BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

Udara adalah elemen penting yang diperlukan oleh semua makhluk hidup di bumi. Udara memiliki fungsi sebagai penopang kehidupan bagi semua makhluk di ekosistem, memungkinkan kelangsungan hidup yang optimal di permukaan bumi. Komposisi udara mencakup berbagai elemen, termasuk senyawa gas dan partikulat yang dapat berwujud padat, cair, atau bersifat tersuspensi di udara. Namun, saat ini kualitas udara semakin menurun dikarenakan adanya polusi udara yang disebabkan oleh aktivitas manusia contohnya emisi gas buang, limbah pabrik, dan pembakaran bahan bakar fosil. Untuk memonitor kualitas udara, dibutuhkan sebuah sistem klasifikasi udara yang akurat dan efisien. Untuk memperjelas ruang lingkup penelitian ini klasifikasi udara menggunakan *machine learning* (Jongdi J, 2022).

Prediksi adalah metode yang digunakan sebagai teknik untuk membuat model klasifikasi berdasarkan data pelatihan yang diberikan. Proses klasifikasi melibatkan analisis data input dan pembentukan model dengan mendefinisikan kelas-kelas data. Ini merupakan salah satu teknik dalam *machine learning* yang bertujuan untuk mengelompokkan data ke dalam kategori atau kelas tertentu. Klasifikasi dalam *Machine learning* adalah sebuah metode atau teknik dalam pembelajaran mesin yang digunakan untuk mengklasifikasikan atau memprediksi. Dalam penelitian ini, pendekatan klasifikasi yang terapkan adalah *Ensemble learning*, yang menggabungkan kekuatan dari dua algoritma, yaitu *K-Nearest Neighbor*, *Support Vector Machine* dan *Random forest*. Pendekatan ini memanfaatkan kombinasi kemampuan masing-masing algoritma untuk meningkatkan performa dan akurasi dalam melakukan klasifikasi pada data yang dianalisis.

KNN merupakan metode dalam *supervised learning* yang sampel uji baru di prediksi berdasarkan kelas terbanyak pada K tetangga terdekat. Tujuannya adalah memprediksi nilai baru berdasarkan variable dan sampel latih, tanpa menggunakan model yang cocokkan, melainkan berdasarkan memori. Ketika

diberikan titik uji, K objek (titik pelatihan) terdekat dengan titik uji akan diidentifikasi. Klasifikasi dilakukan dengan menghitung mayoritas *voting* dari klasifikasi K objek tersebut. Algoritma K-NN menggunakan prinsip ketetanggaan untuk memprediksi nilai sampel uji baru, di mana kedekatan atau jarak tetangga umumnya diukur menggunakan jarak *Euclidean*. (Sarereake, 2019)

Klasifikasi KNN dalah metode non-parametrik sederhana yang efektif untuk klasifikasi. Meskipun algoritmanya sederhana, kinerjanya terbukti sangat baik, menjadikannya metode standar yang penting. Klasifikasi K-NN memerlukan metrik dan parameter integer positif K. Algoritma ini mempertahankan posisi sampel pelatihan beserta kelasnya. Saat harus mengklasifikasikan data masukan baru, algoritma ini mencari K tetangga terdekat berdasarkan nilai atribut, dan tujuannya adalah mengklasifikasikan objek baru berdasarkan nilai-nilai atribut dan data latih yang telah disimpan sebelumnya. (Agrawal, 2014) Metode *K-Nearest Neighbor* diperkenalkan oleh Thomas Cover dan Peter Hart. KNN dikenal sebagai metode klasifikasi "pemalas" atau "lazy" karena algoritma ini tidak membentuk model klasifikasi pada tahap pelatihan. Sebaliknya, K-NN menyimpan semua nilai data pelatihan dan menunda proses pembentukan model hingga data uji diberikan untuk prediksi. Ketika data uji diberikan, algoritma ini mencari K tetangga terdekat dari data uji tersebut di antara data pelatihan yang telah disimpan, dan klasifikasi dilakukan berdasarkan mayoritas kategori dari tetangga-tetangga tersebut. Pendekatan "pemalas" ini membuat K-NN fleksibel dan mudah beradaptasi dengan data yang berubah atau berkembang. Mulak & Talhar (2013), mengidentifikasi k sampel dalam data pelatihan yang memiliki variabel x mirip dengan sampel u, dan menggunakan sampel-sampel tersebut untuk mengklasifikasikan sampel baru ini ke dalam kelas v. Fungsi F dalam konteks ini adalah fungsi yang halus. Konsep yang masuk akal dalam K-NN adalah mencari sampel-sampel dalam data pelatihan yang mirip (dalam hal variabel independen) dengan sampel u yang baru, dan kemudian menghitung nilai kelas v dari sampel-sampel tersebut. Jarak atau ukuran ketidakseragaman antara sampel-sampel dapat dihitung dengan mengukur jarak Euclidean, yang menggambarkan seberapa dekat atau jauhnya sampel-sampel tersebut dalam ruang variabel independen. (Hu, dkk, 2016)

