JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

Analysis Of The K-Nearest Neighbor (KNN) Algorithm For Gender Classification Based On Voice Characteristics

Bintang Hutagalung*, Sriani

Faculty of Science and Technology, Department of Computer Science, State Islamic University of North Sumatra, Medan, Indonesia
Email: [1*]bintanghutagalung121@gmail.com,[2]sriani@uinsu.ac.id
Corresponding Author Email: bintanghutagalung121@gmail.com
Submitted **05-10-2025;** Accepted **29-10-2025;** Published **31-10-2025**

Abstract

The gender recognition system based on voice still faces challenges such as dependence on MFCC (Mel Frequency Cepstral Coefficients) features, which are not yet able to fully represent the complexity of human voice patterns. To overcome this, this study uses 20 voice characteristics and the K-Nearest Neighbor (KNN) algorithm because it is non-parametric, capable of handling non-linear relationships between features, and works intuitively by grouping data based on similarity of distance in the feature space, making it suitable for voice patterns that are not always linearly distributed. The purpose of this study is to analyze and develop a KNN model in classifying gender based on voice characteristics. Based on testing 50 variations of K values using K-Fold Cross Validation and Euclidean Distance, the evaluation results at K = 3, 5, and 7 showed average accuracies of 0.9740, 0.9700, and 0.9712. K = 3 was selected as the optimal parameter because it produced the highest accuracy. The results show that testing on 634 test data samples using K = 3 produced 619 correct predictions and 15 incorrect predictions, with an accuracy of 98% (0.9740), as well as precision, recall, and F1-score for the Female class of 0.98, 0.97, and 0.98, while for the Male class they were 0.97, 0.98, and 0.98.

Keywords: K-Nearest Neighbor; Classification; Gender; Voice Characteristics; Knowledge Discovery in Databases

1. INTRODUCTION

Voice-based gender recognition systems still face a number of challenges, one of the main limitations of which is the reliance on conventional voice features, such as MFCC (Mel Frequency Cepstral Coefficients) [1]. Although MFCC has been widely used in various speech recognition applications, it has not been able to fully capture the complexity and variation of human voice patterns [2]. Voice signals have high complexity because they are influenced by various factors, such as noise, intonation variations, accents, and recording quality [3]. Human voice patterns also have many aspects, including intonation nuances, accents, and other unique characteristics that differentiate male and female voice [4][5]. Therefore, the limitation of MFCC in fully representing these aspects often results in suboptimal accuracy, especially when dealing with voice data with wide variations [6][7].

To address this, this research proposes a more comprehensive approach by utilizing 20 different voice characteristics [3]. These characteristics include various acoustic parameters, such as mean frequency, standard deviation, median frequency, Q25, Q75, IQR, skewness, kurtosis, spectral entropy, spectral flatness, mode frequency, frequency centroid, average of fundamental frequency, minimum of fundamental frequency, maximum of fundamental frequency, average of dominant frequency, minimum of dominant frequency, maximum of dominant frequency, range of dominant frequency, and modulation inde [7]. The selection of these features is based on their ability to represent biological and acoustic differences between male and female voices, such as the fundamental frequency (F0) differences of male and female voices that have been studied[4][5][8]. As such, the use of this rich feature set is expected to capture more information from human voice patterns, allowing for a more accurate and detailed representation that is then used for classification modeling [6][9].

This research selected the K-Nearest Neighbor (KNN) algorithm as the classification method for modeling [10]. This selection is based on several advantages of KNN that suit the nature of voice data [11]. First, KNN is a non-parametric algorithm, which means it makes no assumptions about the distribution of the data [12][13]. This is suitable for human voice patterns that are not always linearly distributed or follow a particular statistical distribution [3][9]. Secondly, KNN is able to handle non-linear relationships between features [14]. Voice characteristics often have complex and non-linear interactions, so KNN can identify similarities between samples more flexible [6]. In addition, the KNN's proximity-based approach in feature space makes it a relevant choice for modeling gender classification based on voice characteristics [15][16].

Although various methods and datasets have shown diverse voice classification performance, some previous studies still have limitations. For example, the research of [9] used a relatively small dataset (100 samples), which risks causing overfitting and is less representative for more diverse real data. Although the results achieved 90% accuracy on the test data, the limited size of the dataset makes it difficult to generalize the model to new voice variations. The study by [17], despite using a large dataset, still focuses on multi-class classification (age and gender) simultaneously, which can increase complexity and potentially reduce precision in gender-specific classification. The CNN model used did show high accuracy of 92%, but the multi-class approach made the gender classification results less than optimal because the model had to balance two prediction categories at once. And the study by [16] relies entirely on visual features (facial data); whereas, voice characteristics offer advantages as a biometric that is more flexible, does not require visual devices, and is easily integrated in everyday use, such as for authentication on camera-less devices or interaction with virtual assistants. Although the visual approach produces very high accuracy (AUC 0.996 for Naïve Bayes and 0.992 for KNN),

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

its limitations lie in its dependence on camera devices and visual context, making it irrelevant for voice-based systems, and the Naïve Bayes algorithm assumes independence between features, which is not consistent with the complex interactions between human voice characteristics.

Unlike previous studies conducted by [9] and [17], which still have limitations in terms of dataset size, classification focus, feature types, and model assumptions, this study offers a more adaptive alternative solution through the development of a gender classification model (male and female) based on voice characteristics using the KNN algorithm [10][18][19]. Previous studies have shown that RNN and CNN based models perform well but are less than optimal in data generalization or classification precision, while visual feature-based approaches require additional devices that limit their application to non-visual systems [9][16][17]. Therefore, this study was designed to overcome these limitations by utilizing a large and balanced voice dataset (3,168 samples) that is more representative of voice variations [3]. This model relies on 20 acoustic voice characteristics that represent aspects of frequency, intensity, spectral distribution, and others, enabling it to comprehensively describe the characteristics of the human voice[4][7]. The KNN algorithm was chosen because of its non-parametric nature and its ability to group data based on the proximity between samples, making it more flexible in handling non-linear relationships between voice features [11][14]. Unlike the research by [16], which used Naïve Bayes, which assumes independence between features, KNN allows for a more realistic exploration of interactions between voice characteristics [12][20][21]. This study also optimized the K value and applied the K-Fold Cross Validation technique to ensure the stability and reliability of the classification results [15][22][23].

In line with this approach, the objective of this study is to analyze and develop an effective KNN model for classifying gender based on voice characteristics [10][11]. The analysis process will include exploring how each of the 20 voice characteristics contributes to the classification process AND how the KNN algorithm processes this information to generate predictions [3][7]. This methodological difference is expected to result in improved accuracy compared to previous studies while demonstrating the advantages of a non-parametric approach in handling the complexity of voice signals [6][14]. Thus, this research aims to contribute to the field of voice recognition by offering a more robust gender classification solution that can be applied to voice-based systems, such as virtual assistants, biometric voice recognition, or automated customer service, which will ultimately improve the accuracy and efficiency of voice-based identification systems [1][9][17].

2. RESEARCH METHODOLOGY

2.1 Research Stages

This research implements the Knowledge Discovery in Databases (KDD) method, a systematic process that aims to discover valid, novel, and useful knowledge from data sets (Fayyad et al., 1996, cited in Shu & Ye, 2023) [13]. This research approach is also quantitative, where conclusions are drawn based on the results of statistical hypothesis testing using empirical data obtained through measurement [24]. The stages of the KDD process applied in this study are illustrated in **Figure 1**, which describes the sequential flow from data selection to interpretation and evaluation.

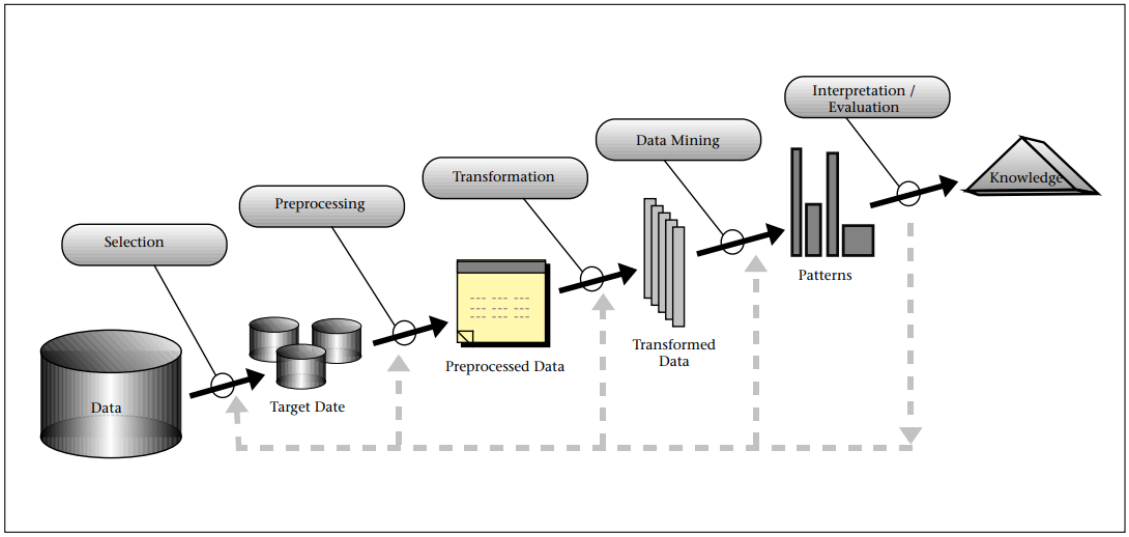

Figure 1. Knowledge Discovery in Databases (KDD)

a. Selection

This research data uses *Gender Recognition by Voice* from Kaggle in CSV format. The dataset consists of 3,168 samples (1,584 male and 1,584 female) with 20 numerical features derived from acoustic extraction and one target label representing gender (male or female). Table 1 presents the first ten rows of the dataset to describe the variable structure and feature composition used in the analysis. Each column represents an acoustic attribute that serves as an input variable in the classification process.

