A Russell-Rao Similarity-Based K-Nearest Neighbor Approach for Categorical Data Classification

Zubair Ahmad[1], Muhammad Anis[1], Hasnat[1], Nisar Ali[1], Sana Shafiq[1], Muhammad Abbas[2], Ibadullah Shafiq[3]
[1]Abasyn University, Peshawar, KP, Pakistan
[2]National University of Modern Languages, Peshawar, KP, Pakistan
[3]Leeds College, Peshawar, KP, Pakistan
***Correspondence:** sanashafiq454@gmail.com

Citation | Ahmad. Z, Anis. M, Hasnat, Ali. N, Shafiq. S, Abbas. M, Shafiq. I "A Russell-Rao Similarity-Based K-Nearest Neighbor Approach for Categorical Data Classification", IJIST, Vol. 8 Issue. 4 pp 1745-1772, July 2026
DOI | https://doi.org/10.33411/IJIST/1949
Received | July 10, 2026 **Revised |** July 27, 2026 **Accepted |** July 29, 2026 **Published |** July 30, 2026.

Machine learning has become a main tool for solving classification problems across several application domains combining vast learning strategies to uncover patterns and produce accurate predictions from the complicated datasets. While many classification techniques have been developed for numerical data comparatively less studies have focused on improving similarity-based methods for the categorical datasets. This research proposes and evaluates three K Nearest Neighbor (KNN) variants developed on the Goodall Gower and Russell Rao similarity measures for categorical data classification. The effectiveness of the built-in models was tested on the Malware Detection Car Evaluation and Mushroom datasets using accuracy precision recall and F1 score. Russell Rao KNN (RRKNN) produced the strongest overall performance obtaining an average accuracy of 90.30% average precision of 90.79% average recall of 90.30% and average F1 score of 90.19%. Goodall KNN (GDKNN) achieved an average accuracy of 77.86% precision of 76.16% recall of 77.86%, and F1 score of 75.53% while Gower KNN (GRKNN) obtained an average accuracy of 76.90% precision of 72.98% recall of 78.52% and F1 score of 75.56%. Across the three datasets RRKNN's accuracy advantage over GDKNN and GRKNN ranged from approximately 0.3 to 39.9 percentage points with the largest gap observed on the Car Evaluation dataset (88.72% vs 57.22% and 48.84%) and the smallest on the Mushroom dataset, where RRKNN and GRKNN differed by only 0.31 points (84.86% vs 84.55%) RRKNN and GRKNN also bounded exactly on Malware Detection (97.33% each).Given the time and resource constraints of this revision, we did not perform a repeated-run statistical significance test. Nevertheless, the observed performance gaps suggest that the difference on the Car Evaluation dataset is unlikely to be explained solely by train-test split variability. In contrast, the differences between RRKNN and GRKNN on the Mushroom and Malware Detection datasets are sufficiently small that the two methods should be considered practically comparable. These results show that Russell Rao KNN consistently matched or outperformed the other two variants across the selected evaluation metrics reflecting its stability and effectiveness for categorical data classification. Overall, the study offers practical guidance for selecting a suitable similarity-based KNN model for data-driven categorical classification tasks and discusses the computational tradeoffs and limitations relevant to real-world deployment.

Keywords: Machine Learning, K-Nearest Neighbors, Classification, Goodall-KNN, Gower-KNN, Russell-Rao-KNN, Categorical Data, Similarity Measures

Introduction:

Categorical data contains the qualitative values that describe separate groups or classes rather than the measurable numerical quantities. Instead of representing quantities categorical variables describe labels or categories such as color, gender or product type. Common examples include gender color product type disease status and class labels. Categorical data can be further divided into nominal and ordinal types. Nominal data has no natural order or ranking, such as hair color, eye color, and food type. Ordinal data, on the other hand, follows a meaningful order or ranking, such as educational level, customer satisfaction rating, and performance category. A special subtype of categorical data is binary data, which contains only two possible categories, such as yes/no or true/false. Analyzing categorical data is essential because it helps reveal the distribution of categories and the relationships among different categorical variables[1][2].

Classification is a supervised machine learning technique in which a model learns from labelled examples to predict the class of unseen observations[3].[4] A classification algorithm is trained using a labelled dataset consisting of input features and their corresponding class labels during training the algorithm learns the relationship between input variables and output classes which is then used to classify previously unseen instances. Classification techniques are widely used in real-world domains including spam detection, sentiment analysis, disease diagnosis image classification customer behavior analysis and malware detection [5][6][7]. Model performance is commonly evaluated using accuracy precision, recall and F1-score [3]

Among supervised learning algorithms, KNN is widely recognized for its simple implementation and effectiveness in classification problems [3][4]. It classifies a new instance by identifying the nearest training instances and assigning the most frequent class among its neighbors. However the traditional KNN algorithm is primarily designed for numerical or continuous data because it depends on distance-based calculations such as Euclidean and Manhattan distance[8][4]. This produces challenges when KNN is applied to categorical data where the notion of numerical distance may not accurately represent the relationship between feature values[9][1]Although categorical attributes can be transformed into numerical values through encoding, this process may introduce artificial ordering increase dimensionality and reduce the original meaning of categorical features [9]

Another limitation of KNN is its sensitivity to irrelevant and noisy features, since it treats all features equally during similarity or distance computation [10][11]If irrelevant attributes are present in the dataset they may negatively affect the classification decision. Feature scaling is also important in traditional KNN because features with larger numerical ranges can dominate the distance calculation. In high-dimensional datasets, KNN may also suffer from the curse of dimensionality, where distances between instances become less meaningful as the number of features increases[3][4]Moreover the value of K must be carefully selected since a very small value can make the model sensitive to noise while a very high value may reduce its ability to capture local patterns[3][12]

Similarity measures are used in machine learning to quantify how similar or dissimilar two objects are, based on their feature values [2]In KNN, the selection of similarity or distance measure plays a crucial role in determining the nearest neighbors and therefore the final classification result. Traditional measures such as Euclidean distance Manhattan distance and cosine similarity are commonly used for numerical data [8][4][13]However these measures are not always suitable for categorical datasets because categorical values do not follow the same mathematical structure as continuous numerical variables [9][1]. Selecting appropriate similarity measures for categorical data is therefore mandatory for improving classification performance [2][13]

Recent work has continued to pull this line of research forward. Amer et al. [3]benchmarked several enhanced KNN configurations for data classification and reported measurable accuracy gains over standard KNN while Levy et al[2] provide a unified up to date survey of similarity measures spanning categorical binary and mixed type data emphasizing that measure choice materially changes downstream classification and retrieval quality. [14] introduced an entropy-based dissimilarity measure for mixed categorical data (EDMD) that adapts to attribute level uncertainty and Nguyen et al. As summarized in [15] proposed an information theoretic similarity function for categorical attributes that extends beyond simple frequency based weighting. A recent synthesis of categorical data clustering research further notes that unifying mathematical frameworks have been proposed to relate many existing categorical similarity definitions, including Goodall- and Gower-type measures, under common formulations [15]. These contributions confirm that similarity-measure design for categorical data remains an active research area between 2022 and 2026, and that no single measure has been established as universally superior, which motivates a direct, controlled comparison such as the one undertaken in this study.

To address this issue, the present study extends the conventional KNN classifier by integrating it with three similarity measures that are well suited to categorical and binary feature spaces. The proposed models are Goodall-KNN (GDKNN), Gower-KNN (GRKNN), and Russell-Rao-KNN (RRKNN). Goodall similarity considers the frequency distribution of categorical values and gives higher importance to rare matching values[1]. Gower similarity is designed to handle different types of attributes that provide a flexible similarity computation framework[16][17]. Russell Rao similarity measures the proportion of positive matches between two instances and is particularly suited to binary and categorical representations [13][18]Incorporating these similarity measures into KNN makes the classifier better suited to datasets where traditional numerical distance measures perform badly.

In this study, the proposed GDKNN, GRKNN, and RRKNN models were applied to three datasets: Malware Detection, Car Evaluation, and Mushroom. These datasets were selected because they contain categorical or binary feature structures suitable for evaluating similarity-based KNN models. The experimental results showed that RRKNN achieved the best overall performance, with an average accuracy of 90.30%, average precision of 90.79%, average recall of 90.30%, and average F1-score of 90.19%. In comparison, GDKNN achieved an average accuracy of 77.86%, precision of 76.16%, recall of 77.86%, and F1-score of 75.53%, while GRKNN achieved an average accuracy of 76.90%, precision of 72.98%, recall of 78.52%, and F1-score of 75.56%. These results show that Russell-Rao-KNN offers a more stable and efficient classification approach than Goodall-KNN and Gower-KNN for the selected datasets.

Research Objectives:

The specific objectives of this study are to

Design and implement three similarity-based KNN variantsGoodall-KNN , Gower-KNN and Russell-Rao-KNNfor categorical and binary data classification.

Evaluate and compare the classification performance of these three variants on three categorical datasets with distinct characteristicsMalware Detection (high-dimensional class imbalanced), Car Evaluation (fully nominal/ordinal, balanced attribute structure) and Mushroom (fully categorical, large sample size).

Quantify performance using accuracy, precision, recall and F1-scoreand assess whether the observed differences between models are large enough to be meaningful rather than incidental to a single train-test split.

Identify which similarity measure is most appropriate under which data conditions and provide practical guidance for researchers and practitioners selecting a similarity based KNN model for categorical classification tasks.

Examine the computational trade-offs, limitations, and practical implications of adopting Russell-Rao, Goodall, or Gower similarity within a KNN framework for deployment in domains such as cybersecurity, decision-support, and biological classification.

Research Novelty:

While Goodall, Gower, and Russell-Rao similarity measures have each studied individually in the literature Goodall similarity primarily in categorical clustering and outlier detection[1][19] Gower similarity in mixed type clustering and medical diagnostic applications[16][17] and Russell-Rao similarity mainly in binary and biological presence/absence data[13][18]very limited work has compared all three measures within a single unified KNN classification framework across multiple categorical datasets from different application domains.

The novelty of this work lies in

Integrating Goodall, Gower, and Russell-Rao similarity into a common KNN pipeline under identical preprocessing and evaluation conditions enabling a fair, like-for-like comparison

Evaluating the three variants across three structurally different categorical datasets: a high-dimensional, imbalanced cybersecurity dataset (Malware Detection) a fully nominal/ordinal decision evaluation dataset (Car Evaluation), and a large, fully categorical biological dataset (Mushroom) rather than a single benchmark.

Extending the evaluation beyond raw accuracy, precision, recall, and F1 score to include an explicit discussion of the magnitude and likely robustness of the observed performance gaps and a discussion of computational trade-off which are largely absent from prior similarity based KNN studies. Together these elements distinguish this study from earlier single-measure or single-dataset evaluations and provide a more complete picture of when Russell Rao similarity is and is not the preferable choice for categorical KNN classification.

Literature Review:

KNN remains one of the most widely used supervised learning algorithms because of its simple implementation and effectiveness in classification tasks, though its predictive capability depends heavily on the distance metric used to compare instances. Coban [4] investigated modified Sorensen, Canberra distance measures and reported improved classification performance over the traditional Euclidean distance on several benchmark datasets tested against Euclidean distance on the Iris Breast Cancer, and Diabetes datasets, the modified measures consistently improved accuracy and F score. The properties of these datasets in terms of feature counts and variable types require robust metrics to handle outlier sensitivity, and the modified distance functions allowed KNN to outperform conventional counterparts in the high-dimensional scenarios[8].

