

PENERAPAN PEMBOBOTAN *TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY* DAN ALGORITMA *K-NEAREST NEIGHBOR* UNTUK ANALISIS ULASAN HOTEL DI SITUS TRIPADVISOR

Khairul Huda¹, Sry Dhina Pohan², Youfih Herlina³

^{1,3} Universitas Karya Husada Semarang; Jl. Kompol R. Soekanto No.46, Kec. Tembalang, Kota Semarang, Jawa Tengah; telp/Fax (024)6724581

² Universitas Nahdlatul Ulama Indonesia; Jl. Taman Amir Hamzah No.5, Kec. Menteng, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta; telp/Fax 081258881926

Received: 9 Juli 2024

Accepted: 31 Juli 2024

Published: 7 Agustus 2024

Keywords:

text mining; term frequency-inverse document frequency; k-nearest neighbor; random over-sampling; imbalanced

Correspondent Email:

khairulhuda@unkaha.ac.id

Abstrak. Penelitian ini latarbelakangi oleh masalah evaluasi produk dan layanan menggunakan metode tradisional seperti survei, kuisioner dan wawancara yang sering menghasilkan analisis yang tidak konsisten dan tidak akurat. Salah satu pendekatan untuk mengatasi masalah tersebut adalah dengan menerapkan Teknik pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF) dan algoritma *K-Nearest Neighbor* untuk menganalisis ulasan pelanggan hotel dari situs TripAdvisor, yang dikategorikan menjadi 3 kelas sentimen yaitu netral, negatif dan positif menggunakan *text mining*. Algoritma *K-Nearest Neighbor* dipilih karena kemampuannya dalam komputasi yang efisien, mudah beradaptasi dengan berbagai data yang besar, serta relative rendah untuk kompleksitas algoritmanya. Hasil penelitian menunjukkan bahwa sistem ini mampu mengklasifikasikan ulasan hotel dengan tingkat akurasi yang optimal, mencapai 76% untuk data pelatihan dengan $K=31$, dan meningkatkan akurasi hingga 84% setelah melalui penerapan teknik *random over-sampling* untuk mengatasi *imbalanced* dataset.

Abstract. This research is motivated by issues in evaluating products and services using traditional methods such as surveys, questionnaires, and interviews, which often result in inconsistent and inaccurate analyses. One approach to address these issues is by applying the Term Frequency-Inverse Document Frequency (TF-IDF) weighting technique and the K-Nearest Neighbor algorithm to analyze hotel customer reviews from TripAdvisor, categorized into three sentiment classes: neutral, negative, and positive using text mining. The K-Nearest Neighbor algorithm was chosen for its efficient computation, adaptability to large datasets, and relatively low algorithmic complexity. The research findings indicate that this system is capable of classifying hotel reviews with optimal accuracy, achieving 76% accuracy on the training data with $K=31$, and improving accuracy to 84% after applying random over-sampling techniques to address the imbalanced dataset.

1. PENDAHULUAN

Revolusi industri generasi keempat telah menyebabkan perubahan besar-besaran di berbagai sektor, termasuk industri perhotelan. Sektor ini sekarang menghadapi tekanan untuk beradaptasi dan bertahan dalam persaingan yang semakin ketat. Salah satu kunci sukses

dalam menghadapi tantangan ini adalah dengan memperhatikan kebutuhan pelanggan melalui evaluasi berkelanjutan terhadap produk dan layanan mereka. Saat ini, metode evaluasi seperti wawancara, survei, dan kuisioner masih dominan dalam prakteknya. Namun, penggunaan metode ini dapat mengakibatkan

hasil analisis yang kurang akurat dan konsisten.[1].

Penggunaan teknologi *web 2.0*, seperti yang terlihat pada situs seperti TripAdvisor, telah mempengaruhi cara evaluasi dalam industri perhotelan. Situs ini memungkinkan pengguna untuk berbagi pengalaman mereka melalui ulasan *online*, menggunakan skala peringkat dari 1 sampai 5 sebagai indikator kualitas. Meskipun demikian, skala peringkat ini tidak selalu dapat diandalkan untuk menilai secara akurat isi dari ulasan tersebut. Untuk mengatasi masalah ini, diperlukan penggunaan metode komputasi yang mampu menganalisis data ulasan dengan efisien melalui teknik *text mining*. Metode ini memungkinkan untuk mengekstrak informasi yang lebih mendalam dari teks ulasan, memperbaiki akurasi evaluasi, dan meningkatkan konsistensi dalam menilai kualitas layanan dan produk yang direview[2]. *Text mining* adalah bagian dari *data mining* yang khusus mengeksplorasi dan menganalisis data dalam bentuk teks. Tujuannya adalah untuk mengenali gagasan, pola, topik, kata kunci, dan karakteristik lain yang terdapat dalam teks tersebut. Dengan menggunakan teknik *text mining*, informasi yang berguna dapat diekstrak dari data teks, seperti melakukan analisis sentimen terhadap ulasan pelanggan di media *online*. Hal ini memungkinkan perusahaan untuk memahami pandangan, opini, atau perasaan pelanggan terhadap produk atau layanan mereka dengan lebih mendalam dan akurat[3].

Analisis sentimen, juga dikenal sebagai penambangan opini, adalah metode untuk menganalisis, dan mengekstrak informasi dari teks yang berisi opini, preferensi, dan sentimen pengguna. Tujuannya adalah untuk mendapatkan wawasan dan pendapat dari pengguna. Analisis sentimen bertujuan untuk mengkategorikan teks sebagai positif, negatif, atau netral berdasarkan pola teks yang mendasar dari suatu kumpulan data. Sentimen ini sangat berguna bagi bisnis yang untuk memahami pandangan umum konsumen terhadap produk atau layanan mereka[4].

