Perbandingan Improved K-Nearest Neighbour Dengan K-Nearest Neighbour Pada Analisis Sentimen Moda Raya Terpadu Jakarta

Fahmy Akhmad Firdaus, Yulison Herry Chrisnanto, Puspita Nurul Sabrina

Program Studi Informatika, Fakultas Sains dan Informatika Universitas Jenderal Achmad Yani
Jl.Terusan Jenderal Sudirman, 148 Cimahi, Jawa Barat, Indonesia
Email: fahmy.akhmad@student.unjani.ac.id, y.chrisnanto@lecturer.unjani.ac.id dan puspita.sabrina@lecture.unjani.ac.id

ABSTRAK

K-Nearest Neighbour merupakan algoritma klasifikasi yang dikenal sebagai metode berbasis jarak. Improved K-Nearest Neighbour merupakan perkembangan dari K-Nearest Neighbour yang memiliki perbedaan pada bobot nilai k yang memiliki nilai tetap Permasalahan dalam penelitian ini adalah bagaimana akurasi metode iknn dibandingan pada analisis sentimen Moda Raya Terpadu Jakarta (MRT). MRT Jakarta merupakan sebuah transportasi umum yang menggunakan listrik di Jakarta yang diharapkan dapat mengurangi angka kemacetan di daerah Jakarta. Pengoprasian MRT yang sudah secara resmi banyak menimbulkan respon dari masyarakat, baik itu respon yang positif, negatif, maupun netral. Untuk mengetahui hal tersebut, analisis sentiment dapat digunakan untuk mengklasifikasikan sebuah kalimat. Hasil penelitian dengan eksperimen dataset yang tidak balance dan yang balance di setiap kelasnya, eksperimen nilai K dan beberapoa splitting data menunjukkan bahwa peningkatan akurasi metode Improved K-Nearest Neighbour terhadap K-Nearest Neighbour pada kasus analisis sentiment moda raya terpadu tidak signifikan, dengan akurasi 77.24%, precission sebesar 0.77, Recall sebesar 0.77, dan F1 Score sebesar 0.77. Sedangkan metode K-Nearest Neighbour memiliki akurasi sebesar 76.12%, dengan precission sebesar 0.76, Recall sebesar 0.76, dan F1 Score sebesar 0.76.

Kata kunci: Improved K-Nearest Neighbour. K-Nearest Neighbour, MRT Jakarta

ABSTRACT

K-Nearest Neighbor is a classification algorithm known as the distance-based method. Improved K-Nearest Neighbor is a development of K-Nearest Neighbor which has a difference in the weight of the value of k which has a fixed value. The problem in this research is how the accuracy of the Improved K-Nearest Neighbor method is compared to the sentiment analysis of the Jakarta Integrated Raya Mode (MRT). MRT Jakarta is a public transportation that uses electricity in Jakarta which is expected to reduce congestion in the Jakarta area. The operation of the MRT which has officially elicited many responses from the public, be it positive, negative or neutral responses. To know this, sentiment analysis can be used to classify a sentence. The results of the research with unbalanced and balanced dataset experiments in each class, experiments on K values and some data splitting show that the increase in accuracy of the Improved K-Nearest Neighbor method against K-Nearest Neighbor in the case of integrated modal sentiment analysis is not significant, with an accuracy of 77.24 %, precision of 0.77, Recall of 0.77, and F1 Score of 0.77. While the K-Nearest Neighbor method has an accuracy of 76.12%, with a precision of 0.76, Recall of 0.76, and F1 Score of 0.76.

Keywords: *Improved K-Nearest Neighbor. K-Nearest Neighbor, MRT Jakarta*

PENDAHULUAN

Jakarta merupakan kota metropolitan terbesar di Indonesia yg mempunyai segala daya tarik bagi penduduk Indonesia, banyak sekali latar masyarakat dari berbagai daerah Indonesia beraktivitas didalamnya(Pambudi & Hidayati, 2020). Banyaknya jumlah penduduk akan berbanding lurus dengan padatnya kegiatan sehari-hari dan tentu saja

meningkatnya kemacetan. Salah satu penyebab kemacetan yang terjadi di Jakarta adalah banyaknya masyarakat yang masih menggunakan kendaraan pribadi untuk setiap orang dari pada menggunakan kendaraan umum. Moda Raya Terpadu (MRT) menjadi moda transportasi terbaru untuk mengatasi kemacetan di ibu kota(Agustina & Rahmah, 2022).

