. A. Al-Hamdany, L. F. A. Bernardo, dan H. Imran, "Shear strength prediction of slender steel fiber reinforced concrete beams using a gradient boosting regression tree method", *Buildings*, vol. 12, no. 5, 2022.

[23] S. Prusty, S. Patnaik, dan S. K. Dash, "SKCV: Stratified k-fold cross-validation on ML classifiers for predicting cervical cancer", *Front. Nanotechnol.*, vol. 4, Agt. 2022.

[24] A. Kavitha, P. Suganya, A. G. Prakash, S. Jeevan, dan R. V. Kumar, "A survey on credit card fraud detection using holdout cross validation and stratified k-fold cross-validation", *Journal of Science Transactions in Environmental Technovation*, vol. 17, no. 1, hlmn. 1–9, 2023.

[25] R. Rahim, A. S. Ahmar, dan R. Hidayat, "Cross-validation and validation set methods for choosing k in knn algorithm for healthcare case study", *JINAV: Journal of Information and Visualization*, vol. 3, no. 1, hlmn. 57–61, 2022.

[26] K. Pal dan B. V. Patel, "Data classification with k-fold cross validation and holdout accuracy estimation methods with 5 different machine learning techniques", di dalam *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, 2020, hlmn. 83–87.

[27] S. D. A. Putri, M. O. Ibrohim, dan I. Budi, "Abusive language and hate speech detection for indonesian-local language in social media text", di dalam *Lecture Notes in Networks and Systems*, Ser. Lecture notes in networks and systems, Cham: Springer International Publishing, 2021, hlmn. 88–98.

[28] I. Nti, O. Nyarko-Boateng, dan J. Aning, "Performance of machine learning algorithms with different k values in k-fold cross-validation", *International Journal of Information Technology and Computer Science*, vol. 6, hlmn. 61–71, Des. 2021.

[29] H. A. A. Alfeilat, A. B. A. Hassanat, O. Lasassmeh, dkk., "Effects of distance measure choice on k-nearest neighbor classifier performance: A review", *Big Data*, vol. 6, no. 3, hlmn. 177–185, 2019.

[30] Ş. Kaya dan H. B. Macit, "Classification of lung cancer with deep learning methods using histopathology images", di dalam *Engineering Sciences and Technologies Researches*, H. I. Kurt dan E. Ergul, timed. Lyon, 2023, Chapter VII.

[31] Scikit-learn, *Sklearn.neighbors.kneighborsclassifier*, Diakses: 25-Nov-2024, 2024. sumber: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html.

[32] F. Thabtah, S. Hammoud, F. Kamalov, dan A. Gonsalves, “Data imbalance in classification: Experimental evaluation”, *Information Sciences*, vol. 513, hlmn. 429–441, 2020.

[33] M. Grandini, E. Bagli, dan G. Visani, *Metrics for multi-class classification: An overview*, 2020. arXiv: 2008.05756 [stat.ML].

[34] I. P. Kamila, C. A. Sari, E. H. Rachmawanto, dan N. R. D. Cahyo, “A good evaluation based on confusion matrix for lung diseases classification using convolutional neural networks”, *Advance Sustainable Science, Engineering and Technology*, vol. 6, no. 1, Des. 2023.

[35] A. Nanfak, A. Hechifa, S. Eke, A. Lakehal, C. Kom, dan S. Ghoneim, “A combined technique for power transformer fault diagnosis based on k-means clustering and support vector machine”, *IET Nanodielectrics*, Juli 2024.

[36] D. Valero-Carreras, J. Alcaraz, dan M. Landete, “Comparing two svm models through different metrics based on the confusion matrix”, *Computers Operations Research*, vol. 152, Des. 2022.

[37] A. Muhaimin, W. Wibowo, dan P. Riyantoko, “Multi-label classification using vector generalized additive model via cross-validation”, *Journal of Information and Communication Technology*, vol. 22, hlmn. 657–673, Okt. 2023.

[38] V. Kumar dan B. Subba, “A tfidfvectorizer and svm based sentiment analysis framework for text data corpus”, di dalam *2020 National Conference on Communications (NCC)*, 2020, hlmn. 1–6.

[39] S. Nam, J.-H. Yoo, dan J. W.-K. Hong, “Log-tf-idf for anomaly detection in network switches”, di dalam *NOMS 2024-2024 IEEE Network Operations and Management Symposium*, 2024, hlmn. 1–9.

[40] M. Chiny, M. Chihab, C. Younes, dan O. Bencharef, “Lstm, vader and tf-idf based hybrid sentiment analysis model”, *International Journal of Advanced Computer Science and Applications*, vol. 12, hlmn. 2021, Juli 2021.

[41] D. Bertsimas dan I. Paskov, “Stable regression: On the power of optimization over randomization in training regression problems”, Nov. 2020.

[42] V. R. Joseph, “Optimal ratio for data splitting”, *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 15, no. 4, hlmn. 531–538, Apr. 2022. sumber: http://dx.doi.org/10.1002/sam.11583.

[43] M. Bhagat dan B. Bakariya, “Implementation of logistic regression on diabetic dataset using train-test-split, k-fold and stratified k-fold approach”, *National Academy Science Letters*, vol. 45, Juli 2022.

[44] I. Paryudi, “What affects k value selection in k-nearest neighbor”, *International Journal of Scientific & Technology Research*, vol. 8, no. 7, hlmn. 86–92, 2019.

[45] W. Xing dan Y. Bei, “Medical health big data classification based on knn classification algorithm”, *IEEE Access*, vol. 8, hlmn. 28 808–28 819, 2020.

