Jurnal Riset Statistika (JRS)
e-ISSN 2798-6578 | p-ISSN 2808-3148
https://journals.unisba.ac.id/index.php/JRS

Tersedia secara online di
Unisba Press
https://publikasi.unisba.ac.id/

Aplikasi Algoritma *K-Nearest Neighbor* pada Analisis Sentimen Omicron Covid-19

Alfiari Firdaus*

Prodi Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Islam Bandung, Indonesia.

ARTICLE INFO

Article history :
Received : 07/8/2022
Revised : 16/11/2022
Published : 20/12/2022

Volume : 2
No. : 2
Halaman : 85-92
Terbitan : **Desember 2022**

ABSTRAK

K-Nearest Neighbor (KNN) merupakan salah satu algoritma klasifikasi yang paling banyak digunakan dalam metode *Machine learning*. Klasifikasi KNN merupakan metode klasifikasi non-parametrik konvensional yang telah digunakan sebagai pengklasifikasi dasar dalam banyak masalah klasifikasi pola. Teknik pencarian KNN yang digunakan dalam penelitian ini dengan menggunakan rumus jarak cosine similarity. Keuntungan dari metode ini adalah efektif terhadap data noise dan efektif ketika data training berukuran besar. Namun metode ini masih memiliki kekurangan yaitu masalah tingkat akurasi metode yang digunakan untuk mengukur kemiripan antar objek yang dibandingkan. Tujuan dari penelitian ini adalah untuk mengetahui penerapan metode KNN pada analisis sentimen. Data yang digunakan adalah data tweet sebanyak 12.951 yang diambil dari twitter dengan menggunakan hastag #OmicronVariant dan #Covid19. Hasil penelitian menunjukkan bahwa parameter nilai k terbaik adalah 15. Menggunakan jarak cosine similarity akurasi cukup baik, dan recallnya pun cukup baik kemudian presisinya baik, maka hasil prediksi diperoleh nilai kategori positif lebih tinggi dibandingkan nilai kategori netral dan nilai kategori negatif. Dapat disimpulkan bahwa persepsi masyarakat terhadap Covid-19 Omicron adalah positif, artinya mereka percaya dengan adanya Omicron.

Kata Kunci : KNN; Jarak Cosine Similarity; Analisis Sentimen

ABSTRACT

K-Nearest Neighbor (KNN) is one of the most widely used classification algorithms in *Machine learning* methods. KNN classification is a conventional non-parametric classification method that has been used as a primary classifier in many pattern classification problems. The KNN search technique used in this research is the cosine similarity distance formula. The advantage of this method is that it is effective against noise data and is effective when the training data is large. However, this method still has drawbacks, namely the problem of the accuracy of the method used to measure the similarity between the objects being compared. This study aimed to determine the application of the KNN method to sentiment analysis. The data used is 12,951 tweets taken from Twitter using the hashtags #OmicronVariant and #Covid19. The results showed that the best k value parameter was 15. Using the cosine similarity distance, the accuracy was quite good, and the recall was quite good and the precision was good, so the prediction results obtained that the positive category value was higher than the neutral category value and the negative category value. It can be concluded that the public's perception of Covid-19 Omicron is positive, meaning that they believe in Omicron.

Keywords : KNN; Cosine Similarity Distance; Sentiment Analysis

Corresponding Author : *falfiari@gmail.com
Indexed : Garuda, Crossref, Google Scholar
DOI : https://doi.org/10.29313/jrs.v2i2.1148

A. Pendahuluan

Di tengah kemajuan pesat inovasi penalaran terkomputerisasi (*Artificial Intelligence*) saat ini, kesadaran buatan manusia terdiri dari beberapa cabang, salah satunya adalah *Machine learning*. *Machine learning* bertujuan untuk mengkompilasi data yang diamati dari pengalaman yang dipelajari oleh program untuk menghasilkan informasi yang dapat dimanfaatkan [1]. K-Nearest Neighbor (KNN) merupakan salah satu algoritma klasifikasi dalam metode *Machine learning* yang paling banyak digunakan karena sederhana dan mudah diimplementasikan. KNN adalah suatu metode yang menggunakan algoritma supervised dimana hasil dari query instance yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada KNN [2]. Metode kNN didasarkan pada klasifikasi terhadap objek berdasarkan data pelatihan yang jaraknya paling dekat dengan objek tersebut. kNN [3]. Selain itu, biasanya digunakan sebagai pengklasifikasi dasar dalam banyak masalah domain [4].

