

An Improved Hybrid Geometric Similarity-Based Retrieval Model for Case-Based Reasoning Systems

O. O. Omonijo^{1*}, S. O. Akinola², D. B. Bolou¹ and A. O Akinleye³

¹Computer Science Unit, Nigeria Maritime University, Okerenkoko, Delta State, Nigeria

²Department of Computer Science, University of Ibadan, Ibadan, Oyo State, Nigeria

³Department of Aquaculture and Fishery Management, Federal University of Agriculture, Abeokuta, Ogun State, Nigeria

ARTICLE INFO

Received: May 2025

Accepted: July 2025

Keywords:

Knowledge-Reuse, k-NN, Similarity Scoring, Binary Features, Disease Diagnosis, Retrieval Accuracy

Corresponding author:

oluwaseyi.omonijo@nmu.edu.ng

DOI: 10.55518/fjpas.ZFQB3920

ORCID ID: 0009-0009-8011-3884

Phone No: +2348034759227

Abstract

This research proposed an improved model for case retrieval in Case-Based Reasoning (CBR) systems by introducing a hybrid geometric similarity measure. Traditional similarity measures often fail to jointly capture magnitude, orientation, and feature significance, hence leading to suboptimal case retrieval, particularly in domains requiring high precision. The proposed hybrid geometric similarity measure (*NHSim*) normalizes Euclidean distance and Cosine similarity before combining with Jaccard similarity by using a simple average approach. The resulting hybrid similarity formula has scores in the range of [0, 1] and was integrated with the k-Nearest Neighbors (k-NN) algorithm to evaluate case similarity retrieval. The model provides consistent, interpretable, scalable, and computationally lightweight similarity values. This approach leverages the strengths of all three measures while minimizing their individual limitations. The model was evaluated using a dataset of 45 catfish disease cases characterized by 33 binary symptoms, with a case base (60%) and query set (40%). Python was employed for the implementation, while evaluation metrics such as accuracy, precision, recall, F1-score, and AUC were used to measure the model's performance. Using a k value of 4 and threshold (τ) of 6.5, experimental results showed that the proposed hybrid model consistently outperformed individual similarity metrics, achieving 85% accuracy, 77% precision, 85% recall, and 80% F1-score. The AUC was 92%, highlighting the model's robustness and reliability. The proposed model demonstrated significant improvements in retrieving relevant cases, particularly for binary-featured domains such as veterinary diagnostics. The findings suggest that hybrid similarity measures offer better accuracy than standalone metrics.

1.0 Introduction

Case-Based Reasoning (CBR) is a problem-solving methodology that leverages previously encountered cases to solve new problems by identifying and reusing similar past experiences [1]. Inspired by cognitive science, CBR mimics how humans resolve new challenges by recalling and adapting solutions from similar historical situations [2], [3]. Each case typically consists of a problem description and a corresponding solution, and these cases are stored in a structured repository known as a case base or case library. When a new problem arises, the system searches for similar cases using predefined similarity measures to retrieve, adapt and eventually retain new solutions, thereby enhancing its knowledge base over time.

Based on the widely cited Aamodt and Plaza 4R model, figure 1 outlines the four key phases of the CBR cycle which include: retrieval, reuse, revise and retain [3]. Among these, the retrieval phase is considered critical, as the effectiveness

of a CBR system largely depends on its ability to identify the most relevant past cases. Accurate retrieval hinges on two main components: the representation of the case and the similarity measure employed [4]. Poor case representation or an inadequate similarity metric can lead to erroneous retrievals, ultimately degrading the performance of the entire system.

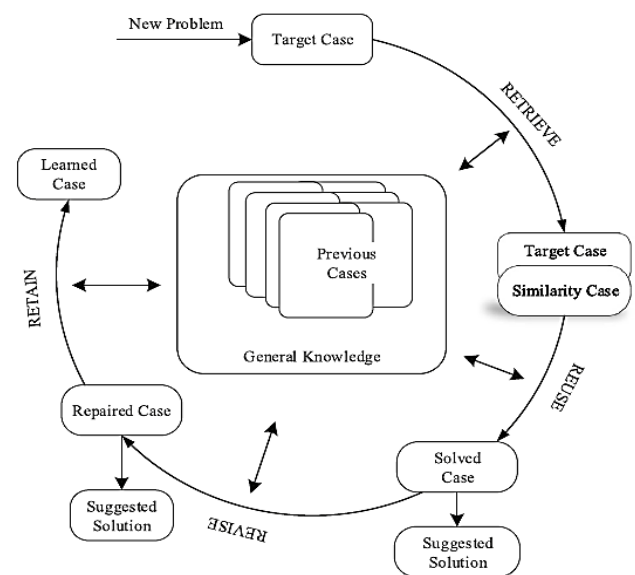


Figure 1: Process of CBR Cycle [19]

To improve retrieval performance, researchers have explored various similarity techniques ranging from geometric measures like Euclidean and Cosine, to semantic and rule-based methods such as fuzzy logic and association rules. For

example, Euclidean and Cosine focus on structural similarity [5], [6], while fuzzy logic and genetic algorithms aim to enhance reasoning flexibility and adaptability [7], [8]. Geometric-based techniques remain dominant due to their simplicity and computational efficiency [9], but often exhibit specific limitations that compromise retrieval accuracy.

