BAB II
TINJAUAN PUSTAKA

2.1 Studi Pustaka

Dalam penelitian ini penelitian sejenis terdahulu yang dipakai oleh penulis sebagai bahan untuk referensi sekaligus untuk mengintopeksi penelitian yang akan dilakukan. Berikut beberapa tinjauan studi sebelumnya yang menggunakan metode *K-Nearest Neighbor*:

Tabel 2.1 Studi Literatur

No	Permasalahan	Metode Penelitian	Hasil Penelitian
1.	Ramainya pemilihan Gubernur DKI Jakarta yang menjadi trending topik pada media sosial twitter. Penelitian dibuat untuk menganalisis sentimen pengguna twitter terhadap setiap kandidat pilgub.	Metode yang digunakan adalah (K-NN) dengan pembobotan TF-IDF dan fungsi *cosine similarity*	Dalam penelitian tersebut dihasilkan tingkat akurasi terbesar adalah 67,2% dengan nilai k=5. Sedangkan nilai presisi tertinggi sebesar 56,94% saat k=5 dan recall terbesar 78,24 % ketika k=15 (Akhmad Deviyanto, 2018).

2.	Banyaknya judul topik dalam sebuah forum memerlukan sebuah teknik klasifikasi untuk membedakan masing-masing kelasnya.	Penelitian yang dilakukan oleh imdriani menggunggu nakan algoritma klasifikasi Naïve Bayes (NB) dan *K-Nearest Neighbor* (KNN)	Pada penelitian tersebut dalam pengukuran efektivitas Klasifikasi terhadap 15 data uji didapatkan nilai 80% untuk metode k-NN dan sebesar 73% untuk metode NBC. (Indriani, n.d.-a).
3.	Penilaian kinerja dosen untuk meningkatkan kualitas program kerja dosen di lingkungan kampus melalui analisis sentimen mahasiswa.	Algoritma yang digunakan adalah *K-Nearest Neighbor* (KNN) untuk mengukur tingkat sentimen mahasiswa dalam menilai kinerja dosen.	nilai akurasi training sebesar 100%, hasil analisis sentimen memiliki tingkat akurasi sebesar 80%, presisi training bernilai 1 dan testing bernilai 0.909 kemudian hasil recall training bernilai 1 dan hasil recall testing bernilai 0.889 (Permana and Efendi, 2020).

4.	Banyak artikel dengan berbagai macam judul dan metode yang digunakan sama, namun tidak untuk tingkat kemiripannya.	Menggunakan algoritma vector space model dan algoritma k-nearest neghbor untuk membandingkan.	Menghasilkan dokumen dengan tingkat kemiripan tertinggi pada metode VSM yaitu pada Dok 5, Dok 7, Dok 8 dan Dok 4. Sedangkan untuk KNN menghasilkan tingkat kemiripan pada range Doc7,Doc10 |Doc8,Doc10 | Doc4,D10 | Doc5,Doc10 |Doc3,Doc10 (Fauziah et al., 2019).
5.	Penelitian mengenai sentimen terhadap relokasi Ibukota Nusantara	Algoritma yang digunakan pada penelitian ini adalah Naïve Bayes dan K-NN	Dari penelitian tersebut didapatkan tingkat akurasi pada metode NB sebesar 82.27% kemudian nilai Precision sebesar 86.36% dan nilai Recall sebesar 76.93%, sedangkan pada KNN akurasi sebesar 88,12% dengan Precision sebesar 93.98% dan nilai recall sebesar 81.53% (Syahril Dwi Prasetyo, et. Al., 2023).

6.	Untuk mencari sentimen terhadap pelayanan kantor POS	Menggunakan Algoritma *Lexicon Based*	Dari 185 user, ditemukan 318 opini pelanggan, dimana Sentimen Positif sebanyak 99 opini, Sentimen Negatif sebanyak 184 opini, Sentimen Netral sebanyak 32 opini dan Sentimen Nihil sebanyak 3 opini, artinya hanya sebagian kecil dari pelanggan yang merasa puas dengan pelayanan jasa yang diberikan PT. Pos Indonesia (Matresya Matulatuwa et al., 2017).
	Ramainya pengguna GoPay memunculkan banyak opini dari pengguna dari yang senang maupun tidak. Penelitian dilakukan untuk mencari sentimen para pengguna GoPay.	Menggunakan Metode Lexicon Based Dan Support Vector Machine	Hasil dari pelabelan dengan Lexicon Based berjumlah 923 untuk positif dan 287 untuk negatif. Sedangkan klasifikasi metode SVM menggunakan kernel Linear menghasilkan 89,17% dan 84,38% untuk kernel Polynomial (Mahendrajaya et al., 2019).