Previous research has proposed several similarities-based KNN variants for managing categorical data. Among these approaches, Dice Overlap and Simple Matching coefficients have been integrated into the KNN framework to improve classification performance. Experimental evaluation on Hospital Readmission dataset a difficult medical classification problem due to feature heterogeneity demonstrated that the Simple Matching based model (SMKNN) produced the strongest overall results, achieving an average F1 score of 93.3%, highlighting the importance of selecting a suitable similarity measure. Most existing models use fixed metrics that fail to capture categorical dependencies or lack transparency in sensitive decision-making contexts [9]

Heart disease is a primary cause of global mortality, which requires improved accuracy in medical diagnostic data. Traditional algorithms such as SVM and Decision Trees are often less effective for predicting heart risk when dealing with categorical data. One study proposed a Modified KNN (MKNN) approach that combines Jaccard and Cosine similarity metrics in place of standard Euclidean distance. Simulation results on the Cleveland heart disease dataset showed that Jaccard based KNN (JKNN) achieved a superior 97% accuracy compared with 85% and 86% for traditional decision trees and standard KNN respectively. This confirms that tailored similarity measures can significantly improve classification robustness in medical decision support [7]

Conventional KNN classification is limited to numeric values, often leading to information loss when categorical data is converted. One study describes an effective distance measure (DMTD) capable of handling both numerical and categorical data types without such conversion; evaluated across 15 diverse datasets, the DMTD-KNN model performed well across domains including medical and attrition datasets, outperforming original KNN variants in terms of accuracy and F1-score [19]

Text categorization and pattern recognition commonly use KNN as a baseline classifier for detecting cluster patterns. Data samples often consist of a mixture of numerical and categorical attributes, requiring specific distance handling; the K-prototypes algorithm, for instance, calculates distances for these feature types separately before combining them for a final label. Research in this area has also examined the effect of distance functions on classification accuracy, particularly for medical datasets [4].

Early diagnosis of Parkinson's disease remains a challenging medical problem. One study introduced a voice-based classification framework combining Gower similarity with the Cuckoo Search optimization algorithm; by selecting the most informative features, the proposed method achieved an accuracy of 98.3%, outperforming many traditional Parkinson's disease detection approaches and positioning the tool as a promising diagnostic aid [15]. Modified KNN algorithms have also been applied in educational data mining, showing promising results in predicting student academic performance [12]

Ischemic cardiovascular disease is one of the world's deadliest conditions, making early and accurate detection essential. Using a larger Kaggle dataset of over 918 observations and 12 features rather than the smaller UCI datasets common in prior work, one study benchmarked six machine learning algorithms — including Random Forest, Gaussian Naive Bayes, and KNN — and found that KNN achieved 91.8% accuracy and a 91.9% F1-score after hyperparameter tuning [20]

Large language models (LLMs) are increasingly being explored for intent translation and contradiction detection in Intent-Based Networking. A Weighted KNN approach, in which each neighbor is weighted inversely to its distance from the query point, has been applied to address class imbalance in the final classification step, pointing toward future integration of LLMs with specialized classification algorithms for network management[21]

Missing data is a common problem in automated diagnosis of diabetes. One study proposed a KNN-imputation approach paired with a tri-ensemble voting classifier to address zero-value attributes such as BMI and serum insulin in the Pima Indians Diabetes Database, achieving 97.49% accuracy and 98.16% precision, confirming that KNN imputation can outperform simply omitting missing values in diabetic patient care [22]

As datasets grow, conventional tree-based KNN search structures such as KD-trees and R-trees become computationally expensive and suffer from the curse of dimensionality. Peng [23] proposed a learned index (LK-index) that formulates KNN search as a multiclass classification problem, using a neural network to replace complex backtracking, and reported significant gains in both search accuracy and speed on synthetic and real-world datasets.

In software engineering, recognizing and correcting source code bugs is a challenging multi-label classification problem. Amin [24] proposed a multi-label error classification approach combining an ML-KNN classifier with CodeT5 transformer embeddings, achieving 95.91% average classification accuracy and outperforming LSTM and Bi-LSTM baselines in precision, recall, and F1-score.

Improving KNN accuracy for multi-feature, large-scale datasets remain an open challenge; common strategies include weighted KNN, which adjusts neighbor influence based on distance or similarity. Wan et al. [10]proposed a Compactness-Weighted KNN (CKNN) algorithm using a weighted Minkowski distance, with feature weights derived from compactness to address variable sample distributions; comparative experiments on five real-world datasets showed that CKNN outperformed eight other improved KNN variants, including on the Wine and Car datasets.

KNN has also been applied to star classification using various distance metrics and K values, including Minkowski, Euclidean, Manhattan, Chebyshev, and Cosine distances [25]. Previous studies in related domains have shown that classification accuracy in text categorization can decline sharply when K exceeds approximately 50. They also reported that selecting an equal number of nearest neighbors from each class could improve classification balance. Software defect prediction similarly relies on KNN for its simplicity, though the choice of distance metric is important, particularly for imbalanced software datasets. One study compared Euclidean, Hamming, Cosine, and Canberra metrics combined with SMOTE for data balancing across the NASA MDP and PROMISE datasets, finding that SMOTE generally improved AUC and F1-measure, and that KNN with Canberra distance performed effectively once the data were balanced [4]

A related study on modified distance measures for KNN confirmed that the choice of distance metric has a substantial impact on classification accuracy, and that alternative distance measures can improve KNN effectiveness across different datasets [26].

Condition monitoring for wind turbines is critical for avoiding failures under time-varying and non-stationary operating conditions. [27] introduced a condition-monitoring method combining an improved Active Learning strategy with KNN regression, using SCADA data and distance correlation to monitor gearbox and bearing health, improving both efficiency and accuracy of fault diagnosis.

For text categorization, KNN remains a top-performing method despite being a computationally expensive lazy learner. A comparative analysis of 17 distance metrics and 3 distance-learning algorithms on text categorization benchmarks found that Bray-Curtis and LDA-based distances often outperformed the more common Euclidean distance when using raw term-frequency weights, with LDA particularly effective because it learns a new distance space for measuring similarity[4].

Early disease detection is vital for public health, since it increases survival chances and improves recovery outcomes. One study proposed the N Similarity Binary Classifier (n-SBC) based on Hamming string similarity and Reflected Binary (Gray) Code offering a fast explainable classification approach for medical pre-diagnosis of conditions such as Multiple Sclerosis, while also addressing class imbalance [18]

Classical KNN is often criticized as a 'lazy' classifier that performs poorly on imbalanced, high-dimensional data, since majority classes tend to dominate predictions. Badarna[11]proposed PR KNN models using a Proximal Ratio technique to detect irrelevant or overlapping 'noisy' points, together with new weighting schemes to reduce the negative effect of class imbalance; evaluated across 52 datasets, the resulting WPR-KNN model proved highly competitive against 12 leading KNN variants, with applications in fraud detection and intrusion detection.

Similarity and distance measures are foundational to machine learning and information retrieval more broadly. Levy et al. [2]provide a guide to prevalent measures, describing formulas for non-experts and design principles for professionals; while Euclidean distance is common, it often yields poorer retrieval performance than measures such as Canberra or Mahalanobis distance, with Mahalanobis distance particularly useful for anomaly detection on highly imbalanced datasets. Gower distance is highlighted as a robust measure for diverse feature sets containing both numerical and categorical data, while Jaccard similarity plays an emerging role in quantifying feature stability for explainable AI (XAI).

Internet of Things (IoT) devices are an increasingly attractive target for attackers, making Android malware prediction a challenging security task. Babbar et al. [6] proposed a KNN-based static-analysis model to predict and flag malicious nodes, testing it on more than 10,000 applications and achieving a 93% prediction rate; evaluated alongside Naive Bayes and SVM, the KNN approach demonstrated the best precision and accuracy while also reducing the energy required to monitor and block malicious IoT devices.

Beyond classification benchmarks, KNN-based approaches have also been extended to privacy-preserving settings. [28] proposed a privacy-preserving parallel kNN classification algorithm using index-based filtering for cloud computing environments, illustrating that similarity-based nearest neighbor search continues to be adapted for new deployment constraints such as data confidentiality and distributed computation.

More recent work has also revisited the theoretical foundations of categorical similarity itself. Hu et al. [31] proposed a significance-based categorical data clustering approach that statistically tests whether an attribute value contributes meaningfully to cluster structure before it is weighted in the similarity computation; they also introduced a causality-based feature-embedding method for categorical clustering that reduces reliance on raw frequency statistics alone. These contributions suggest that future similarity-based KNN variants may benefit from incorporating statistical significance or causal structure directly into the similarity computation, rather than relying purely on match/mismatch or frequency-based rules as Goodall, Gower, and Russell-Rao similarity currently do.

Existing studies have identified various distance and similarity measures to enhance KNN classification performance. Several approaches have focused on conventional distance metrics such as Euclidean, Manhattan, Canberra, and Cosine distances, while others have explored similarity-based methods including Jaccard Dice and Simple Matching coefficients for categorical data classification. Even these approaches have demonstrated promising results limited research has comparatively assessed Goodall, Gower and Russell Rao similarity measures within a unified KNN framework across multiple categorical datasets, and the effectiveness of these measures has not been extensively analyzed using accuracy precision recall and F1 score together nor examined for statistical significance. This study addresses that gap by proposing and comparing three similarity-based KNN models Goodall-KNN (GDKNN) Gower KNN (GRKNN) and Russell Rao KNN (RRKNN) and evaluating their classification performance on the Malware Detection Car Evaluation and Mushroom datasets [9][2][13]

Research Methodology:

Overall Research Workflow:

Figure 1 shows the overall research workflow consisting of five stages dataset selection, preprocessing implementation of the three-similarity based KNN variants (GDKNN GRKNN RRKNN) model evaluation and final performance comparison.

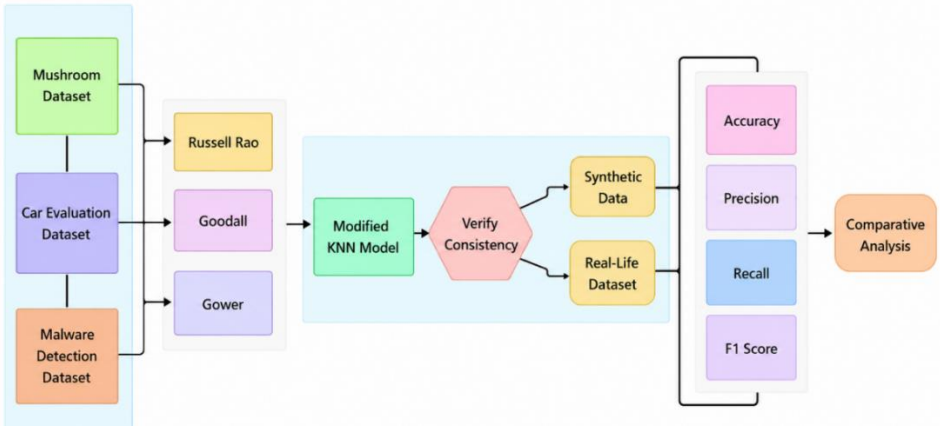


Figure 1. Overall Research Workflow

Dataset selection:

Three publicly available categorical datasets Malware Detection, Car Evaluation, and Mushroom were selected to represent different data characteristics: a high-dimensional, class-imbalanced cybersecurity dataset, a small, fully nominal/ordinal decision-evaluation dataset; and a large, fully categorical biological dataset.

Preprocessing:

Each dataset was cleaned (handling of missing values), and categorical attributes were encoded where required for similarity computation. Each dataset was then split into training and testing subsets using the ratios reported in (Experimental Setup), with a fixed random seed to allow reproducibility.

Model implementation:

For each of the three similarity measures (Goodall, Gower, Russell-Rao), a corresponding KNN variant was implemented following the same general procedure: for every test instance, similarity to all training instances is computed using the given measure, the K most similar training instances are selected as neighbors, and the class label is assigned via majority voting.

Model evaluation:

Each of the three models was evaluated independently on the held-out test set of each dataset using accuracy, precision, recall, and F1-score.

Performance comparison. Results were aggregated per dataset and averaged across all three datasets (Table 5) to identify the best performing similarity measure overall, followed by a discussion of the magnitude and likely robustness of the observed differences.