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

Table 1. Sample of the First 10 Rows from the Dataset

meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx	label
0.059781	0.064241	0.032027	0.015071	0.090193	0.075122	12.863462	274.402906	0.893369	0.491918	0.000000	0.059781	0.084279	0.015702	0.275862	0.007812	0.007812	0.007812	0.000000	0.000000	male
0.066009	0.067310	0.040229	0.019414	0.092666	0.073252	22.423285	634.613855	0.892193	0.513724	0.000000	0.066009	0.107937	0.015826	0.250000	0.009014	0.007812	0.054688	0.046875	0.052632	male
0.077316	0.083829	0.036718	0.008701	0.131908	0.123207	30.757155	1024.927705	0.846389	0.478905	0.000000	0.077316	0.098706	0.015656	0.271186	0.007990	0.007812	0.015625	0.007812	0.046512	male
0.151228	0.072111	0.158011	0.096582	0.207955	0.111374	1.232831	4.177296	0.963322	0.727232	0.083878	0.151228	0.088965	0.017798	0.250000	0.201497	0.007812	0.562500	0.554688	0.247119	male
0.135120	0.079146	0.124656	0.078720	0.206045	0.127325	1.101174	4.333713	0.971955	0.783568	0.104261	0.135120	0.106398	0.016931	0.266667	0.712812	0.007812	5.484375	5.476562	0.208274	male
0.132786	0.079557	0.119090	0.067958	0.209592	0.141634	1.932562	8.308895	0.963181	0.738307	0.112555	0.132786	0.110132	0.017112	0.253968	0.298222	0.007812	2.726562	2.718750	0.125160	male
0.150762	0.074463	0.160106	0.092899	0.205718	0.112819	1.530643	5.987498	0.967573	0.762638	0.086197	0.150762	0.105945	0.026230	0.266667	0.479620	0.007812	5.312500	5.304688	0.123992	male
0.160514	0.076767	0.144337	0.110532	0.231962	0.121430	1.397156	4.766611	0.959255	0.719858	0.128324	0.160514	0.093052	0.017758	0.144144	0.301339	0.007812	0.539062	0.531250	0.283937	male
0.142239	0.078018	0.138587	0.088206	0.208587	0.120381	1.099746	4.070284	0.970723	0.770992	0.219103	0.142239	0.096729	0.017957	0.250000	0.336476	0.007812	2.164062	2.156250	0.148272	male
0.134329	0.080350	0.121451	0.075580	0.201957	0.126377	1.190368	4.787310	0.975246	0.804505	0.011699	0.134329	0.105881	0.019300	0.262295	0.340365	0.015625	4.695312	4.687500	0.089920	male

b. Preprocessing

1. Missing Value Handling

A check was performed on all attributes of the dataset to ensure that there were no missing values that could interfere with the analysis. Table 2 presents the identification of missing values for each attribute, showing that all columns have zero missing entries. This indicates that the dataset is complete and can be directly used in the subsequent analytical process without requiring data imputation.

Table 2. Identification of Missing Values

meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx	label
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

2. Label Encoding

The categorical target variable consisting of *Male* and *Female* labels was transformed into numerical format to enable processing by the classification algorithm. As shown in Table 3, the *Female* class is encoded as 0 and the *Male* class as 1. This encoding process results in the addition of a new column, label_encoded, which stores the numerical representation of each sample for use in subsequent analysis.

Table 3. Label Encoding Results

label	male	female
label_encoded	1	0

3. Outlier Detection

Outliers in numerical features were identified using the Interquartile Range (IQR) method, where data points outside the range $[Q1 - 1.5 \times IQR, Q3 + 1.5 \times IQR]$ were considered outliers. As presented in Table 4, the analysis shows that several features, such as *kurtosis*, *maxfun*, and *mindom*, contain a notable number of outliers, while features such as *sfm*, *mode*, and *meanfun* have none. Despite these variations, the dataset was retained in its entirety because the presence of outliers was not expected to significantly affect the subsequent analysis, and their removal could potentially alter the intrinsic characteristics of the data distribution.

Table 4. Outlier Detection with IQR

No.	Feature	Lower Bound (Q1 - 1.5*IQR)	Upper Bound (Q3 + 1.5*IQR)	Number of Outliers	Percentage of Outliers

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

0	meanfreq	0.110436	0.252372	64	2.020202
1	sd	0.004353	0.104621	10	0.315657
2	median	0.108054	0.272157	109	3.440657
3	Q25	0.013808	0.273217	33	1.041667
4	Q75	0.156376	0.296031	27	0.852273
5	IQR	-0.064863	0.221598	10	0.315657
6	skew	-0.273619	4.854882	230	7.260101
7	kurt	-6.299491	25.617943	332	10.479798
8	sp.ent	0.761457	1.029067	6	0.189394
9	sfm	-0.155413	0.947130	0	0.000000
10	mode	-0.036617	0.375737	0	0.000000
11	centroid	0.110436	0.252372	64	2.020202
12	meanfun	0.038125	0.248454	0	0.000000
13	minfun	-0.026298	0.092426	38	1.199495
14	maxfun	0.218736	0.312689	313	9.880051
15	meandom	-0.716179	2.313173	19	0.599747
16	mindom	-0.085938	0.164062	275	8.680556
17	maxdom	-5.335938	14.414062	42	1.325758
18	dfrange	-5.375977	14.413086	42	1.325758

4. Feature-Label Separation

The dataset was separated into an independent variable (X) containing 20 acoustic features, and a dependent variable (y) containing the *encoded* labels (0=Female, 1=Male). The separation result shows X of size (3168, 20) and y of size (3168,).

5. Data Splitting

The data is divided into *training set* (80%) and *test set* (20%) using stratified split to maintain a balanced class distribution. The result was 2,534 training data samples and 634 test data samples, with a 50:50 class proportion in both subsets. The split was done using *random state* (42) to make it reproducible.

c. Transformation

Transformation is done by *feature scaling* using StandardScaler so that all numeric features are on a comparable scale. The *fit* process is only performed on X_train, and then the result parameters are used to transform X_train and X_test. This prevents information leakage from the test data. The target variable is not subjected to scaling as it is categorical. As shown in Table 5, all numerical features in *X_train* have been successfully standardized around a mean of zero and a standard deviation of one, indicating that the scaling process was correctly applied to the training data.

Table 5. After StandardScaler (X_train)

meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx
-1.224882	1.290441	-0.223137	-1.332380	-1.006838	0.971200	-0.201954	-0.202108	1.126049	1.401893	0.451184	-1.224882	0.819628	-1.034670	-0.784840	1.488656	-0.703356	0.375196	0.387773	0.404547
0.423792	-0.380829	0.849609	0.121298	0.299560	0.022979	-0.244011	-0.210104	-0.126503	-1.488511	0.797065	0.423792	-0.487280	-0.375979	0.560669	0.500670	0.794318	-0.074707	-0.088777	0.216376
-1.717208	0.606718	-1.926168	-1.258614	-1.412305	0.666037	-0.337830	-0.214813	1.502915	1.778316	-0.778713	-1.717208	-1.138348	-0.970269	0.560669	-1.380915	-0.703356	-1.364871	-1.352959	-0.485717
-0.366495	0.576094	-0.159486	0.032999	-0.190087	-0.141258	0.072442	0.001772	0.931776	0.873962	-2.121315	-0.366495	0.426261	-0.344277	0.560669	-1.324650	-0.703356	-1.318557	-1.306628	0.976734
-0.810810	0.391888	-0.339583	-0.904363	-1.282450	0.332684	-0.046520	-0.167117	-0.152498	-0.181655	-0.935062	-0.810810	-1.383076	0.314447	-1.378620	-0.771997	-0.360139	-1.212146	-1.206243	5.722539
1.043179	0.191631	1.091648	0.765760	1.632236	0.016096	0.088849	-0.135121	-0.029932	-0.050230	1.262884	1.043179	0.391818	0.575970	0.667058	2.005707	0.295093	1.674182	1.669605	-0.532967
0.132736	-1.336651	-0.051633	0.670802	-0.999574	-1.310103	0.002506	-0.167209	-1.569865	-0.910552	0.125869	0.132736	0.561978	-0.111065	-1.696991	0.241695	0.419899	0.441358	0.434104	0.649931
-0.836266	0.076543	-1.401433	-0.623186	-0.507579	0.434274	-0.026809	-0.142070	0.684530	0.334002	-0.709298	-0.836266	-1.011965	-0.157229	-0.084485	0.011173	0.419899	-0.480227	-0.487833	1.424285

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

-1.718034	0.790501	-1.372504	-1.423074	-2.302245	0.368574	-0.429125	-0.232923	1.525756	1.773745	0.074587	-1.718034	-0.107805	-0.909741	-1.007811	-1.012931	-0.328938	-0.416545	-0.410890	-0.578648
0.352749	-1.408417	0.274471	0.805270	-0.879736	-1.398185	0.060614	-0.144560	-1.574620	-1.022284	0.412570	0.352749	1.297139	-0.634249	0.560669	-0.710659	-0.703356	0.412688	0.422799	-0.715756

The transformation results for *X_test* are presented in Table 6, which demonstrates the application of the same scaling parameters obtained from the training data. This ensures that both training and testing sets share a consistent feature scale, maintaining the validity of the model evaluation process.