Dari beberapa penelitian terdahulu di atas yang menjadi pembeda dalam penelitian ini adalah konsep penelitian menggunakan metode SEMMA, obyek penelitian yang diambil yakni isu *marketplace* guru serta metode yang digunakan yaitu algoritma *Lexicon* yang dikombinasikan dengan *K-Nearest Neighbor*.

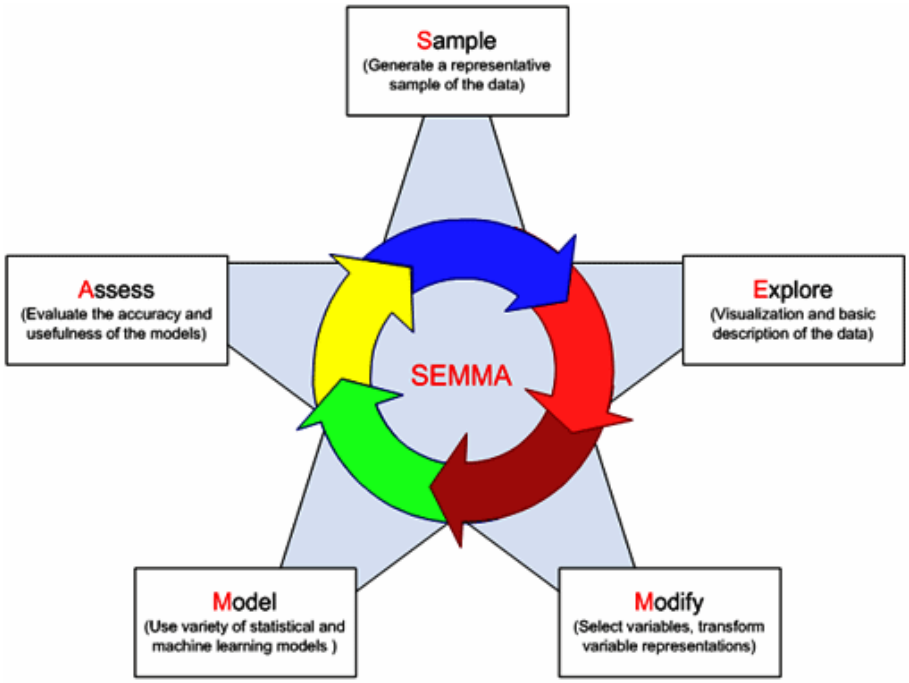
2.2 Analisis Sentimen

Analisis sentimen atau opinion minning (Liu, 2010). Biasa digunakan untuk mendapatkan gambaran persepsi masyarakat terhadap tren isu atau fenomena yang sedang ramai diperbincangkan, baik opini yang cenderung positif maupun negatif ataupun netral. Pendapat umumnya positif atau negatif tetapi juga dapat diklasifikasikan sebagai baik, sangat baik, buruk dan sangat buruk. Analisis sentimen mempunyai kemampuan bagus yang ampuh dalam mengekstraksi pengetahuan dari pengguna media sosial dalam kelas sasaran. Analisis sentimen juga memungkinkan untuk digunakan dalam menganalisis emosi dan suasana hati seseorang dalam melakukan suatu hal. Analisis sentimen dapat menghasilkan apa yang ingin kita ketahui, misalnya tentang mengoreksi kalimat tertentu (Syaifudin dan Irawan, 2018).

