

Implementasi Algoritma *K-Nearest Neighbor* (KNN) untuk Analisis Sentimen Pengguna Aplikasi Tokopedia

M. Rival Ridautal Lillah^{1,*}, Dian Sa'adillah Maylawati², Wildan Budiawan Zulfikar³, Wisnu Uriawan⁴, Agung Wahana⁵

^{1,2,3,4,5}Universitas Islam Negeri Sunan Gunung Djati Bandung, Bandung, Indonesia

Informasi Artikel

Sejarah Artikel:

Dikirim : 23 November 2023

Revisi : 06 Desember 2023

Diterima : 30 Desember 2023

Diterbitkan: 31 Desember 2023

Kata Kunci

K-Nearest Neighbor (KNN), Analisis Sentimen, Aplikasi Tokopedia, CRISP-DM, Akurasi Prediksi.

Correspondence

E-mail: menangeleh123@gmail.com*

A B S T R A K

Marketplace adalah *platform* tempat penjual dapat berkumpul dan menjual barang atau jasa mereka kepada pelanggan tanpa pertemuan fisik. Dalam beberapa dekade terakhir, pasar telah menjadi *platform* paling populer bagi penjual bisnis untuk menjual produk mereka. Menjadi *marketplace* nomor 1 di Indonesia dengan rata-rata pengunjung terbanyak adalah *marketplace* yang tepat di tahun 2023 yaitu Tokopedia. Namun, kebanyakan orang skeptis terhadap produk yang belum pernah mereka beli atau gunakan. Ulasan pengguna memegang peran penting dalam pemasaran produk, terutama di Tokopedia. Ulasan membantu calon pelanggan membangun kepercayaan terhadap produk dan layanan yang ditawarkan oleh penjual. Untuk menganalisis ulasan dengan cepat dan tepat, diperlukan proses analisis sentimen. *Natural Processing Language* (NLP) dan algoritma *text mining* digunakan untuk mengklasifikasikan ulasan sebagai positif, atau negatif. Salah satu metode yang digunakan adalah algoritma *K-Nearest Neighbor* (KNN), yang digunakan untuk mengklasifikasikan ulasan pengguna Tokopedia di *play store* dan *app store*. *Dataset* terdiri dari 1000 data komentar dari *play store* dan 1000 data dari *app store*. Total 2000 komentar yang terdiri dari 2 label yaitu positif dan negatif untuk pemodelan. Sedangkan untuk pengujian 885.092 komentar dari *play store* dan 4000 komentar dari *app store*. Total 889.092 untuk data pengujian yang belum dilabeli. Hasil prediksi pada *dataset app store* menunjukkan terdapat 97,0% prediksi label positif dan hanya 3,0% prediksi label negatif.

Abstract

A marketplace is a platform where sellers can come together and sell their goods or services to customers without physical meetings. In the past few decades, marketplaces have become the most popular platform for business sellers to sell their products. Becoming the number 1 marketplace in Indonesia with the most visitors on average is the right marketplace in 2023, namely Tokopedia. However, most people are skeptical of products they have never purchased or used. User reviews play an important role in product marketing, especially on Tokopedia. Reviews help potential customers build trust in the products and services offered by the seller. To analyze reviews quickly and precisely, a sentiment analysis process is needed. Natural Processing Language (NLP) and text mining algorithms are used to classify reviews as positive, or negative. One of the methods used is the K-Nearest Neighbor (KNN) algorithm, which is used to classify Tokopedia user reviews in the Play Store and App Store. The dataset consists of 1000 comment data from the Play Store and 1000 data from the App Store. A total of 2000 comments consisting of 2 labels, namely positive and negative for modeling. Meanwhile, for testing, there were 885,092 comments from the Play Store and 4000 comments from the App Store. Total 889,092 for unlabeled test data. The prediction results on the app store dataset show that there are 97.0% positive label predictions and only 3.0% negative label predictions.

This is an open access article under the CC-BY-SA license



1. Pendahuluan

Dalam beberapa dekade terakhir, pasar *online* telah menjadi *platform* paling populer bagi penjual bisnis untuk menjual produk mereka. Pembeli dan penjual dapat bertemu langsung dalam satu *platform* untuk transaksi yang cepat, mudah, dan sederhana. Meskipun banyak bermunculan toko *online*, namun *marketplace* tidak bisa diremehkan. Bahkan, beberapa *marketplace* ternama seperti *Amazon*, *eBay*, *Alibaba* dan lain sebagainya telah berperan penting dalam menghidupkan kembali *marketplace online* sedunia [1].