Algoritma *K-Nearest Neighbor* digunakan untuk klasifikasi Karakterisasi Kualitas Udara metode yang diusulkan karakterisasi AQI menggunakan pembelajaran mesin KNN algoritma berhasil diimplementasikan. Prototipe proyek terdiri dari berbagai sensor dikembangkan. Mesin KNN model pembelajaran dibangun dengan akurasi 99,56%. Keseluruhan, statistik model sangat baik dan akurat (Ferreira, dkk, 2018).

Random forest adalah pengembangan dari metode *Decision Tree* yang menggunakan beberapa pohon keputusan. Setiap pohon keputusan dalam *Random forest* telah dilatih dengan sampel individu, dan setiap atribut dipecah pada pohon yang dipilih secara acak dari antara subset atribut. *Random forest* memiliki sejumlah keunggulan, seperti meningkatkan akurasi hasil bahkan ketika data hilang, serta memiliki ketahanan terhadap data yang outlier. *Random forest* juga efisien dalam penyimpanan data. Metode ini juga melibatkan seleksi fitur, yang memungkinkan pengambilan fitur terbaik untuk meningkatkan kinerja model klasifikasi. Dengan seleksi fitur, *Random forest* dapat bekerja secara efektif pada big data dengan parameter yang kompleks (Devella dan Rahmawati, 2020).

Random forest digunakan untuk membangun sistem klasifikasi topik pada tweet dengan menerapkan ekspansi fitur metode Word2Vec. Ekspansi fitur Word2Vec diterapkan dalam sistem klasifikasi untuk mengatasi ketidakcocokan kosakata pada kalimat tweet. Proses ekspansi fitur melibatkan penggunaan tiga corpus Word2Vec (tweet, berita, dan gabungan antara tweet dan berita), serta tiga variasi ekspansi fitur (Top 1, Top 5, dan Top 10) untuk mencari model terbaik.

Hasil penelitian menunjukkan bahwa model terbaik diperoleh dengan menggunakan fitur top 5, yang menghasilkan nilai akurasi sebesar 99,49% dan nilai F1-Score sebesar 0,9949. Ekspansi fitur ini dilakukan dengan memanfaatkan kamus kata gabungan dari data berita dan tweet. Penerapan model ekspansi fitur ini berhasil meningkatkan nilai akurasi metode klasifikasi *Random forest* dari sebelumnya hanya mencapai 98,44% dan nilai F1-Score sebesar 0,9842. (Ramli & Sibaroni, n.d.).

Random forest juga digunakan dalam penelitian untuk diagnosis penyakit ginjal kronis Dalam penelitian ini, teknik penambangan data yang berbeda

digunakan untuk mendiagnosis orang sehat dan tidak sehat. Penelitian menggunakan algoritma *Random forest , Deep Learning network dan Neural Network* mencapai tingkat akurasi tertinggi dengan 99,09% (*Random forest*), 98,04% *(Deep Learning network*) dan 96,52% (*Neural Network*) (Rezayi, dkk, 2021).

Dari penggunaan *ensemble learning* dalam klasifikasi untuk prediksi cuaca, beberapa kesimpulan dapat diambil. Hasil akurasi dan Mean Squared Error (MSE) dari metode *ensemble learning* mencapai 81.21% untuk akurasi dan 18.79% untuk MSE. Dalam kategori decision tree, *Random forest* menunjukkan performa terbaik dengan akurasi sebesar 82.38% dan MSE sebesar 17.62%. Sementara itu, algoritme deep learning menunjukkan performa dengan akurasi 82.92% dan MSE 17.08%. Algoritma dengan performa tertinggi di antara yang lain adalah sebesar 84.06% untuk akurasi dan 15.94% untuk MSE. Kesimpulan ini memberikan gambaran bahwa penggunaan *ensemble learning* dalam prediksi cuaca memberikan hasil yang baik, dengan performa tertinggi diperoleh dari salah satu algoritma yang digunakan (Siregar, 2020).