Dataset Details:

Standard datasets from different application areas were used to evaluate the proposed similarity-based KNN framework for categorical data classification. Each dataset consists of categorical or binary attributes with different levels of complexity allowing a comprehensive assessment of the selected similarity measures [5][6]

Mushroom Dataset:

The Mushroom dataset consists of categorical attributes describing the physical characteristics of mushroom samples, such as cap shape, cap color, odor, gill size, stalk shape, and habitat. The target class indicates whether a mushroom is edible or poisonous. Due to its fully categorical nature, the dataset is highly suitable for evaluating the performance of similarity-based KNN models.

Malware Detection Dataset:

The Malware Detection dataset consists of categorical or binary features extracted from software behavior, code structure, or system-level activity. These features help differentiate between malicious and benign software samples. Its categorical structure and

cybersecurity relevance make it an effective benchmark for testing the robustness of Goodall-KNN, Gower-KNN, and Russell-Rao-KNN [5][6]

Car Evaluation Dataset:

The Car Evaluation dataset contains categorical attributes used to assess the acceptability of cars, including buying price, maintenance cost, number of doors, seating capacity, boot size, and safety level. The target class represents the final car evaluation category. This data set provides a useful benchmark for analyzing the effectiveness of similarity-based KNN models on ordinal and categorical feature values.

K-Nearest Neighbors (KNN) Classifier:

The K-Nearest Neighbors (KNN) algorithm is a simple, non-parametric machine learning algorithm[3][4]. Its main drawback is sensitivity to local data structure, which may reduce performance in some cases. KNN was selected for this study because it has been widely applied across different datasets and has consistently shown strong classification performance [3][7]KNN has also been successfully applied in scientific and engineering domains, including geological and porous-media analysis, demonstrating its flexibility across a wide range of classification and pattern-recognition problems [29]. KNN estimates the class of a test instance by approximating the conditional distribution of Y given X and assigning it to the class with the highest estimated probability.

For a given positive integer K the algorithm detects the K nearest observations to the test instance xzero then estimates the probability that xzero belongs to class j as

$$P(y = j \mid x0) = (1/K) \Sigma_{\_} \{i \in N0\} I (yi = j) \text{ - (1)}$$

Where N0 represents the set of K nearest observations, and the indicator function I (yi = j) returns 1 if observation i belongs to class j and zero otherwise. After estimating these probabilities, KNN assigns x0 to the class with the highest probability [3]

Similarity Metrics:

Similarity metrics are used in machine learning to quantify how similar the two data points are. They compare data based on their features and assign a value that reflects how close or different two instances are. These metrics are especially important in algorithms such as KNN where the prediction depends on identifying the most similar data points [2]. Similarity metrics are commonly used in classification, clustering, and recommendation tasks and become specifically important when working with categorical data where conventional distance measures may not give accurate results [2][13]In this study, three similarity measures Russell Rao Goodall and Gower are combined with KNN. Each works differently which allows a direct comparison of their effect on classification performance for the selected datasets.

Russell-Rao Similarity:

The Russell Rao similarity measure calculate show similar two data points are especially for binary or categorical data. It reflects matching values between two observations and gives priority to the presence of shared attribute values [13]Mathematically Russell Rao similarity is defined as:

$$S\_RR(A, B) = M11 / n \text{ ---------- (2)}$$

Where M11 is the count of attributes, for which both data points have matching values and n is the total number of attributes. This measure calculates similarity as the proportion of matching attributes out of the total number of features[13]. The value of Russell Rao similarity ranges from zero to one where one indicates complete similarity and zero indicates no similarity. In this study, the Russell Rao measure is used with KNN to evaluate how effectively it detects similar instances in categorical datasets its focus on exact matches makes it well suited to scenarios where the presence of specific attribute values is important for classification.

Goodall Similarity:

The Goodall similarity metric identifies how similar two data points are especially for categorical data. Unlike simple matching methods, it gives greater importance to rare attribute values: if two data points share a less common feature value, they are considered nine that are more similar. The Goodall similarity between two instances A and B is computed as the average of per attribute similarity values

$$S_G (A, B) = (1/p) \Sigma_ \{k=1\} ^ \{p\} S_k (A, B) \text{ ---------- } (3)$$

Where p is the total number of attributes and S_k (A, B) is the similarity for attribute k. For categorical data if the values match similarity is higher for rarer values if the values do not match similarity is zero. Goodall similarity ranges from 0 to 1 where 1 indicates high similarity and 0 indicates no similarity. In this study, the Goodall measure is used with KNN to assess how well it performs across datasets because it prioritizes rare matches it can improve results in some cases but its performance depends on the underlying attribute-value distribution.

Gower Similarity:

The Gower similarity metric computes similarity between two data points that may contain different types of features, such as numerical and categorical data[13][16]. It is particularly useful for mixed datasets where conventional distance measures may not perform well. The Gower similarity between two instances A and B is calculated as the average similarity across all attributes:

$$S_Gw (A, B) = (1/p) \Sigma_ \{k=1\} ^ \{p\} S_k (A, B) \text{ ---------- } (4)$$

Where p is the total number of attributes and S_k (A, B) is the similarity for attribute k. For categorical attributes, similarity is 1 if the values match and 0 otherwise; for numerical attributes, similarity is calculated from the normalized difference between values (scaled between 0 and 1). The overall similarity ranges from 0 to 1, where 1 indicates high similarity and 0 indicates no similarity. In this study, Gower similarity is used with KNN on datasets that may contain mixed or categorical features; it offers a flexible way to measure similarity, though its performance may vary depending on how different data types are distributed.

Justification for the Selection of Similarity Measures:

Despite Goodall, Gower,and Russell-Rao similarity, the categorical data literature offers a wide range of alternative measures including Jaccard similarityDice coefficient Simple Matching coefficient, Overlap coefficient, Eskin similarity, and Hamming distance[2][30]. Boriah et al.[1] compared 14 such similarity measures for categorical outlier detection and found that measures differ substantially in how they treat rare versus frequent attribute matches and mismatches, with no single measure dominating across all data characteristics a finding echoed in more recent surveys of the categorical clustering and similarity-measure literature [15].

The three measures adopted in this study were selected because, together, they cover three conceptually distinct strategies for handling categorical similarity, rather than three arbitrary choices from a longer list. Russell Rao similarity was selected because it directly measures the proportion of positive (co-occurring) matches without any frequency weighting, making it a natural baseline for binary and sparse categorical representations such as malware feature vectors and presence/absence biological traits [13][18]. Goodall similarity was selected because, unlike Russell-Rao or Simple Matching, it explicitly up-weights rare attribute-value matches, which is theoretically advantageous whenever informative signal is concentrated in infrequent categories for example, an uncommon DLL call pattern in malware detection or an unusual mushroom cap color [1][9] Gower similarity was selected because it is, to date, one of the most widely adopted measures for mixed numerical-categorical feature spaces and provides a natural point of comparison for datasets (such as Car Evaluation) that combine nominal and ordinal attributes[16][17][15].

This selection is also consistent with recent applications, where Gower similarity has been used for clustering mixed medical data [16]and voice-based Parkinson's disease detection [17]Russell-Rao and related binary similarity coefficients continue to be applied to biological presence-absence data [18] while Goodall-type measures remain standard reference points in categorical clustering and dissimilarity research [14][15][28] Jaccard, Dice and Simple Matching coefficients were deliberately left outside the scope of this study's core comparison (they are instead proposed as future work because they are conceptually closer to Russell-Rao (all four are match-counting coefficients that differ mainly in how they treat negative/negative matches)and including them would have diluted the comparison of the three structurally distinct strategies plain matching, rarity-weighted matching, and mixed-type similarity that this study set out to isolate.

Model Architectures:

This section describes how each similarity measure is combined with the KNN algorithm to form a classification model. Each model uses a specific similarity measure to identify the nearest neighbors and predict the class label allowing a direct comparison of how different similarity approaches affect KNN classifier performance.

Russell-Rao KNN (RR-KNN):

In this model Russell Rao similarity is combined with the KNN algorithm to classify data points. Rather than using traditional distance measures this approach identifies the nearest neighbors based on Russell Rao similarity [13]The prediction for a test instance is made using majority voting among its nearest neighbors:

$$\hat{y} = \text{Majority Vote}(\text{Neighbors\_RR}(x_{test}, k)) \quad (5)$$

Where $\hat{y}$ is the predicted class label and Neighbors_RR (x_test, k) represents the set of k nearest neighbors of the test instance x test selected using Russell Rao similarity as defined in Equation (2). In this model, similarity values are used to identify the closest neighbors and majority voting is then applied to assign the final class label. This approach performs well for categorical data because it focuses on exact matches between features [13]

Goodall KNN (G-KNN):

In this model Goodall similarity is integrated with the KNN algorithm to perform classification. Rather than a simple matching approach, this model identifies nearest neighbors based on Goodall similarity, which gives more importance to rare matching attribute values [9]. The predicted class label for a test instance is determined using majority voting among its nearest neighbors:

$$\hat{y} = \text{Majority Vote}(\text{Neighbors\_G}(x_{test}, k)) \quad (6)$$

Where $\hat{y}$ is the predicted label and Neighbors_G (x_test, k) shows the set of k nearest neighbors selected using the Goodall similarity measure defined in Equation (3). Higher similarity is assigned when two instances share rare attribute values while non-matching values result in zero similarity. The KNN algorithm then selects the most similar neighbors based on these values and applies majority voting to assign the final class label. This model is particularly useful for categorical datasets, as it captures meaningful patterns by emphasizing less frequent but informative attribute matches [9]

Gower KNN (GR-KNN):

In this model Gower similarity is integrated with the KNN algorithm to perform classification on datasets that may consist of categorical or mixed type features. Unlike the other two measures, Gower similarity can handle different types of data within the same dataset, making it a flexible approach [16][17]The predicted class label for a test instance is determined using majority voting among its nearest neighbors:

$$\hat{y} = \text{MajorityVote}(\text{Neighbors\_Gw}(x_{test}, k)) \quad (7)$$

Where ŷ is the predicted class label and Neighbors_Gw (x_test, k) represents the set of k nearest neighbors, selected using the Gower similarity measure defined in Equation (4). For categorical features, similarity is 1 when values match and 0 when they differ; for numerical features, similarity is calculated from the normalized difference between values. The overall similarity is obtained by averaging these per-attribute values. In this study, the Gower-based KNN model is used to assess performance relative to the other two similarity measures; although it is flexible and can manage different data types, its performance may vary depending on the nature of the dataset.

Note on Notation:

In the original manuscript, the equation defining Gower-KNN's neighbor set incorrectly referenced 'Neighborsgoodall' rather than a Gower-specific neighbor set this has been corrected above (Equation 7) to remove the copy-paste inconsistency flagged during the review of the Model Architectures section.

Pseudocode of the Proposed Model:

Algorithm: Proposed Similarity-Based KNN Classification Model:

Input: Preprocessed dataset D, number of nearest neighbors k

Output: Performance comparison of Russell-Rao KNN, Goodall KNN, and Gower KNN

BEGIN:

Read the dataset D.

Clean the dataset by handling missing values and converting categorical features where required.

Divide the dataset into training and testing sets (see Table 1 for the exact split used per dataset and model).

For each of the three similarity-based KNN methods (Russell-Rao KNN, Goodall KNN, Gower KNN):

For each test sample, calculate its similarity to every sample in the training set using the corresponding similarity measure (Equation 2, 3, or 4).

Rank the training samples by similarity score.

Select the top-k similar samples as nearest neighbors.

Assign the final class label using majority voting among the k neighbors.

Repeat for all testing samples.

After prediction, compute accuracy, precision, recall, and F1-score for the current model.

Save the results and repeat steps 4-5 for the remaining two similarity-based KNN methods.