Klasifikasi KNN adalah klasifikasi metode non-parametrik konvensional yang telah digunakan sebagai pengklasifikasi dasar dalam banyak masalah klasifikasi pola. Hal ini didasarkan pada pengukuran antara jumlah data testing dan data training untuk memutuskan klasifikasi akhir. Kelebihan metode KNN efektif terhadap data yang noise dan efektif apabila data training besar. Data noise yaitu random error atau varians dalam variable yang diukur, artinya terdapat kesalahan pada data yang bisa disebabkan oleh human error atau outlier yang menyimpang dari normal [5]. Pada umumnya, metode pencarian KNN diselesaikan dengan menggunakan jarak euclidean. Jarak euclidean adalah formula untuk melacak jarak antara dua fokus dalam ruang dimensi dua. Dalam literatur, ada beberapa jenis fungsi jarak lainnya, seperti cosinus similarity [6], minkowski distance [7], manhattan distance, dan linear least square distance [8]. Ukuran kesamaan cosinus biasanya digunakan untuk menghitung nilai kesamaan antara dokumen dalam pencarian teks [6]. Dalam proses analisisnya, penggunaan teknik KNN untuk memutuskan jumlah k yang digunakan untuk mengkarakterisasi informasi baru. Besaran k, idealnya bilangan ganjil, misalnya k = 1, 3, 5, dst. Kepastian nilai k dilihat berdasarkan seberapa banyak informasi yang ada dan ukuran aspek yang dibentuk oleh informasi tersebut. Semakin banyak informasi yang ada, semakin rendah jumlah k yang seharusnya diambil. Namun, semakin besar ukuran aspek informasi, semakin tinggi jumlah k yang harus diambil [6].

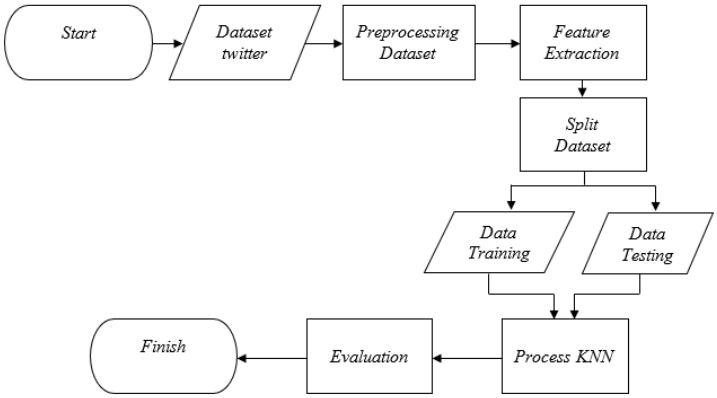
Dengan munculnya web 3.0 berbagai platform seperti facebook, twitter, linkedIn, instagram memungkinkan masyarakat untuk berbagi komentar, pandangan, perasaan, penilaian mereka tentang berbagai topik mulai dari pendidikan hingga hiburan. Platform ini berisi sejumlah besar data dalam bentuk tweet, blog, dan pembaruan status, posting, dan lain – lain. Banyaknya opini atau persepsi masyarakat di platform tersebut memunculkan berbagai tanggapan positif, negatif, atau bahkan netral. Twitter menjadi platform yang sering digunakan untuk mengungkapkan opini atau persepsi tentang berbagai hal. Dalam sehari twitter mampu menghasilkan jumlah tweet kurang lebih sebanyak 500 juta cuitan yang dikirimkan oleh penggunanya dari seluruh penjuru dunia [9]. Sentimen adalah istilah yang digunakan untuk menggambarkan topik yang subjektif dan objektif dan topik faktual atau non-faktual yang melampaui perbedaan antara topik positif atau negatif [10]. Analisis sentimen adalah pendekatan analitis yang digunakan untuk menganalisis sebuah teks. Tujuan dari analisis sentimen adalah untuk mengetahui subjektivitas opini, hasil review atau tweet. Berdasarkan analisis sentimen, opini dari seseorang dapat diklasifikasikan ke dalam berbagai kategori berdasarkan ukuran data dan jenis dokumen [11]. KNN merupakan metode pengklasifikasian, sehingga analisis sentimen dengan menggunakan metode KNN dapat menjadi solusi untuk menentukan hasil klasifikasi dari cuitan pada platform twitter [3].

