Terakreditasi SINTA Peringkat 4
Surat Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti No. 28/E/KPT/2019
masa berlaku mulai Vol.3 No. 1 tahun 2018 s.d Vol. 7 No. 1 tahun 2022

Terbit online pada laman web jurnal:
http://publishing-widyagama.ac.id/ejournal-v2/index.php/jointecs

JOINTECS

JOINTECS
(Journal of Information Technology and Computer Science)

Vol. 6 No. 3 (2021) 145 - 152

e-ISSN:2541-6448 p-ISSN:2541-3619

Pebandingan Performa *Naïve Bayes* dan KNN pada Klasifikasi Teks Sentimen Jasa Ekspedisi

Zuda Pradana Putra[1], Aryo Nugroho[2]
Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Narotama
[1]zudapradanaputra.17@fasilkom.narotama.ac.id, [2]aryo.nugroho@narotama.ac.id

Abstract

This study aims to determine the performance of the naive bayes model and KNN (K-Nearest Neighbor) in classifying sentiment text expedition services. Twitter user reviews on @jntexpressid, @JNE_ID, and @posindonesia will be mined and classified based on positive, neutral, and negative sentiments Data was mined for two months from October1 to December 1 2020, the results obtained on the JNT data are 46.220, JNE 5.364, and Pos Indonesia 11.194. The raw data obtained will be cleaned and labeled before entering using TF-IDF for term weighting. The clean data obtained from the pre-processing of the text will be divided into 70% training data and 30% test data. The over-sampling method is used to balance the dataset and avoid overfitting training data. Naive ayes model and KNN there was a significant increase after the over-sampling method was implemented. The greatest accuracy in naive bayes is found in JNT data of 82%, while KNN on POS data is 71%. In KNN a high K value does not determine the accuracy results, it is proven that K = 2 has the highest accuracy compared to K = 4, K = 6, K = 8, K = 10 after the resampling technique is carried out.

Keywords: KNN; naïve bayes; TF-IDF; over-sampling; sentiment.

Abstrak

Penelitian ini bertujuan untuk mengetahui performa dari model *naïve bayes* dan KNN (*K-Nearest Neighbor*) dalam mengklasifikasikan dataset teks sentimen jasa ekspedisi. Ulasan pengguna twitter pada akun @jntexpressid, @JNE_ID, dan @posindonesia akan di tambang dan diklasifikan berdasarkan sentimen positif, netral, dan negatif. Data digali dari 1 Oktober hingga 1 Desember 2020, hasil didapat pada data JNT sebanyak 46.220, JNE 5.364, dan Pos Indonesia 11.194. Data mentah yang didapat akan dibersihkan dan dilabeli sebelum masuk ketahap pembobotan kata menggunakan TF-IDF. Data bersih yang didapat dari pra-pemrosesan teks akan dibagi menjadi data latih sebanyak 70% dan data uji 30% untuk diuji akurasinya kesetiap model. Metode *over-sampling* digunakan untuk meyeimbangkan dataset dan menghindari data latih yang *overfitting*. Pada model *naïve bayes* dan KNN terjadi peningkatan cukup signifikan setelah metode *over-sampling* diimplementasikan. Akurasi terbesar pada *naïve bayes* terdapat pada data JNT 82%, sedangkan KNN pada data POS 71%. Pada KNN nilai K tinggi tidak menentukan hasil akurasi, terbukti K=2 memiliki akurasi tertinggi dibanding K=4, K=6, K=8, K=10 setelah teknik resampling dilakukan.

Kata kunci: KNN; *naïve bayes*; TF-IDF; *over-sampling*; sentimen.

Diterima Redaksi : 01-08-2021 | Selesai Revisi : 18-09-2021 | Diterbitkan Online : 30-09-2021

1. Pendahuluan

Pemanfaatan data pada klasifikasi teks sentimen di media sosial memiliki potensi besar dalam menghasilkan informasi. Twitter biasa digunakan sebagai tempat promosi, menguraikan isi pikiran, politik, memberikan opini atau kritik atas suatu masalah. Cuitan yang tersedia dari twitter dapat dimanfaatkan untuk mendapatkan infromasi (*text mining*)[1]. Sentimen akan diproses untuk menghasilkan informasi atau pola yang dapat diubah sebagai ilmu pengetahuan baru[2]. Tujuan dari pengklasifikasian sentimen adalah untuk menemukan kecendurangan pola opini dari ulasan pengguna untuk menganalisa dan menemukan masalah[3]. Analisis sentimen adalah bidang studi yang digunakan untuk menganalisa pendapat yang dicurahkan seseorang dengan tersurat terkait topik masalah tertentu[4]. Terdapat banyak model klasifikasi yang dapat digunakan sebagai pengolahan sentimen seperti naïve bayes dan KNN.