Selain itu, *marketplace* juga banyak digunakan di Indonesia, bahkan banyak *merchant* yang berjualan melalui *e-commerce* dan *marketplace* di Indonesia, tidak hanya oleh pelaku UMKM, tetapi juga *brand-brand* besar telah masuk dan menjual *marketplace online*. Hal ini terlihat dari laju pertumbuhan pasar Indonesia yang terus berkembang pesat, terutama di masa pandemi. 18 *marketplace* terbesar di Indonesia saja memiliki pengunjung bulanan dengan total lebih dari 500 juta pengguna [2].

Menjadi *marketplace* nomor 1 di Indonesia dengan rata-rata pengunjung terbanyak adalah *marketplace* yang tepat di tahun 2023 yaitu Tokopedia. Menurut studi *EcommerceIQ*, salah satu alasan banyak konsumen memilih Tokopedia adalah banyaknya pilihan produk yang tersedia. Di Tokopedia sendiri, kelompok produk yang paling banyak dicari adalah elektronik, disusul *fashion* dan sembako [3]. Dalam dunia *marketplace* terdapat sebuah istilah yang dinamakan ulasan. Ulasan adalah salah satu jenis testimoni yang diberikan oleh konsumen atau pengguna yang merasa puas atau kecewa dengan produk atau jasa yang dibelinya.

Secara umum, kebanyakan orang *skeptis* terhadap produk yang belum pernah mereka beli atau gunakan. Apalagi untuk beberapa produk yang dapat mempengaruhi kehidupan, seperti produk kesehatan, produk kecantikan dan lain- lain. Semua pebisnis tahu ini. Oleh karena itu, para pedagang biasanya memasang ulasan pengguna di halaman penjualan mereka sehingga testimoni pengguna mereka akan meningkatkan kepercayaan calon pelanggan lain terhadap produk yang mereka jual. Berdasarkan penjelasan tersebut dapat disimpulkan bahwa ulasan memegang peranan yang sangat penting dalam proses pemasaran suatu produk di pasar, khususnya di Tokopedia.

Ulasan atau komentar pengguna bertujuan untuk memberikan manfaat dan dapat dipercaya. Mengulas konten di *play store* dan *app store* adalah cara efektif untuk berbagi masukan bermanfaat dan membantu pengguna lain menemukan sebuah produk, konten, dan layanan yang bagus. Dengan berbagai macam dan banyaknya pengguna Tokopedia memberikan ulasan perlu sebuah proses untuk menganalisis ulasan tersebut secara cepat dan tepat yang biasa disebut dengan analisis sentimen [4].

Pendekatan analisis sentimen digunakan untuk menganalisis informasi berupa opini publik untuk mendukung pengambilan keputusan. Tugas analisis sentimen adalah mengklasifikasikan kumpulan kutub dari teks, kalimat, atau *fitur* dari keseluruhan dokumentasi, yang aspeknya positif, *netral*, atau negatif. Dalam analisis sentimen juga harus menerapkan sebuah metode atau algoritma dalam kasus klasifikasi respon pengguna Tokopedia di *play store* dan *app store*.

Algoritma *K-Nearest Neighbor* (KNN) adalah metode atau pendekatan untuk mengklasifikasikan objek berdasarkan data *training* yang paling dekat dengan objek tersebut. Data *training* diproyeksikan ke dalam ruang multidimensi, di mana setiap dimensi menggambarkan properti data [5].

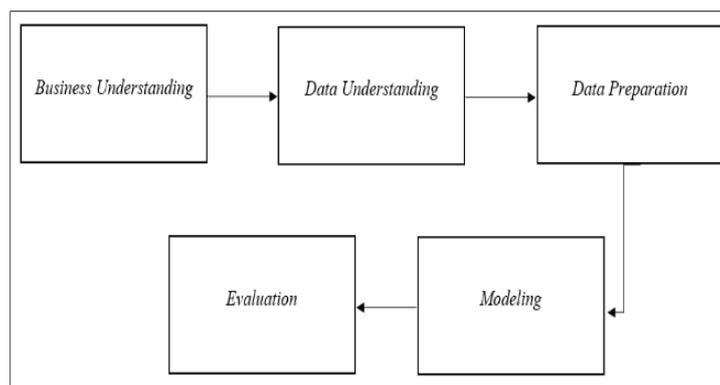
Tujuan penelitian yang dipaparkan dalam penelitian ini adalah mengetahui hasil klasifikasi yang didapatkan dari analisis sentimen pengguna Tokopedia di *play store* dan *app store* dengan menggunakan algoritma *K-Nearest Neighbor* (KNN).