Analisis Sentimen merupakan metode yang Opinion mining merupakan suatu proses deteksi polaritas dan ekstraksi fitur (variabel) dalam bentuk klasifikasi dengan kategori positif dan negatif(Indriya Dewi Onantya, Indriati, 2019). Selama proses klasifikasi dilakukan analisis sentimen dengan *Improved K-Nearest Neighbour* perkembangan *K-Nearest Neighbour*. Perubahan tersebut dengan cara mendefinisikan titik-k berubah. Namun nilai k tetap ditentukan karena perbedaan nilai K pada setiap kelasnya(Arafah & Fathoni, 2022)

K-Nearest Neighbour merupakan metode yang bekerja dengan mencari jarak terdekat antara suatu data dengan sejumlah k data terdekat. Selanjutnya, data yang paling dominan di antara data-data yang memiliki jarak terdekat akan ditentukan(Amardita, Adiwijaya, & Purbolaksono, 2022). *Improved K-Nearest Neighbour* merupakan perkembangan dari metode *K-Nearest Neighbour*, *Improved K-Nearest Neighbour* ini digunakan karena tangguh dengan data noisy. Perbedaan *Improved K-Nearest Neighbour* dengan metode *K-Nearest Neighbour* ada pada penentuan nilai k yang berbeda pada setiap kategorinya(Arafah & Fathoni, 2022).

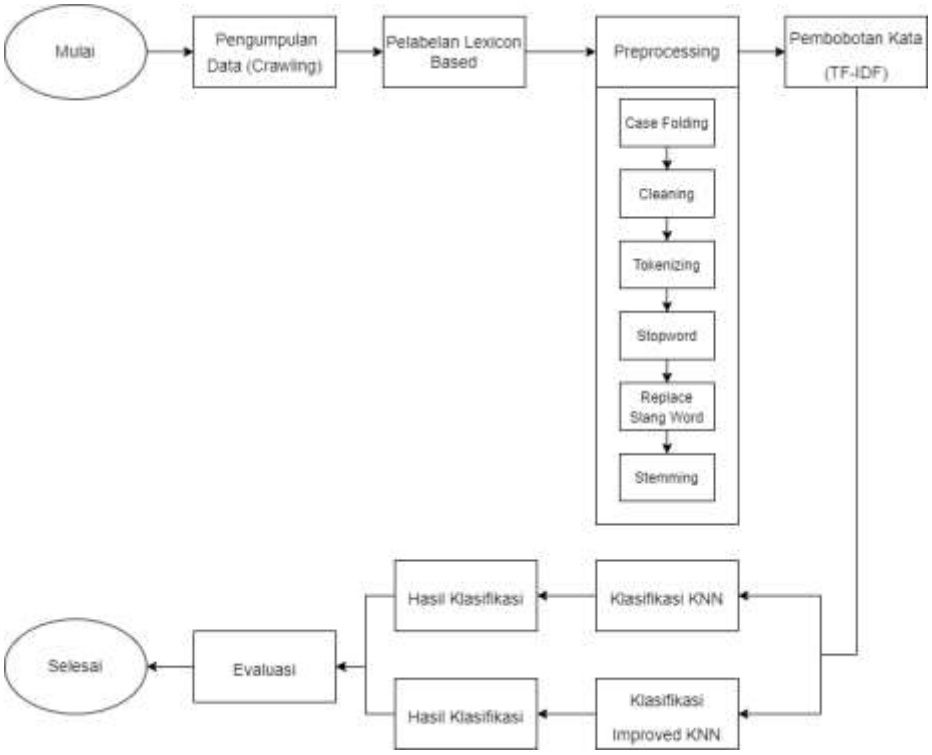
Salah satu kajian yang dilakukan Okfalisa pada tahun 2017 adalah perbandingan *K-Nearest Neighbour* dan *Improved K-Nearest Neighbour* unit pelaksana kiriman uang bersyarat dengan 7395 record. Hasil dari penelitian ini adalah *K-Nearest Neighbour* mencapai presisi tertinggi dengan rata-rata akurasi 94,95, dengan akurasi tertinggi 93,94, *Improved K-Nearest Neighbour* dengan tingkat akurasi tertinggi 99,51, dan rata-rata tingkat akurasi sebesar 99,51. 99,20%(Arafah & Fathoni, 2022).

Perkembangan metode Improved K-Nearest Neighbour dari K-Nearest Neighbour ini tentunya dapat dilihat dari sejauh mana hasil akhir dari penggunaan kedua metode ini pada studi kasus tertentu, apakah terdapat peningkatan yang signifikan pada metode Improved K-Nearest Neighbour ini.