Table 6. After StandardScaler (X_test)

meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx
-0.986834	0.612163	-0.367233	-0.786061	-1.212481	0.235861	0.087986	-0.081651	1.176424	1.471757	-0.567897	-0.986834	0.602729	-0.279679	-0.029619	-0.572236	-0.703356	-0.937024	-0.924946	0.772854

d. Data Mining

The classification model was built using the K-Nearest Neighbor algorithm with the Euclidean metric. The K parameter value was determined through Stratified K-Fold Cross Validation (5-fold, k=1-50). The final model was trained using the entire train data with the optimal K value.

$$d(x^{test}, x^{train}) = \sqrt{\sum_i^n (x_i^{test} - x_i^{train})^2} \quad (1)$$

As shown in Figure 2, the *K-Fold Cross Validation* procedure divides the dataset into five folds, where each fold is used once as the test set while the remaining folds serve as training data. This process ensures that every sample is evaluated and contributes equally to model validation.

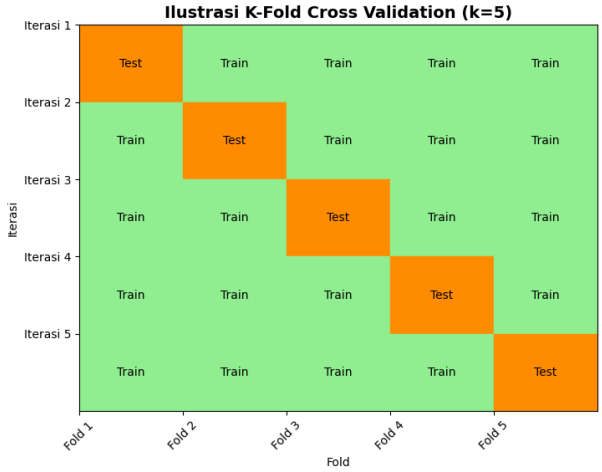

Figure 2. Illustration of K-Fold Cross Validation

e. Interpretation/Evaluation

Model performance was measured using Confusion Matrix, Accuracy, Precision, Recall, and F1-Score [25]. Evaluation was performed on test data to assess the consistency and effectiveness of the model in classifying gender [6].

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+FP+FN+TN}} \quad (3)$$

$$Precision = \frac{\text{TP}}{\text{TP+FP}} \quad (4)$$

$$Recall = \frac{\text{TP}}{\text{TP+FN}} \quad (5)$$

$$\text{F1} - \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \quad (6)$$

Description:

TP (True Positive): Correctly predicts the positive class.

TN (True Negative): Correctly predicts the negative class.

FP (False Positive): Predicts the positive class incorrectly (Type I Error).

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

FN (False Negative): Predicts the negative class incorrectly (Type II Error) [22][23][25].

f. Knowledge

The research results in the form of a KNN model with high accuracy and a methodological contribution in showing how a combination of 20 acoustic voice characteristics can be used for gender classification [3][11]. This knowledge can be utilized in voice-based applications such as biometric systems and virtual assistants [1][9].

2.2 Theoretical Framework

a. K-Nearest Neighbor (KNN)

K K-Nearest Neighbor (KNN) is an instance-based learning classification algorithm that assigns a data point to a class based on its proximity to a number of K nearest neighbors [10][11]. The algorithm belongs to the category of supervised learning because it relies on labeled data for prediction [14][15][26]. In this study, KNN is employed as the core method for gender classification using voice features, with Euclidean Distance as the distance metric, while the optimal value of K is determined through Stratified K-Fold Cross Validation to ensure stable classification performance [15][27].

b. Classification

Classification in the context of machine learning is the process of grouping data into specific categories based on patterns learned from labeled datasets [28][29]. This study focuses on binary classification, namely distinguishing between male and female voices [12][20]. Such an approach is appropriate for the research objective and has practical applications in areas such as voice biometrics, virtual assistants, and automated customer service systems [3][9][18].

c. Gender

Gender is biologically determined by structural differences in the human body, including the vocal folds [3][16][18][30]. Male vocal folds are generally longer and thicker, producing a lower fundamental frequency (90-155 Hz), while female vocal folds are shorter and thinner, producing a higher fundamental frequency (165-255 Hz). These acoustic differences form the basis for utilizing voice signals in classification tasks [3][5][17].

d. Voice Characteristics

Human voice reflects biological attributes, including gender, through acoustic parameters that can be digitally measured [4]. This study uses 20 acoustic voice features, encompassing spectral and statistical measures such as mean frequency, standard deviation, skewness, kurtosis, spectral entropy, and spectral flatness, as well as parameters related to fundamental and dominant frequencies [2][7]. These features serve as numerical representations used as input for the KNN model [1][8]. As detailed in Table 7, each acoustic feature represents a specific characteristic of the speech signal that captures the spectral distribution, frequency variation, and energy balance of the voice. Spectral features describe the structure of sound energy across frequencies, while statistical features explain the variation and distribution patterns. Meanwhile, features related to the fundamental and dominant frequencies provide information about pitch, tone stability, and voice modulation, which are essential indicators for distinguishing male and female voices in classification models.

Table 7. Acoustic Features and Definitions

Feature	Description
Mean Frequency	Weighted average frequency of the spectrum.
Standard Deviation (SD)	Measures the dispersion of frequencies.
Median Frequency	The midpoint of the spectral energy.
Q25 (First Quartile Frequency)	Frequency at which 25% of the spectral energy is accumulated.
Q75 (Third Quartile Frequency)	The frequency at which 75% of the spectral energy is accumulated.
IQR (Interquartile Range)	Frequency range between Q25 and Q75.
Skewness	Degree of asymmetry in the distribution of spectral energy.
Kurtosis	Sharpness or peakedness of the spectral energy distribution.
Spectral Entropy	Measure of the distribution of spectral energy (ordered vs. random).
Spectral Flatness	Measure of the flatness of the spectrum (tonal vs. noisy).
Mode Frequency	Frequency with the highest occurrence or intensity.
Frequency Centroid	Spectral center of mass or average frequency.
Average Fundamental Frequency	Average of the fundamental frequency of the voice.
Minimum Fundamental Frequency	The lowest value of the fundamental frequency.
Maximum Fundamental Frequency	Highest value of the fundamental frequency.
Average Dominant Frequency	Average of the most prominent (dominant) frequency.
Minimum Dominant Frequency	Lowest value of the dominant frequency.
Maximum Dominant Frequency	Highest value of the dominant frequency.
Range of Dominant Frequency	Difference between maximum and minimum dominant frequencies.
Modulation Index	Measure of variation in the dominant frequency of the voice.

3. RESULT AND DISCUSSION

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

3.1 Modeling

The K-Nearest Neighbor (KNN) model in this study was built with two main stages, namely optimizing the K parameter through the *Stratified K-Fold Cross Validation* technique and forming the final model using the best K value obtained. Cross validation with five folds was chosen because it is able to maintain a balance in the proportion of classes in each fold so that the evaluation results are more representative and can be reproduced consistently. The range of K values analyzed was 1 to 50, with the distance between samples measured using the Euclidean metric. As an illustration, a manual calculation between one test sample(X_test[0]) and the first 10 training samples(X_train[0 − 9]) was performed:

a. Distance Calculation

Xtest[0] compared to the entire Xtrain[0-9]. Each distance is calculated with the formula:

$$d(x^{test}, x^{train}) = \sqrt{\sum_{i}^{20} (x_i^{test} - x_i^{train})^2} \qquad (2)$$

Before performing the manual distance computation, the detailed feature values for the test sample X_test[0] are presented in Table 8, which provides a reference for all feature attributes used in the calculation process.

Table 8. Feature Values of Sample X_test[0]

meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx
0.150956	0.067478	0.171749	0.101665	0.196236	0.094572	3.505392	25.322624	0.948315	0.673163	0.120441	0.150956	0.162247	0.031311	0.258065	0.531494	0.007812	1.765625	1.757812	0.264828

Subsequently, to ensure the comparability of the manual computation, the corresponding feature values for the ten training samples X_train[0–9] are summarized in Table 9. This table presents the same feature attributes for each training sample to be used in computing the Euclidean distance relative to the test sample.