Beberapa algoritma klasifikasi dapat digunakan untuk analisis sentimen, yaitu K-Nearest Neighbor (KNN), Support Vector Machine (SVM) maupun Artificial Neural Network (ANN). Salah satu kelemahan dari algoritma ANN adalah ketergantungan pada

data besar untuk *training*. Sementara itu, kelebihan dari algoritma SVM adalah dapat diterapkan pada data berdimensi tinggi, akan tetapi kelemahannya adalah tidak mudah diterapkan untuk dataset yang besar[5]. KNN adalah algoritma yang menghitung jarak terdekat antara atribut data. Algoritma ini memiliki beberapa keunggulan yaitu dalam hal kinerja komputasi tinggi, sederhana, mudah adaptasi terhadap berbagai jenis data yang besar, serta memiliki akurasi yang baik dibandingkan dengan algoritma yang lain, dan kompleksitas algoritma yang lumayan kecil[6].

Penelitian sebelumnya telah menggunakan metode KNN antara lain untuk klasifikasi teks dalam bahasa Inggris, klasifikasi ulasan *user* hotel, klasifikasi sentimen komentar di media sosial Twitter dengan menggunakan bahasa Inggris dan klasifikasi teks berita di situs BBC News dengan menerapkan pembobotan TF-IDF [7], [8], [9], [10]. Namun, sebuah permasalahan umum yang timbul dalam proses klasifikasi adalah ketidakseimbangan dataset, di mana satu kelas memiliki jumlah data yang signifikan lebih banyak dibanding kelas lainnya. Keadaan ini dapat mengakibatkan hasil klasifikasi cenderung menguntungkan kelas dengan jumlah data yang lebih besar, sehingga mengurangi akurasi klasifikasi secara keseluruhan[11].

Tahapan awal untuk melakukan klasifikasi data berbentuk teks adalah pembobotan kata (*term weighting*), dengan tujuan untuk konversi data yang berbentuk teks menjadi data angka. Salah satu teknik pembobotan adalah *Term Frequency-Inverse Document Frequency* (TF-IDF). Metode TF-IDF memiliki beberapa keunggulan yaitu efektif dalam menghasilkan pembobotan yang baik dan mempunyai waktu pemrosesan yang cepat[12].

Berdasarkan pada masalah yang dihadapi dan kelebihan metode yang terbukti dalam studi sebelumnya, penelitian ini mengusulkan menerapkan metode TF-IDF untuk pembobotan dan algoritma KNN untuk menganalisis sentimen dari ulasan media *online*. Untuk mengatasi ketidakseimbangan dalam dataset, dalam penelitian ini digunakan *random oversampling* untuk menyeimbangkan kembali data, dan diharapkan kinerja dari algoritma klasifikasi dapat dioptimalkan, sehingga hasil analisis sentimen ulasan menjadi lebih efektif dan efisien.

2. TINJAUAN PUSTAKA

Beberapa studi telah dilakukan oleh peneliti sebelumnya mengenai metode text mining. Penelitian oleh Hilal dkk (2016) telah mengaplikasikan metode tersebut untuk mengklasifikasi teks dalam bahasa Romawi-Urdu dan bahasa Inggris. Digunakan sampel data sampel data 300 yang diambil dari *blog*, kemudian menganalisisnya menggunakan beberapa algoritma klasifikasi antara lain KNN, *Decision Tree* dan juga *Naive Bayes*. Hasil penelitian menunjukkan bahwa penggunaan algoritma KNN dalam klasifikasi teks mencapai tingkat akurasi sebesar 95% [13].

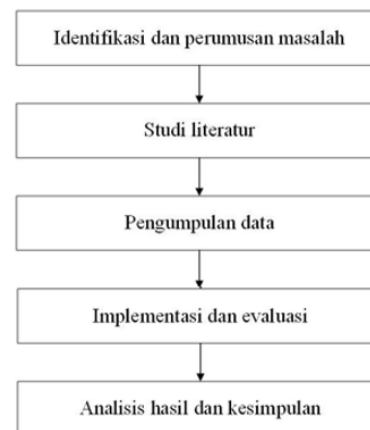
Studi sebelumnya telah melaksanakan penelitian menggunakan metode *text mining* untuk menganalisis ulasan pelanggan hotel. Khorsand dkk (2020) menggunakan data ulasan dari 64 hotel di kota Teheran Iran yang diambil dari situs TripAdvisor. Penelitian ini bertujuan untuk melakukan prediksi skor ulasan pelanggan baru berdasarkan fasilitas hotel dan juga informasi profil. Penelitian ini membandingkan delapan algoritma klasifikasi: *Neural network*, KNN, *naive Bayes*, *logistic regression*, *decision tree*, SVM, *random forest*, dan juga *gradient boosting*. Hasil penelitian menunjukkan bahwa KNN adalah algoritma terbaik di antara beberapa metode tersebut untuk mengolah data ulasan hotel di kota Teheran [14].

Shah dkk (2020) melakukan penelitian menggunakan *text mining* untuk mengklasifikasikan berita dari laman BBC News. Mereka menggunakan pembobotan TF-IDF dan membandingkan tiga algoritma klasifikasi: *Logistic regression*, KNN dan *random forest*. Penelitian menggunakan data 2200 teks dengan bahasa Inggris yang dibagi ke dalam beberapa kategori antara lain politik, olahraga bisnis, hiburan dan juga teknologi. Hasil studi menunjukkan algoritma KNN adalah metode yang paling efektif karena menghasilkan tingkat akurasi hingga 92% [15].

3. METODE PENELITIAN

2.1 Tahapan penelitian

Tahapan-tahapan penelitian yang akan dilakukan ditunjukkan pada gambar 1.