Dari paparan diatas maka penelitian ini menerapkan metode Metode K-*Nearest Neighbour* dan *Improved K-Nearest Neighbour* pada Analisis sentiment MRT Jakarta untuk mengetahui sejauh mana peningkatan performa dari metode *Improved K-Nearest Neighbour* dan dapat mengetahui apakah ulasan pada data yang didapat tersebut positif, negatif, atau netral.

METODE PENELITIAN

Berikut merupakan alur metode penelitian yang dilakukan pada penelitian ini.



Gambar 1 Metode Penelitian

1. Pengumpulan Data
   Metode pengumpulan data menggunakan teknik crawling pada media sosial twitter dengan kata kunci "MRTJakarta" pada CMD menggunakan Node.js dan menghasilkan 1500 data.

2. Pelabelan Lexicon Based
   Dalam tahap pelabelan data ini menggunakan kamus Lexicon Based yang memuat kamus kata dalam bahasa Indonesia agar dapat menentukan kelas sentimennya, lexicon sudah cukup baik dalam pelabelan data sentiment, karena kamus lexicon sudah memiliki cukup kesesuaian antar kata yang bersifat positif, negatif, dan netral. Lexicon adalah metode yang menggunakan kata-kata opini untuk memberikan nilai emosional pada sebuah teks. Pendekatan ini efisien dan dapat digunakan untuk menganalisis teks dalam skala dokumen, kalimat, atau entitas. Pendekatan berbasis leksikon menggunakan kamus sebagai sumber informasi saat mempelajari bahasa baru(Syakur, 2021).

3. Preprocessing
   Pada tahap ini ada beberapa tahapan dalam prosesnya yaitu case folding, cleaning, tokenizing, filtering, replace slang word dan stemming. Tujuan dari preprocessing data adalah untuk membersihkan, mengubah format, dan mengatur data sehingga

lebih sesuai dan siap digunakan dalam tahapan analisis selanjutnya(Hendrastuty et al., 2021).

a. Case Folding
   Pada tahapan ini proses penyamaan bentuk huruf pada teks apakah ingin huruf kecil semua atau huruf kapital.
b. Cleaning
   Proses ini merupakan proses untuk menghilangkan dokumen yang tidak diperlukan, seperti username (@), URL, hastag (#), dan kata yang tidak diperlukan lainnya.
c. Tokenizing
   Pada Tahap ini melakukan pemotongan istilah kata dalam kalimat atau disebut *parsing* yang menggunakan spasi sebagai penanda pemisah kata, kemudian menghasilkan token yang akan memudahkan proses mengetahui frekuensi.
d. Stopword
   Proses Filtering merupakan proses pengambilan kata-kata dari hasil tokenizing dan membuang kata yang tidak penting dalam suatu kalimat.
e. Replace Slang Word
   Tahap ini merupakan tahapan yang mengubah kata – kata singkatan menjadi kata yang lebih jelas, seperti yg menjadi yang, gmn menjadi gimana, dan skrng menjadi sekarang.
f. Stemming
   Stemming adalah suatu proses yang mengubah kata yang memiliki imbuhan menjadi kata dasar. Tujuannya agar data tersebut mampu untuk diolah diproses selanjutnya.

4. Pembobotan Kata

Pembobotan TF-IDF (Term Frequency-Inverse Document Frequency) adalah sebuah proses untuk mengubah data teks menjadi data numerik yang digunakan untuk memberikan bobot pada setiap kata atau fitur. TF-IDF merupakan metrik statistik yang digunakan untuk menilai seberapa penting suatu kata dalam sebuah dokumen. TF mengukur frekuensi kemunculan kata dalam dokumen, menunjukkan pentingnya kata tersebut dalam dokumen tersebut. DF mengukur frekuensi dokumen yang mengandung kata tersebut, memberikan gambaran seberapa umum kata tersebut. IDF merupakan nilai kebalikan dari DF. Hasil pembobotan kata menggunakan TF-IDF didapatkan dengan mengalikan nilai TF dengan nilai IDF. Bobot kata akan semakin tinggi jika kata tersebut sering muncul dalam suatu dokumen dan akan semakin rendah jika kata tersebut muncul dalam banyak dokumen(Septian, Fachrudin, & Nugroho, 2019).

Tahapan perhitungan TF-IDF yaitu

1. Menghitung *Term Frequency* (TF)

Yaitu frequensi setiap pada kalimat yang sudah dipisah menjadi kata. kemdian kata yang muncul pada dokumen maka diberi nilai 1.

2. Menghitung *Document Frequency* (DF)

   Mengihtung jumlah nilai TF yang muncul pada dokumen.