Table 9. Feature Values of Samples X_train[0-9]

meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx
0.143809	0.078843	0.177024	0.074958	0.201040	0.126082	2.267833	8.811947	0.946048	0.660670	0.199402	0.143809	0.169236	0.016807	0.235294	1.620739	0.007812	6.414062	6.406250	0.220935
0.193306	0.050840	0.216294	0.146104	0.231553	0.085450	2.088317	7.715960	0.889679	0.143810	0.226213	0.193306	0.127126	0.029466	0.275862	1.098558	0.101562	4.820312	4.718750	0.198510
0.129028	0.067387	0.114681	0.078564	0.191569	0.113005	1.687865	7.070532	0.963008	0.727981	0.104069	0.129028	0.106148	0.017998	0.275862	0.104083	0.007812	0.250000	0.242188	0.114839
0.169580	0.066874	0.179354	0.141705	0.220116	0.078412	3.439047	36.757162	0.937305	0.566266	0.000000	0.169580	0.156444	0.030075	0.275862	0.133821	0.007812	0.414062	0.406250	0.289125
0.156240	0.063787	0.172761	0.095881	0.194602	0.098721	2.931277	13.608140	0.888509	0.377501	0.091950	0.156240	0.098263	0.042735	0.217391	0.425914	0.029297	0.791016	0.761719	0.854701
0.211902	0.060432	0.225155	0.177526	0.262680	0.085155	3.509079	17.993717	0.894025	0.401002	0.262320	0.211902	0.155451	0.047761	0.279070	1.894015	0.070312	11.015625	10.945312	0.109208
0.184568	0.034825	0.183302	0.172884	0.201209	0.028326	3.140537	13.595432	0.824722	0.247160	0.174186	0.184568	0.161185	0.034557	0.207792	0.961682	0.078125	6.648438	6.570312	0.250178
0.155476	0.058503	0.133890	0.109627	0.212701	0.103074	3.015410	17.041213	0.926178	0.469710	0.109450	0.155476	0.109973	0.033670	0.256410	0.839844	0.078125	3.383789	3.305664	0.342461
0.129003	0.070466	0.134949	0.070524	0.170783	0.100259	1.298188	4.588312	0.964036	0.727164	0.170211	0.129003	0.139353	0.019208	0.228571	0.298573	0.031250	3.609375	3.578125	0.103764
0.191174	0.033622	0.195240	0.179457	0.204008	0.024551	3.388562	16.699887	0.824508	0.227180	0.196409	0.191174	0.184621	0.024502	0.275862	0.458333	0.007812	6.546875	6.539062	0.087424

a. Calculation for X_train[0]

$$d(x^{test}, x^{train}) = \sqrt{\sum_{i=0}^{20} (x_0^{test} - x_0^{train})^2}$$

$d(x_0^{test} - x_0^{train}) = \sqrt{}((0.151 - 0.144)^2 + (0.067 - 0.079)^2 + (0.172 - 0.177)^2 + (0.102 - 0.075)^2 + (0.196 - 0.201)^2 + (0.095 - 0.126)^2 + (3.505 - 2.268)^2 + (25.323 - 8.812)^2 + (0.948 - 0.946)^2 + (0.673 - 0.661)^2 +$

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

$(0.120 - 0.199)^2 + (0.151 - 0.144)^2 + (0.162 - 0.169)^2 + (0.031 - 0.017)^2 + (0.258 - 0.235)^2 + (0.531 - 1.621)^2 + (0.008 - 0.008)^2 + (1.766 - 6.414)^2 + (1.758 - 6.406)^2 + (0.265 - 0.221)^2)$
$= \sqrt{318.5475}$
$= 17.8479$

b. Calculation for X_train[1]

$$d(x^{test}, x^{train}) = \sqrt{\sum_{i=0}^{20}(x_0^{test} - x_1^{train})^2}$$

$d(x_0^{test} - x_1^{train}) = \sqrt{((0.151 - 0.193)^2 + (0.067 - 0.051)^2 + (0.172 - 0.216)^2 + (0.102 - 0.146)^2 + (0.196 - 0.232)^2 + (0.095 - 0.085)^2 + (3.505 - 2.088)^2 + (25.323 - 7.716)^2 + (0.948 - 0.890)^2 + (0.673 - 0.144)^2 + (0.120 - 0.226)^2 + (0.151 - 0.193)^2 + (0.162 - 0.127)^2 + (0.031 - 0.029)^2 + (0.258 - 0.276)^2 + (0.531 - 1.099)^2 + (0.008 - 0.102)^2 + (1.766 - 4.820)^2 + (1.758 - 4.719)^2 + (0.265 - 0.199)^2)$
$= \sqrt{(330.7413)}$
$= 18.1863$

c. Calculation for X_train[2]

$$d(x^{test}, x^{train}) = \sqrt{\sum_{i=0}^{20}(x_0^{test} - x_2^{train})^2}$$

$d(x_0^{test} - x_2^{train}) = \sqrt{((0.151 - 0.129)^2 + (0.067 - 0.067)^2 + (0.172 - 0.115)^2 + (0.102 - 0.079)^2 + (0.196 - 0.192)^2 + (0.095 - 0.113)^2 + (3.505 - 1.688)^2 + (25.323 - 7.071)^2 + (0.948 - 0.963)^2 + (0.673 - 0.728)^2 + (0.120 - 0.104)^2 + (0.151 - 0.129)^2 + (0.162 - 0.106)^2 + (0.031 - 0.018)^2 + (0.258 - 0.276)^2 + (0.531 - 0.104)^2 + (0.008 - 0.008)^2 + (1.766 - 0.250)^2 + (1.758 - 0.242)^2 + (0.265 - 0.115)^2)$
$= \sqrt{(341.2540)}$
$= 18.4731$

After the calculations for X_train[0]–X_train[2] are presented as examples, the same procedure is applied to the remaining training samples up to X_train[9]. To provide a complete overview of the Euclidean distance values obtained, the full results between the test sample X_test[0] and all ten training samples are summarized in Table 10, which presents each training index, the corresponding Euclidean distance, and its class label.

Table 10. Feature Values of Sample X_test[0]

X_train[i]	Euclidean Distance	Label
0	17.8479	1
1	18.1863	1
2	18.4731	1
3	11.6015	0
4	11.8304	1
5	15.0229	0
6	13.6040	0
7	8.6017	1
8	21.0140	1
9	10.9709	0

b. Sorting the Distance Results

To facilitate interpretation, the calculated Euclidean distance values were arranged in ascending order to identify the nearest training instances. The sorted results are shown in Table 11, which lists all samples ordered from the smallest to the largest distance, providing a clearer view of proximity relationships within the dataset.

Table 11. Euclidean Distance Calculation Results (After Sorting)

X_train[i]	Euclidean Distance	Label
7	8.6017	1
9	10.9709	0
3	11.6015	0
4	11.8304	1
6	13.6040	0
5	15.0229	0
0	17.8479	1
1	18.1863	1
2	18.4731	1
8	21.0140	1

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

c. Selection of K Nearest Neighbors

1. Based on the ordered distances, the nearest neighbors for several K values were then identified. For the case K = 3, the three closest neighbors to X_test[0] are presented in Table 12, showing their rank, training index, distance value, and corresponding label.

Table 12. Three Closest Neighbors for Xtest[0] (k = 3)

Rank	X_train[i]	Euclidean Distance	Label
1	7	8.6017	1
2	9	10.9709	0
3	3	11.6015	0

2. For the scenario K = 5, the five nearest neighbors are summarized in Table 13, arranged according to their proximity ranking to the test sample.

Table 13. Five Closest Neighbors for Xtest[0] (k = 5)

Rank	X_train[i]	Euclidean Distance	Label
1	7	8.6017	1
2	9	10.9709	0
3	3	11.6015	0
4	4	11.8304	1
5	6	13.6040	0

3. For the case K = 7, the complete list of seven nearest neighbors obtained from the sorted distance values is provided in Table 14, which shows the most similar instances contributing to the final KNN classification decision.

Table 14. Seven Closest Neighbors for Xtest[0] (k = 7)

Rank	X_train[i]	Euclidean Distance	Label
1	7	8.6017	1
2	9	10.9709	0
3	3	11.6015	0
4	4	11.8304	1
5	6	13.6040	0
6	5	15.0229	0
7	0	17.8479	1

d. Majority Voting of Classes

Analysis of the neighbor composition shows consistent results. At K = 3, the majority belongs to class 0 (Female). The same outcome is observed for K = 5 and K = 7, with Female continuing to dominate.

e. Class Prediction

Across all tested values of K (3, 5, and 7), the majority voting consistently produces a prediction of label 0 (Female) for the test sample X_test[0].

Evaluation of model performance was conducted using four metrics, namely *accuracy*, *precision*, *recall*, and *F1-score*. The value of each metric is calculated at each fold, then averaged to obtain the final performance of each *k* value. The results of this calculation become the basis for determining the optimal parameters. Numerical calculations, data management, and visualization of performance trends were performed using computational tools, which were all used to illustrate the relationship between variations in K values and average model accuracy. The optimal parameters were then set at the K value with the highest average accuracy during cross-validation.

The test results at K = *1* showed excellent performance, with an average *accuracy of* 0.9692, *precision of* 0.9641, *recall of* 0.9747, and *F1-score* of 0.9694. The consistency between validation folds confirms that the model is able to distinguish Male and Female classes stably, while the distribution on the confusion matrix shows the dominance of correct predictions. The computation time required to complete all folds is relatively short, at 1.17 seconds, as shown in Table 15.

Table 15. Result of Testing at K = 1

Fold	Accuracy	Precision	Recall	F1-Score	Confusion Matrix	Processing Time (seconds)
1	0.9704	0.9579	0.9843	0.9709	[[242, 11], [4, 250]]	0.0200
2	0.9763	0.9764	0.9764	0.9764	[[247, 6], [6, 248]]	0.0221
3	0.9684	0.9611	0.9763	0.9686	[[244, 10], [6, 247]]	0.0221
4	0.9684	0.9647	0.9723	0.9685	[[245, 9], [7, 246]]	0.0188
5	0.9625	0.9606	0.9644	0.9625	[[243, 10], [9, 244]]	0.0179
Average	**0.9692**	**0.9641**	**0.9747**	**0.9694**	—	**0.1100 (total)**

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

At K = 3, the model performance shows excellent performance with an average accuracy of 97.40%, precision 96.67%, recall 98.18%, and F1-Score 97.42%. The high recall value indicates that the model is able to recognize most of the positive samples well, while the high precision indicates that the positive prediction error rate is relatively low. Consistency between validation folds can be seen from the stable distribution of metrics in each fold. The total processing time for all folds was 0.12 seconds, indicating good computational efficiency, as presented in Table 16.