While Euclidean distance effectively captures magnitude-based differences, it ignores directional relationships between feature vectors. Conversely, Cosine similarity focuses on orientation but disregards magnitude [10], [11]. For binary and sparse data, Jaccard similarity is often preferred, as it emphasizes feature overlap and is tolerant of missing values [12]. However, it treats all features equally, failing to account for the intensity or weight of individual symptoms.

These observed limitations in single-method approaches have motivated a shift towards hybrid similarity models, which combine the

strengths of multiple techniques to deliver more robust and accurate retrieval results. Some advanced methods, like the Triangular Similarity Measure (TSM), attempt to capture both angle and magnitude through geometric constructions, but they struggle with sparse or unevenly distributed case bases [9]. Other models, such as Triangle area Similarity and Sector area Similarity (TS-SS) metric, multiply geometric features to prevent distortion, yet often suffer from high computational overhead and lack real-world testing [13]. Similarly, multi-view frameworks like RDSim1-5, which integrate cosine, Euclidean and geometric spatial features, offer precision gains but are limited by their scalability, flexibility and parameter sensitivity in high-dimensional or noisy datasets [14].

This study contributes to this evolving landscape by proposing *NHSim*, a hybrid similarity measure that combines Euclidean, Cosine and Jaccard similarity scores. Embedded within a k-

Nearest Neighbour (k-NN) framework, *NHSim* is designed to capture magnitude, orientation and binary overlap in a single normalized similarity score. Unlike many existing hybrid models that rely on complex weighting schemes or parameter tuning, *NHSim* uses normalization and a simple average approach, making it scalable and computationally lightweight. The model was evaluated using a domain-specific dataset of catfish disease cases, therefore demonstrating its practical applicability and effectiveness in real-world case retrieval tasks.

2.0 Literature Review

2.1 Overview of Case-Based Reasoning (CBR)

The quest to capture human expertise in computer systems led to the birth of Knowledge-Based Systems. These systems were designed to replicate expert-level reasoning using either a set of rules as seen in Rule-Based Systems (RBS) or through the reuse of past experiences, as is the case with CBR systems [5], [9]. While RBS can

be effective in well-defined domains, they tend to struggle when the problem space becomes too complex or dynamic. They don't learn from experience and updating the rules can be tedious and expensive [15].

CBR, on the other hand emerged as a more flexible alternative. Instead of relying on hardcoded rules, it learns incrementally by storing and adapting past problem-solving experiences. This mirrors how humans think, that is, drawing on previous situations to find solutions to new ones. This capability makes CBR particularly suited to domains characterized by uncertainty, variability or a lack of formalized rules [16].

Since its formal introduction by Schank in 1982 [17], CBR has been successfully deployed in numerous fields such as medical diagnosis [18], agriculture [19], customer support [20] and legal reasoning [21], thanks to its cyclic reasoning model. The classic 4R cycle; Retrieve, Reuse,

Revise and Retain was formalized by Aamodt and Plaza [3].

The following are the purpose of each step:

1. Retrieval phase, wherein cases that have some form of similarity to the problem at hand are retrieved
2. Reuse phase, where the solutions from the retrieved cases are utilized to create a new solution [22]
3. Revise phase which adapts prospective solutions to fit where available solution does not perfectly fit and needs more revision
4. Retain phase, where revised case(s) learned by the system is finally stored.

The cyclic model implies that, given a new problem, the relevant cases are retrieved from the case base, the cases are reused if they are a solution to the new case else, they are revised or fine-tuned and thereafter the new solution is retained in memory. As a result of the retained solution, the case-based cardinality keeps growing, which enhances the system's capacity

to solve problems and be reused in subsequent episodes. However, this can lead to a gradual increase in case base cardinality over time as new cases are added without careful consideration or evaluation [23]. Thereby, lead to serious bottleneck in time complexity when retrieving cases in the CBR system.

While case base growth is a common limitation of CBR systems [24], other equally critical challenges include case adaptation, where the retrieved case may require significant modification to fit new problem contexts and case base maintenance, which involves removing outdated or redundant cases to prevent retrieval noise and performance degradation [25]. Additionally, issues such as feature selection, scalability to large datasets and domain transferability remain pressing concerns [26], particularly in real-world CBR deployments. These challenges highlight the need for similarity measures that are not only

accurate but also adaptable and computationally efficient across evolving case structures. [27].

2.2 Case Representation and Retrieval

At the heart of every CBR system is the case base. This is a memory of previously solved problems. A case typically consists of a problem description, a solution and optionally, an outcome [28], [29]. The way these cases are represented directly affects how efficiently and accurately the system can retrieve and reuse them. Common structures for case representations include using simple attribute-value pairs or more complex structures like trees, graphs or relational objects, depending on the problem domain [29].