B. Metode Penelitian

Peneliti menggunakan metode *K-Nearest Neighbour*. Populasi yang dipilih dalam penelitian ini adalah cuitan *tweets* masyarakat Indonesia pada bulan Desember 2021 hingga Februari 2022 yang berjumlah 12.951 tweets yang diperoleh dari website netlytics. Variabel penelitian yang digunakan dalam penelitian ini adalah tweet atau opini yang dituangkan masyarakat dalam media sosial twitter dengan hashtag #Covid19 dan #OmicronVariant.

Dalam pengambilan data tweet ini menggunakan teknik crawling sehingga diperoleh sebanyak 12.951 tweets. Pada penelitian ini akan diklasifikasikan data tweets tersebut dengan melihat tingkat akurasi, presisi,

dan recall menggunakan cosine similarity distance dengan bantuan software python. Berikut merupakan flowchart untuk penelitian ini:



Gambar 1. *Flowchart* Analisis Penelitian

Dari *flowchart* diatas dijelaskan beberapa tahapan, yaitu: (1) Input data tweet; (2) Melakukan pre-processing pada data tweet; (a) Cleaning data dengan cara menghapus terlebih dahulu akun-akun bot yang dilihat dari sisi followers, jam post tweet, source tweet yang tidak dikenal. Diasumsikan followers kurang dari 100 merupakan akun bot, jam post tweet dari jam 00.00 – 04.00 merupakan bot; (b) Melakukan *Case Folding* yaitu mengubah kalimat yang didapat menjadi format yang sama dalam artian menjadi lower case semua; (c) Melakukan tokenisasi yaitu menghilangkan whitespace dan membuang karakter tertentu seperti tanda baca, emoji dan url; (d) Melakukan *Stemming* yaitu menyederhanakan kata yang berisi imbuhan; (e) Melakukan normalisasi kata yaitu untuk mengurangi huruf berturut-turut dari suatu kata; (f) Melakukan stopword removal yaitu menghilangkan kata umum yang sering muncul tetapi tidak memiliki arti penting dan tidak digunakan, contoh has, and, he, being dan sebagainya

(3) Merubah kalimat data tweet menjadi kategori, nilai 1 sebagai label sentimen positif, nilai 2 sebagai label sentimen negatif, dan nilai 3 sebagai label sentimen netral; (4) Membagi data tweet menjadi data training dan data testing dengan proporsi 80:20. Berdasarkan hasil penelitian oleh (Prakasa & Lhaksmana, 2018) dalam mengklasifikasikan data tweet menggunakan proporsi data training dan data testing sebesar 80:20 memberikan hasil akurasi yang paling baik yaitu 90.50% menggunakan metode K-Nearest Neighbour; (5) Menghitung jarak cosine similarity distance antara record data testing dan data training; (6) Setelah mendapatkan jarak cosine similarity distance selanjutnya menentukan jumlah k atau tetangga terdekat, k yang digunakan yaitu 1 sampai 40 dengan menggunakan trial and error. Semakin banyak informasi yang ada, semakin rendah jumlah k yang seharusnya diambil. Namun, semakin besar ukuran aspek informasi, semakin tinggi jumlah k yang harus diambil [6]; (7) Membuat confusion matrix untuk mengevaluasi penggunaan jarak dengan melihat tingkat akurasi, presisi dan recall.

C. Hasil dan Pembahasan

Preprocessing Data

Tahapan *preprocessing data* perlu dilakukan karena beberapa kalimat tweet yang didapatkan tidak sepenuhnya menggunakan kata baku dan menggunakan bahasa inggris yang baik.