Table 16. Result of Testing at K = 3

Fold	Accuracy	Precision	Recall	F1-Score	Confusion Matrix	Processing Time (seconds)
1	0.9704	0.9509	0.9921	0.9711	[[240, 13], [2, 252]]	0.0196
2	0.9803	0.9766	0.9843	0.9804	[[247, 6], [4, 250]]	0.0203
3	0.9684	0.9611	0.9763	0.9686	[[244, 10], [6, 247]]	0.0201
4	0.9803	0.9765	0.9842	0.9803	[[248, 6], [4, 249]]	0.0182
5	0.9704	0.9685	0.9723	0.9704	[[245, 8], [7, 246]]	0.0196
Average	**0.9740**	**0.9667**	**0.9818**	**0.9742**	—	**0.1000 (total)**

Testing at K = 5 shows that the classification performance remains high with an average accuracy of 97.00%, precision 96.27%, recall 97.79%, and F1-Score 97.02%. The high recall value shows the model's ability to recognize most of the positive samples, while the stable precision shows that the positive prediction error rate is still low. Consistency between validation folds is evident from the relatively uniform distribution of metrics despite slight variations. The total processing time of all folds at this value was recorded at 0.11 seconds demonstrates maintained efficiency, as presented in Table 17.

Table 17. Result of Testing at K = 5

Fold	Accuracy	Precision	Recall	F1-Score	Confusion Matrix	Processing Time (seconds)
1	0.9803	0.9692	0.9921	0.9805	[[245, 8], [2, 252]]	0.0218
2	0.9763	0.9654	0.9882	0.9767	[[244, 9], [3, 251]]	0.0205
3	0.9586	0.9567	0.9605	0.9586	[[243, 11], [10, 243]]	0.0205
4	0.9724	0.9614	0.9842	0.9727	[[244, 10], [4, 249]]	0.0192
5	0.9625	0.9606	0.9644	0.9625	[[243, 10], [9, 244]]	0.0188
Average	**0.9700**	**0.9627**	**0.9779**	**0.9702**	—	**0.1000 (total)**

Testing at K = 7 resulted in an average accuracy of 97.12%, precision 96.35%, recall 97.95%, and F1-Score 97.14%. The high recall value shows that the model is still able to recognize most of the positive samples well, while the stable precision indicates that the positive prediction error rate is still under control. The consistency of the metrics in each validation fold shows that the model's performance is relatively balanced despite slight variations between folds. The total processing time of all folds at this value was recorded at 0.11 seconds, as shown in Table 18.

Table 18. Result of Testing at K = 7

Fold	Accuracy	Precision	Recall	F1-Score	Confusion Matrix	Processing Time (seconds)
1	0.9744	0.9617	0.9882	0.9748	[[243, 10], [3, 251]]	0.0200
2	0.9763	0.9690	0.9843	0.9766	[[245, 8], [4, 250]]	0.0207
3	0.9684	0.9611	0.9763	0.9686	[[244, 10], [6, 247]]	0.0235
4	0.9763	0.9689	0.9842	0.9765	[[246, 8], [4, 249]]	0.0186
5	0.9605	0.9569	0.9644	0.9606	[[242, 11], [9, 244]]	0.0182
Average	**0.9712**	**0.9635**	**0.9795**	**0.9714**	—	**0.1100 (total)**

A comparison of the test results across all K values (1–50) demonstrates a clear and systematic pattern in model performance. The overall trend shows that as the number of neighbors (*K*) increases, the classification accuracy gradually declines. This relationship highlights the balance between model flexibility and generalization, where smaller K values make the model more sensitive to local data structures, while larger K values produce smoother decision boundaries that may overlook finer distinctions between classes. The visualization provided in Figure 3 illustrates this overall behavior of the K-Nearest Neighbor model throughout the entire range of tested K values.

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

	K	Rata-rata Accuracy	Rata-rata Precision	Rata-rata Recall	Rata-rata F1-Score
0	1	0.9692	0.9641	0.9747	0.9694
1	2	0.9692	0.9838	0.9542	0.9687
2	3	0.9740	0.9667	0.9818	0.9742
3	4	0.9728	0.9769	0.9684	0.9726
4	5	0.9700	0.9627	0.9779	0.9702
5	6	0.9708	0.9693	0.9724	0.9708
6	7	0.9712	0.9635	0.9795	0.9714
7	8	0.9688	0.9648	0.9732	0.9689
8	9	0.9633	0.9529	0.9747	0.9637
9	10	0.9641	0.9586	0.9700	0.9643
10	11	0.9629	0.9515	0.9755	0.9634
11	12	0.9629	0.9564	0.9700	0.9631
12	13	0.9613	0.9499	0.9739	0.9618
13	14	0.9605	0.9540	0.9676	0.9608
14	15	0.9609	0.9492	0.9739	0.9614
15	16	0.9601	0.9505	0.9708	0.9605
16	17	0.9582	0.9441	0.9739	0.9588
17	18	0.9582	0.9462	0.9716	0.9587
18	19	0.9566	0.9413	0.9739	0.9573
19	20	0.9574	0.9434	0.9731	0.9580
20	21	0.9554	0.9378	0.9755	0.9562
21	22	0.9570	0.9413	0.9747	0.9577
22	23	0.9542	0.9343	0.9771	0.9552
23	24	0.9554	0.9391	0.9739	0.9562
24	25	0.9515	0.9308	0.9755	0.9526
25	26	0.9546	0.9377	0.9739	0.9554
26	27	0.9530	0.9316	0.9779	0.9542
27	28	0.9530	0.9335	0.9755	0.9540
28	29	0.9522	0.9296	0.9787	0.9535
29	30	0.9522	0.9315	0.9763	0.9533
30	31	0.9499	0.9267	0.9771	0.9512
31	32	0.9511	0.9294	0.9763	0.9522
32	33	0.9499	0.9261	0.9779	0.9512
33	34	0.9503	0.9280	0.9763	0.9515
34	35	0.9495	0.9260	0.9771	0.9508
35	36	0.9495	0.9267	0.9763	0.9508
36	37	0.9479	0.9220	0.9787	0.9495
37	38	0.9479	0.9239	0.9763	0.9493
38	39	0.9467	0.9212	0.9771	0.9483
39	40	0.9475	0.9226	0.9771	0.9490
40	41	0.9467	0.9206	0.9779	0.9483
41	42	0.9479	0.9232	0.9771	0.9494
42	43	0.9463	0.9199	0.9779	0.9479
43	44	0.9463	0.9217	0.9755	0.9478
44	45	0.9455	0.9185	0.9779	0.9472
45	46	0.9444	0.9183	0.9755	0.9460
46	47	0.9440	0.9170	0.9763	0.9457
47	48	0.9447	0.9190	0.9755	0.9464
48	49	0.9447	0.9171	0.9779	0.9465
49	50	0.9447	0.9184	0.9763	0.9464

Figure 3. Performance Results of K-Fold Cross Validation (1-50)

To further analyze the relationship between *K* and performance stability, Figure 4 visualizes the average accuracy obtained from the 5-Fold Cross Validation process for each K value. The curve indicates that accuracy reaches its peak at K = 3 with an average accuracy of 0.9740, after which the performance gradually declines as K increases. This finding empirically confirms that when *K* is too small, such as at K = 1, the model is prone to overfitting and becomes overly sensitive to noise and minor data variations. Conversely, as *K* grows larger, the decision boundary becomes overly generalized, reducing the model's precision in distinguishing between male and female voice characteristics. Thus, the experiment confirms that K = 3 provides the optimal trade-off between stability and performance, ensuring balanced accuracy, precision, recall, and F1-score metrics across folds.

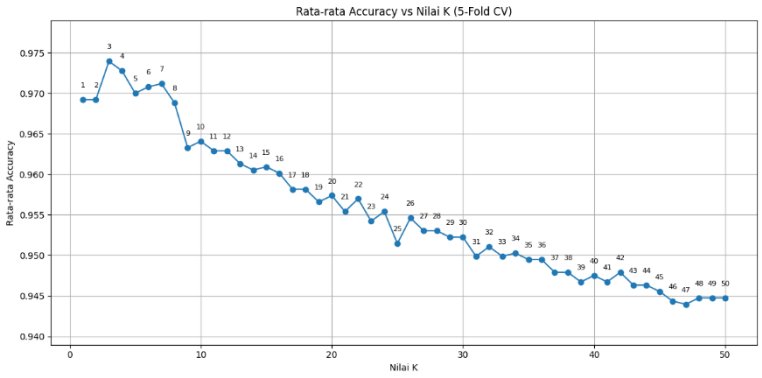

Figure 4. Average Accuracy vs K Value (5-Fold Cross Validation)

In addition to overall performance trends, the consistency of the model at K = 3 across different folds of the cross-validation process is demonstrated in Figure 5. The figure illustrates the minimal variation between folds, where each fold maintains similar levels of accuracy, precision, recall, and F1-score. This consistency indicates that the KNN model exhibits robust generalization behavior under the optimal parameter setting and is not overly dependent on a specific data partition. Therefore, it can be concluded that K = 3 provides the most stable classification results across all validation sets.

```
Nilai K terbaik (berdasarkan rata-rata Accuracy): 3
Rata-rata Accuracy tertinggi: 0.9740
Waktu total proses K-Fold CV untuk semua K: 29.37 detik
```

Figure 5. Performance Evaluation

The detailed results of testing at K = 3 are further presented in Figure 6, which summarizes the evaluation metrics obtained for each of the five folds. The results show that all folds achieve performance above 96% across all metrics, reinforcing the stability of the model and the reliability of K = 3 as the best configuration. This visualization also highlights