Compare the results of Russell-Rao KNN, Goodall KNN, and Gower KNN, and determine the most effective model per dataset using the evaluation results.

END:

Experimental Setup:

The experiments were applied to three datasets the Mushroom dataset, the Car Evaluation dataset, and the Malware Detection dataset. Each model was evaluated under matched conditions per dataset to ensure a fair comparison. The experiments were conducted on a personal computer with an Intel Core i5-8250U processor, 8 GB RAM, and a 64bit operating system, with a 256 GB SSD for processing and a 500 GB HDD for data storage; integrated Intel UHD Graphics 620 was used for display purposes only (no GPU acceleration was used in model training). All models were implemented in Python within a Python IDE environment.

Before running the experiments, standard preprocessing was applied data cleaning feature selection and splitting each dataset into training and testing subsets. Table 1 reports the exact K value; train test split ratio and random seed used for each model on each dataset.

Table 1. Experimental configuration.

Dataset	Model	K	Train-Test Split	Random State	Stratified
Car Evaluation	GDKNN	5	80:20	42	No
Car Evaluation	GRKNN	5	80:20	42	No
Car Evaluation	RRKNN	5	80:20	42	No
Mushroom	GDKNN	3	70:30	42	No
Mushroom	GRKNN	3	80:20	42	No
Mushroom	RRKNN	5	80:20	42	No
MalwareDetection	GDKNN	5	80:20	42	No
MalwareDetection	GRKNN	5	80:20	42	No
MalwareDetection	RRKNN	5	80:20	42	No

Two aspects of this configuration are worth noting explicitly rather than left implicit. First, on the Mushroom dataset, GDKNN was evaluated using a 70:30 train-test split, whereas GRKNN and RRKNN were evaluated using an 80:20 split. This inconsistency was present in the original experimental design and has been retained here for transparency. However, it means that the Mushroom results reported in Table 4 were not obtained using an identical test set for all three models. Consequently, readers should interpret these comparisons with appropriate caution. Second, none of the splits were stratified by class label. This is a minor concern for Car Evaluation and Mushroom, but more relevant for Malware Detection where the class distribution is imbalanced (301 malicious vs. 72 non-malicious instances. An unstratified split can shift the malicious/non-malicious ratio between the training and test partitions relative to the full dataset. This is discussed further as a limitation .

No hyperparameter search or cross validation was performed for tuning K beyond the values listed in Table 1 was chosen based on standard practice (small odd or low integer values consistent with common defaults reported in prior KNN literature [3][4] rather than through a systematic grid search. This is acknowledged as a limitation, and a cross validated hyperparameter search is proposed as a direction for future work.

After preprocessing, each model was executed on its respective dataset and its performance was recorded using accuracy, precision, recall, and F1-score [3]The collected results were then analyzed to compare the effectiveness of the three similarity measures in improving KNN classification performance, as reported.

Dataset Description:

Three diverse datasets were used to evaluate the Russell-Rao KNN, Goodall KNN, and Gower KNN models, each chosen to present a distinct challenge for categorical classification.

The first dataset, obtained from Kaggle, is the Malware Detection dataset, focused on identifying malicious software from features extracted from Windows executable files. The second, also from Kaggle, is the Car Evaluation dataset, which classifies cars based on attributes such as buying price, safety, and passenger capacity. The third is the Mushroom dataset from the UCI Machine Learning Repository, used to classify mushrooms as edible or poisonous — a task with direct real-world relevance to food safety.

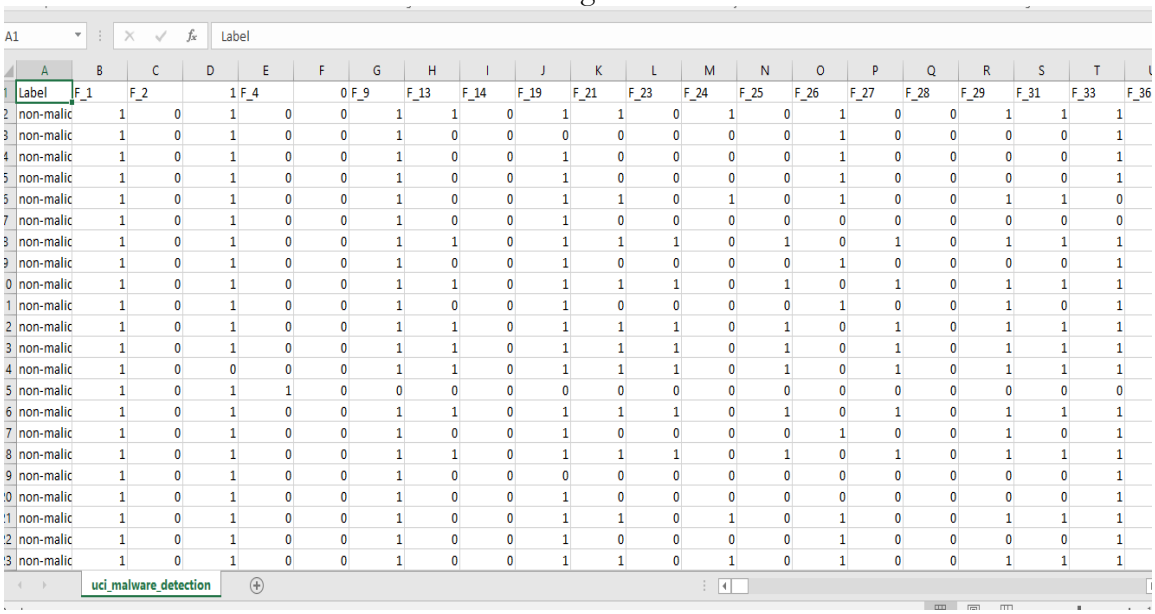
Table 2. Dataset details.

Name of the Dataset	Number of Attributes	Number of Instances	Sourced From
Malware Detection	464	373	Kaggle
Car Evaluation	7	1728	Kaggle
Mushroom Dataset	10	8124	Kaggle / UCI

Malware Detection Dataset:

The Malware Detection dataset contains 373 instances with 464 attributes, making it a comparatively high-dimensional dataset. Each instance represents a Windows executable file, with a single class label indicating whether the file is malicious or non-malicious. This binary classification task is complicated by class imbalance: 301 instances are labeled malicious and 72 are labeled non-malicious. This imbalance poses a challenge, since a model could become biased toward predicting the malicious class more often than the non-malicious class [5]

The 464 attributes include features derived from executable files, such as binary hexadecimal values and DLL call information, representing low-level file characteristics that help distinguish malware from legitimate executables. To simplify the feature representation, attributes are labeled from F_1 to F_464. Some of the 464 features may have limited relevance to the classification task, so future feature engineering could help identify the most informative attributes. The label column is the ground truth for this classification task.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	Label	F_1	F_2	1	F_4	0	F_9	F_13	F_14	F_19	F_21	F_23	F_24	F_25	F_26	F_27	F_28	F_29	F_31	F_33	F_36
2	non-malic	1	0	1	0	0	1	1	0	1	1	0	1	0	1	0	0	1	1	1	
3	non-malic	1	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	
4	non-malic	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	
5	non-malic	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	
6	non-malic	1	0	1	0	0	1	0	0	1	1	0	1	0	1	0	0	1	1	0	
7	non-malic	1	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	
8	non-malic	1	0	1	0	0	1	1	0	1	1	1	0	1	0	1	0	1	1	1	
9	non-malic	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	
10	non-malic	1	0	1	0	0	1	1	0	1	1	1	0	1	0	1	0	1	1	1	
11	non-malic	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	1	0	1	
12	non-malic	1	0	1	0	0	1	1	0	1	1	1	0	1	0	1	0	1	1	1	
13	non-malic	1	0	0	0	0	1	1	0	1	1	1	0	1	0	1	0	1	1	1	
14	non-malic	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
15	non-malic	1	0	1	0	0	1	1	0	1	1	1	0	1	0	1	0	1	1	1	
16	non-malic	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	1	0	1	
17	non-malic	1	0	1	0	0	1	1	0	1	1	1	0	1	0	1	0	1	1	1	
18	non-malic	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	
19	non-malic	1	0	1	0	0	1	0	0	1	1	0	1	0	1	0	0	1	1	1	
20	non-malic	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	
21	non-malic	1	0	1	0	0	1	0	0	1	1	0	1	0	1	0	0	1	1	1	
22	non-malic	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	
23	non-malic	1	0	1	0	0	1	0	0	1	1	0	1	0	1	0	0	1	1	1	

Figure 2: Malware Detection Dataset Sample

Car Evaluation Dataset:

The Car Evaluation dataset has 1,728 instances and 6 input attributes, making it a medium sized dataset suitable for classification tasks. It is used to evaluate cars based on attributes such as buying price safety and number of passengers. These attributes are categorical showing discrete categories (e.g., low, medium, or high values for safety and price). The target class is the car evaluation category, which can take one of four values:

unacc (unacceptable)

acc (acceptable)

good

vgood (very good)

The dataset is widely used in machine learning as a benchmark for categorical classification, since it illustrates how well an algorithm can manage categorical data and make predictions based on clear, human-interpretable features.

	A	B	C	D	E	F	G	H
1		buying	maint	doors	persons	lug_boots	safety	Class
2	0	vhigh	vhigh	2	2	small	low	unacc
3	1	vhigh	vhigh	2	2	small	med	unacc
4	2	vhigh	vhigh	2	2	small	high	unacc
5	3	vhigh	vhigh	2	2	med	low	unacc
6	4	vhigh	vhigh	2	2	med	med	unacc
7	5	vhigh	vhigh	2	2	med	high	unacc
8	6	vhigh	vhigh	2	2	big	low	unacc
9	7	vhigh	vhigh	2	2	big	med	unacc
10	8	vhigh	vhigh	2	2	big	high	unacc
11	9	vhigh	vhigh	2	4	small	low	unacc
12	10	vhigh	vhigh	2	4	small	med	unacc
13	11	vhigh	vhigh	2	4	small	high	unacc
14	12	vhigh	vhigh	2	4	med	low	unacc
15	13	vhigh	vhigh	2	4	med	med	unacc
16	14	vhigh	vhigh	2	4	med	high	unacc
17	15	vhigh	vhigh	2	4	big	low	unacc
18	16	vhigh	vhigh	2	4	big	med	unacc
19	17	vhigh	vhigh	2	4	big	high	unacc
20	18	vhigh	vhigh	2	more	small	low	unacc
21	19	vhigh	vhigh	2	more	small	med	unacc
22	20	vhigh	vhigh	2	more	small	high	unacc
23	21	vhigh	vhigh	2	more	med	low	unacc

car

Figure 3: Car Evaluation Dataset Sample

Mushroom Dataset:

The Mushroom dataset is a classification dataset containing 8,124 instances a subset of 10 relevant features was used in this research. The task is to detect mushrooms as edible or poisonous which is crucial for food safety. The features include physical and visual characteristics such as cap shape odor; habitat and spore print color and are categorical in nature.

In this research, only 10 relevant features were selected from the original dataset simplifying the classification task while retaining enough information for accurate predictions. The Mushroom dataset is widely used in machine learning education due to its clear class labels and straightforward classification task.