Ensemble learning merupakan teknik atau model sebuah algoritma mempelajari dataset dengan menggabungkan atau mengkombinasikan beberapa algoritma atau model untuk menghasilkan akurasi yang lebih baik apabila dibandingkan dengan hanya menggandalkan *single* algoritma.

Hasil pengujian menunjukkan bahwa algoritme stacking mampu meningkatkan kinerja dari berbagai sisi, termasuk akurasi, True Positive Rate (TPR), True Negative Rate (TNR), Geometric Mean (G-Mean), dan Area Under the Curve (AUC). Dengan demikian, dapat diakui bahwa penggunaan algoritme stacking mampu menghasilkan performa yang lebih baik dibandingkan dengan menggunakan single classifier lainnya dalam konteks deteksi dini penyakit jantung (Nurmasani dan Pristyanto, 2021)

Beberapa penelitian terkait yang telah dilakukan sebelumnya dalam bidang prediksi udara menggunakan algoritma *machine learning*, antara lain:

Sistem perkiraan kualitas udara otomatis berbasis *machine learning* untuk memperkirakan konsentrasi harian enam polutan umum. Dengan menggunakan

model pembelajaran mesin umum (MLR, MLP, RF, GBDT, SVR) dan model ansambel SG diintegrasikan ke dalam sistem. sistem yang diusulkan adalah menemukan yang terbaik secara otomatis "Model + *Hyperparameters*" tanpa campur tangan manusia dengan menggunakan parameter acak CV atau pencarian grid CV. Dan di evaluasi menggunakan matrix ME, RMSE, dan MAPE. Sistem yang dibuat mendapatkan hasil perkiraan yang andal dan menyiratkan prospek penerapan yang baik di bidang kualitas lingkungan atmosfer perkotaan peramalan (Ke, dkk, 2022)

Michael, Leonardo (2019) mengadopsi metode Deep Neural Network untuk mengimplementasikan sistem klasifikasi index kualitas udara. Data yang telah terkumpul kemudian digunakan untuk proses pelatihan (training) dan pengujian (testing) dengan tujuan menghasilkan hasil klasifikasi indeks kualitas udara. Hasil akurasi sebesar 84% menunjukkan kemampuan model Deep Neural Network dalam mengklasifikasikan indeks kualitas udara dengan tingkat keberhasilan yang signifikan. Penelitian ini memberikan gambaran mengenai konfigurasi dan performa model DNN dalam konteks klasifikasi kualitas udara.

Penelitian yang dilakukan oleh (Gladkova dan Saychenko, 2022). memprediksi perubahan konsentrasi PM2.5 untuk pemantauan dan pencegahan kualitas udara menggunakan ARIMA, Prophet dan LSTM analisis nilai ukuran MSE dan RMSE untuk model prediksi ARIMA dan Prophet, menyimpulkan bahwa kedua model memberikan prediksi dengan akurasi yang kira-kira sama. Akan tetapi mereka masih kalah dengan prakiraan model *deep learning*.

2.2 Dasar Teori

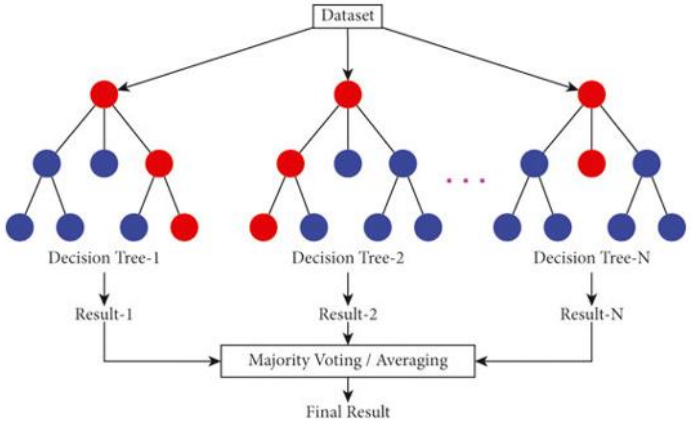
2.2.1 Machine learning

Pengertian *Machine learning* menurut (Mahesh, 2018) adalah teknik untuk melakukan inferensi dalam konteks machine learning menekankan pada hubungan variabel yang ada dalam data dengan menggunakan pendekatan matematis. Inti dari machine learning adalah pembuatan model matematis yang dapat merefleksikan pola-pola yang terdapat dalam data. *Machine learning* memungkinkan komputer atau program untuk menemukan pengetahuan atau pola-pola tersebut tanpa harus diprogram secara eksplisit.