3. Menghitung *Invers Document Frequency (IDF),*

   Setelah menadapatkan nilai TF dan DF maka dilakukan perhitungan yang ditunjukan pada persaman (1).

$$IDF_j = \log\left(\frac{N}{df_j}\right) \qquad (1)$$

   Keterangan :

   IDF : Merupakan Hasil *Invers Dokumen Frequency*

   N : Jumlah Dokumen

   df : Jumlah Dokumen pada kata yang dicari

4. Menghitung *Weight*

   Melakukan perhitungan bobot pada setiap kata berdasarkan hasil nilai TF dikali dengan Nilai IDF, perhitungannya ditunjukan pada persamaan (2).

$$w_{ij} = tf_{ij} \times idf_j \qquad (2)$$

   Setelah mendapatkan hasil pembobotan maka dapat digunakan sebagai input pada proses klalsfikasinya

5. Proses Algoritma *K-Nearest Neighbour*

Pada tahapan ini proses klasifikasi ditentukan dengan mengitung kemiripan menggunakan *Cosine Similarity*, lalu menentukan kelas nya berdasarkan banyaknya data sejumlah k disekitarnya(Septian et al., 2019).

Untuk menghitung jarak antara fitur dalam *K-Nearest Neighbor*, digunakan perhitungan *Cosine Similarity*. Perhitungan *Cosine Similarity* yang digunakan untuk menghitungnya terdapat pada persamaan (3).

$$\text{CosSim}(x, dj) = \frac{\sum_i^m = \text{xi.dji}}{\sqrt{(\sum_i^m = 1 x^2)} \times \sqrt{(\sum_i^m = 1 dji^2)}} \qquad (3)$$

Keterangan :

x : Dokumen Uji

dj : Dokumen Latih

xi : Bobot setiap term pada dokumen Uji

dji : Bobot setiap term pada dokumen latih

Langkah – langkah proses untuk menggunakan *K-Nearest Neighbour* adalah sebagi berikut :

a. Menghitung *Cosine Similarity* dengan rumus (3).
b. Mengurutkan data *Cosine Similarity.*
c. Menentukan jarak klasifikasi terdekat

Langkah-langkah untuk menghitung metode *K-Nearest Neighbour* antara lain:

1. Menentukan *K* (Jarak tetangga terdekat).
2. Menghitung jarak antara data yang diuji dengan dataset sebelumnya. Pada tahap ini menghitung jarak data yang ingin diprediksi dihitung dengan semua dataset sebelumnya.
3. Mengurutkan nilai jarak atau me-ranking berdasarkan *K* ke dalam kelompok yang mempunyai nilai *Cosine* terkecil (mengurutkan hasil jarak secara dari terkecil ke terbesar).
4. Mengumpulkan kategori Y (Klasifikasi *Nearest Neighbour*) berdasarkan nilai *K* atau ambil data tetangga terdekat setelah diurutkan sebelumnya
5. Langkah selanjutnya, penentuan klasifikasi. Dari hasil training, akan diperoleh data yang mendominasi

6. Proses Algoritma *Improved K-Nearest Neighbour*

Pada tahapan ini proses klasifikasi dengan malakukan perhitungan *cosine similarity* terlebih dahulu, setelah itu menetukan nilai K pada setiap kategorinya. Setelah itu menghitung probablitas data uji pada kelas kategori(Andriani & Tanzil Furqon, 2019).

Langkah – Langkah untuk menggunakan klasifikasi *Improved K-Nearest Neighbour* adalah sebagai berikut

1. Melakukan vectorisasi pada data, pada penelitian ini menggunakna TF-IDF untuk mencari bobot kata.
2. Setelah mendapatkan hasil vector, dilakukan perhitungan *cosine similarity.* Perhitungan *cosine similarity* ada pada persamaan (3).

$$\text{CosSim}(x, dj) = \frac{\sum_i^m = \text{xi.dji}}{\sqrt{(\sum_i^m = 1 x^2)} \times \sqrt{(\sum_i^m = 1 dji^2)}} \qquad (3)$$

Keterangan :

x : Dokumen Uji

dj : Dokumen Latih

xi : Bobot setiap term pada dokumen Uji

dji : Bobot setiap term pada dokumen latih

3. Setelah mengetahui nilai *cosine similarity* maka dilakukan pengurutan nilai mulai dari yang terbesar hingga nilai yang terkecil. Nilai dengan *cosine similarity* yang besar menandakan kedua dokumen memiliki kemiripan.