Keuntungan analisis sentimen adalah Anda bisa mendapatkan umpan balik mengenai produk, layanan, dan topik lainnya tanpa berkonsultasi langsung dengan audiens. Dengan menambang data dari platform yang tersedia di Internet, pemasok bisa mendapatkan penilaian penting untuk perkembangan mereka sendiri. Pada saat yang sama, kelemahan atau kekurangan dari analisis sentimen adalah adanya kata-kata yang memiliki nilai positif dalam satu situasi dan mungkin memiliki nilai negatif dalam situasi lain. Kemudian, pengguna tidak selalu mengutarakan pendapat yang sama tentang suatu kata. Ada beberapa jenis analisis sentimen, seperti analisis detail, deteksi emosi, analisis berbasis aspek (Asri, 2021).

2.3 Metode SEMMA

SEMMA merupakan singkatan dari *Sample, Explore, Modify, Model and Asses* adalah proses penambangan data yang dikenalkan oleh SAS *Institute* untuk menentukan pola yang belum diketahui dari data yang jumlahnya besar guna keuntungan dalam bisnis. SEMMA memiliki lima tahapan proses yaitu *sample, explore, modify, model, asses* (Azevedo and Santos, 2008). Berikut adalah gambaran lima tahap dalam metode SEMMA:



Gambar 2.1 Metode SEMMA (Poona, 2013)

2.4 Klasifikasi

Klasifikasi adalah sebuah metode yang digunakan untuk mengelompokan sebuah teks untuk kelompok kelas tertentu. Algoritma klasifikasi yang sering dipakai diantaranya Naïve Bayes classifiers, *K-Nearest Neighbor*, *Metode Rule Based*, *Memory Based Reasoning*, dan *Support Vector Machines* (SVM), Analisa Statistik, Algoritma Genetika, *Rough Sets* (Leidiyana 2013).

Tujuan dijalankannya tahapan ini ialah supaya data yang ada dapat diklasifikasikan ke dalam kelas tertentu yang telah disiapkan. Dalam klasifikasi ada dua tugas utama yakni pembangunan model sebagai prototipe untuk disimpan sebagai memori. Penggunaan model tersebut untuk melakukan pengenalan/klasifikasi/ prediksi pada suatu objek data lain agar diketahui di kelas mana objek data tersebut dalam model yang sudah disimpannya (Putro et al., 2020).

Model klasifikasi memiliki arti yang sama dengan *black box*, yaitu terdapat model yang mengambil *input*, kemudian dapat mencerminkan *input* tersebut, dan dapat memberikan jawaban sebagai *output* dari refleksinya. Sebuah sistem yang menjalankan klasifikasi dikatakan mampu mengklasifikasikan semua dataset dengan benar, namun tidak dapat dipungkiri bahwa kinerjanya tidak bisa 100% akurat, sehingga suatu classifier perlu mengukur kinerjanya. Secara umum, pengukurannya dilakukan dengan menggunakan confusion matrix. Confusion matrix adalah tabel pencatat kinerja klasifikasi (Ali Imron, 2019).

Dari penjabaran matriks konfusi terlihat bahwa jumlah data dalam matriks konfusi dapat diringkas menjadi dua nilai yaitu akurasi dan tingkat kesalahan. Berdasarkan jumlah data yang diklasifikasikan dengan benar maka dapat ditentukan tingkat keakuratan hasil prediksi, dan dengan mengetahui jumlah data yang diklasifikasikan salah maka dapat ditentukan tingkat kesalahan prediksi (Prasetyo, 2012).

Dalam menentukan akurasi digunakan rumus formula sebagai berikut:

$$\frac{jumlah\ data\ yang\ diprediksi\ secara\ benar}{jumlah\ prediksi\ yang\ dilakukan} = \frac{f_{11} - f_{00}}{f_{11} + f_{10} + f_{01} + f_{00}}$$

Kemudian, untuk mencari error (kesalahan prediksi) digunakan formula:

$$\frac{jumlah\ data\ yang\ diprediksi\ secara\ salah}{jumlah\ prediksi\ yang\ dilakukan} = \frac{f_{10} - f_{01}}{f_{11} + f_{10} + f_{01} + f_{00}}$$