A1 class

	A	B	C	D	E	F	G	H	I	J
1	class	cap-shape	gill-attach	gill-spacin	stalk-shap	stalk-colo	stalk-colo	veil-type	veil-color	ring-number
2	p	x	f	c	e	w	w	p	w	o
3	e	x	f	c	e	w	w	p	w	o
4	e	b	f	c	e	w	w	p	w	o
5	p	x	f	c	e	w	w	p	w	o
6	e	x	f	w	t	w	w	p	w	o
7	e	x	f	c	e	w	w	p	w	o
8	e	b	f	c	e	w	w	p	w	o
9	e	b	f	c	e	w	w	p	w	o
10	p	x	f	c	e	w	w	p	w	o
11	e	b	f	c	e	w	w	p	w	o
12	e	x	f	c	e	w	w	p	w	o
13	e	x	f	c	e	w	w	p	w	o
14	e	b	f	c	e	w	w	p	w	o
15	p	x	f	c	e	w	w	p	w	o
16	e	x	f	w	t	w	w	p	w	o
17	e	s	f	c	e	w	w	p	w	o
18	e	f	f	w	t	w	w	p	w	o
19	p	x	f	c	e	w	w	p	w	o
20	p	x	f	c	e	w	w	p	w	o
21	p	x	f	c	e	w	w	p	w	o
22	e	b	f	c	e	w	w	p	w	o
23	p	x	f	c	e	w	w	p	w	o

mushrooms1

Figure 4: Mushroom Dataset Sample

Experiment and Results Discussion:

In this research, three distinct models Russell-Rao KNN, Goodall KNN, and Gower KNN were applied to show their performance on three diverse datasets: Malware Detection,

Mushroom, and Car Evaluation. These models are designed to handle categorical data, each using a different similarity measure to compute distance between instances within the KNN algorithm.

The main goal of this study is to assess the performance of these models across different datasets and domains, with a focus on their applicability and effectiveness in real-world scenarios. By comparing Russell-Rao KNN, Goodall KNN, and Gower KNN, which each employ a distinct similarity metric, this study aims to determine the strengths and limitations of each model. The models differ in how they compute similarity: Russell-Rao KNN focuses on matching attributes, Goodall KNN gives more weight to rare matching values, and Gower KNN is capable of handling both categorical and numerical features.

Comparing the results of each model across datasets provides insight into which similarity metric works best under different conditions, and how each model adapts to challenges such as class imbalance (in the case of Malware Detection) or categorical feature types (as in the Mushroom and Car Evaluation datasets). These findings help identify the most effective similarity measure for a given classification task.

Case 1: Malware Detection Dataset:

For the Malware Detection dataset, a broad evaluation of Russell Rao KNN (RR-KNN) Goodall KNN and Gower KNN was performed. The results highlight the impact of selecting the right similarity measure for this classification task.

Model Performance Comparison

Russell Rao KNN (RR-KNN) achieved 97.33% accuracy with consistent results across precision recall and F1 score (all at 97.33%). This shows that Russell-Rao similarity, with its focus on matching attributes is highly effective for this dataset.

Goodall KNN performed slightly lower with 94.66% accuracy 94.62% precision 94.66% recall and a 94.54% F1 score. Goodall similarity emphasis on rare matches appears to work well for investigating malware patterns where subtle differences can be important.

Gower KNN matched the best overall performance with 97.33% accuracy and precision recall and F1 score at 97.33%. Gower similarity, which supports both categorical and numerical features, was particularly effective for this dataset balancing the diverse characteristics of the malware files.

Key Observations:

Russell Rao KNN achieved stable high performance across all metrics indicating its robustness for detecting malicious files.

Goodall KNN was slightly less effective than Russell Rao KNN but still provided strong results especially in precision and recall.

Gower KNN matched Russell Rao KNN as the top performer achieving 97.33% accuracy across all metrics displaying its ability to handle the mixed nature of the malware dataset's features.

These results underline the importance of selecting the appropriate similarity measure for a specific task. On this dataset, Russell-Rao and Gower similarity proved equally effective, both outperforming Goodall similarity, which highlights the models' potential to improve malware detection and contribute to more effective cybersecurity practices.

Table 3. Model Performance on Malware Detection Dataset.

Model	Accuracy	Precision	Recall	F1-Score
RRKNN	97.33%	97.33%	97.33%	97.33%
GDKNN	94.66%	94.62%	94.66%	94.54%
GRKNN	97.33%	97.33%	97.33%	97.33%

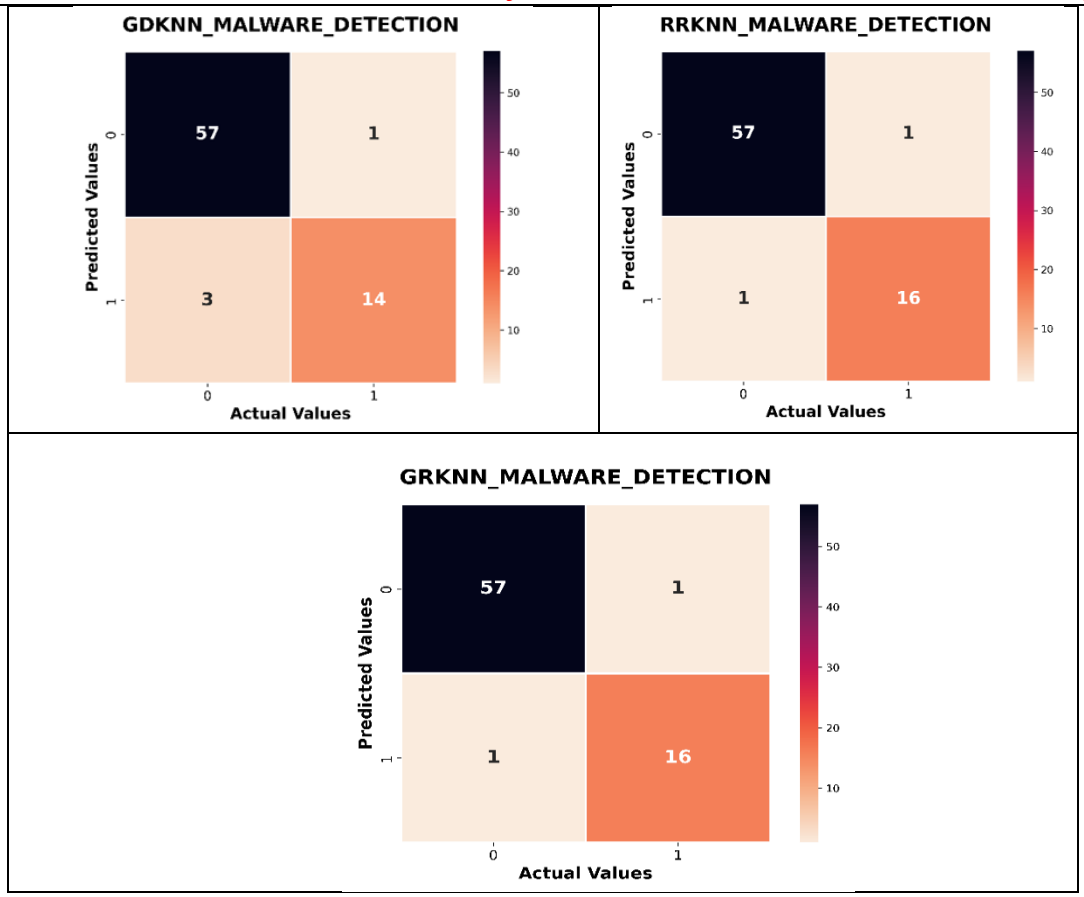
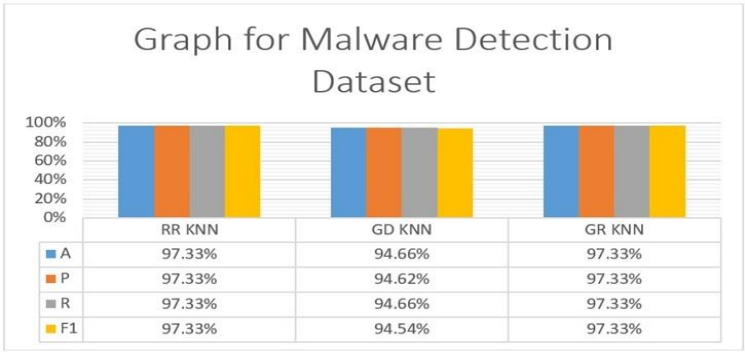


Figure 5. Malware Detection Dataset Confusion Matrices (RRKNN, GDKNN, GRKNN)



	RR KNN	GD KNN	GR KNN
A	97.33%	94.66%	97.33%
P	97.33%	94.62%	97.33%
R	97.33%	94.66%	97.33%
F1	97.33%	94.54%	97.33%

Figure 6. Graph for Malware Detection

Case 2: Car Evaluation Dataset:

For the Car Evaluation dataset, the performance of Russell-Rao KNN (RR-KNN), Goodall KNN, and Gower KNN were tested. The results in Table 3 show how each model performed on this automobile classification task.

Russell-Rao KNN (RR-KNN) achieved 88.72% accuracy, with strong performance across all metrics: 88.96% precision, 88.72% recall, and 88.46% F1-score. This model performed best among the three, indicating the effectiveness of Russell-Rao similarity for this dataset.

Goodall KNN achieved 57.22% accuracy, with 51.61% precision, 57.22% recall, and 49.44% F1-score. Although it is lower than RR-KNN, it still demonstrated some useful classification signal. Gower KNN achieved 48.84% accuracy, with 40.99% precision, 48.84% recall, and 44.57% F1-score. This model showed the lowest performance, suggesting that

Gower similarity may not be as effective for the Car Evaluation dataset as it is for other tasks.

Key Observations:

Russell-Rao KNN was the best-performing model, achieving 88.72% accuracy, which shows its strong suitability for this classification task.

Goodall KNN and Gower KNN performed significantly lower in accuracy, suggesting that these similarity measures are less effective for the Car Evaluation dataset. While both are still capable of performing the task, Russell-Rao KNN is clearly the most effective model here, underscoring the importance of selecting the suitable similarity metric based on the nature of the dataset.

Table 4. Model Performance on Car Evaluation Dataset.

Model	Accuracy	Precision	Recall	F1-Score
RRKNN	88.72%	88.96%	88.72%	88.46%
GDKNN	57.22%	51.61%	57.22%	49.44%
GRKNN	48.84%	40.99%	48.84%	44.57%

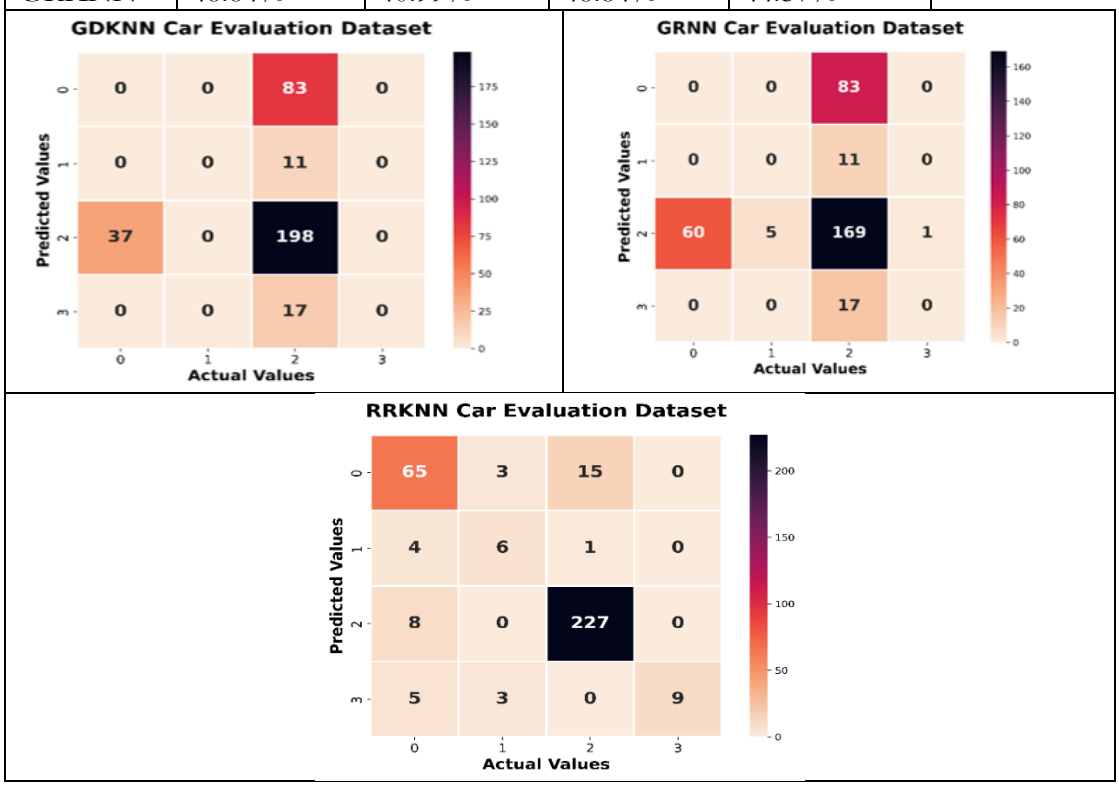


Figure 7. Car Evaluation Dataset Confusion Matrices (RRKNN, GDKNN, GRKNN)