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

the balanced distribution of prediction outcomes, as reflected in the confusion matrices, indicating that both male and female classes are classified accurately and consistently.

```
========== Menguji K = 3 ==========
Fold 1
- Accuracy  : 0.9704
- Precision : 0.9509
- Recall    : 0.9921
- F1-Score  : 0.9711
- Confusion Matrix:
[[240  13]
 [  2 252]]
- Waktu Proses Fold: 0.0561 detik

Fold 2
- Accuracy  : 0.9803
- Precision : 0.9766
- Recall    : 0.9843
- F1-Score  : 0.9804
- Confusion Matrix:
[[247   6]
 [  4 250]]
- Waktu Proses Fold: 0.0566 detik

Fold 3
- Accuracy  : 0.9684
- Precision : 0.9611
- Recall    : 0.9763
- F1-Score  : 0.9686
- Confusion Matrix:
[[244  10]
 [  6 247]]
- Waktu Proses Fold: 0.0658 detik

Fold 4
- Accuracy  : 0.9803
- Precision : 0.9765
- Recall    : 0.9842
- F1-Score  : 0.9803
- Confusion Matrix:
[[248   6]
 [  4 249]]
- Waktu Proses Fold: 0.0608 detik

Fold 5
- Accuracy  : 0.9704
- Precision : 0.9685
- Recall    : 0.9723
- F1-Score  : 0.9704
- Confusion Matrix:
[[245   8]
 [  7 246]]
- Waktu Proses Fold: 0.0618 detik

- Rata-rata Accuracy  : 0.9740
- Rata-rata Precision : 0.9667
- Rata-rata Recall    : 0.9818
- Rata-rata F1-Score  : 0.9742
- Waktu Proses K = 3  : 0.31 detik
```

Figure 6. Result of Testing at K = 3

Based on these optimal parameters, the model is rebuilt using the entire training dataset to maximize information utilization. The optimal configuration applies K = 3 with the Euclidean distance metric, ensuring consistency with the cross-validation phase. As illustrated in Figure 7, the model is then fitted to the full training data to strengthen its generalization

```
1  final_knn = KNeighborsClassifier(n_neighbors=best_k, metric='euclidean')
2  final_knn.fit(X_train, y_train)
✓ 0.0s

KNeighborsClassifier
KNeighborsClassifier(metric='euclidean', n_neighbors=3)
```

Figure 7. Training of KNN Model Using Entire Training Data

3.2 Evaluation

odel evaluation produces a confusion matrix in a 2×2 matrix format that presents a comparison between predicted and actual class values. As shown in Figure 8, this visualization illustrates the distribution of correct and incorrect predictions generated by the model on the test dataset. Based on the prediction results on the test data, 307 True Negative (TN) were obtained, which is the number of voice samples that are actually female and have been correctly predicted as female by the model. The False Positive (FP) is 10, which is the number of voice samples that are actually female (Female) but wrongly predicted as male (Male) by the model. This condition is known as Type I Error. False Negative (FN) of 5, which is the number of voice samples that are actually male but incorrectly predicted as female by the model. This condition is known as Type II Error. True Positive (TP) is 312, which is the number of voice samples that are actually male (Male) and successfully predicted correctly as male (Male) by the model.

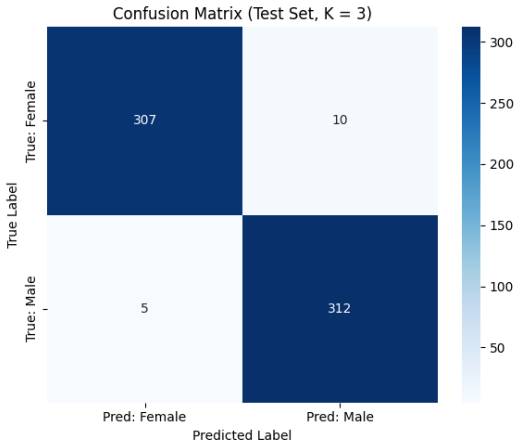

Figure 8. Confusion Matrix

From this matrix, it can be seen that the model performs well in distinguishing between *Male* and *Female* voices, as the number of correct predictions (TN and TP) is significantly higher than incorrect predictions (FP and FN). The relatively small number of misclassifications, 10 FP and 5 FN, indicates that the model exhibits high reliability in

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

gender recognition. Furthermore, the lower number of FN compared to FP shows that the model tends to misclassify *Female* voices as *Male* more often than the opposite case. Out of 634 total predictions, the model produced 619 correct predictions (307 TN + 312 TP) and only 15 misclassifications (10 FP + 5 FN), resulting in high classification consistency.

The distribution of predictions on the confusion matrix is then used as the basis for calculations that utilize the True Positive, True Negative, False Positive, and False Negative values to describe the model's performance quantitatively.

1. Accuracy

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} = \frac{312 + 307}{312 + 10 + 5 + 307} = \frac{619}{634} = 0.9763 = 0.98$$

2. Precision

$$Precision\ (1) = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{312}{312 + 10} = \frac{156}{161} = 0.9689 = 0.97$$

$$Precision\ (0) = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{307}{307 + 5} = \frac{307}{312} = 0.9839 = 0.98$$

3. Recall

$$\text{Recall}\ (1) = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{312}{312 + 5} = \frac{312}{317} = 0.9842 = 0.98$$

$$Recall\ (0) = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{307}{307 + 10} = \frac{307}{317} = 0.9684 = 0.97$$

4. F1-Score

$$\text{F1} - \text{Score}\ (1) = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = 2 \times \frac{0.9689 \times 0.9842}{0.9689 + 0.9842} = \frac{47679569}{48827500} = 0.9765 = 0.98$$

$$\text{F1} - \text{Score}\ (0) = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = 2 \times \frac{0.9839 \times 0.9684}{0.9839 + 0.9684} = \frac{23820219}{24403750} = 0.9760 = 0.98$$

Based on these calculations, the model achieved an accuracy of 0.98 with a relatively balanced performance distribution between the Male and Female classes. The precision of the Female class is 0.98, which is slightly higher than that of Male, which is 0.97, so Female predictions are less likely to be wrong. In contrast, the recall of Male class is higher (0.98) than Female (0.97), which means the model is more sensitive in recognizing Male samples. The F1-score value for both classes was 0.98, showing a balance between precision and recall. This shows that the model is not only accurate overall, but also consistently maintains a stable performance in both voice categories. This result, as illustrated in Figure 9, is in line with the classification report produced by the model, which is a summary form that confirms that the model's performance is not only accurate overall, but also evenly distributed in each class.

	precision	recall	f1-score	support
Female	0.98	0.97	0.98	317
Male	0.97	0.98	0.98	317
accuracy			0.98	634
macro avg	0.98	0.98	0.98	634
weighted avg	0.98	0.98	0.98	634

Figure 9. Classification Report

This result is consistent with the overall classification report produced by the model, confirming that its performance is not only accurate globally but also well distributed across individual classes. To provide a more detailed interpretation, Figure 10 presents an example of prediction results for several test samples. This table displays the acoustic feature values, actual labels, and predicted labels, enabling direct observation of how the model classifies individual voice data and evaluates the closeness between predicted and actual results.

Tabel Hasil Prediksi 10 Data Pertama (Fitur X, Aktual, Prediksi):

	meanfreq	sd	median	Q25	Q75	IQR	skew	kurt	sp.ent	sfm	mode	centroid	meanfun	minfun	maxfun	meandom	mindom	maxdom	dfrange	modindx	Aktual	Prediksi
0	-0.986834	0.612160	-0.367233	-0.786061	-1.212481	0.235861	0.087986	-0.081651	1.176424	1.471757	-0.567897	-0.986834	0.602729	-0.279967	-0.029619	-0.572236	-0.703356	-0.937022	-0.924946	0.772854	female	female
1	1.304040	-1.403656	0.891472	1.238381	0.920108	-0.911239	-0.349048	-0.229809	-0.923693	-1.245554	0.461334	1.304040	1.143007	0.549064	0.667058	1.399583	-0.453744	1.283824	1.292336	-0.346303	female	female
2	1.095994	-1.348344	0.684377	1.214011	0.165464	-1.294774	0.077122	-0.136281	-1.680872	-1.239926	0.451653	1.095994	0.970581	0.673628	0.613556	0.628831	2.541604	0.298007	0.253192	-0.498031	female	female
3	-1.285264	1.224012	-1.157617	-1.169400	-0.924551	0.830123	-0.483388	-0.232326	1.731318	2.089284	-0.992060	-1.285264	-1.464485	-0.981049	-2.489153	-0.985878	-0.703356	-1.252395	-1.240440	0.617735	male	male
4	-0.051068	0.039409	-0.007258	0.261253	-0.491520	-0.565957	-0.370860	-0.228686	1.008890	1.005219	0.461644	-0.051068	0.212021	-0.825155	0.624207	1.381680	-0.750158	-0.338253	-0.325122	2.015425	female	female
5	-0.122701	0.518836	-0.672518	-0.357713	1.041769	0.975925	-0.076847	-0.167470	0.364897	-0.144199	-0.477009	-0.122701	-0.801461	-0.892905	-3.151614	-1.025667	0.263892	-1.216282	-1.221411	1.617751	male	male
6	-0.191196	0.561935	0.159550	-0.658751	0.091465	0.801369	2.211330	1.733002	0.599163	0.249744	-1.346749	-0.191196	-2.185883	-1.061385	-0.297093	-1.389555	0.295093	-1.234752	-1.240440	-0.835357	male	male
7	-0.654460	0.600888	0.063940	-0.791898	-0.344978	0.715372	-0.242138	-0.213066	1.078608	1.053839	0.748399	-0.654460	0.592017	0.539424	0.667058	0.979666	-0.453744	1.270592	1.279099	-0.427237	female	male
8	-2.524426	1.669893	-2.931305	-2.285010	-1.646438	1.709348	-0.166508	-0.184088	1.368893	1.725083	-2.011756	-2.524426	1.264362	-1.053544	0.560669	-1.014106	-0.703356	-1.192849	-1.180872	1.662955	female	female
9	0.606041	-0.027827	0.694491	0.401253	0.730257	-0.059712	0.726394	0.361011	0.235702	-0.570517	-0.958707	0.606041	-2.193949	-0.977935	-3.582485	-1.302440	0.669511	-1.316352	-1.328690	-0.385604	male	male