Manfaat dari penelitian ini meliputi pemahaman yang lebih mendalam tentang cara mengklasifikasikan opini pengguna Tokopedia di *platform Play Store* dan *App Store* menggunakan algoritma *K-Nearest Neighbor* (KNN). Selain itu, penelitian ini juga memungkinkan pengklasifikasian

opini dalam jumlah besar menjadi kategori positif dan negatif dengan cepat. Hasil dari analisis sentimen yang dihasilkan dari penelitian ini memiliki potensi besar untuk menjadi sumber informasi yang berharga bagi Tim Tokopedia dalam mengevaluasi perkembangan dan perbaikan aplikasi Tokopedia di masa yang akan datang.

Penelitian ini menggunakan 1000 data komentar dari *play store* dan 1000 data dari *app store*. Total 2000 komentar yang terdiri dari 2 label yaitu positif dan negatif untuk pemodelan. Sedangkan untuk pengujian 885.092 komentar dari *play store* dan 4000 komentar dari *app store*. Total 889.092 untuk data pengujian yang belum dilabeli.

2. Metodologi Penelitian



Gambar 1. Metode Penelitian

Gambar 1 merupakan alur dari metode CRISP-DM yang akan diimplementasikan dalam penelitian ini. Metode ini harusnya terdiri dari 6 tahap. Akan tetapi dalam penelitian ini pada tahap *deployment* itu dihapus karena tahap ini tidak diperlukan dalam penelitian. Jadi tahapan dalam metode ini ada 5 tahap, yaitu *business understanding*, *data understanding*, *data preparation*, *modeling*, dan *evaluation*.

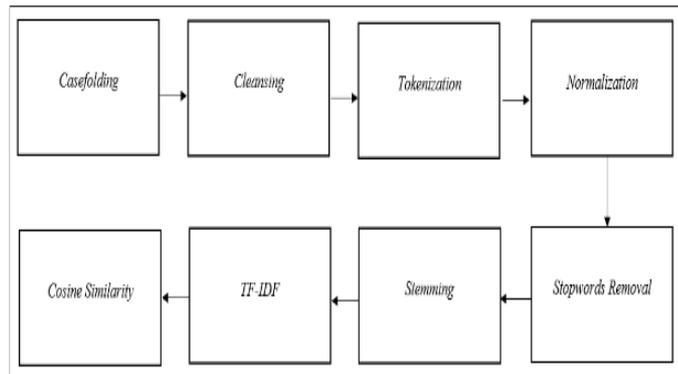
2.1. Business Understanding

Tahap *Business Understanding* dalam penelitian ini mengidentifikasi dua masalah bisnis utama: meningkatkan kualitas layanan dan memahami persepsi pelanggan Tokopedia. Untuk mencapai tujuan ini, peneliti memahami konteks bisnis, termasuk persaingan dengan *platform e-commerce* lainnya dan kebutuhan pelanggan. *Stakeholder* seperti manajemen Tokopedia, tim *developer*, dan pelanggan juga diidentifikasi. Tujuan analisis data pada tahap ini adalah memahami sentimen pelanggan dan meningkatkan layanan. Hasil dari tahap ini menjadi dasar pemilihan metode analisis data, yaitu algoritma *K-Nearest Neighbor (KNN)* dalam *Supervised Learning*.

2.2. Data Understanding

Setelah melakukan tahap *Business Understanding* selanjutnya adalah tahap *Data Understanding*. Pada penelitian ini yaitu akan melakukan kegiatan analisis sentimen Tokopedia menggunakan 1000 data komentar dari *play store* dan 1000 data dari *app store*. Total 2000 komentar yang terdiri dari 2 label yaitu positif dan negatif untuk pemodelan. Sedangkan untuk pengujian 885.092 komentar dari *play store* dan 4000 komentar dari *app store*. Total 889.092 untuk data pengujian yang belum dilabeli.

2.3. Data Preparation



Gambar 2. Data Preparation

Terdiri dari proses *casefolding*, *cleansing*, *tokenization*, *normalization*, *stopwords removal*, *stemming*, *tf-idf*, dan *cosine similarity*.