4. Selanjutnya dilakukan pengelompokan hasil *cosine similarity* sesuai dengan katergori.
5. Melakukan penetapan *k-values* yang baru pada setiap kategori. Perhitungan yang akan digunakan untuk menghitungnya pada persaman (4).

$$n = \left[\frac{k*N\ (c_m)}{mak\{N(c_m)|j = 1 \ldots . N_c\}}\right] \qquad (4)$$

Keterangan:

n : nilai *k-values* baru.

K : nilai *k-values* yang ditetapkan di awal.

$N(c_m)$: jumlah dokumen/data latih pada kategori *m.*

$mak\{N(c_m)|j = 1 \ldots . N_c\}$: jumlah dokumen / data latih terbanyak pada semua kategori yang ada

6. Kemudian hitung perbandingan kemirpan dokumen uji dengan dokumen latih pada setiap ketegori, lalu jumlahkan nilai similaritas sebanyak top n tetangga yang termasuk dalam suatu kategori.

$$\sum_{d\ j \in top\ n\ KNN(Cm)} sim(x, dj) y(dj. Cm) \qquad (5)$$

$$\sum_{d\ j \in top\ n\ KNN(Cm)} sim(x, dj) \qquad (6)$$

7. Lalu hitung nilai maksimum perbandingan antara kemiripan data uji dengan dokumen latih sebanyak top n tetangga pada suatu kategori kategori dengan kemiripan data uji dengan dokumen latih sebanyak top n tetangga pada data latih.

$$P(x, c_m) = argmax_m \frac{\sum_{dj \in top\ n\ kNN\ (c_m)} sim(x, d_j) y(d_j. c_m)}{\sum_{dj \in top\ n\ kNN\ (c_m)} sim(x, d_j)} \qquad (7)$$

Keterangan :

$p(x, c_m)$: probabilitas dokumen *X* menjadi anggota kategori c_m.

$sim(x, d_j)$: kemiripan antara dokumen *X* dengan dokumen latih d_j.

$top\ n\ kNN$: *top n* tetangga

$y(d_j. c_m)$: fungsi atribut dari kategori tertentu yang memenuhi persamaan (9).

$$y(d_j. c_m) = \left\{\frac{1, d_j \in c_m}{0, d_j \in c_m}\right\} \qquad (8)$$

Berdasarkan perhitungan pada persamaan (7) maka akan diketahui nilai probabilitas terbesar. Hasil perhitungan probabilitas tersebut yang akan menjadi penentuan terhadap dokumen uji(Putriani, Umbara, & Sabrina, 2022).

7. Evaluasi

Dalam tahap ini Informasi keluaran yang didapatkan dari sebuah proses data mining harus diperlihatkan dalam bentuk yang umum dan dapat dipahami yang terkait. Pada tahapan ini akan mencakup pemeriksaan terhadap pola informasi jika bertentangan dengan fakta atau hipotesis yang sebelumnya. Pada penelitian ini menggunakan evaluasi *Confusion Matrix.*

HASIL DAN PEMBAHASAN

Analisis Perolehan Data

Perolehan data terkait Analisis Sentiman Moda Raya Terpadu Jakarta ini diperolah dari hasil *crawling* data Twitter. Dengan kata kunci pencarian "MRTJakarta" dataset yang diperoleh berjumlah 1500 data. Crawling data ini dilakukan pada tangal 29 Mei 2023, jadi dataset berisi opini mengenai MRT Jakarta ini hanya sampai tanggal 29 Mei 2023. Contoh data yang telah di *crawling* terdapat pada table

Tabel 1 Perolehan Data

No	Teks
1	informasi cara membeli tarif tiket MRT Jakarta baru
2	Min @mrtjakarta saran aja sih kalo bisa mushola untuk pria wanita dipisah tempatnya. Udah beberapa kali solat di MRT bundaran HI hai kak berangkat telat hari ini dari stasiun mrt senayan ke stasiun mrt bundaran hai jam 08.00 pagi kami menghimbau anda untuk datang semenit lebih awalweekdays masih aman ga terlalu numpuk, cuma kalo di weekend kaya gini numpuk bgt ditambah pria wanita di gabung gitu tempatnya.
3	Kartu MRT Jakarta keren desiainnya bagus dan lucuu @mrtjakarta
4	hai kak, terima kasih atas masukan dan sarannya. Tolong sebutkan ini sebagai motivasi kami untuk tingkat layanan yang lebih baik di masa depan, terima kasih
5	keran mesin mrt mah Bos teknologi Jepang mampu membuat sat set sat setn yang tidak mampu kartu bank elektronik yang banyak mohon ampun dan kami terjebak pada sistem itu dan bahkan tidak berbicara tentang menggunakan kode qr aplikasi seluler untuk tiket mereka lambat