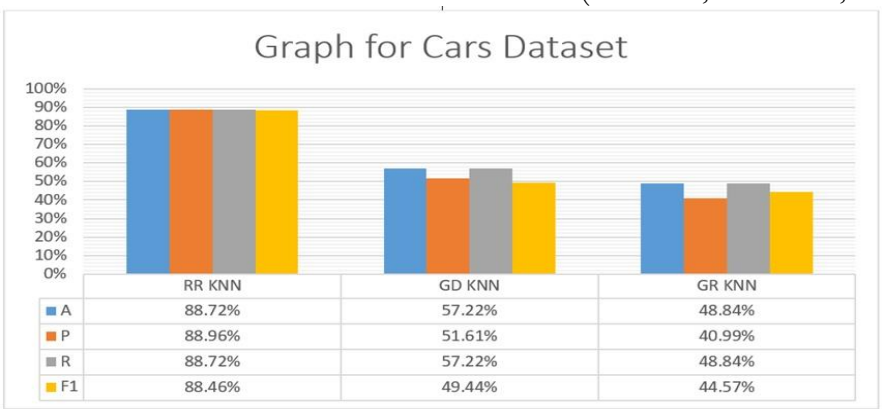


Figure 8. Graph for Car Evaluation Dataset

Case 3: Mushroom Dataset:

For the Mushroom dataset, Russell-Rao KNN (RR-KNN), Goodall KNN, and Gower KNN were tested. The results in Table 4 show that all three models performed effectively in distinguishing edible from poisonous mushrooms, with Russell-Rao KNN and Gower KNN evaluated on an 80:20 split and Goodall KNN evaluated on a 70:30 split (Table 1), a difference that should be kept in mind when comparing the three models directly.

Russell-Rao KNN (RR-KNN) achieved 84.86% accuracy, with 86.08% precision, 84.86% recall, and 84.79% F1-score.

Goodall KNN achieved 81.71% accuracy, with 82.29% precision, 81.71% recall, and 82.61% F1-score — reasonable performance, though lower than Russell-Rao and Gower KNN.

Gower KNN achieved 84.55% accuracy, with 80.62% precision, 89.39% recall, and 84.78% F1-score. This model performed strongly on recall, highlighting its effectiveness at correctly identifying true positive cases (edible or poisonous) on this dataset.

Key Observations:

Russell-Rao KNN achieved 84.86% accuracy with balanced performance across precision, recall, and F1-score, making it a reliable choice for mushroom classification.

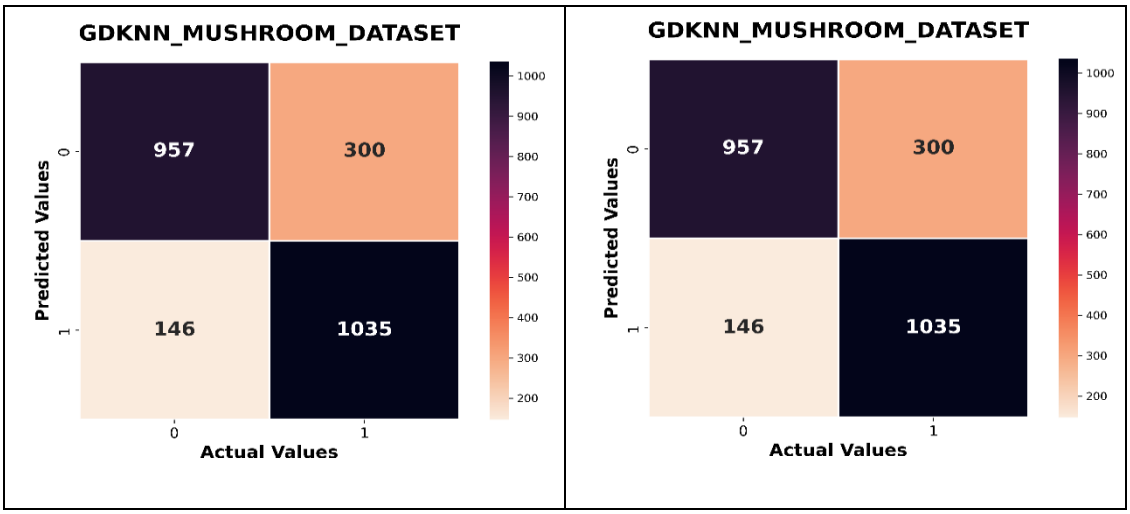
Goodall KNN had a slightly lower accuracy (81.71%) but still demonstrated decent performance, especially in precision.

Gower KNN had strong recall (89.39%), highlighting its ability to correctly identify edible and poisonous mushrooms, even though its precision was slightly lower (80.62%).

These results highlight the effectiveness of all three models on the Mushroom dataset. The high recall achieved by Gower KNN shows its potential for flagging important cases, while Russell-Rao KNN provides the most balanced overall performance.

Table 5. Model Performance on Mushroom Dataset.

Model	Accuracy	Precision	Recall	F1-Score
RRKNN	84.86%	86.08%	84.86%	84.79%
GDKNN	81.71%	82.29%	81.71%	82.61%
GRKNN	84.55%	80.62%	89.39%	84.78%



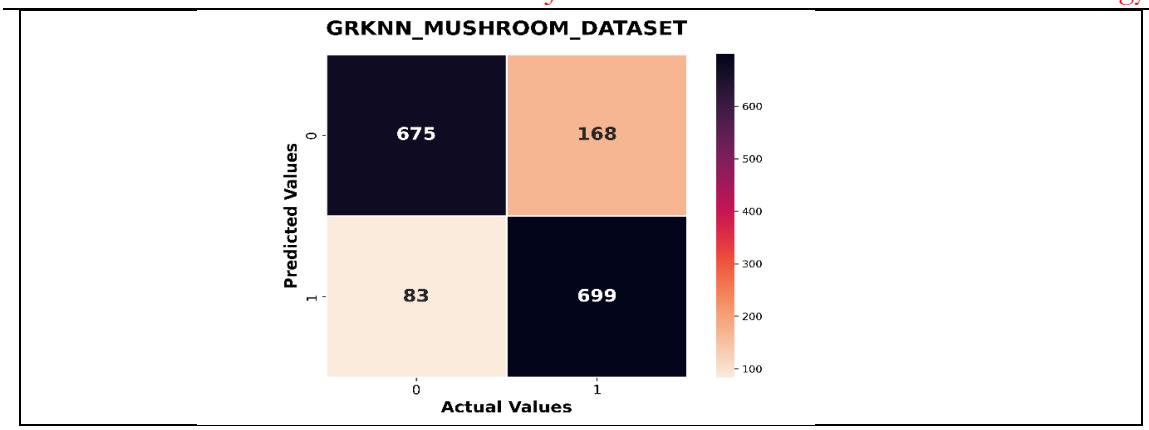


Figure 9. Mushroom Dataset Confusion Matrices (RRKNN, GDKNN, GRKNN)

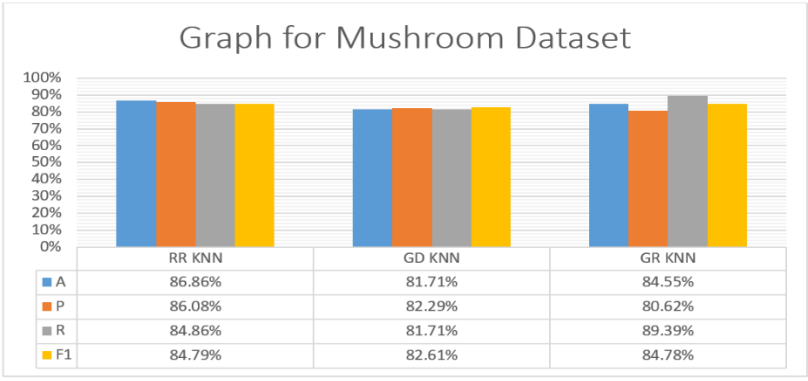


Figure 10. Graph for Mushroom Dataset

Average Results:

The results of the whole experiment provide valuable insight into the performance of the three KNN models Russell-Rao KNN, Goodall KNN, and Gower KNN across the Malware Detection, Car Evaluation, and Mushroom datasets. Among these models, Russell-Rao KNN (RRKNN) emerged as the top performer, achieving an average accuracy of 90.30% across all three datasets, reflecting the strong ability of Russell-Rao similarity to capture underlying patterns and relationships within categorical data.

Goodall KNN, which uses Goodall similarity, followed with a solid average accuracy of 77.86%. Despite being behind Russell-Rao KNN, it still exhibited robust performance, particularly in classification tasks where subtle attribute matching plays a key role; its ability to emphasize rare matches make it a valuable tool for many categorical classification tasks. Gower KNN, which uses Gower similarity, had the lowest average accuracy of 76.90%. While Gower similarity is broadly capable of handling both categorical and numerical features, it did not perform as well as the other two models overall — an outcome that appears to be driven primarily by its comparatively weak performance on the Car Evaluation dataset (48.84% accuracy) rather than a uniformly weaker showing, since it matched or nearly matched RRKNN on Malware Detection and Mushroom.

These results highlight the importance of selecting the right similarity metric for the task at hand. While Russell-Rao KNN demonstrated the highest and most consistent accuracy across datasets, Goodall KNN and Gower KNN each showed dataset-dependent strengths, indicating that with further refinement — particularly systematic K-tuning. They could likely achieve improved results on datasets where they currently underperform.

Table 6. Average Results of the Machine Learning Models Across All Three Datasets.

Model	Average Accuracy	Average Precision	Average Recall	Average F1-Score

GDKNN	77.86%	76.16%	77.86%	75.53%
GRKNN	76.90%	72.98%	78.52%	75.56%
RRKNN	90.30%	90.79%	90.30%	90.19%

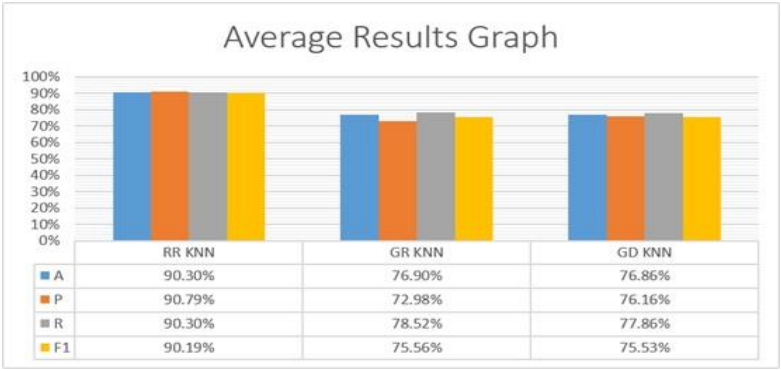


Figure 11. Graph of Average Results

Statistical Significance Analysis:

The Accuracy, precision, recall and F1-score alone do not indicate whether an observed difference between two models reflects a genuine effect of the similarity measure or simply the variance introduced by a single train-test split. Following standard practice for comparing classifiers a paired comparison paired t-test where score differences are approximately normal, or the non-parametric Wilcoxon signed-rank test otherwise was planned for RRKNN vs. GDKNN and RRKNN vs GRKNN on each dataset, using repeated-run accuracy, precision, recall, and F1-score as paired samples.