Gambar 1. Tahapan Penelitian

Gambar 1 menunjukkan tahapan penelitian dan implementasi algoritma untuk menganalisis ulasan pengguna hotel yang dijabarkan pada penjelasan berikut ini:

1. Tahap Identifikasi dan Perumusan Masalah
Identifikasi masalah dilakukan untuk memahami tujuan yang ingin dicapai dilakukannya penelitian tentang analisis sentimen dengan menggunakan metode TF-IDF dan algoritma KNN.
2. Tahap Studi Literatur
Studi literatur dilakukan penggalan ilmu berdasarkan perkembangan ilmu pengetahuan pada publikasi-publikasi jurnal internasional dan nasional yang relevan dengan penelitian sebagai referensi utama dan pendukung.
3. Tahap Pengumpulan Data
Pengumpulan data yang dibutuhkan dalam penelitian ini meliputi pengambilan sumber ulasan pengguna pada laman TripAdvisor yang digunakan untuk data masukan sistem yang akan dibangun. Penelitian ini menggunakan data teks ulasan pengguna hotel pada laman TripAdvisor dalam Bahasa Inggris. Data yang digunakan berjumlah 1314 terdiri dari beberapa kolom yaitu: *Review*, merupakan isi ulasan yang diberikan oleh pengguna; *rating*, merupakan skor yang diberikan pengguna dalam skala 1 sampai 5; *year*, merupakan tahun atau periode penulisan ulasan. Untuk membatasi ruang lingkup analisis dalam penelitian ini, data yang digunakan untuk sampel dalam penelitian ini adalah sejumlah 1313 ulasan. Untuk kolom *review* merupakan

sumber informasi yang akan dianalisis dan digunakan dalam tahap selanjutnya yaitu pra pengolahan teks. Tabel 1 menunjukkan sampel data ulasan yang digunakan dalam penelitian ini.

Tabel 1. Sampel data yang digunakan

Dn	Review	Rating	Year
D1	Terrible hotel, bad service outdated room, mediocre food and overpriced,	2	2021
...
D1313	Clean convenient, friendly, perfect location and excellent price,	4	2017

4. Tahap Implementasi dan Evaluasi

Tahap implementasi dilakukan beberapa dasar kegiatan yang meliputi: Prapengolahan data, pelabelan data, implementasi algoritma TF-IDF dan KNN, serta evaluasi menggunakan *confusion matrix*.

5. Tahap Prapengolahan Data

Data masukan di dalam penelitian ini adalah data teks yang tidak terstruktur, sehingga dibutuhkan proses prapengolahan dengan tujuan membuat data teks sebagai masukan menjadi lebih terstruktur. Tahap prapengolahan teks terdiri dari tokenisasi, *stopword* dan *stemming*. Pustaka NLTK dan *spacy* dalam pemrograman *python* digunakan untuk proses prapengolahan teks tersebut secara otomatis. Tabel 2 menunjukkan hasil prapengolahan teks dalam penelitian ini.

Tabel 2. Prapengolahan data

Dn	Ulasan	Hasil Prapengolahan
D1	Terrible and bad service, mediocre food and overpriced.	terrible bad service mediocre food overprice
...
D1313	Really bad, terrible food cold and dirty foodsteps sheets,	bad terrible food cold dirty foodstep sheet

6. Tahap Pelabelan Data

Setelah melalui tahap prapengolahan, data masukan perlu diberikan pelabelan sentimen yang terkandung didalamnya. Pemberian label sentimen akan digunakan untuk pemrosesan algoritma KNN. Pemberian label kelas ini berfungsi untuk memberikan parameter pelatihan sistem dan untuk perhitungan dalam rangka mencari nilai akurasi proses algoritma tersebut. Proses pelabelan sentimen dilakukan secara otomatis dengan menggunakan pustaka *vader sentiment* dalam pemrograman *python*. Pada penelitian ini label sentimen untuk pemrosesan algoritma KNN dibagi menjadi 3 label kelas yaitu positif, negatif dan netral sesuai dengan informasi yang terkandung pada data ulasan. Tabel 3 menunjukkan hasil pelabelan otomatis data masukan.

Tabel 3. Pelabelan data

Dn	Ulasan	Hasil Prapengolahan	Sentimen
D1	Terrible and bad service, mediocre food and overpriced.	terrible bad service mediocre food overprice	0
...
D1313	Really bad, terrible food cold and dirty foodsteps sheets,	bad terrible food cold dirty foodstep sheet	0

7. Implementasi TF-IDF

TF-IDF adalah teknik untuk mengubah data teks menjadi representasi numerik, memungkinkan untuk pengolahan data menggunakan algoritma klasifikasi. Metode ini digunakan untuk menghitung seberapa penting suatu kata dalam sebuah dokumen dengan mempertimbangkan frekuensi kata tersebut dalam dokumen tersebut serta dalam seluruh koleksi dokumen. Perhitungan tersebut digunakan untuk menentukan seberapa penting suatu kata dalam dokumen tersebut. TF-IDF merupakan metode yang sering digunakan untuk menganalisis data teks dengan jumlah besar. Algoritma ini memberikan bobot pada kata kunci dalam setiap kategori untuk menilai seberapa mirip kata kunci dengan kategori yang

ada. Langkah-langkah algoritma TF-IDF adalah sebagai berikut:[16] .

1. TF (Term Frequency)

Term frequency (TF) menghitung seberapa sering suatu term muncul selanjutnya hasilnya dilogartmikan untuk mengurangi skala nilai. Dengan menggunakan logaritma, nilai yang besar akan diperkecil, karena logaritma mengurangi jumlah digit. Persamaan untuk menghitung TF adalah sebagai berikut:

$$TF_{t,d} = 1 + \log (TF_{t,d}) \quad (2.1)$$

dengan $TF_{t,d}$ adalah banyaknya kata t pada dokumen d , sebagai contoh *term* di sebuah dokumen adalah 6 kali, dapat dihitung bobot = $1 + \log (6) = 1.778$. Kemudian apabila *term* tidak dijumpai dalam suatu dokumen maka nilai bobotnya adalah 0. Untuk menghitung nilai TF juga bisa digunakan rumus TF biner, di mana perhatian diberikan pada keberadaan kata atau term dalam dokumen. Apabila suatu *term* terdapat dalam dokumen maka nilainya akan satu; jika tidak ada, nilainya akan nol.