2.5 Preprocessing

Preprocessing adalah tahapan yang dijalankan di awal penelitian, sebelum melabeli dan mengklasifikasi pada data opini atau sentimen dataset hasil crawling data sebanyak 1006 yang telah terkumpul tentunya belum terstruktur dan membutuhkan tahapan awal yang disebut preprosesing atau tahapan sebelum proses. Preprocessing dalam klasifikasi dokumen digunakan untuk mengindeks kumpulan dokumen. Indeks adalah seperangkat istilah yang menunjukkan konten atau topik yang terkandung dalam dokumen. Penerapan *inverted indeks* harus sesuai dengan konsep pemrosesan bahasa yang bertujuan untuk mengekstraksi istilah-istilah penting dari dokumen yang disajikan sebagai sekumpulan kata (Indriani, n.d.-b). Beberapa langkah dalam preprocessing diantaranya adalah:

1. *Case Folding.*

   Case Folding adalah sebuah tahapan dalam preprocessing yang dijalankan untuk menyamakan karakter di dalam data dimana pada proses *case folding* ini adalah proses mengubah semua huruf menjadi huruf kecil.

   Contoh tahapan *case folding*:

   Sebelum :

   @namamu Demi Keselamatan Pengendara di mlm hari, hrs dipasang Lampu di pinggir jln. Supaya tidak Membahayakan. #lampu

   Sesudah :

   @namamu demi keselamatan pengendara di mlm hari, hrs dipasang lampu di pinggir jln. supaya tidak membahayakan. #lampu

2. *Cleaning.*

   Cleaning membantu mengidentifikasi dan memperbaiki kesalahan yang timbul saat memproses data dari berbagai sumber. Dengan menghilangkan inkonsistensi dan ketidakakuratan, pembersihan data dapat menjamin keandalan kumpulan data. Data yang akan

dipakai harus melewati proses pembersihan data seperti symbol, tautan URL, angka. Data asli dari twitter tentu banyak terdapat data kotor seperti tagar, angka, nama pengguna, URL dan teks retweet.

Contoh tahapan *cleaning*:

Sebelum:

@namamu demi keselamatan pengendara di mlm hari, hrs dipasang lampu di pinggir jln. supaya tidak membahayakan. #lampu

Sesudah:

demi keselamatan pengendara di mlm hari hrs dipasang lampu di pinggir jln supaya tidak membahayakan

3. *Tokenizing*.

   Tokenizing adalah proses memecahkan setiap deretan kata di dalam kalimat menjadi token atau penggalan kata tunggal. Langkah ini juga menghapus beberapa karakter seperti tanda baca dan mengubah semua token menjadi huruf kecil.

   Contoh tahapan *tokenizing*:

   Sebelum:

   demi keselamatan pengendara di mlm hari hrs dipasang lampu di pinggir jln supaya tidak membahayakan

   Sesudah:

   ‘demi’ ‘keselamatan’ ‘pengendara’ ‘di’ ‘mlm’ ‘hari’ ‘hrs’ ‘dipasang’ ‘lampu’ ‘di’ ‘pinggir’ ‘jln’ ‘supaya’ ‘tidak’ ‘membahayakan’

4. *Normalize*

   Normalize adalah proses membakukan kata-kata yang mempunyai makna yang sama dengan cara mengubah ejaan akronim dan tidak baku sehingga mempunyai kesatuan makna.

   Contoh :

   Aplg : apalagi

   Hrs : harus

Skrg : sekarang

Bgt : banget

Sebelum:

‘demi’ ‘keselamatan’ ‘pengendara’ ‘di’ ‘mlm’ ‘hari’ ‘hrs’ ‘dipasang’ ‘lampu’ ‘di’ ‘pinggir’ ‘jln’ ‘supaya’ ‘tidak’ ‘membahayakan’

Sesudah:

‘demi’ ‘keselamatan’ ‘pengendara’ ‘di’ ‘malam’ ‘hari’ ‘harus’ ‘dipasang’ ‘lampu’ ‘di’ ‘pinggir’ ‘jalan’ ‘supaya’ ‘tidak’ ‘membahayakan’

5. *Stopword.*

   Stopword adalah langkah filter untuk setiap kata yang tidak diperlukan pada pemrosesan. Dalam tahapan ini, setiap kata akan diperiksa, apabila mengandung kata hubung, kata depan atau kata ganti maka akan dihilangkan. Contoh :