Preprocessing data dilakukan dengan tahap *Case Folding*, Tokenisasi, *Stemming*, normalisasi kata, dan *Stopwords Removal* sehingga menghasilkan data bersih dan siap untuk lanjut pada proses berikutnya. Selanjutnya dilakukan proses *duplicate* dengan cara menghapus kalimat tweet yang sama. Setelah itu dilanjutkan dengan proses *filtering* melalui tiga tahapan, diantaranya: (1) Melakukan proses filtering pada peubah source dengan cara menghapus source tweet yang tidak dikenal; (2) Melakukan proses filtering pada peubah user_followers_count dengan cara menghapus followers kurang dari 100 merupakan akun bot; (3) Melakukan proses filtering pada peubah pubdate dengan cara menghapus jam post tweet dari jam 00.00 – 04.00.

Case Folding

Case Folding adalah proses merubah data tweet menjadi lowercase. Berikut merupakan contoh data penelitian yang dilakukan proses *Case Folding*. Pada tahapan *Case Folding library* yang digunakan yaitu library pandas dengan menggunakan fungsi str.lower(). Fungsinya untuk menyeragamkan kalimat-kalimat tweet menjadi huruf kecil dan juga agar memudahkan proses preprocessing selanjutnya menjadi lebih mudah.

Tokenisasi

Tokenisasi dalam penelitian ini merupakan tahapan dalam memecah string atau input terhadap suatu teks yang telah melewati tahap *Case Folding* berdasarkan tiap kata yang menyusunnya dan menghilangkan URL, @mention dan hashtag. Tahap tokenisasi dilakukan dengan menggunakan fungsi nltk_tokenize(), library pada bahasa pemrograman Python yang bernama Natural Language Tools Kit (NLTK). Dilakukan Import library terlebih dahulu.

Library string digunakan untuk memuat satu karakter atau lebih yang ada pada data tweet. Terlebih dahulu import library Regular Expression (re) untuk melakukan tahapan atau deretan karakter yang digunakan untuk pencarian teks dengan menggunakan pola (*pattern*). Dengan menggunakan library re dapat memudahkan dalam mencari string tertentu dari teks yang banyak. Selain itu pada tahap ini juga dilakukan proses removing number, whitespace dan punctuation (tanda baca). Pada tahapan ini, terlebih dahulu harus membuat function agar proses tokenisasi sesuai dengan apa yang dibutuhkan. Berikut merupakan hasil dari proses tokenisasi.

Stemming

Stemming adalah tahap mencari root (dasar) kata dari tiap kata hasil filtering sebelumnya dengan menghapus kata imbuhan di depan maupun imbuhan di belakang kata. Tahap *Stemming* dilakukan dengan menyederhanakan kata yang berisi imbuhan untuk mencari kata dasar dari hasil proses sebelumnya yaitu tokenisasi.

Pada proses syntax *Stemming* sedikit menjadi lebih mudah karena cuitan tweet menggunakan bahasa inggris. Mengapa demikian karena pada software python library yang digunakan untuk proses ini defaultnya adalah bahasa inggris, oleh sebab itu tidak perlu menggunakan library tambahan seperti sastrawi.

Normalisasi Kata

Normalisasi kata merupakan tahapan untuk mengurangi huruf berturut-turut dari suatu kata. Normalisasi teks pada tahapan ini merupakan pengubahan kata yang sebelumnya memiliki karakter huruf yang berturut-turut. Pada proses syntax normalisasi kata ini dimaksudkan agar memudahkan proses selanjutnya untuk mengidentifikasi cuitan tweet yang ganda atau lebih, karena proses normalisasi kata ini menghilangkan kata yang memiliki huruf berlebih.

Stopwords Removal

Fungsi *Stopwords Removal* merupakan tahap preprocessing yang akan menghilangkan kata-kata tidak penting. Tahap pertama pada proses *Stopwords Removal* adalah mengambil setiap kata pada dokumen kemudian dibandingkan dengan kamus stopwords yang berada di variable directory atau library nltk.corpus, jika ditemukan kata tidak penting maka akan dihapus, sehingga nantinya akan menghasilkan kata unik yang disebut keyword.