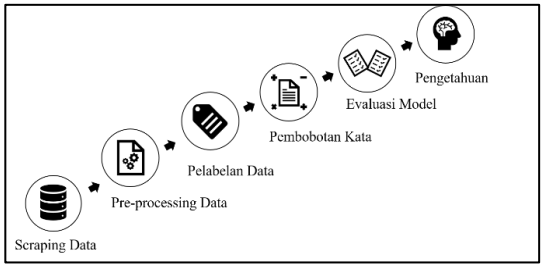
Permasalahan yang sering dialami penjual daring adalah penilaian buruk dari pelanggan karena keterlambatan pengiriman ekspedisi. Masalah tersebut dapat diatasi dengan menganalisa ulasan atau sentimen pengguna twitter terhadap akun jasa ekspedisi untuk menghasilkan informasi mengenai performa perusahaan. Pada penelitian ini data yang digali berkaitan tentang sentimen pengguna jasa ekspedisi di Indonesia pada akun media sosial twitter @posindonesia, @JNE_ID, dan @jntexpressid. Pemilihan ketiga jasa ekspedisi tersebut didasari aktifnya akun twitter sehingga memudahkan pengambilan dari data. Peneliti juga membandingkan ulasan positif maupun negatif dari pemain baru JNT, dan pemain lama yaitu JNE, dan Pos. Dalam pengklasifikasian akan digunakan model naïve bayes dan KNN, kedua model ini digunakan karena mudah dalam implementasinya. Pengujian perbandingan akurasi dari kedua model tersebut adalah untuk mengetahui akurasi terbaik dalam menyelesaikan klasifikasi dengan data teks sentimen jas ekspedisi. Naïve bayes adalah model yang sangat efektif digunakan dalam penyelesaian masalah klasifikasi[5]. Konsep dari bayesian ini berdasarkan probabilitas data yang ada. Sedangkan model klasifikasi KNN memiliki ciri penyelesaian dengan cara menghitung kedekatan objek K atau ketetanggaan paling mendekati. KNN atau yang biasa disebut lazy learning adalah metode klasifikasi sederhana yang berfokus mencari ketetanggan terdekat dengan perhitungan sederhana[6].

Teks klasifikasi menggunakan model naïve bayes dan KNN pernah dibahas oleh [7][8][9] untuk membandingkan akurasi. Pada penelitian Devita[7] pada tahun 2018 yang membahas perbandingan kinerja naïve bayes dan KNN dalam klasifikasi artikel berbahasa indonesia. Hasil akhir menunjukkan naïve bayes memiliki performa yang baik dengan 70% akurasi, sedangkan KNN cukup rendah hanyak 40% akurasi. Penelitian Syarifuddin[8] pada tahun 2020 membahas tentang perbandingan metode naïve bayes dan k-nearest neighbor untuk mengetahui opini publik pengguna twitter terhadap kasus covid-19 di Indonesia. Hasil pada penelitian ini diketahui naïve bayes memiliki akurasi tertinggi dengan 63.21% akurasi. Sedangkan k-nearest neighbor memiliki akurasi sedikit lebih rendah dengan 58.10%. Opini positif juga mendapati akurasi naïve bayes lebih tinggi dengan 66.40%, dibanding KNN 58.40%. Penelitian lain juga menguji dari kedua kinerja model naïve bayes dan KNN oleh Indriana[9] pada tahun 2020 membahas perbandingan kedua model naïve bayes dan KNN pada klasifikasi data. Hasil dari penelitian ini diketahui KNN memiliki keunggulan dalam akurasi pada data sebesar 80%, sedangkan naïve bayes memiliki akurasi 73%.

Perbedaan ketiga tinjauan penelitian diatas adalah penelitian ini menggunakan metode resampling dalam menyeimbangkan dataset, sedangkan pada penelitian sebelumnya data label belum diseimbangkan. Metode resampling dengan teknik over-sampling diimplementasikan untuk menghindari dari terjadinya overfitting (data latih terlalu baik) pada data. Perbedaan juga terlihat pada tahap pra-pemrosesan teks, pada penelitian sebelumnya hanya dilakukan tahap dasar seperti casefolding, stopword, dan stemming, sedangkan penelitian ini ditambahkan proses normalisasi kata untuk membenarkan kata yang salah pada setiap cuitan sehingga proses pelabelan akan lebih mudah. Keluaran penelitian ini adalah untuk mengetahui performa model klasifikasi antara naïve bayes dan KNN dalam pengklasifikasian teks berupa sentimen jasa ekspedisi yang didapat pada media sosial twitter.

2. Metode Penelitian

Penelitian ini menggunakan data yang diambil dari twitter dengan bantuan pustaka bahasa pemrograman python berdasarkan penyebutan (mention) pengguna konsumen ke akun jasa ekspedisi. Cuitan yang diambil hanya cuitan berbahasa Indonesia dan hanya diambil satu cuitan per satu akun. Hal ini dilakukan untuk mengurangi banyaknya cuitan yang dianggap spam aatau tida dapat digunakan pada penelitian. Berikut adalah digram alur dari metode penelitian seperti pada Gambar 1.



Gambar 2.1. Diagram Metode Penelitian

Gambar 1 diawali dengan penggalian data dari web (scraping), data yang didapat akan diproses pada tahap

pra-pemrosesan (pre-processing text) teks untuk mendapatkan teks yang lebih terstruktur dan akan dilabeli setiap cuitan yang didapat berdasarkan polaritas dan subjektifitas. Kata dari seluruh dokumen akan diberi bobot (term weighting) menggunkan TF-IDF bertujuan untuk memudahkan dalam proses evaluasi model. Dataset yang sudah bersih akan dikelompokkan menjadi dua bagian yaitu data latih dan data uji. Model yang digunakan naïve bayes dan KNN sebagai perbandingan untuk mendapatkan akurasi terbaik dalam pengklasifikasian data teks sentimen.