2.2.2 Random forest

Random forest merupakan sebuah teknik yang dapat meningkatkan tingkat akurasi karena pembentukan simpul pada setiap node dilakukan secara acak.. Hasil dengan mudah menyesuaikan dengan linearitas yang terletak di data, dan dengan demikian cenderung meramalkan efektif daripada regresi linier. Lebih baik lagi, metode pembelajaran *ensemble* sepertihutan acak diadaptasi dengan baik untuk kumpulan data sedang hingga besar. Ketika jumlah variabel independen lebih besar dari jumlah observasi , mekanisme regresi logistik dan regresi linier tidak berjalan karena jumlah estimasi parameter lebih besar dari jumlah observasi. *Random forest* berfungsi karena tidak semua variabel perkiraan digunakan secara bersamaan. *Random forest* adalah salah satu metode pembelajaran fungsional yang berhasil dalam pembelajaran mesin. Keuntungan dalam teknik ini hanya bisa efektif pada tingkat yang mendekati fungsi algoritma untuk peneliti sosial (Schonlau. M,2020)

Random forest (RF) adalah sebuah teknik yang dapat meningkatkan tingkat akurasi karena pembentukan simpul pada setiap node dilakukan secara acak. Pendekatan ini digunakan untuk konstruksi Decision Tree yang melibatkan root node, internal node, dan leaf node dengan cara memilih atribut dan data secara acak sesuai dengan aturan yang telah ditetapkan.



Gambar 2.1 *Random forest* (Khan , dkk, 2021)

$$Entropy(S) = \sum_{i=1}^{n} -pi * log\ 2\ pi \quad (2.1)$$

Keterangan:

S = Himpunan Kasus

n = Sampel data

pi = Proporsi Si terhadap S

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy(S_i) \quad (2.2)$$

Keterangan:

S = Himpunan Kasus

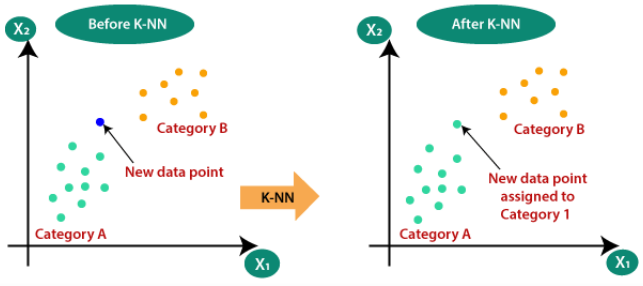
A = Atribut

$|S_1|$ = jumlah kasus pada partisi ke-1

|S| = Proporsi Si terhadap S

2.2.3 K-Nearest Neighbors (KNN)

Metode klasifikasi algoritma K-Nearest Neighbors (KNN) dikenal sebagai salah satu metode yang menunjukkan konsistensi yang kuat. Algoritma KNN melakukan prediksi kelas suatu objek berdasarkan mayoritas dari kelas-kelas tetangga terdekatnya, dengan mencari kelompok objek terdekat tersebut. Proses kerja algoritme KNN melibatkan klasifikasi sampel berdasarkan suara terbanyak (*majority voting*) dari tetangga terdekat, yang diukur dengan menggunakan fungsi jarak. *Euclidean Distance* merupakan rumus jarak yang umumnya digunakan dan menjadi pengaturan default untuk parameter metrik pada library scikit-learn untuk algoritma KNN. Dalam penelitian ini, parameter n_neighbors atau jumlah tetangga terdekat divariasikan mulai dari 1 hingga 50 sepeti gambar 2.1.



Gambar 2.2 *K-Nearest Neighbor (sumber : https://www.javatpoint.com)*

Pengolahan data dilakukan dengan membagi dataset yang telah melewati tahap sebelumnya menjadi data latih dan data uji. Setelah itu, diterapkan algoritma *K-Nearest Neighbor* (KNN). Cara kerja algoritma ini adalah dengan melakukan klasifikasi terhadap objek baru melalui perhitungan jarak terdekat objek tersebut terhadap data yang sudah ada dalam dataset. Jarak antara kedua objek data dihitung menggunakan rumus *Euclidean Distance*. Rumus perhitungan *Euclidean Distance* adalah sebagai berikut: (Handayani, 2019):

$$d_i = \sqrt{\sum_{i=1}^{n} (X_i - Y_i)^2 + (X_i - Y_i)^2} \qquad (2.3)$$

Keterangan:

X_i = Sampel data atau Data train

Y_i = Data uji atau data testingi

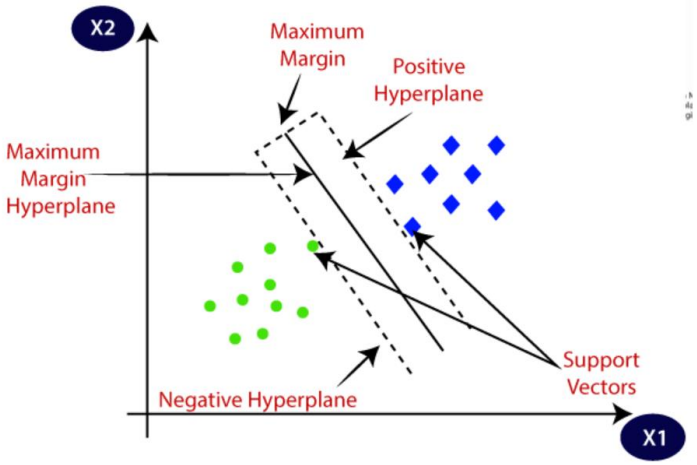
i = Variabel data

d = Jarak

n = Jumlah data

2.2.4 Support Vector Machine (SVM)

Vapnik bersama rekannya, Bernhard Boser dan Isabelle Guyon, memperkenalkan Support Vector Machine (SVM) pada tahun 1992. SVM adalah algoritma yang menggunakan pemetaan nonlinier untuk merubah data latih asli ke dalam dimensi yang lebih tinggi. Dimensi baru ini, SVM mencari *hyperplane* untuk memisahkan secara linier. Dengan menggunakan pemetaan nonlinier yang tepat ke dimensi yang lebih baik, data dari kelas akan selalu bisa dipisahkan oleh *hyperplane* tersebut, Widodo (2013).



Gambar 2.3 *Support Vector Machine (Sumber : Javapoint.com)*

2.2.5 Parameter

Parameter adalah variabel dari model yang dapat dirubah nilainya selama proses pelatihan. Nilai parameter diatur oleh model selama proses pelatihan untuk mengoptimalkan kinerja model, dan dapat berubah selama proses pelatihan seiring model memperbarui pengetahuannya tentang data. Nilai parameter dapat dioptimalkan selama proses pelatihan untuk meningkatkan kinerja model.

Contoh parameter KNN yaitu jarak tetangga terdekat / nilai K, SVM tidak ada parameter spesifik yang dipelajari selama proses pelatihan , parameter RF yaitu jumlah pohon keputusan (n_estimator) .

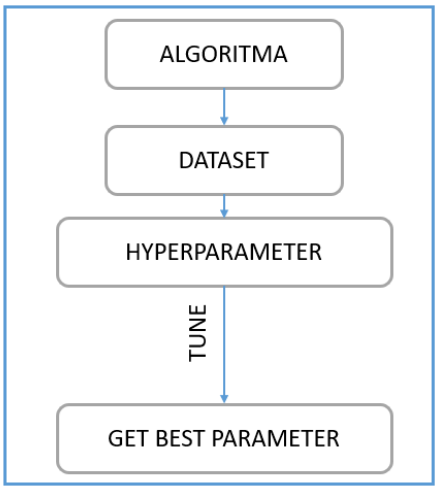
2.2.6 Hyperparameter

Hyperparameter adalah bagian dari parameter model yang menentukan bagaimana model dilatih. variabel yang nilainya ditetapkan sebelum proses pelatihan dimulai dan mempengaruhi cara model belajar dari data. Nilai *hyperparameter* harus ditentukan secara manual sebelum pelatihan dan tidak dapat dipelajari oleh model. Nilai *hyperparameter* memengaruhi cara model belajar dari data dan dapat memengaruhi kinerja model secara keseluruhan. Nilai *hyperparameter* tidak berubah selama proses pelatihan dan harus diatur sebelum pelatihan dimulai. Untuk menemukan nilai yang optimal, tuning *hyperparameter* memerlukan pengaturan manual atau otomatis(Arnold, dkk, 2024).