Feature Extraction

Feature Extraction data hasil crawling dan telah melalui tahapan preprocessing dilakukan secara manual dengan menggunakan textblob dengan melihat polarity, subjectivity yang dimiliki oleh teks tweet yang telah dikumpulkan. Textblob adalah salah satu library yang disediakan oleh Python untuk pemrosesan dibidang Natural Language Processing yang dapat memberikan tag kata, ekstraksi kata, analisis sentimen. Saat ini textblob hanya tersedia dalam bahasa inggris. Penentuan kelas positif, netral dan negatif didasari oleh nilai polaritas. Nilai polaritas pada analisis sentimen berada pada rentang 1 sampai -1.

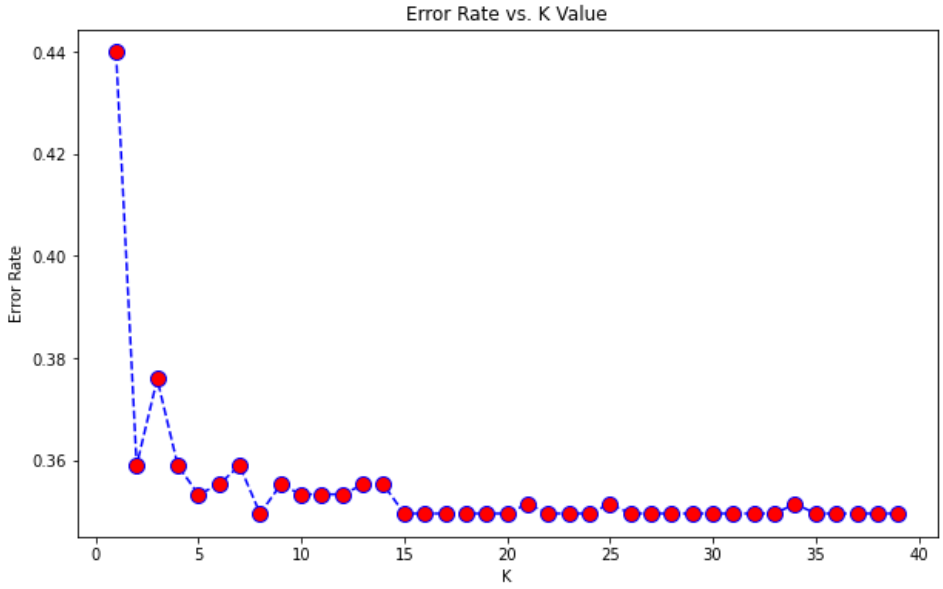
Pada proses syntax *Feature Extraction* ini dibantu menggunakan textblob. Untuk proses ekstraksinya disini apabila nilai polaritasnya kurang dari nol masuk kedalam kategori sentimen negatif, kemudian apabila nilai polaritasnya sama dengan nol masuk kedalam kategori sentimen netral, dan apabila nilai polaritasnya lebih besar dari nol maka masuk ke dalam kategori sentimen positif.

	CleanText	Polarity	Analysis
0	discuss mask back england case wale higher m...	0.187500	Positive
1	sigh relief south africa omicron variant app...	0.333333	Positive
2	new covid variant everyone spook cautious mi...	0.136364	Positive
3	mandatory thread hope hard two year million ...	-0.089394	Negative
4	operation international arrival run smooth i...	0.178788	Positive
5	response emergence include uk case jcvi toda...	0.066667	Positive
6	reflect threat prolonged vaccine injustice l...	-0.325000	Negative
7	sigh relief south africa omicron variant app...	0.333333	Positive
8	analyst expect grow range year end march con...	0.136364	Positive
9	fact vaccine provide good reliable immune pr...	0.700000	Positive