2.1. Pengumpulan Data

Data yang akan diambil sebelumnya akan di obsevasi dengan memanfaatkan fitur pencarian lanjut pada twitter untuk mengetahui ketersediaan data yang diinginkan. Penggalian data akan dilakukan dengan mengambil data dari website menggunakan program secara otomatis atau biasa dikenal dengan scrapping. Alat bantu scrapping menggunakan twint yang berbasis python. Data diambil selama dua bulan sejak 1 Oktober hingga 1 Desember pada tahun 2020, hasil banyaknya data akan berbeda bergantung pada intensitas cuitan yang mention akun jasa ekspedisi. Pada penelitian ini data yang didapat sebanyak 46.220 JNT, 5.364 JNE, dan Pos 11,194. Atribut yang diambil meliputi tanggal, nama identitas, nama pengguna, dan cuitan (*tweet*).

2.2. Pra-pemrosesan Teks

Dataset yang belum tersturktur hasil dari scraping web twitter perlu dilakukan pra-pemrosesan untuk mengeliminasi redudansi data dan kata yang tidak diperlukan. Pra-pemrosesan teks bertujuan untuk mengeliminasi data duplikasi dan memperbaiki data yang tidak konsisten[10]. Beberapa tahapan yang perlu dilakukan pada pra-pemrosesan teks seperti case folding, pembersihan data, tokenisasi, normalisasi, stopword removal, dan stemming. Case folding adalah proses perubahan seluruh kata pada dokumen menjadi huruf kecil (lower case). Pembersihan data dilakukan untuk eliminasi tanda baca, simbol, angka, emoticon pada keseluruhan dokumen. Tokenisasi adalah proses pemisahan kata dalam setiap dokumen menjadi sebuah token. Normalisasi merupakan proses mengubah kata gaul atau kekinian, singkatan, atau kata yang tidak baku dirubah ke bentuk aslinya. Stopword removal berfungsi sebagai eliminasi kata yang tidak mempengaruhi hasil seperti kata imbuhan dan hubung (jika, sedangkan, yang, dan lain-lain.). Stemming adalah Proses dimana kalimat pada cuitan yang didapat akan diubah kata menjadi bentuk kata dasar.

2.3. Pelabelan Data

Pelabelan data membagi label sentimen pada tiap cuitan menjadi tiga bagian yaitu positif, netral, dan negatif. Dalam penelitian ini dataset dari tiap cuitan akan dilabeli secara manual dan bantuan pustaka. TextBlob dibangun diatas pustaka NLTK (Natural Language Toolkit) berbasis bahasa python. TextBlob merupakan alat yang dapat menentukan sebuah teks berbahasa Inggris dalam dokumen mempunyai sentimen negatif, positif, atau netral berdasarkan pada polaritas dan subjektifitas. Peneliti akan meninjau ulang hasil dari label pustaka TextBlob sebelum digunakan untuk penelitian.

2.4. Pembobotan Kata

Disarankan Term weighting atau pembobotan kata adalah proses perhitungan bobot tiap kata yang akan dicari pada keseluruhan dokumen untuk mengetahui ketersediaan dan kemiripan kata pada dokumen[11]. TF (*Term Frequency*) adalah banyaknya frekuensi kemunculan kata pada keseluruhan dokumen[6]. IDF (*Inverse Document Frequency*) adalah kebalikan dari TF (inverse) dimana kata yang jarang muncul dan memiliki bobot tinggi, dan dapat memiliki makna yang spesifik atau penting dalam keseluruhan dokumen[12]. Pembobotan kata adalah bagian dimana teks kalimat akan dirubah menjadi numerik atau setiap kalimat mempunyai nilai tergantung keunikannya sehingga perhitungan model klasifikasi akan mudah. TF-IDF sendiri adalah perkalian dari TF dan IDF, rumus TF-IDF dapat dilihat pada rumus 1.

$$TF - IDF = \frac{f_{t,d}}{\sum_{t^1 \in d} ft^1,_d} \times \log\left(\frac{N}{df(t)}\right) + 1\rrbracket \quad (1)$$

Rumus 1 digunakan untuk mendapatkan bobot dari suatu kata pada dokumen. Dengan $f_{t,d}$ adalah jumlah frekuensi kata t pada dokumen. $\sum_{t^1 \in d} ft^1, d$ merupakan jumlah keseluruhan kata dalam dokumen. N adalah jumlah keseluruhan dokumen. Sedangkan df(t) yaitu jumlah dokumen yang mengandung term pada t. TF-IDF sendiri adalah hasil perkalian dari TF (*Term Frequency*) dan IDF (*Inverse Dcoument Frequency*). *Laplace smoothing* diterapkan pada IDF untuk menghindari kata yang belum pernah muncul sebelumnya dalam dokumen dengan cara penambahan nilai satu.

2.5. Pelabelan Data

Pada tahap pembagian data akan dilakukan pemisaan dua bagian dokumen untuk 70% data latih dan 30% data uji. Data latih adalah data yang akan dipelajari oleh mesin atau model untuk menemukan suatu pola sebagai acuan dari data uji. Data uji adalah data yang akan diujikan atau di dipakai untuk mengetahui performa sebuah model yang sudah dilatih sebelumnya ketika menemukan data baru yang belum pernah dilihat sebelumnya. Dalam pembagian data dilakukan menggunakan split test dari pustaka sk-learn.