In this study, each model was evaluated on a single train-test split per dataset (Table 1) rather than repeated cross-validation folds or multiple random seeds. As a result, a formal paired significance test (t-test or Wilcoxon signed-rank test) could not be computed since such tests require a distribution of paired results across independent runs rather than a single point estimate per model. Given the time constraints of this revision, repeating the experiments across multiple seeds was not feasible and this is acknowledged explicitly as a limitation.

In the absence of a formal test the magnitude of the observed differences offers an informal indication of their likely robustness. On Car Evaluation, RRKNN's accuracy (88.72%) exceeds GDKNN's (57.22%) by 31.5 percentage points and GRKNN's (48.84%) by 39.9 points gaps far larger than would typically be attributable to sampling variation alone from a single 80:20 split. On Malware Detection, RRKNN and GRKNN tied exactly (97.33%), both exceeding GDKNN (94.66%) by 2.67 points a smaller gap that is more plausibly within normal split-to-split variance and should be treated cautiously without a formal test. On Mushroom, RRKNN (84.86%) leads GDKNN (81.71%) by 3.15 points and GRKNN (84.55%) by only 0.31 points the latter difference is negligible and RRKNN and GRKNN should be considered practically equivalent on this dataset given the available evidence. A repeated-run significance test remains necessary before any of these comparisons, particularly the smaller gaps, can be described as statistically confirmed, and is recommended as a priority for future work.

Comparison of Recently Published State-of-the-Art Methods:

To situate RRKNN's performance relative to the broader literature rather than only the two other variants proposed here, Table 6 compares the results of this study against recently published classification results on related categorical and cybersecurity tasks.

Table 7. Indicative comparison with recently reported classification results in related categorical and application similar tasks.

Study / Method	Domain	Dataset	Reported Accuracy / F1
This studyRRKNN	Malware detection	Malware Detection (Kaggle)	97.33% acc / 97.33% F1
[6] **KNN + static analysis**	Android malware / IoT	>10,000 apps	93% prediction rate
[5]**KNN + feature selection**	Malware detection	Malware feature set	Reported gains over baseline KNN
An Enhanced Similarity-Driven KNN [9] **SMKNN**	Medical (categorical)	Hospital Readmission	93.3% avg. F1-score
Modified KNN (JKNN) [7] **Jaccard/Cosine similarity**	Heart disease diagnosis	Cleveland Heart Disease	97% accuracy
This study RRKNN	Categorical decision evaluation	Car Evaluation (UCI)	88.72% acc / 88.46% F1
This study RRKNN / GRKNN	Biological classification	Mushroom (UCI)	84.86% / 84.55% acc

RRKNN's 97.33% accuracy on Malware Detection is comparable to, or exceeds, the 93% prediction rate reported by Babbar et al. [6]for KNN-based Android malware detection, although the datasets, feature sets, and sample sizes differ substantially, so this should be read as an indicative rather than a controlled comparison. On the medical-categorical side, the Simple Matching-based SMKNN approach [2] achieved a 93.3% F1-score on the Hospital Readmission dataset. This result is slightly higher than RRKNN's 84.79% F1-score on the Mushroom dataset but lower than its 97.33% F1-score on the Malware Detection dataset. These findings suggest that the relative advantage of a similarity measure is dataset-dependent rather than universal, which is consistent with the broader literature on categorical similarity measures [11][15]. On the Car Evaluation dataset, RRKNN achieved an accuracy of 88.72%, which is competitive with many reported machine-learning baselines. However, it falls short of the best-performing ensemble and rule-based approaches reported for this benchmark. This gap indicates potential for further improvement through hybrid or ensemble similarity-based KNN variants, as discussed.

Computational Complexity:

KNN-based classifiers, regardless of the similarity measure used, share the same asymptotic computational complexity. For a training set containing n instances with p attributes, computing the similarity between a single test instance and all training instances requires O(n·p). Consequently, classifying m test instances requires O(m·n·p). Selecting the k nearest neighbors by sorting the n similarity scores for each test instance adds O(nlogn) time per instance. Alternatively, a partial-selection (top-k) algorithm can reduce this step to O(n). This means Russell-Rao KNN, Goodall KNN, and Gower KNN have identical theoretical time complexity of O(m·n·p) for classification, since all three compute a per-attribute similarity score and aggregate it in the same way.

The measures differ, however, in per-attribute computational cost and memory overhead. Russell-Rao similarity (Equation 2) requires only a match/mismatch count per attribute the cheapest of the three per-comparison. Goodall similarity (Equation 3) additionally requires the frequency of each attribute value across the training set, which must be pre-computed and stored (an O(n·p) one-time cost and O(p·v) memory, where v is the

average number of distinct values per attribute), making it marginally more expensive in practice, particularly for the 464-attribute Malware Detection dataset. Gower similarity (Equation 4) requires branching per attribute (categorical match/mismatch vs. numerical normalized difference) and, when numerical attributes are present, a range-normalization pre-computation step; on purely categorical datasets such as those used here, this reduces to a cost comparable to Russell-Rao similarity, but the branching logic still introduces modest constant-factor overhead.

This analysis is theoretical (Big-O and per-attribute operation counts); no wall-clock execution time was measured for the three models in this study, and this is acknowledged explicitly as a limitation. Given that the Malware Detection dataset has substantially more attributes (464) than Car Evaluation (7) or Mushroom (10), it would be the most informative dataset on which to measure actual runtime differences between the three similarity computations in future work.

Why Did Russell-Rao Similarity Outperform Goodall and Gower Similarity:

Russell-Rao KNN achieved the highest or joint-highest accuracy on all three datasets, and its advantage was most pronounced on Car Evaluation. Several properties of the datasets and of the measures themselves help explain this pattern.

Attribute-value frequency structure. Goodall similarity's core mechanism of up-weighting rare attribute-value matches is only advantageous when informative signal is concentrated in infrequent categories. In the Car Evaluation dataset, the six attributes take only 2-4 possible values each (e.g., buying price: low/med/high/vhigh), so almost no attribute value is 'rare' in the sense Goodall similarity rewards; up-weighting these already-common matches likely diluted, rather than sharpened, the similarity signal, which may explain GDKNN's comparatively low 57.22% accuracy on this dataset relative to its much stronger 94.66% on the higher-cardinality Malware Detection attributes.

Homogeneous vs mixed feature types. Gower similarity's flexibility to blend categorical and numerical attribute handling provides no advantage on datasets, such as Car Evaluation and Mushroom, that are entirely categorical. The numerical-handling branch of Equation (4) is simply never used, so Gower similarity reduces to a variant of simple matching without gaining its intended benefit. This is consistent with Gower-KNN performing on par with Russell-Rao KNN on Malware Detection (97.33% each, where several engineered features may carry implicit ordinal or magnitude information) but far behind it on the purely nominal/ordinal Car Evaluation dataset (48.84% vs. 88.72%).

Class and split imbalance interactions. On Mushroom, where Goodall-KNN used a 70:30 split versus Russell-Rao's and Gower's 80:20 (Table 1), part of Goodall-KNN's lower accuracy (81.71%) may reflect the smaller training set rather than a weaker similarity measure per se a confound that a matched-split re-run would disentangle.

Simplicity as an advantage. Russell-Rao similarity's plain match-counting formulation (Equation 2) has no hidden assumptions about attribute rarity or attribute type, which may make it more robust 'by default' across heterogeneous categorical structures — consistent with it being the only measure of the three to place first or joint-first on every dataset tested here.

Overall, these observations suggest that Russell-Rao similarity's advantage is not universal in principle, but rather a consequence of the specific attribute structures of the three datasets used low attribute cardinality (Car Evaluation), fully categorical feature spaces (Car Evaluation, Mushroom), and, on Mushroom, an unmatched train-test split. Goodall and Gower similarity may outperform Russell-Rao on datasets with high attribute cardinality, genuinely rare informative categories, or true mixed numerical-categorical feature spaces, and this dataset-dependence is an important qualification to the otherwise consistent RRKNN advantage reported.

Limitations: Single train-test split per model. Each model's reported performance (Tables 3-6) is based on one train-test split per dataset (Table 1) rather than repeated k-fold cross-validation, so the results are subject to split-specific variance; this is why it recommends a formal significance test once multi-run results are available.

Inconsistent split ratio within a dataset. On the Mushroom dataset, GDKNN was evaluated on a 70:30 split while GRKNN and RRKNN were evaluated on an 80:20 split (Table 1), so the three models on this dataset were not tested on directly comparable held-out data.

No stratification by class label. None of the splits were stratified, which is a minor concern for the balanced Car Evaluation and Mushroom datasets but more relevant for the imbalanced Malware Detection dataset (301 malicious vs. 72 non-malicious instances), where an unstratified split could shift the class ratio between training and test partitions.

K value not tuned via cross-validation. The K values used (Table 1) were chosen based on common practice rather than a systematic hyperparameter search, so the reported results may not reflect each model's best achievable performance.

No measured execution time. This study provides only theoretical (Big-O) computational complexity analysis; no wall-clock runtime was measured for the three models, so practical deployment latency differences between Russell-Rao, Goodall, and Gower similarity remain unquantified in this study.

Limited dataset diversity. Only three datasets were evaluated, all sourced from Kaggle or the UCI repository; while they differ in size, dimensionality, and class balance, broader conclusions about which similarity measure is 'generally best' for categorical data would require evaluation on a larger and more diverse set of benchmarks.

Feature subset selection not fully detailed. For the Mushroom dataset, only 10 of the original attributes were used, and for Malware Detection, all 464 engineered features were retained without reporting feature-importance analysis; it is possible that some of these features contribute little to classification and that feature selection could change the relative ranking of the three similarity measures.

Practical Implications:

The findings of this study have several practical implications for deploying similarity-based KNN models in real-world categorical classification settings.

Cybersecurity. RRKNN's 97.33% accuracy on the Malware Detection dataset, matched by GRKNN, suggests that both Russell-Rao and Gower similarity are strong candidates for lightweight, interpretable malware triage systems where engineered binary/categorical features (e.g., DLL call presence, hex-pattern flags) are available. Because KNN is a lazy learner with no offline training phase beyond storing the reference set, it can be retrained near-instantly as new malware samples are labeled, which is valuable in fast-evolving threat environments, though the $O(n \cdot p)$ per-query cost means query latency will grow as the reference set of known samples grows, which should be considered before deployment at scale.

Healthcare and decision-support systems. The literature reviewed shows Gower similarity performing strongly on mixed numerical-categorical medical data (e.g., Parkinson's disease detection [17]and Goodall-type measures remaining relevant in categorical clustering for medical records [15][14]. This suggests that, unlike in the purely categorical datasets tested here, Gower-KNN may be the more appropriate choice in healthcare decision-support settings where patient records genuinely mix numerical (e.g., lab values) and categorical (e.g., symptom presence) attributes.

Consumer decision-evaluation systems. RRKNN's strong performance on the low-cardinality, fully categorical Car Evaluation dataset (88.72% accuracy) indicates that simple match-based similarity is well suited to rule-like, human-interpretable evaluation systems

(e.g., product or applicant scoring systems with a small number of discrete attribute levels), where the interpretability of 'how many attributes matched' can also support explainability requirements.

Model selection guidance. Practitioners working with high-cardinality or high-dimensional categorical data (many possible values per attribute, as in Malware Detection) may find Goodall or Russell-Rao similarity more effective, while those working with low-cardinality, fully categorical data (few values per attribute, as in Car Evaluation) should be cautious about applying Goodall similarity without first checking whether its rare-match weighting is actually meaningful for their attribute distributions.