Analisi Pelabelan Data

Pelabelan data ini merupakan tahapan yang berfungsi untuk memberi label atau kelas pada data agar bisa dilakukan proses selanjutnya. Pada penelitian ini pelabelan yang dugunakan menggunkan *Lexicon Based* dengan tiga kelas yaitu positif, negatif, dan netral. Kamus yang digunakan yaitu kamus kata positif dan kamus kata negatif dalam bahasa Indonesia. Contoh dataset yang telah diberikan label atau kelas terdapat pada tabel

Tabel 2 Pelabelan Data

No	Teks	Label
1	informasi cara membeli tarif tiket MRT Jakarta baru	Netral
2	Min @mrtjakarta saran aja sih kalo bisa mushola untuk pria wanita dipisah tempatnya. Udah beberapa kali solat di MRT bundaran HI weekdays masih aman ga terlalu numpuk, cuma kalo di weekend	Netral

	kaya gini numpuk bgt ditambah pria wanita di gabung gitu tempatnya.	
3	Kartu MRT Jakarta keren desiainnya bagus dan lucuu @mrtjakarta	Positif
4	hai kak, terima kasih atas masukan dan sarannya. Tolong sebutkan ini sebagai motivasi kami untuk tingkat layanan yang lebih baik di masa depan, terima kasih	Positif
5	keran mesin mrt mah Bos teknologi Jepang mampu membuat sat set sat setn yang tidak mampu kartu bank elektronik yang banyak mohon ampun dan kami terjebak pada sistem itu dan bahkan tidak berbicara tentang menggunakan kode qr aplikasi seluler untuk tiket mereka lambat	Negatif

Preprocessing Data

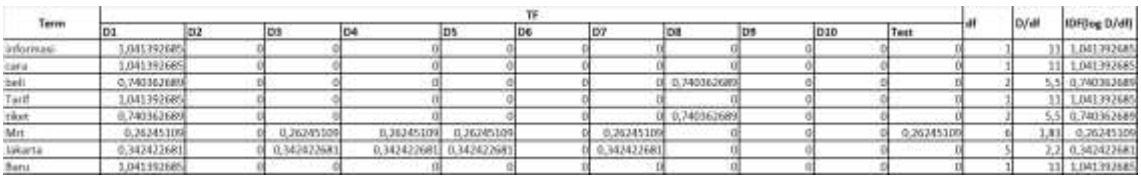
Preprocessing data merupakan teknik awal data mining untuk mengubah data mentah yang dikumpulkan dari berbagai sumber menjadi informasi yang lebih bersih dan bisa digunakan untuk pengolahan selanjutnya. Berikut ini merupakan hasil preprocessing pada penelitian ini.

Tabel 3 Preprocessing Data

No	Teks	Label
1	informasi cara beli tarif tiket mrt jakarta baru	Netral
2	saran aja kalau bisa mushola untuk pria wanita pisah tempat udah berapa kali solat stasiun mrt bundaran hi hari kerja masih aman tidak terlalu numpuk kalau akhir pekan numpuk banget tambah pria wanita gabung tempat	Netral
3	kartu mrt jakarta keren desain bagus lucu	Positif
4	terima kasih atas masukan saran tolong sebut ini sebagai motivasi kami untuk tingkat layanan lebih baik masa depan terima kasih	Positif
5	mesin mrt teknologi jepang mampu buat tidak mampu kartu bank elektronik banyak mohon ampun kami jebak pada sistem bahkan tidak bicara tentang menggunakan kode qr aplikasi seluler untuk tiket mereka lambat	Negatif

Pembobotan Kata TF-IDF

Setelah melakukan preprocessing data maka Langkah selanjutnya yaitu melakukan pembobotan kata menggunakan TF-IDF, dibawah ini merupakan sample hasil dari pembobotan kata.



Term	TF											df	D/df	IDF(log D/df)
	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	Test			
informasi	1,041392685	0	0	0	0	0	0	0	0	0	0	1	11	1,041392685
cara	1,041392685	0	0	0	0	0	0	0	0	0	0	1	11	1,041392685
beli	0,740362689	0	0	0	0	0	0	0,740362689	0	0	0	2	5,5	0,740362689
Tarif	1,041392685	0	0	0	0	0	0	0	0	0	0	1	11	1,041392685
tiket	0,740362689	0	0	0	0	0	0	0,740362689	0	0	0	2	5,5	0,740362689
Mrt	0,262451099	0	0,262451099	0,262451099	0,262451099	0	0,262451099	0	0	0	0,262451099	6	1,83	0,262451099
Jakarta	0,342422681	0	0,342422681	0,342422681	0,342422681	0	0,342422681	0	0	0	0	5	2,2	0,342422681
Baru	1,041392685	0	0	0	0	0	0	0	0	0	0	1	11	1,041392685