2.6. Evaluasi Model

Pada proses evaluasi model akan dibantu pustakan sk-learn yang biasa digunakan untuk pemodelan data, pra-pemrosesan data, dan lain-lain dalam mengatasi persoalan pada pembelajaran mesin. Pemodelan pengklasifikasian data hasil splitting akan di uji menggunakan pada setiap model. Naive bayes termasuk dalam model algoritma supervised learning yang berarti memiliki label sebagai acuan[13]. Model naïve bayes

yang digunakan pada penelitian ini adalah multinomial naïve bayes yang memanfaatkan nilai dari setiap token kata pada dokumen.

$$P(C|td_n) = P(X_1|C) \times P(X_n|C) \times P(C) \quad (2)$$

Pada rumus 2 digunakan untuk mendapatkan nilai token dari setiap kata menggunakan model naïve bayes. Dimana $P(C|td_n)$ merupakan probabilitas *kelas* pada kata dokumen ke-n. $P(X_1|C)$ adalah probabilitas kata pertama pada kelas *C*. $P(C)$ adalah probabilitas prior pada kelas *C*. Variabel $P(X_n|C)$ yaitu jumlah kata X pada kelas C.

Sedangkan K-Nearest Neighbor merupakan algoritma untuk pengklasifikasian suatu objek, berdasarkan k data latih yang memiliki tetangga atau jarak paling dekat dengan objek tersebut. Nilai k tidak boleh melebih besar dari data latih[14]. K adalah jumlah tetangga yang dekat, Algoritma KNN menggunakan aturan ketetanggaan dalam pengklasifikasian objek atau data sebagai nilai prediksi berdasarkan data uji. Untuk menghitung kemiripan kasus, Rumus 3 algoritma KNN didapat sebagai berikut.

$$Similarity(T,S) = \frac{(\sum_{i=1}^{N} f(Ti,Si)*wi)}{wi} \quad (3)$$

Rumus 3 adalah penerapan perhitungan dari model klasifikasi KNN. Pada variabel T adalah kasus baru yang akan dijadikan target. Variabel S yaitu kasus yang ada pada penyimpanan. Variabel n merupakan jumlah atribut dalam setiap kasus. i merupakan variabel atribut individu antara 1 sampai ke-n. Sedangkan variabel w adalah bobot yang diberi pada attribut ke-i.

Tabel *confusion matrix* digunakan untuk mendapatkan hasil akurasi model. Tabel *confusion* membandingkan antara hasil dari label asli dan label prediksi. Dengan bantuan sk-learn Tabel *confusion* matrix bisa didapatkan dengan memanggil fungsi *classification_report*. Berikut adalah gambar dari tabel *confusion matrix* 2x2 yang dapat dilihat pada gambar 2.

		Label Asli	
		Positif	Negatif
Label Prediksi	Positif	True Positif	False Positif
	Negatif	False Negatif	True Negatif

Gambar 2. *Confusion Matrix* 2x2

Gambarl 2 *confusion matrix* bertujuan untuk menjadi alat visualisasi menentukan performa dari suatu model. Alat tersebut juga dapat untuk menentukan suatu nilai dari presisi, *recall*, dan f1-*score*. Pada penlitian ini yang akan diuji dan dibandingkan adalah akurasi dari kedua model yaitu naïve bayes dan KNN. Dari hasil Gambar 2 dapat diketahui rumus untuk menentukan akurasi sebagai berikut.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

Rumus 4 digunakan untuk menghitung nilai akurasi sebuah model. TP adalah teks dari dokumen yang diprediksi bernar (*True Positif*). FP merupakan teks dari dokumen yang seharusnya diprediksi negatif (*False Negatif*). FN yaitu teks dari dokumen yang seharusnya diprediksi positif (*False Positive*). Variabel TN adalah teks dari dokumen yang diprediksi bernar (*True Negative*).

Didapatkannya nilai Akurasi bertujuan untuk mengukur keakuratan dari suatu model dalam pengklasifikasian yang didapatkan dari kesamaan label asli dengan label prediksi. Label data yang tidak berimbang (imbalance data) dapat mempengaruhi hasil akurasi pada model. Hal ini dapat menyebabkan *overfitting* pada data karena data latih dianggap terlalu baik. Teknik yang sering digunakan untuk menangani data label yang tidak seimbang adalah menggunakan teknik *resampling*. Resampling adalah bentuk menyamakan distribusi data label kelas minoritas[15]. Ada dua cara dalam melakukan teknik resampilng yaitu dengan *over-sampling* dan *under-sampling*. *Under-sampling* adalah pengurangan data dari kelas mayoritas menyesuaikan dari data kelas minoritas, sedangkan *over-sampling* adalah menduplikasi record secara aacak dari kelas minoritas agar terhindar dari *overfitting*. Pada penelitian ini metode yang akan diterapkan adalah over-sampling dimana data label keseluruhan akan disetarakan bergantung pada data yang memiliki label tertinggi.

3. Hasil dan Pembahasan

3.1. Pengumpulan Data dengan Teknik *Scaping*

Data yang diambil pada penelitian ini sepenuhnya bersumber dari website media sosial twitter. Setelah data atau cuitan yang diinginkan sudah ditemukan dari pencarian lanjut twitter, penerapan dilakukan ke twint sesuai yang ada pada dokumentasinya. Lama penggalian data dari web (*scraping*) ditentukan oleh kuantitas data yang diambil. Dalam penerapannya atribut yang diambil adalah akan ditargetkan pada ketiga akun jasa ekspedisi. Untuk menetapkan waktu awal pengambilan. Atribut yang diambil berupa tangal dibuatnya cuitan, username, nama akun, cuitan, jumlah suka, jumlah dibagikan, dan jumlah komentar. File hasil pengambilan menggunakan twint akan disimpan dengan format *comma separated values* (CSV). Hasil *scraping* dapat dilihat pada Tabel 1.
.