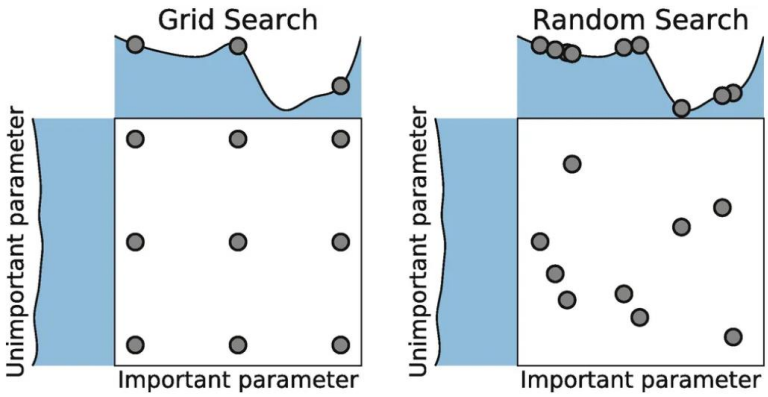
Contoh *hyperparameter* KNN yaitu euclidean, Manhattan, *hyperparameter* SVM yaitu jenis kernel (*linear, polynomial*), *hyperparameter* RF yaitu n_estimator, criterion, max_depth, dll.

2.2.7 Hyperparameter Tuning

Untuk optimalisasi *hyperparameter* dalam algoritma, berbagai strategi telah dikemukakan. *Hyperparameter* tuning adalah teknik pengembangan model *machine learning* untuk memastikan bahwa model *machine learning* memiliki penyetelan *hyperparameter* yang tepat sehingga dapat memberikan hasil yang baik dapat dilihat pada gambar 2.4 . *Grid search* dan *random search* adalah dua metodologi yang banyak digunakan yang menjadi dasar penelitian di tampilkan pada gambar 2.5. *Hyperparameter* mempunyai peran yang amat penting untuk dapat mengoptimalkan kinerja evaluasi dari model machine learning (Moreno, dkk, 2018).



Gambar 2.4 *Hyperparameter*

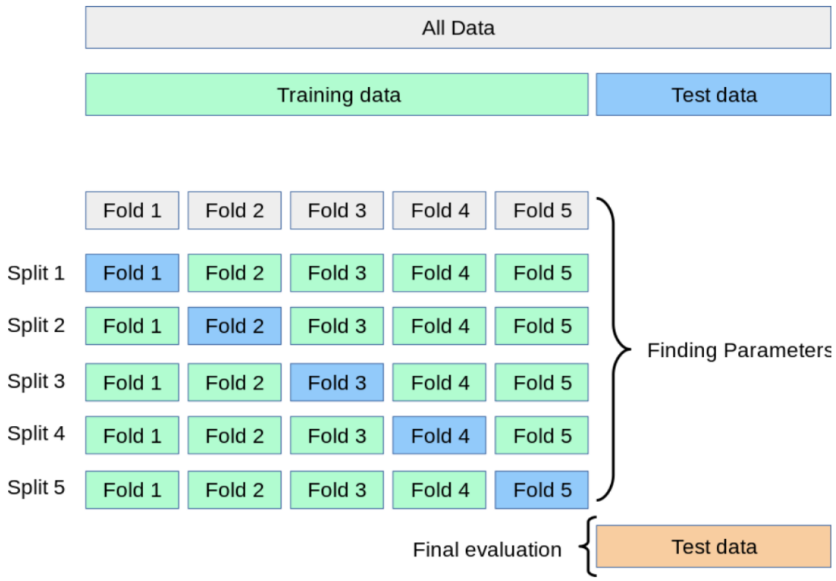


Gambar 2.5 *Grid Search* dan *Random Search*

2.2.8 Cross validation

Cross validation merupakan sebuah metode tambahan dari teknik *data mining* yang digunakan untuk memperoleh hasil akurasi yang lebih optimal, menghindari overfitting, dan memastikan bahwa model memiliki kemampuan generalisasi yang baik. *k-fold cross validation* dengan menggunakan data latih dibagi menjadi K subset (biasanya disebut "fold") yang sama ukurannya (Santosa dan Umam, 2018).

Teknik ini merupakan sebuah metode validasi yang berkembang dari model *Split Validation*, di mana validasinya melibatkan pengukuran kesalahan pelatihan dengan menguji data menggunakan test data atau data uji. Pengembangan teknik ini menjadi respons terhadap kelemahan pada model sebelumnya, di mana pengambilan sampel dilakukan secara acak dan pengujian test error tidak mampu mendistribusikan kelas secara terstruktur. Meskipun model sebelumnya dapat menghasilkan hasil yang optimal, namun tidak mampu memberikan pengujian yang efisien. Dengan menerapkan *Cross Validation*, proses validasi menjadi lebih robust karena melibatkan pengujian model pada beberapa set data uji yang berbeda, yang dapat memberikan gambaran yang lebih baik mengenai seberapa baik kinerja model secara umum, dan tidak hanya tergantung pada satu set data uji saja.