Berikut pada Tabel 1 merupakan proses *casefolding*:

Tabel 1. Proses *Casefolding*

Sebelum	Sesudah
Aplikasi Tokopedia sangat berguna!	aplikasi tokopedia sangat berguna!
Tokopedia memiliki layanan pelanggan yang luar biasa	tokopedia memiliki layanan pelanggan yang luar biasa
Saya kesal dengan aplikasi TOKOPEDIA	saya kesal dengan aplikasi tokopedia
Tokopedia adalah platform yang hebat	tokopedia adalah platform yang hebat
Aplikasi Tokopedia sudah memudahkan hidup saya.	aplikasi tokopedia sudah memudahkan hidup saya

Pada proses *casefolding*, semua ulasan dieja dalam huruf kecil (*lowercase*) untuk memudahkan analisis sentimen tanpa memedulikan perbedaan huruf besar dan kecil. Dalam proses ini membantu menyamakan ulasan dengan mengubah semua huruf menjadi huruf kecil, sehingga kita dapat menganalisis sentimen dengan lebih akurat tanpa terganggu oleh perbedaan huruf besar dan kecil dalam teks.

Tabel 2. Proses *Cleansing*

Sebelum	Sesudah
aplikasi tokopedia sangat berguna!	aplikasi tokopedia sangat berguna
saya kesal dengan tokopedia, ini buruk.	saya kesal dengan tokopedia ini buruk
pengalaman belanja di tokopedia menyenangkan.	pengalaman belanja di tokopedia menyenangkan
pelanggan tokopedia yang puas.	pelanggan tokopedia yang puas
kok makin kesal ya??	kok makin kesal ya

Setelah proses *casefolding* selanjutnya yaitu proses *cleansing*. Pada Tabel 2 merupakan contoh proses dari *cleansing* yang berfungsi untuk membersihkan dan memproses teks dengan menghilangkan karakter-karakter tertentu yang tidak diperlukan atau mengganggu dalam analisis teks atau pemrosesan teks lebih lanjut.

Selanjutnya yaitu proses *tokenization* yang bisa dilihat pada Tabel 3.

Tabel 3. Proses *Tokenization*

Sebelum	Sesudah
aplikasi tokopedia sangat berguna	["aplikasi", "tokopedia", "sangat", "berguna"]
saya kesal dengan tokopedia ini buruk	["saya", "kesal", "dengan", "tokopedia", "ini", "buruk"]
pengalaman belanja di tokopedia menyenangkan	["pengalaman", "belanja", "di", "tokopedia", "menyenangkan"]
pelanggan tokopedia yang puas	["pelanggan", "tokopedia", "yang", "puas"]
kok makin kesal ya	["kok", "makin", "kesal", "ya"]

Tokenisasi adalah proses kunci dalam pemrosesan teks yang bertujuan memisahkan teks menjadi unit-unit lebih kecil, dikenal sebagai token, untuk memungkinkan analisis dan pemahaman teks yang lebih efisien. Ini adalah langkah krusial dalam mengurai teks menjadi elemen-elemen yang dapat diolah dan dianalisis lebih lanjut dalam berbagai konteks aplikasi seperti pemrosesan bahasa alami, analisis sentimen, dan pencarian informasi.

Sedangkan pada proses *normalization* mengubah data ke dalam format yang seragam atau standar. Ini bertujuan untuk mengurangi ambiguitas, meningkatkan efisiensi analisis, dan memudahkan perbandingan data. Dalam konteks pengolahan basis data, normalisasi dapat merujuk pada proses merancang struktur basis data agar lebih efisien dan meminimalkan duplikasi data serta menjaga integritas data. Dalam analisis data, normalisasi dapat berarti mengubah skala data agar seragam, memungkinkan variabel dengan skala yang berbeda untuk dibandingkan dengan adil.