Tabel 1. Hasil *Scraping* Cuitan

Jasa Ekspedisi	Jumlah Cuitan
JNT Ekspress	46.220
JNE Indonesia	5.364
Pos Indonesia	11.194

Tabel 1 merupakan hasil cuitan dari masing-masing data cuitan yang didapat dari ketiga perusahaan jasa ekspedisi

selama dua bulan sejak 1 Oktober hingga 1 Desember. Diketahui ekspedisi JNT Ekspress memiliki cuitan sebanyak 46.220 dari pengguna twitter. Pada data Pos Indonesia mendapati hasil dengan cuitan sebanyak 11.194, JNE Indonesia memiliki hasil cuitan paling sedikit diantara dua kompetitornya dengan 5.364 cuitan.

3.2 Pra-Pemrosesan Teks

Tahap pra-pemrosesan teks adalah tahap dimana data yang diambil dari twitter akan dibersihkan sebelum dianalisa lebih lanjut. Data mentah yang didapat dari twitter akan melalui proses *case folding*, pembersihan data (*clean data*), tokenisasi, normalisasi, *stopword removal*, dan *stemming*. Kata yang dinormalisasikan perlu dibuat kamus manual, kamus tersebut didapatkan dari cuitan yang sering muncul pada dokumen teks. Adapun kamus untuk normalisasi data terdapat pada Tabel 2.

Tabel 2. Hasil Normalisasi Kata

Sebelum	Sesudah
deket	dekat
alesan	alasan
sampe	sampai
pkt	paket
okt	Oktober
jkt	Jakarta

Tabel 2 adalah beberapa kata yang dibuat dan dijadikan *corpus* secara manual berdasarkan cuitan yang didapat. Normalisasi perlu dilakukan untuk mempermudah proses stemming *stemming* (merubah kek kata dasar). Pada proses stemming algoritma yang digunakan adalah Nazief dan Andriani dimana kata yang memiliki imbuhan akan dihilangkan.

Tabel 3. Tahapan Pra-pemrosesan Teks

Sebelum	Input	Output
Case Folding	@jntexpressid keren bgt punya karyawan yang bertanggung jawab	@jntexpressid keren bgtt punya karyawan yang bertanggung jawab
Pembersihan Data		keren bgt punya karyawan yang bertanggung jawab
Tokenisasi		['keren',bgt','punya', 'karyawan','yang', 'bertanggung' ,'jawab']
Normalisasi		['keren','banget', 'punya' ,'karyawan','yang', 'bertanggung' ,'jawab']
Stopword Removal		['keren', 'banget', 'karyawan', 'bertanggung' ,'jawab']
Stemming		['keren', 'banget', 'karyawan','tanggung' ,'jawab']

Tabel 3 merupakan hasil pra-pemrosesan teks pada salah satu cuitan dataset JNT Ekspress. Keenam proses tersebut dilakukan untuk mendapatkan data yang bersih sehingga memudahkan proses pembobotan kata dan evaluasi model. Pada tahapan ini juga akan mengeliminasi redudansi data yang tidak dibutuhkan dalam penelitian. Setelah keseluruhan dataset melewati tahapan pra-pemrosesan teks jumlah cuitan akan mengalami perubahan yang dapat dilihat pada Tabel 4.

Tabel 4. Jumlah Cuitan setelah Tahap Pra-pemrosesan

Perusahaan	Jumlah Cuitan
JNT	5104
JNE	918
POS	747

Tabel 4 menampilkan hasil dari ketiga dataset setelah tahapan pra-pemrosesan teks dilakukan. Data JNT memiliki pengurangan data yang cukup signifikan menjadi 5104. Pada data JNE dan Pos Indonesia juga mengalami penurunan tetapi tidak terlalu signifikan. Penurunan data ini diakibatkan cuitan yang didapat mengalami banyak kesamaan sehingga perlu dielminasi. Hasil data setelah tahap pra-pemrosesan teks akan dilakukan pelabelan pada cuitan pengguna.

3.3 Pelabelan Data

Pada model klasifikasi data perlu memiliki label sebagai acuan atau guru. Pelabelan data sentimen dilakukan secara manual berdasarkan kamus bahasa Indonesia. Pustaka texblob dari python juga digunakan untuk pelabelan otomatis yang menhasilkan nilai subjektifitas dan polaritas. Texblob hanya dapat memproses kalimat berbahasa Inggris sehingga cuitan bahasa Indonesia perlu ditranslate terlebih dahulu. Hasil dalam penentuan label dapat dilihat pada Tabel 5.

Tabel 5. Hasil Pelabelan Data

Cuitan	Polaritas	Subjektifitas	Sentimen
@JNE_ID lama banget pengirimannya|	-0,2	0,4	Negatif
@PosIndonesia min, kirim dokumen ke jepang kondisi covid gini brp hari? RLN/ EMS	0	0,3	Netral
Cuy ini pas bgt pas nanya ke jnt eh semenit kemudia kurirnya datang ahahaha kayanya aku ada keterikatan dengan jnt @jntexpressid	0,14	0,27	Positif

Hasil pelabelan data Tabel 5 menjelaskan cuitan dataset JNT yang memiliki nilai subjektifitas dan polaritas. Nilai polaritas menentukan sebuah sentimen, dimana jika <0 bernilai negatif, jika 0 bernilai netral, dan >0 bernilai positif. Subjektifitas adalah nilai yang didasarkan atas pikiran peneliti.

Tabel 6. Hasil Dataset yang telah Dilabeli

Perusahaan	Positif	Netral	Negatif
JNT	2594	821	1689
JNE	190	255	473
POS	239	224	284

Tabel 6 diketahui hasil sentimen dari tiap akun jasa ekspedisi setelah dilabeli berdasarkan polaritas dan

subjektifias. JNT memiliki sentimen positif lebih banyak dibanding yang lain sedangkan sentimen negatif terbanyak ada di akun JNE yang melebihi sentimen positifnya. Hasil dataset yang telah dilabeli akan di beri bobot pada setiap cuitannya menggunakan TF-IDF.