Gambar 2.6 *K-Fold Cross-Validation* (Maredia, n.d.)

Dari gambar 2.5 adalah suatu ilustrasi yang terdapat 5 fold yang berarti melalukan 5 kali evaluasi .

- Percobaan ke-1, dapat dilihat pada partisi split 1 digunakan untuk validasi dan Sisanya digunakan untuk melatih model.
- Percobaan ke-2, dapat dilihat pada partisi split 2 digunakan untuk validasi dan Sisanya digunakan untuk melatih model.
- Percobaan ke-3, dapat dilihat pada partisi split 3 digunakan untuk validasi dan Sisanya digunakan untuk melatih model.

Pada percobaan ke 5 akan mendapatkan hasil tiap fold, hasil tersebut diambil rata-rata untuk mendapatkan kinerja model secara keseluruhan pada matrix evaluasi

2.2.9 Matrix MSE & RMSE

Mean Squared Error (MSE) dan Root Mean Squared Error (RMSE) adalah metrik evaluasi yang umum digunakan dalam pemodelan statistik dan machine learning untuk mengukur seberapa baik model memprediksi nilai yang sebenarnya. MSE dihitung sebagai rata-rata dari kuadrat selisih antara nilai prediksi model dan nilai sebenarnya dari data yang diuji.

Rumus MSE adalah:

$$MSE = \frac{1}{n}\sum_{i}^{n}(yi-yi)2 \quad (2.4)$$

di mana n adalah jumlah data uji, yi adalah nilai sebenarnya, dan y^i adalah nilai prediksi. RMSE adalah akar kuadrat dari MSE dan memberikan gambaran yang lebih intuitif tentang seberapa besar kesalahan prediksi dalam satuan aslinya. Rumusnya adalah:

RMSE=$\sqrt{MSE}$

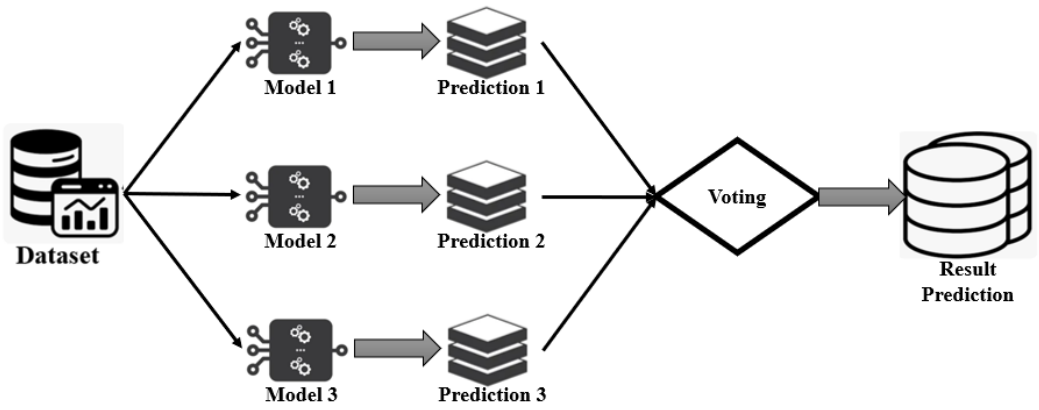
Semakin kecil nilai MSE atau RMSE, semakin baik model memprediksi nilai yang sebenarnya.

2.2.10 Ensemble learning

Ensemble learning adalah sebuah teknik dalam machine learning di mana beberapa algoritma atau model digabungkan untuk meningkatkan kinerja klasifikasi atau prediksi. Biasanya, model *ensemble* memiliki kemampuan belajar lebih baik apabila dibandingkan dengan hanya menggandalkan algoritma tunggal. Selain untuk meningkatkan kinerja dari model ensemble juga sering digunakan untuk mengatasi *overvitting* (Wu, jiaju,2022)