   Sebelum:

   ‘demi’ ‘keselamatan’ ‘pengendara’ ‘di’ ‘malam’ ‘hari’ ‘harus’ ‘dipasang’ ‘lampu’ ‘di’ ‘pinggir’ ‘jalan’ ‘supaya’ ‘tidak’ ‘membahayakan’

   Sesudah:

   ‘demi’ ‘keselamatan’ ‘pengendara’ ‘malam’ ‘hari’ ‘harus’ ‘pasang’ ‘lampu’ ‘pinggir’ ‘jalan’ ‘supaya’ ‘tidak’ ‘membahayakan’

6. *Stemming.*

   Stemming adalah langkah untuk merubah kata majemuk ke kedalam bentuk dasarnya. *Stemming* juga dibutuhkan untuk mengurangi indeks yang berbeda yang terdapat dalam sebuah data. Stemming juga dipakai untuk mengelompokkan kata-kata lain yang mempunyai akar kata dan makna yang sama, namun bentuknya berbeda karena mempunyai imbuhan yang berbeda (Maulana & Witanti, 2023). Teks yang keluar dalam dokumen seringkali

memiliki banyak variasi. Oleh karena itu, kata tersebut dikembalikan ke bentuk aslinya dengan menghapus atau menghilangkan imbuhan di awal atau di akhir setiap kata. Dengan tahapan ini akan didapatkan kelompok kata yang sesuai dimana kata-kata dalam kelompok tersebut merupakan varian sintaksis satu sama lain dan hanya dapat berisi satu kata dalam setiap kelompok dan dapat dianggap sebagai bentuk lain dari kata tersebut (Indriani, 2014).

Tahapan *stemming* dalam penelitian ini menggunakan algoritma porter dimana proses stemming yang dilakukan adalah lima langkah dengan mensimulasikan *infleksi* dan *derivasi* dari kata. Pada setiap langkah, sebuah akhiran tertentu dihapus dengan cara subtitusi (Arie Atwa Magriyanti, n.d.). Lima langkah dalam algoritma porter adalah sebagai berikut (Agusta et al., 2009):

1. Menghilangkan setiap akhiran dalam partikel
2. Menghilangkan akhiran dalam kepemilikan
3. Mengilangkan awalan pertama dalam kata. Jika tidak ada bisa melanjutkan ke langkah 4a, jika ada maka dicari kemudian melanjutkan langkah ke 4b.
4. a.Hilangkan awalan yang kedua, kemudian melanjutkan ke langkah 5a.

   b.Hilangkan akhiran, jika tidak ditemukan maka kata tersebut diartikan sebagai kata dasar. Jika ditemukan maka lanjutkan ke langkah 5b.
5. a.Menghilangkan akhiran. Kemudian kata akhir diasumsikan sebagai kata dasar

   b.Hapus awalan kedua. Kemudian kata akhir diasumsikan sebagai kata dasar.

Setelah melewati lima tahapan diatas maka akan didapatkan kata yang telah bersih dari imbuhan awalan dan akhiran sebagai berikut:

Sebelum:

‘demi’ ‘keselamatan’ ‘pengendara’ ‘malam’ ‘hari’ ‘harus’ ‘pasang’ ‘lampu’ ‘pinggir’ ‘jalan’ ‘supaya’ ‘tidak’ ‘membahayakan’

Sesudah:

‘demi’ ‘selamat’ ‘pengendara’ ‘malam’ ‘hari’ ‘harus’ ‘pasang’ ‘lampu’ ‘pinggir’ ‘jalan’ ‘supaya’ ‘tidak’ ‘bahaya’

2.6 Pembobotan Kata

Pemberian bobot pada setiap kata (istilah) berfungsi sebagai pembobotan pada setiap kata (istilah) yang ada di dalam dokumen teks yang nantinya akan diolah (Deolika & Taufiq Luthfi, 2019). Langkah-langkah menghitung bobot kata adalah sebagai berikut:

1. Term Frekuensi

Term Frekuensi adalah frekuensi suatu kata yang muncul dalam dokumen teks. Term frekuensi (*tf t,d*) dijabarkan sebagai banyaknya kemunculan istilah t dalam dokumen d (Tjipta et al., 2013). Persamaan frekuensi suku (tf t,d) ditunjukkan pada Persamaan berikut:

$$W_{tf_{td}} = \begin{cases} 1 + log_{10}\, tf_{t,d}\,, if\ tf_{t,d} > 0 \\ 0, \qquad\qquad lainnya \end{cases}$$

Keterangan:

$tf_{t,d}$ adalah jumlah kemunculan termtpada dokumen d.