3.4 Pembobotan Kata

Pembobotan kata (*term* weighting) dilakukan untuk memberikan bobot nilai pada setiap kata yang ada pada dokumen. TF akan memberikan nilai bobot tinggi pada kata yang sering muncul pada dokumen. Sebaliknya IDF memberikan nilai tinggi pada kata yang jarang muncul karena dianggap memiliki keunikan pada dokumen. Proses pembobotan kata dapat dihitung menggunakan TF x IDF seperti yang dijelaskan pada Rumus 1.

Tabel 7. Hasil TF-IDF

{'thank': 0.589, 'jnt': 0.294, 'expressku': 0.980, 'abang': 0.363, 'kurir': 0.214, 'daerah': 0.578, 'ganteng': 0.551, 'jntgiveawayiphone': 0.106}
{'transaksi': 0.613, 'marketplace': 0.599, 'shopee': 0.318, 'status': 0.488, 'batal': 0.543, 'tolong': 0.201, 'retur': 0.504, 'kirim': 0.128, 'estimasi': 0.430, 'ubah': 0.379}
{'sebel': 0.94, 'banget': 0.258, 'paket': 0.085, 'gue': 0.592, 'ajak': 0.877, 'muter': 1.230}
{'kacau': 1.106, 'paket': 0.175, 'udh': 0.595, 'stuck': 0.712, 'pekayon': 1.067}

Data teks yang diberi bobot adalah hasil *stemming* dimana teks akan dirubah menjadi numertik untuk mempermudah proses evaluasi model. Dalam mempermudah proses perhitungan bobot untuk setiap kata pada dataset dapat menggunakan bantuan pustaka python sk-learn. Modul yang dimanfaatkan adalah TfidfVectorizer.

3.5 Pembagian Data

Tahap *splitting* data dilakukan untuk memisahkan data latih sebanyak 70% dan data uji sebanyak 30%. Tujuan dari pemisahan data adalah untuk melatih model dari data dan memprediksi untuk mendapatkan performa dari suatu model. Mesin akan melatih berdasarkan data uji dan akan dicocokkan dengan data yang akan dites. Hasil pembagian dapat dilihat pada Tabel 8.

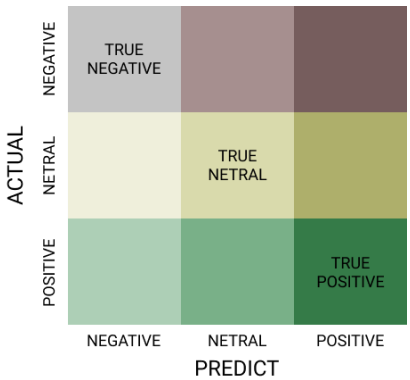
Tabel 8. Jumlah Cuitan setelah Tahap Pra-pemrosesan

Perusahaan	Data Latih	Data Uji
JNT	3569	1535
JNE	641	277
POS	522	225

Tabel 8 menampilkan hasil setelah tahap pembagian data pada ketiga dataset dilakukan, pada data JNT memiliki data latih dan data uji terbanyak sesuai hasil dari data setelah pra-pemrosesan dilakukan. Tahap pembagian data diperlukan karena dalam pemodelan menggunakan metode pembelajaran *supervised learning* dimana dibutuhkan data latih dan data uji untuk jadi bahan pembelajaran mesin. Hasil banyaknya

3.6 Evaluasi Model

Setelah mendapatkan data uji dan data latih model akan dievaluasi menggunakan algoritma klasifikasi naïve bayes dan KNN. Pada penelitian ini evaluasi model dilakukan dengan bantuan pustaka sk-learn. Naive bayes pada penelitian ini menggunakan variasi lain yang sering digunakan untuk klasifikasi teks yaitu multinomial naïve bayes. Baik naïve bayes maupun KNN untuk tahap pertama dilakukan pencocokan (*fitting*) terhadap data latih akan diuji dan dibuat menjadi dua variabel dimana X merupakan kolom hasil stemming yang sudah diberi bobot menggunakan TF-IDF dan variabel y adalah label sentimen. Pada model KNN sebelum masuk ketahap evaluasi model perlu menentukan kedekatan tetangga (n_neighbours) yang pada penelitian ini menggunakan perbandingan n = 2,4,6,8,10. Ketetanggaan yang digunakan pada penelitian ini diuji untuk mengetahui akurasi terbaik dari model KNN, pada penelitian sebelumnya hanya ketetanggaan tertentu yang diujikan. Setelah model diujikan akan dilakukan perhitungan akurasi untuk mengetahui performa pada tiap algoritma menggunakan tabel *confusion matrix*.



Gambar 3. *Confusion Matrix* 3x3

Gambar 3 merupakan Tabel confusion matrix yang memiliki lebih dari dua kelas (*multiclass classification*) untuk memudahkan dalam perhitungan performa dari suatu model algoritma klasifikasi. Untuk mendapatkan akurasi rumus dapat dilihat pada rumus 6. Proses dilakukan untuk mencari seberapa banyak prediksi yang benar dengan bantuan pembelajaran mesin. Data yang didapat ternyata tidak memiliki nilai yang sama pada label sehingga akan diuji menggunakan metode *over-sampling* dimana data kelas minoritas akan diduplikasi hingga setara dengan kelas mayoritas.