Tabel 4. Proses *Stopwords Removal*

Sebelum	Sesudah
[dipakai, di, ip, ngadat, aneh]	[dipakai, ip, ngadat, aneh]
[nyaman, belanja, di, toped]	[nyaman, belanja, toped]
[tidak, pernah, mengecewakan]	[mengecewakan]
[mantap, banget, tokped, aseli]	[mantap, banget, tokped, aseli]
[belanja, selalu, tokopedia]	[belanja, tokopedia]
[sangat, membantu, dalam berbelanja]	[membantu, berbelanja]
[sukses, buat, tokped]	[sukses, tokped]
[cashbacknya, gokil]	[cashbacknya, gokil]
[tokopedia, sangat, baik, dan, terpercaya]	[tokopedia, terpercaya]
[sangat, bagus, sangat, mudah, dan, sangat, cepat]	[bagus, mudah, cepat]

Tabel 4 merupakan contoh proses *stopwords removal* yaitu untuk meningkatkan efektivitas dan keakuratan dalam memproses teks. *Stopwords* adalah kata-kata yang biasanya muncul dalam teks seperti kata depan, kata sambung, dan kata-kata lainnya yang tidak memiliki arti khusus dalam konteks tertentu. Ketika melakukan proses analisis teks, *stopwords* sering tidak memberikan kontribusi signifikan dalam memahami isi teks dan dapat mengurangi efisiensi dan akurasi analisis.

Tabel 5. Proses *Stemming*

Sebelum	Sesudah
[dipakai, ip, ngadat, aneh]	pakai ip ngadat aneh
[nyaman, belanja, toped]	nyaman belanja toped
[mengecewakan]	kecewa
[mantap, banget, tokped, aseli]	mantap banget tokped aseli
[belanja, tokopedia]	belanja tokopedia
[membantu, berbelanja]	bantu belanja
[sukses, tokped]	sukses tokped
[cashbacknya, gokil]	cashbacknya gokil
[tokopedia, terpercaya]	tokopedia percaya
[bagus, mudah, cepat]	bagus mudah cepat

Tujuan utama dari proses *stemming* adalah untuk meningkatkan efisiensi dan akurasi dalam melakukan analisis teks. Dalam beberapa kasus, kata-kata dalam teks mungkin memiliki berbagai variasi bentuk yang berbeda, dan hal ini dapat mengurangi efisiensi dan akurasi analisis. Dengan

melakukan *stemming*, kata-kata yang berasal dari kata dasar yang sama dapat dihitung sebagai satu kesatuan, sehingga proses analisis dapat dilakukan dengan lebih efisien dan akurat.

Tabel 6. Proses *TF-IDF*

Indeks Dokumen	Indeks Istilah (Kata)
(0, 0)	0.5
(0, 13)	0.5
(0, 9)	0.5
(0, 15)	0.5
(1, 20)	0.62583987960545
(1, 5)	0.4654555727358646
(1, 14)	0.62583987960545
(2, 10)	1.0
(3, 1)	0.5182909034319405
(3, 19)	0.44059461896295216
(3, 3)	0.5182909034319405
(3, 11)	0.5182909034319405
(4, 18)	0.7526207669831266
(4, 5)	0.6584542361514052
(5, 4)	0.8024086456077293
(5, 5)	0.5967749705324189
(6, 17)	0.761904967498719
(6, 19)	0.6476888299953735
(7, 8)	0.7071067811865475
(7, 6)	0.7071067811865475
(8, 16)	0.761904967498719
(8, 18)	0.6476888299953735
(9, 7)	0.5773502691896257
(9, 12)	0.5773502691896257
(9, 2)	0.5773502691896257

Tujuan dari proses *TF-IDF* adalah untuk menemukan kata-kata yang paling penting dalam sebuah dokumen dengan cara menghitung frekuensi kemunculan kata dalam dokumen (TF) dan memperhitungkan seberapa umum kata tersebut di seluruh dokumen (IDF).

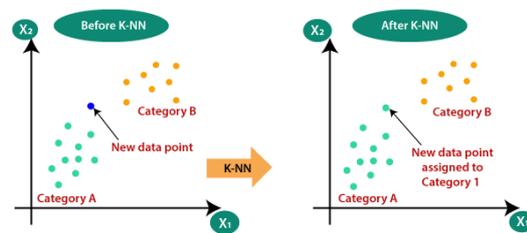
Tabel 7. Proses *Cosine Similariy*

[1.	0.	0.	0.
0.		0.		
0.		0.		0.
[0.	1.	0.	0.
0.30648119	0.27777224			
0.		0.		0.
]				
[0.	0.	1.	0.
0.		0.		
0.		0.		0.
]				
[0.	0.	0.	1.
0.		0.		
0.28536821	0.		0.	0.
]				
[0.	0.30648119	0.	0.
1.	0.39294901			
0.		0.	0.48746406	0.
]				
[0.	0.27777224	0.	0.

```

0.39294901 1.
  0.          0.          0.          0.
]
[0.          0.          0.
0.28536821 0.          0.
  1.          0.          0.          0.
]
[0.          0.          0.          0.
0.          0.
  0.          1.          0.          0.
]
[0.          0.          0.          0.
0.48746406 0.
]
    
```