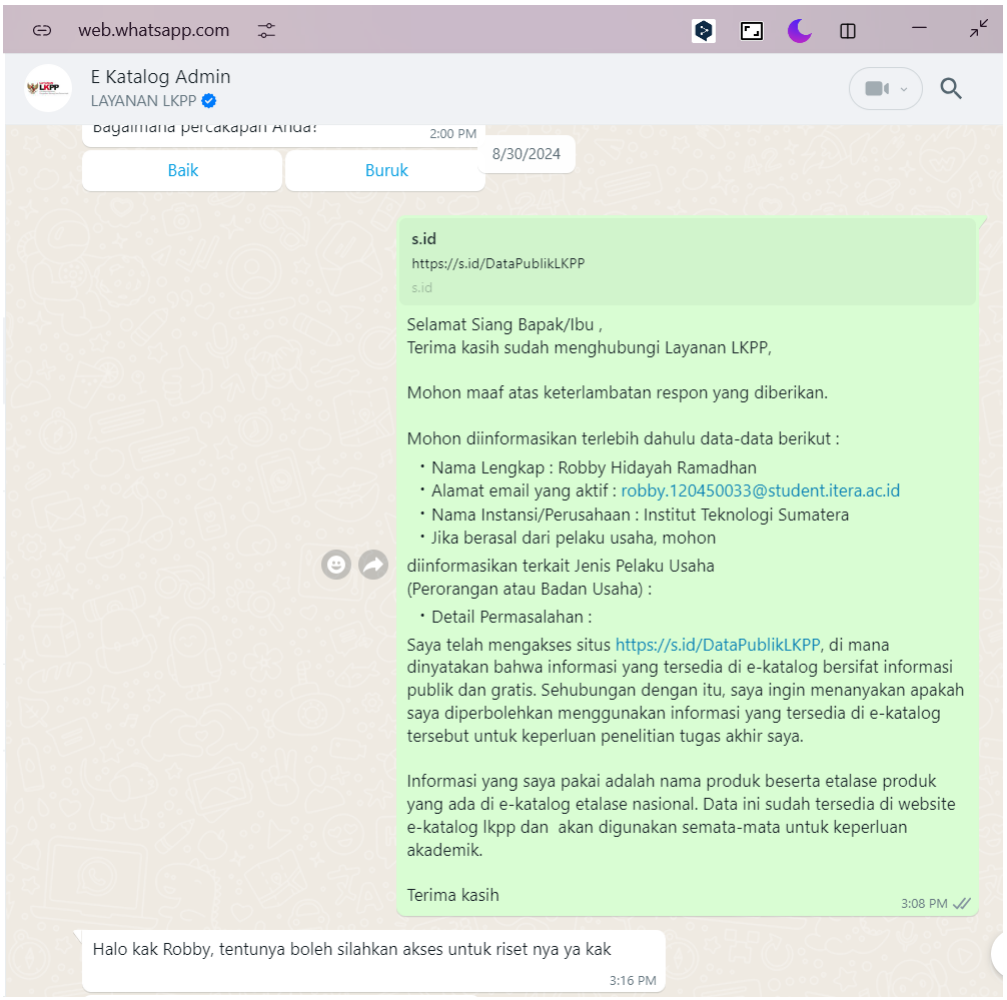
[46] H.-W. Teng dan M. Lee, “Estimation procedures of using five alternative machine learning methods for predicting credit card default”, di dalam *Handbook of Financial Econometrics, Mathematics, Statistics, and Machine Learning*, Bab. Chapter 101, hlmn. 3545–3572.

[47] P. Vuttipittayamongkol, E. Elyan, dan A. Petrovski, “On the class overlap problem in imbalanced data classification”, *Knowladge Based Syst.*, vol. 212, Jan. 2021.

LAMPIRAN

LAMPIRAN A

Keterangan Data Bersifat Publik



Gambar A.1 Keterangan Data bersifat Publik

LAMPIRAN B

Laporan Evaluasi Model pada Data Uji per Etalase

Tabel B.1 Laporan Evaluasi Model pada Data Uji: *Precision*, *Recall*, dan *F1-score* per etalase

Etalase	**Precision**	**Recall**	**F1-Score**	**Support (Jumlah Data Uji)**
0 - Alat Laboratorium	0.91	0.85	0.88	1277
1 - Alat Olahraga	0.91	0.89	0.90	599
2 - Alat Penerangan Jalan	0.96	0.92	0.94	169
3 - Internet Service Provider	0.97	0.97	0.97	145
4 - Jasa Pengiriman (Ekspedisi)	0.96	0.98	0.97	1920
5 - Kendaraan Bermotor	0.97	0.97	0.97	1072
6 - Konsolidasi Logistik Pemilu 2024 Tahap I	1.00	0.93	0.97	15
7 - Konsolidasi Logistik Pemilu 2024 Tahap II	1.00	1.00	1.00	41
8 - Lisensi Perangkat Lunak	0.98	0.96	0.97	1110
9 - Makanan Tambahan	0.88	0.96	0.92	24
10 - Peralatan Elektronik dan Pendukungnya	0.89	0.85	0.87	584
11 - Peralatan Pendidikan	0.75	0.84	0.79	989

Continued on next page

Etalase	Precision	Recall	F1-Score	Support (Jumlah Data Uji)
12 - Peralatan Pengisian Kendaraan Bermotor Listrik (KBL) Berbasis Baterai	0.80	1.00	0.89	4
13 - Peralatan Perkantoran	0.83	0.80	0.82	693
14 - Perangkat Siaran TV Digital	1.00	0.83	0.91	6
15 - Perkakas	0.84	0.86	0.85	982
16 - Sewa Kendaraan	0.92	0.96	0.94	257
17 - Sewa Peralatan Elektronik Perkantoran dan Peralatan Pendukungnya	0.79	0.84	0.81	55