2. *Invers Document Frequency*

Invers Document Frequency adalah frekuensi istilah yang muncul pada semua dokumen teks. Istilah-istilah yang jarang muncul pada seluruh dokumen teks mempunyai nilai *Invers Document Frequency* yang lebih tinggi dibandingkan istilah-istilah yang sering muncul (Ni’mah & Arifin, 2020). Persamaan frekuensi dokumen terbalik (IDF) ditunjukkan pada persamaan berikut:

$$idf_t = log_{10}\left(\frac{N}{df_{(t)}}\right)$$

Penjelasan:

N adalah banyaknya dokumen teks.

df_t adalah banyaknya dokumen yang mengandung *term t*.

3. *Term Frequency -Invers Document Frequency* (TF-IDF)

   Nilai TF-IDF suatu kata adalah gabungan antara nilai tf dan nilai idf pada perhitungan bobot (Refka Muhammad F, 2020). Persamaan TF-IDF dijelaskan pada rumus berikut:

$$W_{t,d} = W_{tf_{t,d}} \times idf_t$$

Keterangan :

$W_{tf_{t,d}}$ merupakan *Term Frequency*.

idf_t merupakan *Invers Document Frequency*.

2.7 *Lexicon Based*

Lexicon based merupakan metode berbasis kamus dengan tujuan untuk mendapatkan bobot kalimat pada data set untuk dapat mengidentifikasi label kelas sentimen pada data set. Leksikon berdasarkan hipotesis bahwa orientasi afektif kontekstual merupakan penjumlahan dari penyesuaian afektif setiap kata maupun pendapat. Metode leksikal bisa dipakai untuk mengekstrak emosi dari blog dengan menggabungkan pengetahuan leksikal dan klasifikasi teks. Metode leksikal bisa dibuat dengan cara otomatis maupun manual dengan memperluas dari kata aslinya (Matresya Matulatuwa et al., 2017).

Keuntungan menggunakan metode ini adalah pelabelan kalimat dapat dilakukan secara otomatis sehingga akan menghemat waktu jika data yang diolah merupakan kumpulan data yang besar. Selain itu, pemberian kesan dengan cara ini juga menghindari bias terkait pendapat pribadi (Azhar, 2019).

Kamus merupakan komponen penting dalam sistem kosakata. Kamus digunakan untuk menormalkan kalimat dan mengekstrasi kata kunci. Dibawah ini adalah contoh kamus positif, negatif beserta isinya:

1. Kata kunci positif: baik, bagus, bisa, ok, cepat, akurat, aman, senang.
2. Kata kunci negatif acuh, ambigu, bodoh, gagal, abnormal, susah, lambat.
3. Kata kunci penyangkalan: seharusnya, bukan, tidak.

2.8 Wordcloud

Wordcloud adalah teknik visualisasi data yang memberikan gambaran umum konten kepada pengguna dalam bentuk kumpulan teks. Semakin besar ukuran font suatu kata, semakin sering kata tersebut ditampilkan dan pentingnya kata tersebut. Sebelum membuat word cloud, Anda harus menghilangkan stop word agar dapat menampilkan informasi yang benar. Manfaat menggunakan Wordcloud adalah untuk memberikan pemahaman penuh tentang suatu ide atau informasi pada waktu tertentu. Wordcloud juga populer karena kemudahan visualisasi datanya (Tessem, 2015).