2. IDF (Inverse Document Frequency)

IDF berperan dalam *reduce* bobot suatu term jika term tersebut sering muncul di semua data teks yang digunakan. Nilai IDF dihitung menggunakan persamaan:

$$IDF = \log \left(\frac{d}{df} \right) \quad (2.2)$$

dengan d adalah jumlah semua dokumen teks yang digunakan, kemudian df adalah jumlah dokumen yang mengandung suatu *term*. Berdasarkan rumus TF dan IDF tersebut, selanjutnya dapat ditentukan nilai bobot akhir sebuah *term* menggunakan rumus:

$$W_{t,d} = tf_{t,d} * idf_t \quad (2.3)$$

dengan $W_{t,d}$ adalah nilai bobot akhir sebuah term, dengan $tf_{t,d}$ adalah banyaknya *term* dalam suatu dokumen, dengan idf_t adalah pembobotan secara keseluruhan.

8. Implementasi KNN

Algoritma *k-nearest neighbor* (KNN) adalah metode untuk mengklasifikasikan data atau dokumen berdasarkan pada data terklasifikasi

sebelumnya. KNN merupakan algoritma supervised learning di mana instance baru diklasifikasikan berdasarkan mayoritas jarak terdekat dengan kategori yang ada dalam KNN. Kelas yang paling sering muncul di antara tetangga terdekat akan menjadi kelas hasil dari klasifikasi tersebut. Algoritma KNN bekerja berdasarkan konsep *learning by analogy*. Data pembelajaran direpresentasikan oleh atribut numerik dalam ruang n-dimensi. Setiap data pembelajaran merupakan titik, ditandai sebagai c , dalam ruang n-dimensi. Ketika sebuah data *query* dengan label yang tidak diketahui dimasukkan, KNN akan mencari k data pembelajaran yang memiliki jarak paling dekat dengan data *query* dalam ruang n-dimensi. Jarak antara data *query* dan data pembelajaran dihitung dengan cara mengukur jarak antara titik yang mewakili data *query* dengan semua titik yang mewakili data pembelajaran menggunakan persamaan *Euclidean Distance* [17]:

$$D_{xy} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.4)$$

dengan D_{xy} adalah jarak antara 2 buah objek yaitu x dan y , dengan x adalah data uji yang akan dilakukan prediksi, dengan y adalah data latih, dengan n adalah banyaknya data latih yang digunakan, dengan i adalah data ke- i . Untuk menghitung tingkat kemiripan antara dua objek x dan y pada proses klasifikasi data berbentuk teks digunakan metode pengukuran jarak *cosine similarity* dengan persamaan[18]:

$$Cos_{ij} = \frac{\sum_k (d_{ik} d_{jk})}{\sqrt{\sum_k d_{ik}^2} \sqrt{\sum_k d_{jk}^2}} \quad (2.5)$$

dengan Cos_{ij} adalah nilai kemiripan data latih dengan data uji, dengan d_{ik} adalah panjang vektor data uji, dengan d_{jk} adalah panjang vektor data latih.

Langkah-langkah penerapan algoritma klasifikasi KNN adalah sebagai berikut:

1. Menentukan nilai K (jumlah tetangga terdekat yang akan dipertimbangkan).
2. Menghitung jarak antara setiap data latih dengan data uji yang digunakan melalui rumus 2.4 atau 2.5.
3. Setelah mendapatkan jarak antara data latih dan data uji, kemudian data diurutkan dari jarak terkecil.

4. Kategori mayoritas dari tetangga terdekat akan menjadi hasil klasifikasi.

Untuk menentukan nilai optimal dari parameter k dalam algoritma klasifikasi KNN, digunakan metode optimasi *k-fold cross-validation*. Metode ini membagi data menjadi k bagian, di mana setiap bagian secara bergantian digunakan sebagai data uji sementara bagian lainnya sebagai data latih. *K-fold cross-validation* membantu mengevaluasi kinerja model dengan memastikan bahwa semua data memiliki kesempatan menjadi data latih dan uji. Jumlah k yang umum digunakan untuk mencapai akurasi optimal adalah 10-fold *cross-validation*. Data dibagi menjadi 10 bagian. Pada tiap iterasi, salah satu dari 10 bagian digunakan sebagai data uji, sedangkan 9 bagian lainnya digunakan sebagai data latih. Proses ini diulang sebanyak sepuluh kali.

9. Pengukuran Kinerja Klasifikasi

Dalam pengukuran kinerja menggunakan confusion matrix, istilah yang mewakili hasil dari proses klasifikasi adalah *True Positive* (TP), yang menunjukkan data yang berhasil diprediksi secara tepat[19]. Dari nilai ini, dapat dihitung nilai akurasi, yang mengindikasikan seberapa baik sistem dapat mengklasifikasikan data dengan membandingkan jumlah data yang diklasifikasikan dengan benar dengan total keseluruhan data. Untuk mendapatkan nilai akurasi, digunakan rumus sebagai berikut:

$$Accuracy = \frac{TP}{\text{Jumlah data uji}} \quad (2.6)$$

10. Tahap Analisis Hasil dan Kesimpulan

Tahap analisis hasil dilakukan untuk menganalisis hasil keluaran berbentuk akurasi sistem prediksi dan pengukuran kinerja dari metode TF-IDF dan KNN. Langkah terakhir dari penelitian ini adalah pengambilan beberapa kesimpulan dari pengaruh penerapan sistem yang menghasilkan beberapa informasi bagi perusahaan.

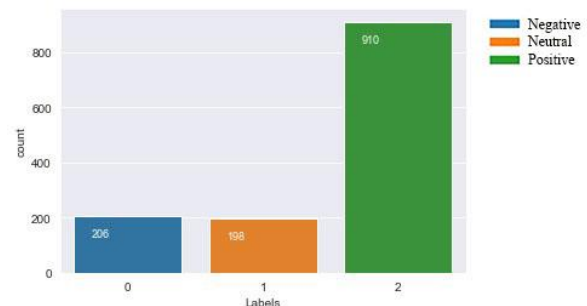
4. HASIL DAN PEMBAHASAN

Pada bagian ini menyajikan pembahasan terkait dengan evaluasi sistem dan faktor memungkinkan dapat mempengaruhi hasil akhir dari penelitian. Penjelasan terkait masing-

masing pembahasan tersebut adalah sebagai berikut:

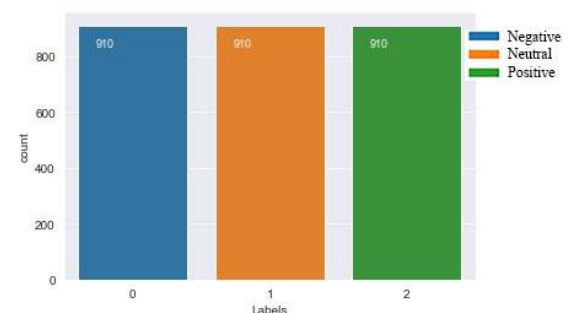
4.1 Analisis Data

Data masukan yang digunakan sebagai data *training* untuk pelatihan sistem sejumlah 1313 dengan kelas sentimen yang terdiri dari kelas positif, negatif dan netral disajikan dalam gambar 2.