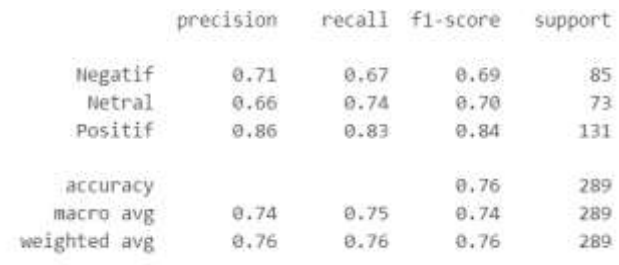
Gambar 2 Hasil Perhitungan TF-IDF

Proses Klasifikasi K-Nearest Neighbour

Pada Proses klasifikasi K-Nearest Neighbour ini menggunakan python yang menggunakan library. Terdapat beberapa eksperimen hasil pada penelitian ini, yaitu menggunakan dataset yang balance dan non balance, lalu eksperiman splitting data 90:10, 80:20, 70:30, 60:40, dan 50:50. Kemudian eksperiman penggunaan nilai K pada klasifikasi K-Nearest Neighbour ini. Hasil tertinggi didapatkan pada Dataset Balance dengan Splitting Data 80:20 dengan K = 17

Tabel 4 Hasil K-Nearest Neighbour

K-Values	K-Nearest Neighbour
K = 5	Accuracy: 68.86% Precision: 0.69 Recall: 0.69 F1 Score: 0.69
K = 10	Accuracy: 71.97% Precision: 0.73 Recall: 0.72 F1 Score: 0.72
K = 13	Accuracy: 73.36% Precision: 0.74 Recall: 0.73 F1 Score: 0.73
K = 15	Accuracy: 73.70% Precision: 0.74 Recall: 0.74 F1 Score: 0.74
K = 17	Accuracy: 76.12% Precision: 0.76 Recall: 0.76 F1 Score: 0.76



	precision	recall	f1-score	support
Negatif	0.71	0.67	0.69	85
Netral	0.66	0.74	0.70	73
Positif	0.86	0.83	0.84	131
accuracy			0.76	289
macro avg	0.74	0.75	0.74	289
weighted avg	0.76	0.76	0.76	289

Gambar 3 Hasil Klasifikasi K-Nearest Neighbour

Proses Klasifikasi Improved K-Nearest Neighbour

Pada Proses klasifikasi Improved K-Nearest Neighbour ini menggunakan python yang menggunakan library. Terdapat beberapa eksperimen hasil pada penelitian ini, yaitu menggunakan dataset yang balance dan non balance, lalu eksperiman splitting data 90:10, 80:20, 70:30, 60:40, dan 50:50. Kemudian eksperiman penggunaan nilai K pada klasifikasi K-Nearest Neighbour ini. Hasil tertinggi didapatkan pada Dataset Balance dengan Splitting Data 90:20 dengan K = 15

Tabel 5 Hasil Improved K-Nearest Neighbour

K-Values	Improved K-Nearest Neighbour
K = 5	Accuracy: 67.59% Precision: 0.69 Recall: 0.68 F1 Score: 0.66
K = 10	Accuracy: 72.41% Precision: 0.73 Recall: 0.72 F1 Score: 0.72
K = 13	Accuracy: 73.79% Precision: 0.74 Recall: 0.74 F1 Score: 0.74
K = 15	Accuracy: 77.24% Precision: 0.77 Recall: 0.77 F1 Score: 0.77
K = 17	Accuracy: 75.86% Precision: 0.76 Recall: 0.76 F1 Score: 0.76

	precision	recall	f1-score	support
Negatif	0.77	0.61	0.68	49
Netral	0.69	0.73	0.71	33
Positif	0.82	0.92	0.87	63
accuracy			0.77	145
macro avg	0.76	0.75	0.75	145
weighted avg	0.77	0.77	0.77	145