2.9 Algoritma *K-Nearest Neighbor*

Algoritma adalah proses komputasi yang dimaknai dengan baik yang mengambil nilai tertentu sebagai input dan menghasilkan nilai yang disebut output. Algoritma adalah prosedur yang terdiri dari tahapan-tahapan guna menyelesaikan suatu masalah. Algoritme adalah alur

pemikiran untuk menyelesaiakan suatu pekerjaan, disajikan dalam bentuk teks yang dapat dimengerti banyak orang (Muchamad Hendrix P, 2018).

Algoritma merupakan jantungnya ilmu komputer atau ilmu komputer. Istilah algoritma mencakup banyak bidang ilmu komputer, seperti algoritma untuk merutekan pesan di dalam sistem komputer, algoritma Bresenham untuk menggambar garis lurus, algoritma KMP untuk menemukan pola dalam teks, dll. (Budiyanto, 2003).

Algoritma *K-Nearest Neighbor* merupakan algoritma yang dibuat guna mengklasifikasikan objek baru berdasarkan atribut dan sampel pelatihan. Apabila hasil data uji baru diurutkan berdasarkan kelas diatas mayoritas K-NN. Algoritma *K-Nearest Neighbor* memakai klasifikasi dari lingkungan sebagai nilai prediksi sampel data uji baru. K-NN merupakan metode clustering yang tugasnya mengelompokkan data baru berdasarkan jarak data baru tersebut dari beberapa data/tetangga terdekat. (Heri Kurniawan, 2017). Berikut pemaparan tentang algortima k-NN:

a. Menghitung jarak data uji dengan data latih yang telah dibangun menggunakan persamaan Cosine Similirity.
b. Memutuskan parameter nilai k adalah jumlah tetanggaan terdekat.
c. Mengurutkan jarak dri yang terkecil di dalam data sample
d. Memasangkan kategori yang sesuai
e. Mencari tetanggaan terdekat yang jumlahnya terbanyak. Kemudian menetapkan kategorinya.

Rumus *euclidean distance* sebagai berikut:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Keterangan :

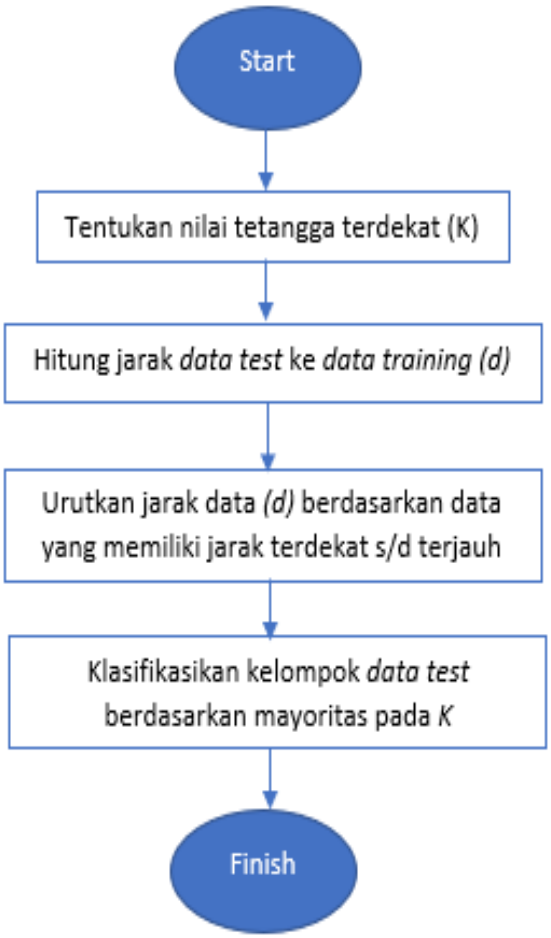
d = jarak *euclidean* data $x_1 y_1$ ke data $x_2 y_2$

$x_1 y_1$ = data ke-1

$x_2 y_2$ = data ke-2

Kelebihan algoritma *K-Nearest Neighbor* adalah salah satu algoritma dalam pembelajaran non-parametrik, jalur keputusan kelas yang dihasilkan oleh model ini bisa sangat fleksibe. Kedua, K-NN tahan terhadap noise dari data sampel pelatihan dan menunjukkan konsistensi yang tinggi. K-NN juga sangat efektif bila digunakan untuk data dalam jumlah besar.