Gambar 2. Statistik data masukan

Berdasarkan gambar 2 terlihat bahwa jumlah kelas untuk sentimen positif memiliki jumlah yang tinggi yaitu 910 dibandingkan dengan jumlah kelas netral yaitu 198 dan jumlah kelas negatif yaitu 206 sehingga dapat dikatakan bahwa kelas positif menjadi kelas mayoritas dan membuat data masukan menjadi tidak seimbang. Untuk mengatasi jumlah kelas yang tidak seimbang tersebut digunakan metode *random over-sampling* untuk *rebalance* data disajikan dalam gambar 4.6.



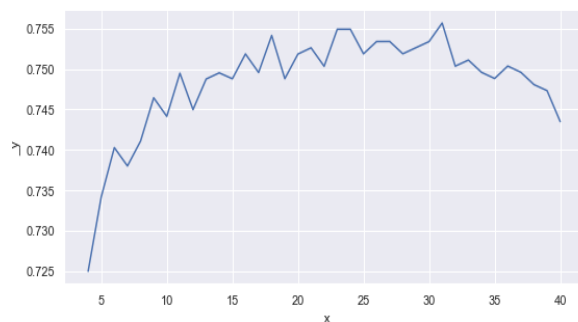
Gambar 3. Statistik hasil *rebalance* data

Berdasarkan gambar 3 terlihat bahwa jumlah kelas setelah proses *rebalance* data menggunakan metode *random over-sampling* untuk masing-masing kelas sudah seimbang.

3.2 Evaluasi Kinerja Algoritma Klasifikasi

Evaluasi *cross validation* digunakan untuk mengetahui kinerja dari algoritma KNN,

skenario percobaan pertama yaitu masukan data *training* yang berjumlah 1313 dan tanpa *rebalance* data terhadap kelas sentimen. Pengujian menggunakan *10-fold cross validation* dan menggunakan nilai parameter 4 sampai 41. Gambar 4 menunjukkan evaluasi kinerja sistem dengan menggunakan *k-fold cross validation* dengan bantuan pustaka *skit-learn* dari pemrograman *python*.

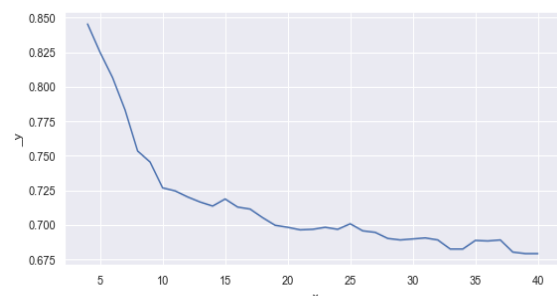


Gambar 4. Hasil evaluasi sistem tanpa *rebalance* data

Berdasarkan gambar 4 evaluasi kinerja sistem dengan menggunakan *10-fold cross validation* dan menggunakan parameter nilai $k=4$ sampai $k=41$, menghasilkan tingkat akurasi paling tinggi mencapai 76% pada nilai $k=31$. Penggunaan parameter nilai k tersebut sangat berpengaruh terhadap proses klasifikasi dan tingkat akurasi yang dihasilkan oleh sistem. Pengujian parameter nilai $k=4$ sampai $k=18$ mengalami grafik kenaikan dalam hal akurasi, untuk parameter $k=19$ sempat mengalami penurunan akurasi kemudian mengalami peningkatan akurasi lagi dan sampai puncaknya pada nilai parameter $k=31$. Pengujian tersebut menjelaskan bahwa semakin tinggi nilai parameter k membuat nilai akurasi semakin baik, berlaku untuk nilai parameter maksimum $k=31$. Setelah itu akurasi akan terus mengalami penurunan, Penurunan akurasi disebabkan oleh beberapa faktor, termasuk kesesuaian data. Data pelatihan yang digunakan memiliki gejala yang sedikit mirip, meskipun berasal dari kelas yang sama. Di sisi lain, gejala yang mirip secara signifikan terdapat pada data dari kelas yang berbeda. Ini mengakibatkan sistem kesulitan dalam mengidentifikasi pola dari data pelatihan yang diperlukan untuk melakukan klasifikasi dengan baik. Semakin tinggi nilai k yang digunakan pada percobaan tersebut sangat berpengaruh terhadap hasil akurasi disebabkan

terjadinya *noise* karena adanya dominasi atau kelas mayoritas pada data latih yang digunakan sehingga hasil klasifikasi akan cenderung diklasifikasikan menjadi kelas mayoritas yaitu kelas positif. Tingkat akurasi tertinggi menggunakan *10-fold cross validation* mengidentifikasi bahwa nilai parameter $K=31$ merupakan kinerja sistem yang paling baik dengan menggunakan skema pengujian tanpa *rebalance* data.

Pada percobaan kedua yaitu dengan masukan data *training* yang berjumlah 1313, kemudian dilakukan *rebalance* data agar distribusi kelas menjadi seimbang. Pengujian menggunakan *10-fold cross validation* dan menggunakan nilai parameter 4 sampai 41. Gambar 5 menunjukkan evaluasi kinerja sistem dengan menggunakan *k-fold cross validation* dengan menggunakan *random over-sampling* untuk *rebalance* data.