Proses *cosine similarity* bertujuan untuk menentukan seberapa dekat dua dokumen atau teks berdasarkan kesamaan antara representasi vektor dokumen tersebut. Dalam hal ini, setiap dokumen atau teks direpresentasikan sebagai vektor berdasarkan jumlah kemunculan kata-kata dalam dokumen atau teks tersebut.



Gambar 3. Ilustrasi Pemodelan KNN

Selanjutnya yaitu tahap *modeling* yaitu fokus berpindah ke pembuatan dan penilaian model berdasarkan pemahaman sebelumnya terhadap data. Dalam konteks pengujian model, terdapat dua pendekatan umum yang digunakan untuk mengukur kinerja dan mengoptimalkan model. Pertama, pengujian dengan nilai k 1 hingga 10, di mana data dibagi menjadi k subset yang saling bergantian digunakan sebagai data pelatihan dan pengujian. Pendekatan ini membantu mengatasi fluktuasi hasil dan memberikan gambaran yang lebih stabil tentang performa model.

Selanjutnya, pengujian dengan pembagian rasio data, seperti 90:10, 80:20, 70:30, atau 60:40, membagi data menjadi dua bagian, satu untuk pelatihan dan yang lainnya untuk pengujian. Evaluasi model melibatkan pengukuran metrik performa, seperti akurasi atau *F1-score*, untuk memilih konfigurasi yang paling sesuai dengan karakteristik data dan tujuan analisis penelitian ini.

Terakhir yaitu tahap *evaluation* dilakukan untuk mengevaluasi kinerja model dalam melakukan klasifikasi sentimen pada data uji yang belum pernah dilihat sebelumnya. Pada tahap ini, model akan diuji dengan data uji dan evaluasi akan dilakukan dengan menggunakan *confusion matrix*, *akurasi*, *presisi*, *recall*, *f1-score*, dan *error rate*. *Confusion matrix* digunakan untuk memahami seberapa baik model klasifikasi dapat memprediksi kelas yang benar dan seberapa besar kemungkinan model tersebut melakukan kesalahan dalam memprediksi kelas.

Akurasi merupakan metrik yang digunakan untuk mengukur seberapa akurat model dalam memprediksi sentimen pada data uji. Presisi dan *recall* digunakan untuk mengukur seberapa baik model dalam mengklasifikasikan sentimen yang sebenarnya positif atau negatif. *F1-Score* merupakan *harmonic mean* dari *presisi* dan *recall*. Selain itu, *error rate* juga digunakan untuk mengetahui ukuran kesalahan model yang menghitung berapa persen data yang salah diklasifikasikan oleh model dari total data yang diprediksi.

3. Hasil dan Pembahasan

Hasil dari proses *splitting dataset* ulasan pengguna Tokopedia di *Play Store* menjadi subset uji dan pelatihan. Subset uji terdiri dari 200 data dengan 1000 fitur, sementara subset pelatihan terdiri dari 800 data dengan 1000 fitur. Selain itu, *output* juga mencantumkan jumlah data dengan sentimen positif dan negatif pada subset uji dan pelatihan. Terdapat 99 data dengan sentimen positif dan 101 data dengan sentimen negatif pada subset uji, sedangkan pada *subset* pelatihan terdapat 377 data dengan sentimen positif dan 423 data dengan sentimen negatif. *Output* terakhir menunjukkan distribusi jumlah data dengan sentimen positif (476) dan sentimen negatif (524) dalam dataset secara keseluruhan.

Sedangkan hasil proses *splitting dataset* ulasan pengguna Tokopedia di *App Store* dan distribusi sentimen dalam dataset. Pertama, subset uji terdiri dari 200 data dengan 1000 fitur, sedangkan *subset* pelatihan terdiri dari 800 data dengan 1000 *fitur*. Selanjutnya, *output* menyajikan jumlah data dengan sentimen positif dan negatif pada subset uji dan pelatihan. Terdapat 163 data dengan sentimen positif dan 37 data dengan sentimen negatif pada subset uji, sedangkan pada subset pelatihan terdapat 632 data dengan sentimen positif dan 168 data dengan sentimen negatif. *Output* terakhir menunjukkan distribusi sentimen keseluruhan dalam dataset, dengan 795 data memiliki sentimen positif dan 205 data memiliki sentimen negatif.