Figure 10. Example of Prediction Results (Test Data)

JURIKOM (Jurnal Riset Komputer), Vol. 12 No. 5,Oktober 2025
e-ISSN 2715-7393 (Media Online), p-ISSN 2407-389X (Media Cetak)
DOI 10.30865/jurikom.v12i5.9117
Hal 705-720
https://ejurnal.stmik-budidarma.ac.id/index.php/jurikom

3.3 Discussion

Based on the evaluation results on the *test set* using the best KNN model (k = 3), the accuracy shows that the overall model is able to correctly classify about 98% of the total samples in the *test set* (both male as male, and female as female). This high accuracy indicates the model's excellent performance in distinguishing between the two classes. For the male class, 97% of the samples predicted as *male* by the model were indeed *male*. The high precision value indicates that the model has a low *False Positive* rate (predicting Male when Female). Recall of 98% shows that the model is able to identify almost all *Male* samples that actually exist in the *test set*, so there are very few *False Negatives* (predicting Female when Male). F1-Score of 98% shows a good balance between precision and recall. Since the dataset is balanced, F1-Score confirms that the model performance is stable in both classes without any significant bias. For the Female class, the precision reached 98%, meaning that almost all Female predictions matched the original label so *False Positive* (predicting Female when Male) was very low. Recall of 97% shows that most of the Female samples were correctly recognized, although there were still a few cases of *False Negative*. The F1-Score value of 98% confirms the balance between precision and recall in the Female class. This result proves that the performance of the model is not only high overall, but also consistent within each class, thus maintaining a fair distribution of performance between Male and Female.

When compared to the other K values also tested (1, 5, and 7), a consistent pattern is seen. At K = 1, the model is able to achieve high accuracy, but it is more sensitive to noise because the classification decision is only determined by one nearest neighbor. This can be seen from the wider variation of the inter-fold metric, so although the performance is strong, the stability of the model is still not guaranteed. In contrast, at K = 5 and K = 7, the classification performance remains competitive with accuracy above 97%. However, the use of a larger number of neighbors makes the model more "delicate" in determining the decision boundary. Consequently, the sensitivity of the model to detailed voice patterns is reduced, which is reflected in a decrease in precision although recall is still relatively high. In other words, the model becomes more stable but loses a bit of sharpness in distinguishing similar classes. K = 3 occupies an ideal middle position. It is small enough to remain sensitive to local variations between samples, but not so small that it is susceptible to *noise*. The stability between validation folds is more consistent than K = 1, and the precision-recall is more balanced than K = 5 and K = 7. This pattern shows that K = 3 provides the best *trade-off*: it is able to maintain peak accuracy, maintain stable performance between folds, and still maintain a balance between sensitivity and accuracy of prediction.

The choice of K value in KNN involves a *trade-off* between bias and variance. A small K value (e.g. K = 1) makes the model more sensitive to *noise* in the *training* data. Each prediction relies heavily on a (single) nearest neighbor sample, which can vary widely. This leads to high variance as the model is highly variable to small changes in the *training* data. The model may have a low bias on the *training* data (because it can "memorize" the *noise*), but tends to *overfitting* and performs poorly on new data (*test set*). The graph shows that the accuracy of K = 1 is already high, but still slightly lower than K = 3, and more susceptible to *noise*. Conversely, large K values (e.g. K = 50) make the model less sensitive to local details or *noise*. Predictions are based on the majority of votes from many neighbors, so the *decision boundary* becomes smoother. This leads to low variance as the model is more stable to small changes. However, the model can have a high bias if K is too large because it includes too many neighbors from other classes, thus ignoring important local patterns. The model is potentially *underfitting* as it is too simple to capture the complexity of the data. The graph shows that the accuracy tends to decrease as the value of K increases.

The optimal K value (e.g. K = 3) is obtained through the *K-fold* cross *validation* process, by finding the K value that provides the best balance between bias and variance. The results are reflected in the evaluation metrics (mainly average accuracy) of the validation folds. In this case, the highest average accuracy is achieved when K = 3. This indicates that K = 3 successfully captures important patterns in the data (low bias) without being too sensitive to *noise* or variation within each fold of the *training* data (variance preserved). The model with K = 3 provides optimal generalization performance, as seen from the high and balanced evaluation metrics on *test sets* that the model has never seen before. As illustrated in Figure 11, the graph “Average Accuracy vs K Value (5-Fold CV)” clearly shows the accuracy peaks at K = 3 (or small K values around it), which supports the choice of K = 3 as the optimal value. This offers the best trade-off between bias and variance for this dataset.

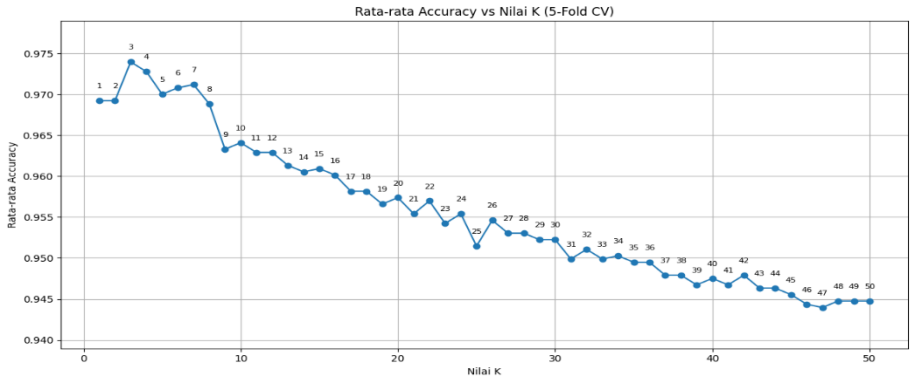

Figure 11. Average Accuracy vs K Value (5-Fold CV)

Thus, the 98% accuracy achieved in this study demonstrates the superiority of the K-Nearest Neighbor (KNN) algorithm in identifying gender based on voice characteristics by utilizing a large and balanced dataset of 3168 samples. This result is better than [9] who only obtained 90% accuracy using Recurrent Neural Network (RNN) on 100 voice samples and surpassed the research of [17] who used Convolutional Neural Network (CNN) for combined classification of gender and age with 92% accuracy. Meanwhile, although [16] study resulted in an AUC of 99.6% for Naïve Bayes and 99.2% for KNN using visual features, the voice characteristic-based approach in this study offers better flexibility as a biometric since it does not require a visual device to be integrated in systems such as virtual assistants or voice-based security. With all these findings, this research can be declared successful in analyzing the application of the K-Nearest Neighbor (KNN) algorithm and developing and evaluating the performance of the model in classifying male and female gender based on voice characteristics.

4. CONCLUSION

Based on the results of the K-Nearest Neighbor (KNN) algorithm analysis in classifying gender based on voice characteristics, it can be concluded that KNN was successfully implemented using 3168 voice samples that were balanced between male and female. From testing 50 variations of k values, the best performance was obtained at K = 3 with an accuracy of 98%, higher than K = 5 and K = 7, which achieved 97% and 97.12%, respectively. The model was then tested on 634 test data and produced an accuracy of 98% with 619 correct predictions and 15 incorrect predictions. The precision, recall, and F1-score values are also balanced for both classes, where Female has a precision of 98%, recall of 97%, and F1-score of 98%, while Male has a precision of 97%, recall of 98%, and F1-score of 98%. These results demonstrate that KNN with the optimal parameter K = 3 can provide high-performance, stable, and reliable performance in distinguishing between male and female voices, making it potentially applicable in practical applications such as virtual assistants, automated customer service, or voice-based security systems. For further development, future research may consider implementing additional preprocessing techniques such as normalization or other relevant methods to address uneven feature distributions. The determination of the number of neighbors should not only be done experimentally but also based on theoretical considerations regarding class distribution and inter-point distances. Computational efficiency can be improved by utilizing KNN variations based on data structures such as KD-Tree or Ball Tree, especially for large datasets. The use of other algorithms such as Support Vector Machine or ensemble classifiers can be used as a comparison to make the model performance more measurable. The application of the model in real-world cases, such as mapping user demographics in human-machine interactions, can also expand the benefits of this research to produce a more adaptive, efficient, and practical voice-based classification system.

REFERENCES

[1] Hermanto and T. W. Sen, "Syllable-Based Javanese Speech Recognition Using MFCC and CNNs: Noise Impact Evaluation," *Jurnal Teknik Informatika*, vol. 18, no. 1, pp. 32–42, 2025.

[2] G. Ajinurseto, L. O. Bakrim, and N. Islamuddin, "Penerapan Metode Mel Frequency Cepstral Coefficients pada Sistem Pengenalan Suara Berbasis Desktop," *INFOMATEK: Jurnal Informatika, Manajemen dan Teknologi*, vol. 25, no. 2, pp. 11–20, 2023, doi: 10.23969/infomatek.v25i1.6109.