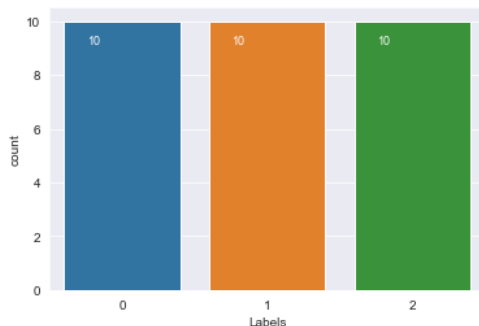
Gambar 5. Hasil evaluasi sistem dengan *rebalance* data

Berdasarkan gambar 5 evaluasi kinerja sistem dengan menggunakan *10-fold cross validation* dan menggunakan parameter nilai $k=4$ sampai $k=41$, menghasilkan tingkat akurasi paling tinggi mencapai 84% pada nilai $k=4$. Penggunaan parameter nilai k tersebut sangat berpengaruh terhadap proses klasifikasi dan tingkat akurasi yang dihasilkan oleh sistem. Semakin tinggi nilai parameter k akan menyebabkan penurunan nilai akurasi, karena semakin besar nilai k mengakibatkan penggunaan lebih banyak tetangga terdekat dalam proses klasifikasi, yang dapat meningkatkan kemungkinan *noise*. Akurasi mencapai nilai maksimum ketika $k=4$. Semakin kecil nilai k yang digunakan mengurangi jumlah tetangga yang digunakan dalam klasifikasi, mengurangi kemungkinan *noise* data dan menghasilkan akurasi yang optimal. Hasil validasi silang *10-fold* menunjukkan

bahwa $k=4$ memberikan tingkat akurasi tertinggi, menunjukkan bahwa kinerja sistem optimal dengan menggunakan skema pengujian yang seimbang untuk data.

3.3 Evaluasi Confusion Matrix

Evaluasi *confusion matrix* digunakan untuk mengukur kinerja dari model dan algoritma yang digunakan dengan membandingkan nilai aktual dengan nilai hasil klasifikasi dengan menggunakan percobaan masukan data uji dan menggunakan model yang dihasilkan dari tahap sebelumnya. Gambar 6 menunjukkan distribusi kelas pengujian pertama pada penelitian ini.



Gambar 6. Distribusi kelas pengujian pertama.

Berdasarkan gambar 6 dapat dijelaskan bahwa untuk percobaan pertama menggunakan 30 masukan data uji dengan distribusi kelas seimbang, kemudian dilakukan pengujian menggunakan 2 model yang dihasilkan sebelumnya, yaitu tanpa *rebalance* data dan setelah *rebalance* data. Tabel 4 menunjukkan hasil evaluasi pada percobaan tersebut.

Tabel 4. Hasil evaluasi pengujian pertama.

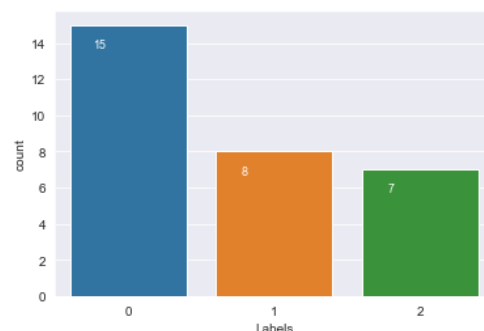
Tanpa <i>rebalance</i> data				
Aktual	Hasil klasifikasi			
	Negatif	Netral	Positif	
Negatif	8	1	1	
Netral	0	0	10	
Positif	0	0	10	

Setelah <i>rebalance</i> data				
Aktual	Hasil klasifikasi			
	Negatif	Netral	Positif	
Negatif	10	0	0	
Netral	4	5	1	
Positif	0	0	10	

Berdasarkan pada tabel 4 sehingga diketahui nilai yang diklasifikasikan dengan benar (TP)

menggunakan model tanpa *rebalance* data adalah 8 pada kelas negatif, 0 pada kelas netral dan 10 pada kelas positif, sehingga menghasilkan akurasi mencapai 60 %. Sedangkan untuk nilai yang diklasifikasikan dengan benar (TP) menggunakan model *rebalance* data adalah 10 pada kelas negatif, 5 pada kelas netral dan 10 pada kelas positif dan menghasilkan akurasi yang meningkat mencapai 83%.

Jika dilihat dari komparasi akurasi antar model diatas, hasil akurasi yang dihasilkan oleh model tanpa *rebalance* data, algoritma KNN menghasilkan hasil klasifikasi 60%. Hasil akurasi tersebut dipengaruhi oleh jumlah kelas yang tidak seimbang pada data latih, dengan netral sebagai kelas minoritas sehingga model kurang bisa mengenali data tersebut secara optimal dan cenderung hasil klasifikasi mengarah ke kelas mayoritas pada data latih. Hal tersebut kemudian dapat dioptimasi menggunakan *rebalance* data sehingga distribusi kelas pada data latih menjadi seimbang dan model mampu untuk mengenali data kelas negatif dan netral dengan lebih baik dengan tingkat akurasi mencapai 83%, terbukti pada jumlah data yang dapat diklasifikasikan dengan benar menjadi 10 pada kelas negatif dan 5 pada kelas netral. Pada percobaan kedua menggunakan masukan data uji dengan distribusi kelas yang ditunjukkan pada gambar 7.



Gambar 7. Distribusi kelas pengujian kedua

Berdasarkan gambar 7 dapat dijelaskan bahwa untuk percobaan kedua menggunakan 30 masukan data uji dengan distribusi kelas tidak seimbang yaitu untuk 50% kelas negatif, 25% untuk kelas netral dan 25% untuk kelas positif. Kemudian dilakukan pengujian menggunakan 2 model yang dihasilkan sebelumnya, yaitu tanpa *rebalance* data dan setelah *rebalance* data.