Kelemahan algoritma K-NN terletak pada peneliti harus menentukan sendiri parameter k (bilangan tetangga terdekat). K-NN juga tidak secara implisit menangani nilai yang hilang dan sangat sensitif terhadap outlier (Anggi PY and Darwis, 2021). Berikut adalah *flowcart* metode *K-Nearest Neighbor*:



Gambar 2.2 flowcart algoritma K-NN

2.10 *K-fold Validation*

K-fold validation merupakan metode tahapan dalam memprediksi kesalahan dalam kinerja model. Pada validasi silang yang disebut estimasi bergantian, caranya dengan membagi data menjadi *k* himpunan bagian yang mempunyai ukuran sekiranya sama, model klasifikasi yang digunakan kemudian dilatih dan diuji sebanyak *k* kali. Pada setiap iterasi, satu subset akan digunakan sebagai data pengujian dan k subset data sisanya akan digunakan sebagai data pelatihan (Nurhayati, 2014).

Validasi K-Fold merupakan sebuah metode yang dipakai dalam menentukan rata-rata dalam keberhasilan suatu sistem menggunakan redundansi dengan mengacak properti input sehingga sistem yang diuji pada beberapa properti input dimasukkan secara acak. Validasi silang *K-Fold* dimulai dengan membagi data sebanyak yang diinginkan. Pada proses validasi silang, data akan dibagi menjadi n partisi dengan ukuran yang sama, variabel data ke-1, variabel data ke-2, variabel data ke-3... Kemudian, pelatihan proses pengujian dan pelatihan dilakukan sebanyak *n* kali. Pada iterasi ke-*i*, *i* akan menjadi data uji dan sisanya menjadi data latih. Untuk menggunakan tingkat validasi terbaik dalam pengujian validitas, sebaiknya gunakan validasi silang 5 kali lipat pada model. Contoh pemisahan kumpulan data dalam prosedur validasi silang 5 kali lipat.

Tabel 2.2 contoh pembagian dataset pada *5-fold validation*

Validasi ke-n	Pembagian dataset				
1					■
2				■	
3			■		
4		■			
5	■				

Performa dari k-fold validation sebagai berikut:

1. jumlah seluruh instance dibagi menjadi *N* bagian
2. *Fold* ke-1 waktu bagian 1 menjadi data uji selanjutnya sisanya menjadi data latih. Selanjutnya berdasarkan informasi tersebut, kemudian menghitung keakuratan, kemiripan atau kedekatan dari hasil pengukuran dengan angka atau data sebenarnya.. Hitung keakuratannya menggunakan persamaan berikut:

$$Akurasi = \frac{\sum data\ uji\ klasifikasi\ benar}{\sum total\ data\ uji} \times 100$$

3. *Fold* ke-2, yang kedua terjadi saat bagian kedua menjadi data uji kemudian sisanya menjadi data latih. Kemudian menghitung keakuratannya menurut data tersebut.
4. Begitu seterusnya sampai *fold* ke-*k* selesai dicapai. Setelah itu menghitung presisi rata-rata dari presisi *k* di atas. Rata-rata akurasi ini yang akan menjadi presisi akhir.

2.11 Evaluasi

Tes ini akan menghasilkan data dari pengklasifikasi yang mengkuantifikasi emosi positif, netral dan negatif yang nantinya akan diuji memakai metode Confusion Matrix. Matriks konfusi adalah tahap pengujian presisi yang menghitung nilai presisi, recall, dan nilai presisi gradasi menggunakan persamaan menggunakan rumus.(Widhi S and Wulan S., 2019). Berikut adalah rumus confusion matrix untuk menghitung nilai tingkat akurasi dan error rate:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

Keterangan:

TP (*True* Positif): banyaknya perkiraan data positif yang benar

TN (*True* Negatif): banyaknya nilai perkiraan data negatif yang benar

FP (*False* Positif): banyaknya data negatif tetapi diperkirakan sebagai data yang tidak relevan

FN (*False* Negatif): banyaknya data positif namun diperkirakan sebagai data yang tidak relevan.