Tabel 8. Hasil Prediksi Sentimen *Play Store*

Nilai K	Jumlah Prediksi	
	Positif	Negatif
1	45	61
2	94	87
3	104	99
4	110	96
5	107	96
6	107	96
7	107	97
8	111	99
9	110	90
10	104	96

Tabel 9. Hasil Prediksi Sentimen *App Store*

Nilai K	Jumlah Prediksi	
	Positif	Negatif
1	84	26
2	128	52
3	194	26
4	135	65
5	137	63
6	142	58
7	135	65
8	138	62
9	145	55
10	144	56

Selanjutnya pada tahap evaluation terdiri dari 7 proses yaitu *confusion matrix*, *accuracy*, *precision*, *recall*, *f1-score*, dan *error rate*.

```
[ ] # Evaluation
print(f'confusion matrix:\n {confusion_matrix(y_test, predicted)}')
print('=====')
tn, fp, fn, tp = confusion_matrix(y_test, predicted).ravel()
print("TN:", tn)
print("FP:", fp)
print("FN:", fn)
print("TP:", tp)
```

Gambar 4. Proses *Confusion Matrix Play Store* dan *App Store*

Dibawah ini Tabel 10 dan 11 merupakan hasil dari proses *confusion matrix* dari setiap pendekatan jumlah K sebanyak 10 kali percobaan :

Tabel 10. Hasil *Confusion Matrix Play Store*

Nilai K	Confusion Matrix			
	TN	FP	FN	TP
1	95	6	16	83
2	99	2	24	75
3	93	8	12	87
4	93	8	15	84
5	91	10	10	89
6	90	11	15	84
7	89	12	12	87
8	88	13	14	85
9	87	14	13	86
10	87	14	15	84

Tabel 11. Hasil *Confusion Matrix App Store*

Nilai K	Confusion Matrix			
	TN	FP	FN	TP
1	24	13	16	147
2	30	7	23	140
3	24	13	11	152
4	26	11	20	143
5	23	14	12	151
6	24	13	16	147
7	23	14	12	151
8	24	13	15	148
9	24	13	14	149
10	25	12	16	147

Dalam analisis sentimen, nilai akurasi digunakan sebagai salah satu metrik evaluasi untuk mengukur seberapa baik model dapat memprediksi sentimen dengan benar. Tujuan dari nilai akurasi dalam analisis sentimen adalah untuk mengukur sejauh mana model dapat membedakan antara sentimen positif dan negatif dengan akurasi yang tinggi.

```
print("Accuracy:" , accuracy_score(y_test,predicted))
```

Gambar 5. Proses *Accuracy Data Play Store dan App Store*

Tabel 12. Hasil Akurasi *Play Store*

Nilai K	Nilai Akurasi
1	0.89
2	0.87
3	0.9
4	0.885
5	0.9
6	0.87
7	0.88
8	0.865
9	0.865
10	0.855

Tabel 13. Hasil Akurasi *App Store*

Nilai K	Nilai Akurasi
1	0.855
2	0.85
3	0.88
4	0.845
5	0.87
6	0.855
7	0.87
8	0.86
9	0.865
10	0.86

Pada Tabel 12 merupakan hasil dari percobaan atau analisis yang dilakukan pada *dataset play store*. Berdasarkan data yang tersedia, dapat disimpulkan bahwa nilai K adalah parameter dalam algoritma *k-nearest neighbors* (k-NN) yang digunakan untuk klasifikasi dan regresi. Nilai K menentukan jumlah tetangga terdekat yang digunakan untuk membuat prediksi pada setiap data poin. Di sisi lain, nilai akurasi mengindikasikan tingkat keberhasilan model k-NN dalam melakukan prediksi yang benar. Untuk mengukur akurasi, prediksi model dibandingkan dengan label yang sebenarnya pada data yang diketahui. Dalam tabel ini, nilai K berkisar dari 1 hingga 10, sementara nilai akurasi berkisar antara 0.855 dan 0.9.