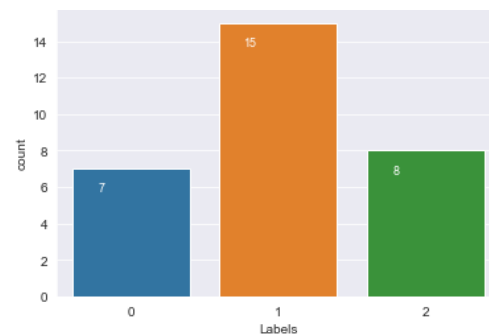
Tabel 5 menunjukkan hasil evaluasi pada percobaan tersebut.

Tabel 5. Hasil evaluasi percobaan kedua

Tanpa <i>rebalance</i> data				
Aktual	Hasil klasifikasi			
	Negatif	Netral	Positif	
Negatif	9	1	5	
Netral	0	0	8	
Positif	0	0	7	
Setelah <i>rebalance</i> data				
Aktual	Hasil klasifikasi			
	Negatif	Netral	Positif	
Negatif	13	2	0	
Netral	2	5	1	
Positif	0	0	7	

Berdasarkan pada tabel 5 dapat diketahui nilai yang diklasifikasikan dengan benar (TP) menggunakan model tanpa *rebalance* data adalah 9 pada kelas negatif, 0 pada kelas netral dan 7 pada kelas positif sehingga menghasilkan akurasi sebesar 53%. Sedangkan untuk nilai yang diklasifikasikan dengan benar (TP) menggunakan model *rebalance* data adalah 13 pada kelas negatif, 5 pada kelas netral dan 7 pada kelas positif, sehingga nilai akurasi meningkat menjadi 83%.

Berdasarkan dari hasil akurasi model diatas tanpa *rebalance* data, algoritma KNN menghasilkan hasil klasifikasi mencapai 53%. Hasil akurasi tersebut dipengaruhi oleh kelas negatif sebagai salah satu kelas minoritas pada data latih yang digunakan, sehingga hasil klasifikasi didominasi oleh kelas mayoritas dan menyebabkan batas-batas yang bias dalam hasil klasifikasi, dibuktikan dengan 9 jumlah data yang diklasifikasi benar pada kelas negatif dan 5 jumlah data diklasifikasikan salah menjadi kelas positif. Hasil akurasi tersebut juga dipengaruhi oleh penggunaan data uji kelas negatif dengan persentase yang tinggi dibandingkan kelas lainnya sehingga model kurang bisa mengenali secara optimal. Untuk percobaan menggunakan *rebalance* data model mampu tetap mampu menghasilkan akurasi yang tinggi mencapai 83% karena distribusi kelas pada data latih seimbang sehingga model dapat mengenali data kelas negatif, netral, maupun positif dengan lebih optimal. Pada percobaan ketiga menggunakan masukan data uji dengan distribusi kelas yang ditunjukkan pada gambar 8.



Gambar 8. Distribusi kelas pengujian ketiga

Berdasarkan gambar 8 dapat dijelaskan bahwa untuk percobaan ketiga menggunakan 30 masukan data uji dengan distribusi kelas tidak seimbang yaitu untuk 50% kelas netral, 25% untuk kelas negatif dan 25% untuk kelas positif. Kemudian dilakukan pengujian menggunakan 2 model yang dihasilkan sebelumnya, yaitu tanpa *rebalance* data dan setelah *rebalance* data. Tabel 6 menunjukkan hasil evaluasi pada percobaan tersebut.

Tabel 6. Hasil evaluasi percobaan ketiga

Hasil klasifikasi				
Aktual	Hasil klasifikasi			
	Negatif	Netral	Positif	
Negatif	5	1	1	
Netral	0	0	15	
Positif	0	0	8	
Hasil klasifikasi				
Aktual	Hasil klasifikasi			
	Negatif	Netral	Positif	
Negatif	7	0	0	
Netral	9	5	1	
Positif	0	0	8	

Berdasarkan pada tabel 6 dapat diketahui nilai yang diklasifikasikan dengan benar (TP) model tanpa *rebalance* data adalah 5 pada kelas negatif, 0 pada kelas netral dan 8 pada kelas positif sehingga menghasilkan akurasi sebesar 43%. Jika dilihat dari hasil akurasi model tanpa *rebalance* data tersebut, kinerja algoritma KNN tersebut dipengaruhi oleh penggunaan data uji dengan kelas netral mempunyai persentase paling tinggi yaitu 50%. Semakin tinggi persentase kelas netral yang digunakan dalam data uji akan semakin menurun tingkat akurasi yang dihasilkan, karena terbatasnya data yang digunakan dalam proses *learning* untuk kelas netral sehingga model menjadi tidak sensitif terhadap kelas netral. Hal tersebut dapat dilihat pada tabel 6,

dengan jumlah data 15 untuk kelas netral diklasifikasikan menjadi kelas positif. Untuk percobaan menggunakan *rebalance* data model menghasilkan peningkatan akurasi yang signifikan menjadi 66%. Penyebabnya adalah penggunaan metode *random over-sampling* dalam proses *rebalance* data, yang menghasilkan sampel dari kelas minor untuk dipelajari dalam proses pembelajaran. Hal ini memungkinkan model pembelajaran untuk menciptakan area pembentukan model klasifikasi yang lebih luas dan membuat model lebih sensitif serta tahan terhadap berbagai skema masukan data uji. Hal tersebut dibuktikan dengan hasil yang diklasifikasikan benar pada kelas netral meningkat dari 0 menjadi 5 data pada skema pengujian.

5. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, disimpulkan bahwa penggunaan baik metode algoritma TF-IDF maupun KNN telah berhasil mengklasifikasikan sentimen ulasan pengguna hotel di laman TripAdvisor ke dalam kategori positif, netral, dan negatif. Implementasi kedua metode tersebut mampu menangani volume data teks yang besar dan mencapai tingkat akurasi sebesar 76%.

Selain itu, penggunaan teknik *random over-sampling* untuk menyeimbangkan data dalam penelitian ini berhasil mengatasi ketidakseimbangan data dan meningkatkan akurasi menjadi 84%. Hasil dari pengolahan menggunakan kedua metode algoritma tersebut disajikan dalam bentuk tabel dan grafik, memberikan informasi yang efektif dan efisien kepada pengguna.