Tabel 9. Hasil Data setelah Menggunakan *Over-sampling*

Akun	Sebelum over-sampling			Sesudah over-sampling		
	Positif	Negatif	Netral	Positive	Negative	Netral
JNT	2594	1689	821	2594	2594	2594
JNE	473	255	190	473	473	473
POS	284	239	224	284	284	284

Dapat dilihat dari Tabel 9 menampilkan perbedaan sebelum dan sesudah *over-sampling* dilakukan. Pada

data sebelum *over-sampling* jumlah dari label tidak setara, sedangkan setelah metode diterapkan setiap label memiliki nilai yang setara. Tujuan diterapkannya metode ini untuk menghindari *overfitting* sehingga mesin akan kesusahan dalam memprediksi data uji baru. Penerapan teknik resampling juga akan mempengaruhi hasil pembagian data uji dan data latih sehingga perlu dilakukan lagi tahap pembagian data.

Tabel 10. Hasil Pembagian Data setelah *Over-sampling*

Perusahaan	Data Latih	Data Uji
JNT	5447	2335
JNE	993	426
POS	596	256

Hasil data pada Tabel 10 menggunakan parameter yang sama seperti sebelumnya dengan 70% data uji dan 30% data latih. Setelah data dibagi akan dilakukan evaluasi menggunakan pustaka sk-learn dengan memanggil fungsi MultinomialNB untuk naïve bayes, sedangkan dalam memanggil fungsi KNN dengan perintah KneighborsClassifier (n_neigbors = nilai ketetanggaan) n akan diisi dengan 2, 4, 6, 8, dan 10. Kedua model akan diuji untuk mengetahui performa akurasi terbaik dalam menyelesaikan kasus klasifikasi sentimen jasa eksepdisi.

Tabel 11. Hasil Akurasi pada Model Algoritma Naïve Bayes

	Akurasi naïve bayes	
Akun	Sebelum over-sampling	Sesudah over-sampling
JNT	78%	82%
JNE	61%	74%
POS	60%	73%

Tabel 11 menjelaskan adanya perbedaan hasil akurasi pada model algoritma naïve bayes sebelum dilakukannya *resampling* dan setelahnya. Terjadi kenaikan akurasi disetiap data akun jasa ekspedisi. Pada dataset JNT mendapatkan akurasi terbesar dengan 82% dibanding data lainnya JNE 74%, dan POS 73%. Sehingga dapat diketahui teknik *resampling* dengan metode *oversampiling* atau menyetarakan label data dengan label tertinggi dapat meningkatkan akurasi suatu model. Hasil pembanding model KNN dapat dilihat pada Tabel 12.

Tabel 12. Hasil Akurasi pada Model Algoritma KNN

	Akurasi K-Nearest Neighbor									
Akun	Sebelum over-sampling					Setelah over-sampling				
	K=2	K=4	K=6	K=8	K=10	K=2	K=4	K=6	K=8	K=10
JNT	39%	37%	44%	51%	54%	60%	51%	44%	40%	37%
JNE	54%	56%	57%	61%	62%	61%	62%	62%	60%	60%
POS	26%	44%	43%	45%	53%	71%	59%	59%	60%	58%

Tabel 12 merupakan hasil akurasi dari pengujian model KNN pada setiap kedekatan atau K dimana memiliki nilai yang menghasilkan akurasi yang berbeda. Berdasarkan ketetanggaanya KNN saat K dimasukkan nilai tinggi belum tentu hasil akurasi tinggi. Akurasi tertinggi didapat pada model KNN pada K=2 setelah metode *over-sampling* dilakukan, sedangkan hasil terendah didapat saat nilai K=10. Untuk hasil sebelum *over-sampling* hasil K tertinggi didapat saat K=10, sedangkan terendah ada pada K=2.

Tabel 13. Perbandingan Akurasi KNN dan Naive Bayes

	Setelah *over-sampling*	
	Naïve bayes	KNN (K=2)
JNT	82%	60%
JNE	74%	61%
POS	73%	71%

Tabel 13 merupakan hasil perbandingan antara kedua model algoritma klasifikasi KNN dan naïve bayes setelah tahap evaluasi diselesaikan. Hasil pada Tabel tersebut diambil dari rata-rata akurasi tertinggi yang didapat pada Tabel 12 dan Tabel 13. Dapat diketahui pada model naïve bayes memiliki akurasi tertinggi pada data JNT sebesar 82%, sedangkan akurasi tertinggi KNN ada pada POS 71%. Dapat disimpulkan bahwa model naïve bayes dalam pengklasifikasian teks sentimen jasa ekspedisi yang diambil dari cuitan twitter memiliki akurasi lebih baik dibanding model K-Nearest Neigbor. Baik sebelum teknik *resampling* dilakukan dan hasil setelah diimplementasikan naïve bayes tetap unggul.