Gambar 2. Proses labeling menggunakan TextBlob

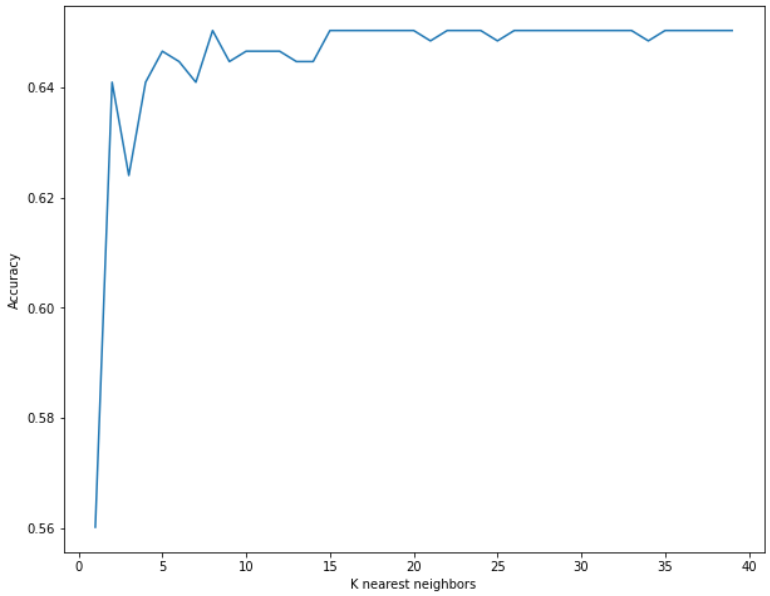
Dapat dilihat pada Gambar 2 proses labeling. Proses ini didasari dari nilai polaritas hasil kalimat teks tweet tersebut. Dapat diambil contoh pada tweet kedua yaitu "sigh relief south africa omicron variant appear super mild' mutation" diperoleh nilai polaritasnya yaitu sebesar 0.333333 yang dapat disimpulkan termasuk pada kategori sentimen positif, artinya tweet tersebut mempercayai adanya omicron. Dari hasil labeling tersebut diperoleh sentimen kategori positif terdapat 1688 tweet, kemudian sentimen kategori negatif terdapat 469 tweet dan sentimen kategori netral terdapat 501 tweet. Setelah didapatkan hasil ketiga kategori sentimen tersebut maka dilakukan proses vectorisasi dengan menggunakan library sklearn.feature_extraction.text dengan memanggil CountVectorizer. Fungsi ini dimaksudkan untuk mengubah kalimat tweet menjadi vektor agar bisa dilakukan proses analisis menggunakan metode KNN.
Implementasi KNN pada Analisis Sentimen menggunakan *Cosine Similarity Distance.*

Proses cosine similarity distance diawali dengan menentukan nilai *k* dengan menggunakan data training dan dilihat dari nilai error ratenya, berikut merupakan grafik menentukan nilai *k* dengan melihat nilai *error rate* terendah.



Gambar 3. Proses penentuan nilai k dilihat dari nilai error rate

Pada Gambar 3 dapat dilihat bahwa jika nilai k ≥ 15 akan memperoleh nilai error rate yang kecil yaitu kurang dari 0.36 dan cenderung konvergen. Maka dapat disimpulkan untuk penentuan nilai k yang diambil yaitu nilai k = 15. Selanjutnya pembuktian dengan menggunakan data training akan dilihat nilai akurasi apabila dilihat dari nilai k = 15. Berikut merupakan hasil grafik nilai k dengan hasil akurasinya:



Gambar 4. Melihat nilai akurasi dengan memasukan nilai k

Pada Gambar 4. dapat dilihat bahwa jika nilai k ≥ 15 akan memperoleh nilai akurasi yaitu lebih besar dari 0.64. Maka dapat disimpulkan benar bahwa dari hasil penentuan nilai k pada Gambar 4.6 dengan menggunakan nilai k ≥ 15 akan memperoleh nilai akurasi yang cukup tinggi. Maka dapat disimpulkan pengambilan nilai k yaitu 15 dengan nilai akurasi menggunakan data training sebesar 64.12%. Proses selanjutnya menggunakan data testing untuk melihat dari penggunaan euclidean distance pada analisis sentimen menggunakan data twitter. Dengan menggunakan k = 15 diperoleh nilai akurasi sebesar 65.04%. Maka setelah diketahui nilai akurasi menggunakan euclidean distance proses selanjutnya akan digunakan keempat fungsi jarak lainnya, fungsi jarak pada metode KNN ini yaitu merupakan perhitungan jarak dari dua buah titik dalam space distance. Penerapan fungsi jarak pada analisis sentimen kasus covid-19 omicron adalah untuk melihat pendapat atau kecenderungan opini terhadap kasis covid-19 omicron terhadap seseorang, apakah cenderung beropini negatif, netral atau positif. Serta dapat memberikan gambaran akurasi yang didapat dari analisis sentimen kasus covid-19 omicron tersebut. Berikut merupakan hasil confusion matrix menggunakan jarak euclidean distance dengan k = 15:

Tabel 1. Confusion Matrix Cosine Similarity Distance dengan k = 15

		Nilai Aktual		
		Positive	***Neutral***	***Negative***
Nilai Prediksi	***Positive***	333	5	8
	Neutral	92	0	0
	Negative	90	3	1

Pada Tabel 1. diperoleh hasil dari proses data testing dengan menggunakan cosine similarity distance berupa confusion matrix. Berikut merupakan penjelasannya: (1) Prediksi terhadap cuitan sentimen kategori positif yang diklasifikasikan dengan benar sebagai sentimen kategori positif ada sebanyak 333; (2) Prediksi terhadap cuitan sentimen kategori netral yang diklasifikasikan dengan benar sebagai sentimen kategori netral ada sebanyak 0; (3) Prediksi terhadap cuitan sentimen kategori negatif yang diklasifikasikan dengan benar sebagai sentimen kategori negatif ada sebanyak 1; (4) Prediksi terhadap cuitan sentimen kategori positif yang diklasifikasikan dengan salah sebagai sentimen kategori netral ada sebanyak 92; (5) Prediksi terhadap cuitan sentimen kategori positif yang diklasifikasikan dengan salah sebagai sentimen kategori negatif ada sebanyak 90; (6) Prediksi terhadap cuitan sentimen kategori netral yang diklasifikasikan dengan salah sebagai sentimen kategori positif ada sebanyak 5; (7) Prediksi terhadap cuitan sentimen kategori netral yang diklasifikasikan

dengan salah sebagai sentimen kategori negatif ada sebanyak 3; (8) Prediksi terhadap cuitan sentimen kategori negatif yang diklasifikasikan dengan salah sebagai sentimen kategori positif ada sebanyak 8; (9) Dan prediksi terhadap cuitan sentimen kategori negatif yang diklasifikasikan dengan salah sebagai sentimen kategori netral ada sebanyak 0.

Dari hasil confusion matrix dapat dihitung nilai akurasi, presisi dan recall. Berikut hasil akurasi, presisi dan recall:

$$Accuracy = \frac{TP + TNa + TNe}{TP + TNa + TNe + FP_1 + FP_2 + FNa_1 + FNa_2 + FNe_1 + FNe_2} * 100\%$$

$$Accuracy = \frac{333 + 0 + 1}{333 + 0 + 1 + 92 + 90 + 5 + 3 + 8 + 0} * 100\% = 62.78\%$$

$$Precision_+ = \frac{TP}{TP + FP_1 + FP_2} * 100\%$$

$$Precision_+ = \frac{333}{333 + 92 + 90} * 100\% = 64.66\%$$

$$Recall_+ = \frac{TP}{TP + FNa_1 + FNe_1} * 100\%$$

$$Recall_+ = \frac{333}{333 + 5 + 8} * 100\% = 96.24\%$$

Dengan menggunakan cosine similarity distance pada data tweet nilai akurasi sebesar 62.78% menandakan model kurang baik dalam melakukan klasifikasi (poor classification). Nilai presisi sebesar 64.66% menandakan persentase cuitan tweet yang benar masuk kategori sentimen positif dari keseluruhan cuitan tweet yang diprediksi sentimen positif. Nilai recall sebesar 96.24% menandakan persentase cuitan tweet positif yang diprediksi sentimen positif dibandingkan keseluruhan cuitan tweet yang sebenarnya masuk kedalam kategori sentimen positif. Adapun hasil perhitungan menggunakan cosine similarity distance prediksi sentimen yang masuk kedalam kategori positif, negatif, dan netral adalah sebagai berikut:

Tabel 2. Persentase Nilai Prediksi Cosine Similarity Distance dengan k = 15

Kategori Prediksi	Nilai Prediksi	Persentase
Positive	346	65.04%
Neutral	92	17.29%
Negative	94	17.67%
TOTAL	**5**	**100%**