UCAPAN TERIMA KASIH

Penulis ingin menyampaikan rasa terima kasih kepada Jurnal Informatika dan Teknik Elektro Terapan (JITET) Universitas Lampung atas fasilitas yang diberikan untuk publikasi jurnal ini. Penulis juga mengucapkan terima kasih kepada semua yang telah membantu dalam penelitian ini. Penulis menyadari bahwa jurnal ini masih memiliki kekurangan, dan oleh karena itu penulis sangat mengharapkan saran dan kritik yang membangun. Semoga jurnal ini dapat memberikan manfaat kepada yang membutuhkan.

DAFTAR PUSTAKA

- [1] Y. Guo, S. J. Barnes, and Q. Jia, "Mining meaning from online ratings and reviews: Tourist satisfaction analysis using latent dirichlet allocation," *Tour Manag*, vol. 59, pp. 467–483, 2017, doi: <https://doi.org/10.1016/j.tourman.2016.09.009>.
- [2] V. O. Tama, Y. Sibaroni, and Adiwijaya, "Labeling Analysis in the Classification of Product Review Sentiments by using Multinomial Naive Bayes Algorithm," *J Phys Conf Ser*, vol. 1192, no. 1, p. 012036, 2019, doi: 10.1088/1742-6596/1192/1/012036.
- [3] V. Kotu and B. Deshpande, *Predictive analytics and data mining: concepts and practice with RapidMiner*.
- [4] I. F. Rahman, A. N. Hasanah, and N. Heryana, "Analisis Sentimen Ulasan Pengguna Aplikasi Samsat Digiital Nasional (Signal) Dengan Menggunakan METODE NAIVE BAYES CLASSIFIER," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4073.
- [5] V. Chandani and R. S. Wahono, "Komparasi Algoritma Klasifikasi Machine Learning Dan Feature Selection pada Analisis Sentimen Review Film," *Journal of Intelligent Systems*, vol. 1, no. 1, 2015, [Online]. Available: <http://journal.ilmukomputer.org>
- [6] S. B. Imandoust and M. Bolandraftar, "Application of K-nearest neighbor (KNN) approach for predicting economic events theoretical background," *Int J Eng Res Appl*, vol. 3, pp. 605–610, Jan. 2013.
- [7] M. Bilal, H. Israr, M. Shahid, and A. Khan, "Sentiment classification of Roman-Urdu opinions using Naive Bayesian, Decision Tree and KNN classification techniques," *Journal of King Saud University - Computer and Information Sciences*, vol. 28, no. 3, pp. 330–344, 2016, doi: <https://doi.org/10.1016/j.jksuci.2015.11.003>.
- [8] R. Khorsand, M. Rafiee, and V. Kayvanfar, "Insights into TripAdvisor's online reviews: The case of Tehran's hotels," *Tour Manag Perspect*, vol. 34, p. 100673, 2020, doi: <https://doi.org/10.1016/j.tmp.2020.100673>.
- [9] E. Suganya and S. Vijayarani, "Sentiment Analysis for Scraping of Product Reviews from Multiple Web Pages Using Machine Learning Algorithms," in *Intelligent Systems Design and Applications*, A. Abraham, A. K. Cherukuri, P. Melin, and N. Gandhi, Eds., Cham: Springer International Publishing, 2020, pp. 677–685.
- [10] S. Kaur, G. Sikka, and L. K. Awasthi, "Sentiment Analysis Approach Based on N-gram and KNN Classifier," in *2018 First International Conference on Secure Cyber*

- Computing and Communication (ICSCCC)*, 2018, pp. 1–4. doi: 10.1109/ICSCCC.2018.8703350.
- [11] V. Ganganwar, “An overview of classification algorithms for imbalanced datasets,” 2012. [Online]. Available: <https://api.semanticscholar.org/CorpusID:7033031>
- [12] U. Erra, S. Senatore, F. Minnella, and G. Caggianese, “Approximate TF–IDF based on topic extraction from massive message stream using the GPU,” *InfSci (N Y)*, vol. 292, pp. 143–161, 2015, doi: <https://doi.org/10.1016/j.ins.2014.08.062>.
- [13] M. Bilal, H. Israr, M. Shahid, and A. Khan, “Sentiment classification of Roman-Urdu opinions using Naïve Bayesian, Decision Tree and KNN classification techniques,” *Journal of King Saud University - Computer and Information Sciences*, vol. 28, no. 3, pp. 330–344, 2016, doi: <https://doi.org/10.1016/j.jksuci.2015.11.003>.
- [14] R. Khorsand, M. Rafiee, and V. Kayvanfar, “Insights into TripAdvisor’s online reviews: The case of Tehran’s hotels,” *Tour Manag Perspect*, vol. 34, p. 100673, 2020, doi: <https://doi.org/10.1016/j.tmp.2020.100673>.
- [15] K. Shah, H. Patel, D. Sanghvi, and M. Shah, “A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification,” *Augmented Human Research*, vol. 5, no. 1, p. 12, 2020, doi: 10.1007/s41133-020-00032-0.
- [16] K. Shah, H. Patel, D. Sanghvi, and M. Shah, “A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification,” *Augmented Human Research*, vol. 5, no. 1, p. 12, 2020, doi: 10.1007/s41133-020-00032-0.
- [17] M. Bramer, *Principles of Data Mining*. Springer London, 2020. doi: 10.1007/978-1-4471-7493-6.
- [18] S. Makki, R. Haque, Y. Taher, Z. Assaghir, M.-S. Hacid, and H. Zeineddine, “A Cost-Sensitive Cosine Similarity K-Nearest Neighbor for Credit Card Fraud Detection,” in *International Conference on Big Data and Cyber-Security Intelligence*, 2018. [Online]. Available: <https://api.semanticscholar.org/CorpusID:115144809>
- [19] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Inf Process Manag*, vol. 45, no. 4, pp. 427–437, 2009, doi: <https://doi.org/10.1016/j.ipm.2009.03.002>.