4. Kesimpulan

Dari hasil cuitan yang didapat berdasarkan mention pengguna, JNT memiliki jumlah pengguna aktif terbanyak dan memiliki opini lebih banyak positif dari pada negatifnya dibanding akun JNE, dan Pos Indonesia. Sedangkan dalam penerapan algoritma naïve bayes dan K-NN pada pengklasifikasian sentimen berdasarkan cuitan ulasan pengguna twitter yang mention ke akun jasa ekspedisi @jntexpressid, @JNE_ID, dan @posindonesia menghasilkan akurasi yang berbeda dari tiap model algoritma yang diuji. TF-IDF memberikan peran dimana setiap kata pada dataset diberi nilai untuk memudahkan dalam perhitungan sesuai rumus model naïve bayes dan K-NN. Pada penelitian ini digunakan teknik *resampling* untuk dataset yang tidak seimbang, teknik ini sukses menaikan akurasi baik naïve bayes maupun K-NN dibanding sebelum menggunakan metode over-sampling. Diketahui akurasi pada algoritma naïve bayes sedikit lebih baik dalam pengklasifikasian terhadap dataset berupa teks.sentimen jasa ekspedisi. Sedangkan pada model K-NN teknik resampling ini menghasilkan akurasi yang berbeda disetiap ketetanggaanya dimana K dengan value tinggi tidak menentukan akurasi yang tinggi. Untuk penelitian selanjutnya, diharapkan dapat menggunakan algoritma klasifikasi lainnya untuk mengetahui performa dari suatu model khususnya pada klasifikasi dataset berupa teks, dan untuk pelabelan sendiri opsi lain dapat menggunakan pendekatan *lexicon based* atau membuat

corpus terlebih dahulu yang nantinya digunakan untuk membantu penentuan sentimen sebagai alternatif apakah sebuah kalimat mengandung sebuah opini positif, negatif, dan netral.

Daftar Pustaka

[1] G. A. Buntoro, "Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter," *INTEGER J. Inf. Technol.*, vol. 1, no. 1, pp. 32–41, 2017.

[2] B. Gunawan, H. S. Pratiwi, and E. E. Pratama, "Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes," *J. Edukasi dan Penelit. Inform.*, vol. 4, no. 2, p. 113, 2018, doi: 10.26418/jp.v4i2.27526.

[3] A. Nugroho, "A Decision Guidance for Solving Success Rate Political Campaign Using Distance Weighted kNN in Nassi-Shneiderman Framework," vol. 14, no. 2, pp. 410–420, 2021, doi: 10.22266/ijies2021.0430.37.

[4] W. E. Nurjanah, R. S. Perdana, and M. A. Fauzi, "Analisis Sentimen Terhadap Tayangan Televisi Berdasarkan Opini Masyarakat pada Media Sosial Twitter menggunakan Metode K-Nearest Neighbor dan Pembobotan Jumlah Retweet," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 1, no. 12, pp. 1750–1757, 2017.

[5] R. Hidayatillah, M. Mirwan, M. Hakam, and A. Nugroho, "Levels of Political Participation Based on Naive Bayes Classifier," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 1, p. 73, 2019, doi: 10.22146/ijccs.42531.

[6] J. A. Septian, T. M. Fahrudin, and A. Nugroho, "Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF - IDF dan K - Nearest Neighbor," *INSYST (JOURNAL Intell. Syst. Comput.*, vol. 1, pp. 43–49, 2019.

[7] R. N. Devita, H. W. Herwanto, and A. P. Wibawa, "Perbandingan Kinerja Metode Naive Bayes dan K-Nearest Neighbor untuk Klasifikasi Artikel Berbahasa indonesia," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 4, p. 427, 2018, doi: 10.25126/jtiik.201854773.

[8] M. Syarifuddin, "Analisis Sentimen Opini Publik Mengenai Covid-19 Pada Twitter Menggunakan Metode Naïve Bayes Dan Knn," *Inti Nusa Mandiri*, vol. 15, no. 1, pp. 23–28, 2020.

[9] A. Indriani, "Analisa Perbandingan Metode Naïve Bayes Classifier Dan K-Nearest Neighbor Terhadap Klasifikasi Data," *Sebatik*, vol. 24, no. 1, pp. 1–7, 2020, doi: 10.46984/sebatik.v24i1.909.

[10] R. P. Sidiq, B. A. Dermawan, and Y. Umaidah, "Sentimen Analisis Komentar Toxic pada Grup Facebook Game Online Menggunakan Klasifikasi Naïve Bayes," *J. Inform. Univ. Pamulang*, vol. 5, no. 3, p. 356, 2020, doi: 10.32493/informatika.v5i3.6571.

[11] A. T. Ni'mah and A. Z. Arifin, "Perbandingan Metode Term Weighting terhadap Hasil Klasifikasi Teks pada Dataset Terjemahan Kitab Hadis," *Rekayasa*, vol. 13, no. 2, pp. 172–180, 2020, doi: 10.21107/rekayasa.v13i2.6412.

[12] S. Qaiser and R. Ali, "Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents," *Int. J. Comput. Appl.*, vol. 181, no. 1, pp. 25–29, 2018, doi: 10.5120/ijca2018917395.

[13] D. Syamsudin, Y. C. D. Halundaka, and A. Nugroho, "Prediksi Status Konsumen Produk Celana Menggunakan Naïve Bayes," *JOINTECS (Journal Inf. Technol. Comput. Sci.*, vol. 5, no. 3, p. 177, 2020, doi: 10.31328/jointecs.v5i3.1435.

[14] A. M. B. Muhammad Rivki, "Implementasi Algoritma K-Nearest Neighbor dalam Pengklasifikasian Follower Twitter yang Menggunakan Bahasa Indonesia," *J. Inf. Syst.*, vol. 13, 2017.

[15] A. Muqiit WS and R. Nooraeni, "Penerapan Metode Resampling Dalam Mengatasi Imbalanced Data Pada Determinan Kasus Diare Pada Balita Di Indonesia (Analisis Data Sdki 2017)," *J. MSA (Mat. dan Stat. serta Apl.)*, vol. 8, no. 1, p. 19, 2020, doi: 10.24252/msa.v8i1.13452.