Sedangkan Pada Tabel 13 merupakan hasil dari percobaan atau analisis yang dilakukan pada *dataset app store*. Dalam tabel ini, nilai K bervariasi antara 1 hingga 10 juga, sedangkan nilai akurasi berkisar antara 0.845 dan 0.88. Dari hasil Tabel 12 akan diambil model yang mempunyai nilai akurasi paling tinggi yaitu 0.9 untuk diimplementasikan dalam pengujian terhadap dataset yang belum mempunyai label untuk diprediksi labelnya. Pada Tabel 13 juga akan diambil nilai akurasi paling tinggi yaitu 0.88. Dengan diambilnya nilai k terbaik dari kedua *dataset* tersebut nanti akan dilakukan pengujian nilai k terbaik dengan data *splitting* berbeda-beda, mulai dari 90:10, 80:20, 70:30, dan 60:40. Setelah itu akan diimplementasikan terhadap dataset yang belum dilabeli.

Tabel 14. Hasil *Error Rate Play Store*

Nilai K	Error Rate
1	0.10
2	0.13
3	0.09
4	0.11
5	0.09
6	0.13
7	0.12
8	0.13
9	0.13
10	0.14

Tabel 15. Hasil *Error Rate App Store*

Nilai K	Error Rate
1	0.14
2	0.15
3	0.12
4	0.15
5	0.13
6	0.14
7	0.13
8	0.14
9	0.13
10	0.14

Setelah melakukan percobaan model K sebanyak 10 kali dengan nilai 1-10, maka dapat disimpulkan nilai akurasi yang terbaik dari seluruh model adalah nilai K = 5 yaitu sebesar 0.9 atau 90% pada *dataset play store* dan nilai K = 3 yaitu sebesar 0.88 atau 88% pada *dataset app store*. Sehingga model KNN terbaik akan diuji untuk diimplementasikan dalam prediksi otomatis *dataset* ulasan pengguna Tokopedia yang belum ada label positif dan negatif pada Mei 2023. Jumlah komentar yang diambil untuk pengujian yaitu sebanyak 885.092 dari komentar pengguna Tokopedia di *play store* dan 4000 komentar penguuna Tokopedia di *app store*.

Untuk *dataset play store* dilakukan data splitting dengan 4 kali pengujian yaitu 90:10, 80:20, 70:30, dan 60:40 dengan nilai K=5. Setiap hasil pengujian diimplementasikan terhadap *dataset* yang belum dilabeli positif dan negatif.

Berikut di bawah ini pada tabel 16 hasil nilai akurasi dari 4 kali pengujian :

Tabel 16. Nilai Akurasi 4 Kali Pengujian *Data Play Store*

Pembagian Data	Nilai Akurasi
90 : 10	0.88
80 : 20	0.905
70 : 30	0.89
60 : 40	0.875

Pada Tabel 16 menunjukkan bahwa nilai akurasi tertinggi yaitu dengan pembagian data latih dan data uji sebanyak 80:20 dengan nilai akurasi sebesar 0.905. Sedangkan untuk nilai akurasi terendah yaitu 60:40 sebesar 0.875. Selanjutnya yaitu menggunakan model terbaik di atas yaitu dengan pembagian data 80:20 digunakan untuk pelabelan otomatis.

```
import joblib
# Simpan model dan vectorizer menggunakan joblib
joblib.dump(vectorizer, 'vectorizer_ps.pkl')
joblib.dump(knn_model, 'model_ps.pkl')
```

Gambar 6. Kode Program Untuk Menyimpan Model Terbaik *Play Store*

Pada Gambar 6 merupakan kode program yang digunakan untuk menyimpan model terbaik *dataset play store* ke dalam *format pickle*. Kedua file itu yang akan dipakai untuk pengujian label pada *dataset* baru.

Tabel 17. Nilai Akurasi 4 Kali Pengujian *Data App Store*

Pembagian Data	Nilai Akurasi
90 : 10	0.9
80 : 20	0.88
70 : 30	0.88
60 : 40	0.87

Nilai akurasi terbaik pada Tabel 17 yaitu dengan pembagian data 90:10 yaitu 0.9 dan paling rendah dengan pembagian data 60:40 yaitu hanya 0.87 nilai akurasinya. Jadi, nilai akurasi 0.9 ini akan digunakan untuk pengujian label otomatis pada *dataset app store* yang baru.

- [5] Yosi Ghea Prasetyowati, "Klasifikasi menggunakan Metode KNN (K-Nearest Neighbor) dalam Python," *medium.com*, 2019.