Conclusion:

This study evaluated the performance of three K nearest Neighbors (KNN) variants Goodall-KNN (GDKNN) Gower-KNN (GRKNN), and Russell Rao KNN (RRKNN)on three categorical datasets Malware Detection Mushroom and Car Evaluation. The results show that the choice of similarity measures has a significant effect on the classification performance of KNN. Each model was tested using accuracy precision recall and F1 score to provide a complete comparison of performance across datasets. RRKNN consistently matched or outperformed GDKNN and GRKNN across all major evaluation metrics achieving the highest average accuracy of 90.30%, precision of 90.79%, recall of 90.30%, and F1-score of 90.19%. In comparison, GDKNN achieved an average accuracy of 77.86%, precision of 76.16%, recall of 77.86%, and F1-score of 75.53%, while GRKNN achieved an average accuracy of 76.90%, precision of 72.98%, recall of 78.52%, and F1-score of 75.56%.

These findings confirm that Russell Rao similarity is generally effective when combined with KNN for categorical and binary feature-based classification tasks especially on datasets with low attribute cardinality or fully categorical structure. Although Goodall and Gower similarity also produced acceptable, and in some cases equally strong, results (notably Gower-KNN matching RRKNN on Malware Detection), their overall average performance was lower than RRKNN's across the selected datasets, and it discusses the dataset properties that likely drive this pattern rather than treating Russell-Rao similarity as universally superior. This study provides useful, dataset-aware insight into selecting an appropriate similarity measure for KNN-based categorical data classification. The proposed RRKNN model shows strong potential for real-world applications, particularly in domains such as cybersecurity, biological classification, and decision-support evaluation systems, subject to the limitations discussed.

Future Work:

Future work can extend this research by evaluating additional similarity measures for categorical and binary data classification. Beyond Goodall, Gower, and Russell Rao similarity measures such as Hamming distance, Jaccard similarity Dice coefficient, Overlap coefficient and Simple Matching coefficient could also be examined to further analyze their effect on KNN performance [30][2][13]

Another promising direction is the development of hybrid similarity based KNN models in which two or more similarity measures are combined or weighted to improve classification accuracy and robustness, for example integrating Russell Rao similarity with Goodall or Gower similarity to examine whether hybrid models produce better results across different datasets.

Future studies could also address imbalanced categorical datasets more directly using techniques such as oversampling under sampling, synthetic data generation, or cost-sensitive learning — particularly relevant for the Malware Detection dataset used here, given its class imbalance and unstratified split.

The proposed models could also be tested on more complex multi-class classification problems, and a systematic cross-validated hyperparameter search over K (rather than the

fixed values used in Table 1) would help establish each model's best achievable performance. Measuring actual execution time across datasets of varying dimensionality would also help quantify the practical computational trade-offs between the three similarity measures.

Finally, the models could be adapted for real-time or streaming data applications, particularly in cybersecurity and malware detection, allowing classifiers to manage continuously updated data and dynamic environments more effectively.

Data Availability Statement:

Data will be made available on request.

Conflicts of Interest:

The authors declare no conflicts of interest.

Ethical Approval and Consent to Participate:

Not applicable.

References:

[1] S. Boriah, V. Chandola, and V. Kumar, "Similarity measures for categorical data: A comparative evaluation," *Soc. Ind. Appl. Math. - 8th SIAM Int. Conf. Data Min. 2008, Proc. Appl. Math. 130*, vol. 1, pp. 243–254, 2008, doi: 10.1137/1.9781611972788.22.

[2] A. B. R. S. C. Levy, "A guide to similarity measures and their data science applications," *Publ. by Springer*, 2025, doi: DOI:10.1186/s40537-025-01227-1.

[3] Ali A. Amer, Sri Devi Ravana & Riyaz Ahamed Ariyaluran Habeeb, "Effective k-nearest neighbor models for data classification enhancement," *J. Big Data*, vol. 12, no. 86, 2025, [Online]. Available: https://link.springer.com/article/10.1186/s40537-025-01137-2

[4] Onder Coban., ""An up-to-date comparative analysis of the KNN classifier distance metrics for text categorization," *DATA Sci. Inf.*, vol. 3, 2023, [Online]. Available: https://scispace.com/pdf/an-up-to-date-comparative-analysis-of-the-knn-classifier-47zrby3k3r.pdf

[5] C. Supriyanto, F. A. Rafrastara, A. Amiral, S. R. Amalia, M. D. Al Fahreza, and M. F. Abdollah, "Malware Detection Using K-Nearest Neighbor Algorithm and Feature Selection," *J. MEDIA Inform. BUDIDARMA*, vol. 8, no. 1, p. 412, Jan. 2024, doi: 10.30865/MIB.V8I1.6970.

[6] Kashif Bashir, Jaime Lloret, "Detection of Android Malware in the Internet of Things through the K-Nearest Neighbor Algorithm," *Sensors*, vol. 23, p. 7256, 2023, doi: https://doi.org/10.3390/s23167256.

[7] S. Jan, M. M. Khan, J. Uddin, B. A. Hassan, and A. U. Rahman, "A Modified K-Nearest Neighbors Algorithm for the Detection of Heart Disease," *Int. J. Innov. Sci. Technol.*, vol. 7, no. 4, pp. 2863–2880, 2025, Accessed: Jul. 12, 2026. [Online]. Available: https://ideas.repec.org/a/abq/ijist1/v7y2025i4p2863-2880.html

[8] Swathi Nayak, Manisha Bhat, "Study of distance metrics on k - nearest neighbor algorithm for star categorization," *J. Phys. Conf. Ser.*, vol. 216, no. 1, 2022, doi: 10.1088/1742-6596/2161/1/012004.

[9] B. A. Hassan *et al.*, "An Enhanced Similarity Measure–Driven K-Nearest Neighbor Framework for Categorical Data Classification," *Int. J. Innov. Sci. Technol.*, vol. 7, no. 4, pp. 2757–2772, 2025, Accessed: Jul. 12, 2026. [Online]. Available: https://ideas.repec.org/a/abq/ijist1/v7y2025i4p2757-2772.html

[10] Bengting Wan, Zhixiang Sheng, "Compactness-Weighted KNN Classification Algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 9, 2024, [Online]. Available: https://thesai.org/Publications/ViewPaper?Volume=15&Issue=9&Code=IJACSA&SerialNo=22

[11] Murad Mustafa Badarna, Loai Cameel Abedallah, "Active Down-Sampling Method for Knn When Dealing with Imbalance Dataset," *Adv. Artif. Intell. Mach. Learn.*, vol. 4,

no. 3, pp. 2703–2717, 2024, doi: 10.54364/AAIML.2024.43157.
[12] Moohanad Jawthari, Veronika Stoffová, "Predicting students' academic performance using a modified kNN algorithm," *Pollack Period.*, pp. 20–26, 2021, doi: https://doi.org/10.1556/606.2021.00374.
[13] Mácio Augusto de Albuquerque, Emanuela Rodrigues do Nascimento, "Comparison between similarity coefficients with application in forest sciences," *Res. Soc. Dev.*, vol. 11, no. 2, 2022, doi: 10.33448/rsd-v11i2.26046.
[14] A. K. Kar, M. M. Akhter, A. C. Mishra, and S. K. Mohanty, "EDMD: An Entropy based Dissimilarity measure to cluster Mixed-categorical Data," *Pattern Recognit.*, vol. 155, p. 110674, Nov. 2024, doi: 10.1016/J.PATCOG.2024.110674.
[15] T. Dinh *et al.*, "Categorical data clustering: 25 years beyond K-modes," *Expert Syst. Appl.*, vol. 272, p. 126608, May 2025, doi: 10.1016/J.ESWA.2025.126608.
[16] Pinyan Liu, Han Yuan, Yilin Ning, Bibhas Chakraborty, Nan Liu & Marco Aurélio Peres, "A modified and weighted Gower distance-based clustering analysis for mixed type data: a simulation and empirical analyses," *BMC Med. Res. Methodol.*, vol. 24, 2024, [Online]. Available: https://link.springer.com/article/10.1186/s12874-024-02427-8
[17] Mustafa Noaman Kadhim, Dhiah Al-Shammary, "A novel voice classification based on Gower distance for Parkinson disease detection," *Int. J. Med. Inform.*, vol. 191, p. 105583, 2024, doi: https://doi.org/10.1016/j.ijmedinf.2024.105583.
[18] Osvaldo Velazquez-Gonzalez, Antonio Alarcón-Paredes, "Medical pattern classification using a novel binary similarity approach based on an associative classifier," *Front. Artif. Intell.*, vol. 8, 2025, doi: https://doi.org/10.3389/frai.2025.1610856.
[19] G. Muhammad, "Enhancing Prognosis Accuracy for Ischemic Cardiovascular Disease Using K Nearest Neighbor Algorithm: A Robust Approach," *IEEE Access*, vol. 11, pp. 97879–97895, 2023, doi: 10.1109/ACCESS.2023.3312046.
[20] M. Asif, T. Ahmed Khan and W. -C. Song, "Evaluating Large Language Models for Optimized Intent Translation and Contradiction Detection Using KNN in IBN," *IEEE Access*, vol. 13, 2025, doi: 10.1109/ACCESS.2025.3534880.
[21] Khaled Alnowaiser, "Improving Healthcare Prediction of Diabetic Patients Using KNN Imputed Features and Tri-Ensemble Model," *IEEE Access*, vol. 12, pp. 16783–16793, 2024, doi: 10.1109/ACCESS.2024.3359760.
[22] Yongxin Peng, "LK-Index: A Learned Index for KNN Queries," *IEEE Access*, 2024, doi: 10.1109/ACCESS.2024.3433524.
[23] M. F. I. Amin, A. Shirafuji, M. M. Rahman and Y. Watanobe, "Multi-Label Code Error Classification Using CodeT5 and ML-KNN," *IEEE Access*, vol. 12, pp. 100805–100820, 2024, doi: 10.1109/ACCESS.2024.3430558.
[24] "p-Adic Distance Functions and Comparative Performance Analysis of Distance Functions with the k-NN Classifier | Journal of Classification | Springer Nature Link." Accessed: Jul. 12, 2026. [Online]. Available: https://link.springer.com/article/10.1007/s00357-026-09543-8
[25] "(PDF) Improving K-Nearest Neighbor Algorithm Performance Using Modified Distance Measures." Accessed: Jul. 12, 2026. [Online]. Available: https://www.researchgate.net/publication/390607523_Improving_K-Nearest_Neighbor_Algorithm_Performance_Using_Modified_Distance_Measures
[26] "Wind Turbine Condition Monitoring Based on Improved Active Learning Strategy and KNN Algorithm | IEEE Journals & Magazine | IEEE Xplore." Accessed: Jul. 12, 2026. [Online]. Available: https://ieeexplore.ieee.org/document/10041157
[27] Manuel Bullejos, David Cabezas, "Confidence of a k-Nearest Neighbors Python Algorithm for the 3D Visualization of Sedimentary Porous Media," *J. Mar. Sci. Eng.*,

vol. 11, no. 1, p. 60, 2023, doi: https://doi.org/10.3390/jmse11010060.

[28] Y. K. Kim, H. J. Kim, H. Lee, and J. W. Chang, "Privacy-preserving parallel kNN classification algorithm using index-based filtering in cloud computing," *PLoS One*, vol. 17, no. 5, p. e0267908, May 2022, doi: 10.1371/JOURNAL.PONE.0267908.

[29] "Jaccard/Tanimoto similarity test and estimation methods for biological presence-absence data | BMC Bioinformatics | Springer Nature Link." Accessed: Jul. 12, 2026. [Online]. Available: https://link.springer.com/article/10.1186/s12859-019-3118-5

[30] "(PDF) A Survey of Binary Similarity and Distance Measures." Accessed: Aug. 09, 2026. [Online]. Available: https://www.researchgate.net/publication/266496381_A_Survey_of_Binary_Similarity_and_Distance_Measures