Pada Tabel 2 menjelaskan nilai persentase prediksi euclidean distance menggunakan k = 15 terhadap analisis sentimen data tweet dengan hashtag #OmicronVariant dan #Covid19 yaitu diperoleh persentase kategori sentimen positif menggunakan cosine similarity distance sebesar 65.04% artinya tweet tersebut tersebut mempercayai adanya omicron, sedangkan persentase kategori sentimen negatif menggunakan cosine similarity distance sebesar 17.67% artinya tweet tersebut kontra atau tidak mempercayai adanya omicron dan persentase kategori sentimen netral menggunakan cosine similarity distance sebesar 17.29% artinya tweet tersebut hanya sebuah berita atau tidak pro dan kontra terhadap omicron.

D. Kesimpulan

Berdasarkan hasil penelitian mengenai penerapan metode KNN dalam analisis sentimen kasus Covid-19 jenis Omicron dengan menggunakan fungsi jarak yaitu, cosine similarity distance yang telah dibahas didapatkan kesimpulan sebagai berikut:

Penerapan metode KNN pada analisis sentimen berhasil dilakukan dengan menggunakan fungsi jarak yaitu, cosine similarity distance

Dengan menggunakan nilai k = 15, diperoleh hasil akurasi tertinggi menggunakan jarak cosine similarity distance dengan nilai akurasi 62.78% kemudian nilai presisi sebesar 64.66% dan nilai recall sebesar 96.24%. Maka penerapan metode KNN pada analisis sentimen menggunakan data tweet akan cocok jika menggunakan jarak cosine similarity distance.

Daftar Pustaka

[1] M. AE, "Comparative Study of Machine Learning Techniques for Supervised Classification of Biomedical Data," *Int. J. Appl. Sci. Technol.*, vol. 7, no. 2, pp. 5–18, 2017.

[2] R. Samuel, R. Natan, and U. Syafiqoh, "Penerapan Cosine Similarity dan K-Nearest Neighbor (K-NN) pada Klasifikasi dan Pencarian Buku," *J. Big Data Anal. Artif. Intell.*, vol. 1, no. 1, pp. 9–14, 2018.

[3] Anggi Priliani Yulianto and S. Darwis, "Penerapan Metode K-Nearest Neighbors (kNN) pada Bearing," *J. Ris. Stat.*, vol. 1, no. 1, pp. 10–18, Jul. 2021, doi: 10.29313/jrs.v1i1.16.

[4] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 4–37, 2000, doi: 10.1109/34.824819.

[5] S. A. Alasadi and W. S. Bhaya, "Review of data preprocessing techniques in data mining," *J. Eng. Appl. Sci.*, vol. 12, no. 16, pp. 4102–4107, 2017, doi: 10.3923/jeasci.2017.4102.4107.

[6] C. D. Manning, P. Raghavan, and H. Schutze, *An Introduction to Modern Information Retrieval*, Online Edi., vol. 53, no. 9. Cambridge: Cambridge University Press, 2009. doi: 10.1108/00242530410565256.

[7] B. G. Batchelor, *Pattern Recognition: Ideas in Practice*. 2011.

[8] J. Singh, G. Singh, and R. Singh, "A review of sentiment analysis techniques for opinionated web text," *CSI Trans ICT*, vol. 4, no. 2–4, pp. 241–7, 2016.

[9] M. H. Syahnur, M. A. Bijaksana, and M. S. Mubarok, "Kategorisasi Topik Tweet di Kota Jakarta , Bandung , dan Makassar dengan Metode Multinomial Naïve Bayes Classifier Program Studi Sarjana Teknik Informatika Fakultas Informatika Universitas Telkom Bandung," vol. 3, no. 2, pp. 3612–3620, 2016.

[10] F. Pozzi, E. Fersini, E. Messina, and B. Liu, *Sentiment Analysis in Social Networks*. Morgan Kaufmann, 2016.

[11] D. S. Rajput, R. S. Thakur, and S. M. Basha, *Sentiment Analysis and Knowledge Discovery in Contemporary Business*. India, 2018.