[3] H. D. Arpita, A. A. Ryan, M. F. Hossain, M. S. Rahman, M. Sajjad, and N. N. I. Prova, "Exploring Bengali speech for gender classification: machine learning and deep learning approaches," *Bulletin of Electrical Engineering and Informatics*, vol. 14, no. 1, pp. 328–337, 2025, doi: 10.11591/eei.v14i1.8146.

[4] R. Cristina Oliveira, A. C. C. Gama, and M. D. C. Magalhães, "Fundamental Voice Frequency: Acoustic, Electroglottographic, and Accelerometer Measurement in Individuals With and Without Vocal Alteration.," *J Voice*, vol. 35, no. 2, pp. 174–180, Mar. 2021, doi: 10.1016/j.jvoice.2019.08.004.

[5] H. Colineaux, L. Neufcourt, C. Delpierre, M. Kelly-Irving, and B. Lepage, "Explaining biological differences between men and women by gendered mechanisms," *Emerg Themes Epidemiol*, vol. 20, no. 1, pp. 1–17, 2023, doi: 10.1186/s12982-023-00121-6.

[6] I. N. Switrayana, S. Hadi, and N. Sulistianingsih, "A Robust Gender Recognition System using Convolutional Neural Network on Indonesian Speaker," *Sistemasi*, vol. 13, no. 3, p. 1008, 2024, doi: 10.32520/stmsi.v13i3.3698.

[7] M. Araya-Salas, "Streamline Bioacoustic Analysis," *warbleR*, pp. 1–149, 2025.

[8] A. Biehl *et al.*, "Scalable and High-Throughput In Vitro Vibratory Platform for Vocal Fold Tissue Engineering Applications.," *Bioengineering (Basel)*, vol. 10, no. 5, May 2023, doi: 10.3390/bioengineering10050602.

[9] D. T. Adherda, M. Hikmatyar, and Ruuhwan, "Gender Classification Based on Voice Using Recurrent Neural Network (RNN)," *Antivirus : Jurnal Ilmiah Teknik Informatika*, vol. 17, no. 1, pp. 111–122, 2023, doi: 10.35457/antivirus.v17i1.3049.

[10] R. W. Dwinanto, A. S. S. A, and R. Ardianto, "Klasifikasi Berisiko Stunting pada Balita: Perbandingan K-Nearest Neighbor, Naïve Bayes, Support Vector Machine," *METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi*, vol. 8, no. 2, pp. 264–273, 2024.

[11] A. A. Baihaqi and M. Fakhriza, "K-Nearest Neighbors (KNN) to Determine BBRI Stock Price," *Sistemasi: Jurnal Sistem Informasi*, vol. 14, no. 2, pp. 969–984, 2025.

[12] MP Firdaus, "Perbandingan Algoritma K-Nearest Neighbor (KNN) dan Naive Bayes Classifier (NBC) dengan pelabelan Transformers serta Ektraksi Fitur TF-IDF dan N-Gram untuk Analisis Sentimen Terhadap Penundaan Pemilu," *Perbandingan Algoritma K-Nearest Neighbor (KNN) dan Naive Bayes Classifier (NBC) dengan pelabelan Transformers serta Ektraksi Fitur*

TF-IDF dan N-Gram untuk Analisis Sentimen Terhadap Penundaan Pemilu, pp. 5–24, 2023, [Online]. Available: https://repository.uinjkt.ac.id/dspace/handle/123456789/72466

[13] X. Shu and Y. Ye, "Knowledge Discovery: Methods From Data Mining And Machine Learning," *Soc Sci Res*, vol. 110, p. 102817, 2023, doi: https://doi.org/10.1016/j.ssresearch.2022.102817.

[14] Z. M. J. Nafis, R. Nazilla, R. Nugraha, and S. 'Uyun, "Perbandingan Algoritma Decision Tree dan K-Nearest Neighbor untuk Klasifikasi Serangan Jaringan IoT," *Komputika: Jurnal Sistem Komputer*, vol. 13, no. 2, pp. 245–252, 2024, doi: 10.34010/komputika.v13i2.12609.

[15] V. I. Sunarko, D. L. S. Dimara, P. S. E. Siagian, D. Manalu, and F. T. Anggraeny, "Implementasi K-Fold Dalam Prediksi Hasil Produksi Agrikultur Pada Algoritma K-Nearest Neighbor (KNN)," *INTEGER: Journal of Information Technology*, vol. 10, no. 1, pp. 10–16, 2025, doi: 10.31284/j.integer.2024.v10i1.6892.

[16] A. Fathah and C. Juliane, "Klasifikasi Gender Menggunakan Data Wajah dengan Algoritma Naïve Bayes dan K-Nearest Neighbors," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 12, no. 1, pp. 99–110, 2025, doi: 10.25126/jtiik.2025128724.

[17] Moh. F. Erinsyah, V. Karenina, and D. S. Wibowo, "Klasifikasi Rentang Usia Dan Gender Dengan Deteksi Suara Menggunakan Metode Deep Learning Algoritma CNN (Convolutional Neural Network)," *Komputika: Jurnal Sistem Komputer*, vol. 12, no. 2, pp. 195–202, 2023, doi: 10.34010/komputika.v12i2.10516.

[18] S. Suwarno, "Gender Classification Based on Fingerprint Using Wavelet and Multilayer Perceptron," *Sinkron*, vol. 8, no. 1, pp. 139–144, 2023, doi: 10.33395/sinkron.v8i1.11925.

[19] A. Muhammad, D. Pratiwi, and A. Salim, "Penerapan Metode Convolutional Neural Networks pada Pengenalan Gender Manusia berdasarkan Foto Tampak Depan," *Jurnal Komtika (Komputasi dan Informatika)*, vol. 7, no. 2, pp. 114–123, 2023, doi: 10.31603/komtika.v7i2.9937.

[20] V. H. Nabilla, D. Fitria, D. Permana, and F. Fitri, "Comparison of Haversine and Euclidean Distance Formula for Calculating Distance Between Regencies in West Sumatra," *UNP Journal of Statistics and Data Science*, vol. 1, no. 3, pp. 120–125, 2023, doi: 10.24036/ujsds/vol1-iss3/39.

[21] V. H. Nabilla, D. Fitria, D. Permana, and F. Fitri, "Jarak Haversine," vol. 1, pp. 120–125, 2023, [Online]. Available: https://ujsds.ppj.unp.ac.id/index.php/ujsds/article/view/39/31

[22] W. Wijiyanto, A. I. Pradana, S. Sopingi, and V. Atina, "Teknik K-Fold Cross Validation untuk Mengevaluasi Kinerja Mahasiswa," *Jurnal Algoritma*, vol. 21, no. 1, pp. 239–248, 2024, doi: 10.33364/algoritma/v.21-1.1618.

[23] A. I. Pradana and Wijiyanto, "Identifikasi Jenis Kelamin Otomatis Berdasarkan Mata Manusia Menggunakan Convolutional Neural Network (CNN) dan Haar Cascade Classifier," *G-Tech : Jurnal Teknologi Terapan*, vol. 8, no. 1, pp. 502–511, 2024, doi: 10.33379/gtech.v8i1.3814.

[24] Wijoyo A, Saputra A, Ristanti S, Sya'ban S, Amalia M, and Febriansyah R, "Pembelajaran Machine Learning," *OKTAL (Jurnal Ilmu Komputer dan Science)*, vol. 3, no. 2, pp. 375–380, 2024, [Online]. Available: https://journal.mediapublikasi.id/index.php/oktal/article/view/2305

[25] S. Sathyanarayanan, "Confusion Matrix-Based Performance Evaluation Metrics," *African Journal of Biomedical Research*, no. November, pp. 4023–4031, 2024, doi: 10.53555/ajbr.v27i4s.4345.

[26] P. Pangestu and R. Setyadi, "Penerapan Metode K-Nearest Neighbor Untuk Pemilihan Rekomendasi Game FPS Pada Aplikasi Google Play Store," *Journal of Information System Research (JOSH)*, vol. 4, no. 2, pp. 742–747, 2023, doi: 10.47065/josh.v4i2.3006.

[27] M. A. Aulia, R. Goejantoro, and M. N. Hayati, "Penerapan Metode Klasifikasi K-Nearest Neighbor (Studi Kasus : Data Status Gizi Balita di Puskesmas Baqa Samarinda Seberang)," *Prosiding Seminar Nasional Matematika, Statistika, dan Aplikasinya*, vol. 3, pp. 128–142, 2023.

[28] I. Belcic, "What is Classification in Machine Learning? | IBM," 2024. [Online]. Available: https://www.ibm.com/think/topics/classification-machine-learning

[29] J. C. Mestika, M. O. Selan, and M. I. Qadafi, "Menjelajahi Teknik-Teknik Supervised Learning untuk Pemodelan Prediktif Menggunakan Python," *Buletin Ilmiah Ilmu Komputer dan Multimedia (BIIKMA)*, vol. 1, no. 1, pp. 216–219, 2023, [Online]. Available: https://jurnalmahasiswa.com/index.php/biikma/article/view/101

[30] M. A. Satriawan and W. Widhiarso, "Klasifikasi Pengenalan Wajah Untuk Mengetahui Jenis Kelamin Menggunakan Metode Convolutional Neural Network," *Jurnal Algoritme*, vol. 4, no. 1, pp. 43–52, 2023, doi: 10.35957/algoritme.xxxx.