2.2.10.1 Ensemble Voting

Prediksi ini adalah untuk menggabungkan model pembelajaran mesin yang sama atau berbeda secara konseptual untuk prediksi melalui pemungutan suara mayoritas. Prediksi dengan menggunakan dua jenis teknik pemungutan suara, keras dan lunak. Dalam keadaan sulit pemungutan suara, prediksi akhir dilakukan dengan suara terbanyak dimana agregator memilih prediksi kelas yang muncul berulang kali di antara model dasar. Dalam soft voting, model dasar harus memiliki Metode Predict_proba. Pemungutan suara menyajikan hasil yang lebih baik secara keseluruhan hasil dibandingkan model dasar lainnya, karena menggabungkan prediksi model yang berbeda (Kumari, dkk, 2021)

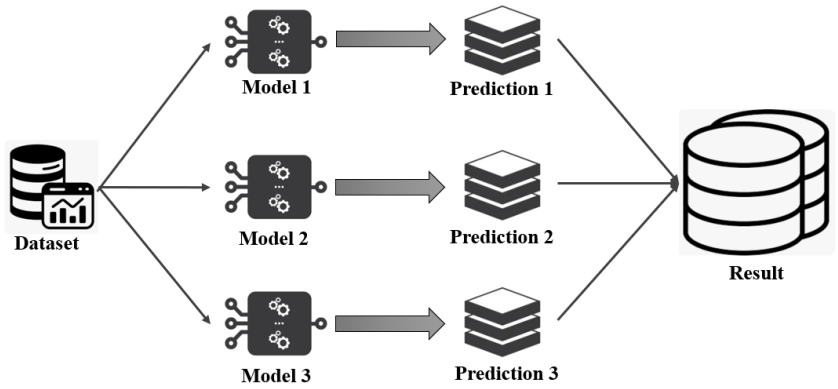


Gambar 2. 7 *Ensemble Voting*
(sumber : https://towardsdatascience.com)

Pengklasifikasi soft voting digunakan untuk proses klasifikasi, yang didasarkan pada kolom atributdict_proba yang menyediakan probabilitas setiap variabel target. Agregator pemungutan suara dan pemungutan suara lunak teknik ini digunakan untuk menghitung prediksi individu, yang kemudian diubah menjadi prediksi akhir menggunakan metode pemungutan suara mayoritas(Tavana, dkk, 2023).

2.2.10.2 Ensemble Stacking

Bagging adalah metode ensemble yang unggul untuk melatih siswa secara individu subset sampel sembarang dari dataset pelatihan asli. Agregasi agregasi beberapa pelajar mengarah ke varians yang lebih rendah dari model sementara biasnya mungkin tetap sama mengingat bias dekomposisi varians kesalahan untuk model pembelajaran mesin. Mengingat beberapa model dari algoritma pembelajaran mesin yang sama yang sama yang dilatih pada data pelatihan yang berbeda, bias dari algoritma pembelajaran algoritma pembelajaran mesin adalah kesamaan antara prediksi rata-rata model dan kebenaran dasar, dan variansnya adalah perbedaan antara prediksi (Ngo, dkk, 2022)



Gambar 2.8 *Ensemble learning Stacking*
(sumber : https://towardsdatascience.com)

2.2.11 Data mining

Data mining merupakan suatu proses mengekstraksi pola, informasi, dan keahlian yang berguna dari kumpulan data yang sangat besar. Proses pencarian secara otomatis informasi yang berguna dalam tempat penyimpanan data berukuran besar. Teknik data mining digunakan untuk memeriksa basis data berukuran besar sebagai cara untuk menemukan pola yang baru dan berguna. Munculnya data mining didasarkan pada jumlah data yang tersimpan dalam basis data semakin besar

2.2.12 Prediksi

Prediksi dalam *machine learning* mengacu pada proses menggunakan model yang telah dilatih untuk membuat prediksi atau ramalan tentang data baru yang belum terlihat sebelumnya. Ini melibatkan pengambilan data input dan menerapkan pola atau hubungan yang dipelajari dari data pelatihan untuk memprediksi output atau hasil (Zhang, dkk, 2021).

2.2.13 Kualitas Udara

Kualitas udara mengacu pada tingkat kebersihan atau polusi udara di suatu wilayah dengan tingkat kebersihan udara yang menunjukkan sejauh mana udara tersebut bebas dari polusi udara. Kualitas udara yang baik berarti udara tersebut bersih dan tidak mengandung kontaminan berbahaya dalam jumlah yang dapat membahayakan kesehatan manusia atau lingkungan. Dengan memperbaiki kualitas udara maka bisa melindungi kesehatan manusia dan lingkungan sekitar. (Fan, dkk, 2024).