Gambar 4 Hasil Klasifikasi Improved K-Nearest Neighbour

KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan menggunakan metode *K-Nearest Neighbour* dan *Improved K-Nearest Neighbour* dengan 1500 data dengan beberapa eksperimen. Dapat disimpulkan dari eksperiman dataset yang tidak *balance* dan yang *balance* di setiap

kelasnya, eksperimen nilai K pada *splitting* data 90 : 10 , *splitting* 80 : 20 . *splitting* 70 : 30, *splitting* data 60 : 40 , dan *splitting* data 50 : 50, secara umum menunjukan bahwa peningkatan akurasi metode *Improved K-Nearest Neighbour* terhadap *K-Nearest Neighbour* pada kasus analisis sentiment moda raya terpadu tidak signifikan. Hasil penelitian menunjukkan bahwa akurasi tertinggi pada metode *Improved K-Nearest Neighbour* terdapat pada Dataset *balance* dengan nilai K = 15 pada *splitting* data 90 : 10 yang memiliki akurasi sebesar 77.24%, dengan *precission* sebesar 0.77, *Recall* sebesar 0.77, dan F1 *Score* sebesar 0.77. Sedangkan metode *K-Nearest Neighbour* memiliki akurasi tertinggi pada dataset *balance* dengan nilai K = 17 pada *splitting* data 80 : 20 dengan akurasi sebesar 76.12%, dengan *precission* sebesar 0.76, *Recall* sebesar 0.76, dan F1 *Score* sebesar 0.76.

DAFTAR PUSTAKA

Agustina, Dina, & Rahmah, Fitri. (2022). *Analisis Sentimen pada Sosial Media Twitter terhadap MRT Jakarta Menggunakan Machine Learning*. *2*, 1–6.

Amardita, Rizki Syafaat, Adiwijaya, Adiwijaya, & Purbolaksono, Mahendra Dwifebri. (2022). Analisis Sentimen terhadap Ulasan Paris Van Java Resort Lifestyle Place di Kota Bandung Menggunakan Algoritma KNN. *JURIKOM (Jurnal Riset Komputer)*, *9*(1), 62. https://doi.org/10.30865/jurikom.v9i1.3793

Andriani, Desy, & Tanzil Furqon, Muhammad. (2019). *Peringkasan Teks Otomatis Pada Artikel Berita Hiburan Berbahasa Indonesia Menggunakan Metode BM25*. *3*(3), 2548–2964. Retrieved from http://j-ptiik.ub.ac.id

Arafah, Siti Nur, & Fathoni, Fathoni. (2022). Sentiment Analysis Pada Masyarakat Terhadap LRT Kota Palembang Menggunakan Metode Improved K-Nearest Neighbor. *Jurnal Media Informatika Budidarma*, *6*(3), 1554. https://doi.org/10.30865/mib.v6i3.4434

Hendrastuty, Nirwana, Rahman Isnain, Auliya, Yanti Rahmadhani, Ari, Styawati, Styawati, Hendrastuty, Nirwana, Isnain, Auliya Rahman, Rahman Isnain, Auliya, Yanti Rahmadhani, Ari, Styawati, Styawati, Hendrastuty, Nirwana, & Isnain, Auliya Rahman. (2021). Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja Pada Twitter Dengan Metode Support Vector Machine. *Jurnal Informatika: Jurnal Pengembangan IT*, *6*(3), 150–155. Retrieved from http://situs.com

Indriya Dewi Onantya, Indriati, Putra Pandu Adikara. (2019). Analisis Sentimen Pada Ulasan Aplikasi BCA Mobile Menggunakan BM25 Dan Improved K-Nearest Neighbor. *J-Ptiik.Ub.Ac.Id*, *3*(3), 2575–2580. Retrieved from http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/4754

Pambudi, Andi Setyo, & Hidayati, Sri. (2020). Analisis Perilaku Sosial Pengguna Moda Transportasi Perkotaan: Studi Kasus Mass Rapid Transit (MRT) DKI Jakarta. *Bappenas Working Papers*, *3*(2), 143–156. https://doi.org/10.47266/bwp.v3i2.74

Putriani, Nelsih, Umbara, Fajri Rakhmat, & Sabrina, Puspita Nurul. (2022). *Analisis Sentimen pada Aplikasi PeduliLindungi dengan Menggunakan Metode Improved K-Nearest Neighbor dan Lexicon Based*. *8*(1), 350–364.

Septian, Jeremy Andre, Fachrudin, Tresna Maulana, & Nugroho, Aryo. (2019). Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor. *Journal of Intelligent System and Computation*, *1*(1), 43–49. https://doi.org/10.52985/insyst.v1i1.36

Syakur, Abdus. (2021). Implementasi Metode Lexicon Base Untuk Analisis

Sentimen Kebijakan Pemerintah Dalam Pencegahan Penyebaran Virus Corona Covid-19 Pada Twitter. *Jurnal Ilmiah Informatika Komputer*, *26*(3), 247–260. https://doi.org/10.35760/ik.2021.v26i3.4720