The retrieval phase is critical in determining how well a CBR system performs. It involves comparing the new case with those in the case base to find the best matches. This comparison relies on similarity measures, which calculate how closely a new case resembles previous ones [30]. Retrieval algorithms include k-NN,

induction, template retrieval and knowledge-guided induction [25], [31], [32]. Among these, k-NN remains the most widely adopted due to its simplicity and flexibility [33].

2.3 Similarity Measures in CBR

The accuracy of CBR heavily depends on the similarity measure employed to compare cases [34]. The challenge, however, lies in choosing the right similarity metric for the job. Euclidean distance works well with continuous numeric data but falls short when direction matters [34]. Cosine similarity on the other hand does the opposite; it measures direction but ignores scale [24], [35]. Meanwhile, Jaccard similarity is better for categorical or binary data and handles missing values more gracefully [25]. Each of these has proven effective in certain situations but inadequate in others [24], [29], [35]. Other methods such as Manhattan distance, Mahalanobis distance and fuzzy-based measures have been explored for domain-specific applications [36].

2.4 Hybrid Similarity Approaches

To address the inherent shortcomings of conventional similarity measures, hybrid similarity models have been developed. These models are especially useful in real-world applications where data types are often mixed; numerical, categorical and even textual. Such models are most times a combination of multiple metrics to better handle heterogeneous data types (numerical, categorical, text) and complex case structures [16], [25], [37].

For instance, [24] proposed a method that combines similarity measurement with a Data Envelopment Analysis (DEA) model for alternative generation, enhancing retrieval decision-making under uncertainty. However, the model's reliance on optimization-based comparisons introduces complexity and limits interpretability for non-expert users. In the manufacturing domain, [25] applied hybrid CBR to injection molding processes by integrating numerical and categorical attributes. Their

method improved case reuse in production scenarios but remained domain-specific and computationally heavy when scaled across heterogeneous problem types.

Similarly, [26] conducted a covariance effect analysis of similarity methods for early construction cost estimation. Their comparative study highlighted how geometric and statistical measures react to input variability, ultimately recommending hybrid use for robustness. However, the model lacked adaptability when extended to non-continuous, binary-feature cases, limiting generalization to broader CBR domains. In the field of mechanical product design, [37] developed a knowledge-based automated system using a hybrid similarity model built on fuzzy sets and general knowledge rules. While the model handled imprecision effectively, it required domain-specific knowledge frameworks and complex tuning, making it less suitable for lightweight or dynamic applications.

Table 1: Comparative Analysis of Hybrid Similarity Models for Case Retrieval in CBR Systems

Author(s)	Method / Model	Strengths	Limitations
Zheng <i>et al.</i> [24]	Similarity + DEA model	Supports multi-criteria decision-making under uncertainty	Optimization logic adds complexity; less interpretable for non-experts
Khosravani & Nasiri [25]	Numerical + categorical hybrid for injection molding	Effective in manufacturing; handles mixed data types	Domain-specific; computationally demanding when generalized
Ahn <i>et al.</i> [26]	Covariance-aware hybrid for cost estimation	Highlights impact of input variability; robust in early estimation contexts	Limited to continuous features; lacks support for binary/categorical domains

Long <i>et al.</i> [37]	Fuzzy logic + knowledge- based similarity	Handles vagueness well; suitable for design automation	Requires domain- specific knowledge base; complex tuning
Zhai <i>et al.</i> [9]	Triangular Similarity Measure (TSM)	Captures both angle and magnitude geometrically	Fails in sparse case bases; sensitive to outliers
Heidarian & Dinneen [13]	Hybrid geometric similarity for document clustering	Combines direction and magnitude; improves clustering accuracy	Suffers in high- dimensional, sparse datasets
Mihajlovic & Xiong [10]	CBR with vector space and geometric models	Effective in text similarity retrieval	Ignores magnitude; less suited for binary data

Diallo <i>et al.</i>	Multi-view	Integrates	High
[14]	geometric model	Cosine, Euclidean and spatial dimensions for better clustering	computational cost; parameter- sensitive; limited scalability

A Triangular Similarity Measure (TSM) that combines distance and angular orientation into a geometric triangle to improve similarity estimation in agricultural decision support systems was proposed by [9]. While TSM enhanced retrieval precision by capturing both direction and magnitude, it relied heavily on consistent case base coverage and struggled in sparse domains, which limited its scalability in real-world applications.

[13] introduced a hybrid geometric approach for document clustering, leveraging angular and

distance-based features to improve document similarity. Their approach outperformed traditional methods in clustering coherence, but its performance degraded when feature vectors were high-dimensional or noisy (conditions common in sparse or binary-featured CBR domains).

Similarly, [10] applied CBR to textual document comparison using geometric vector space models. Although their method achieved accurate retrievals, it emphasized orientation-based similarity and lacked sensitivity to feature

magnitude, making it less effective in domains where binary presence or intensity matters.

To address multi-perspective similarity, [14] proposed a multi-view clustering method based on geometric similarity that integrates cosine, Euclidean and spatial representations. Their model demonstrated improved performance for document clustering across feature spaces but required significant computational resources and extensive parameter tuning hence, making it impractical for lightweight decision-support systems.

Table 1 summarizes the key contributions and trade-offs of recent hybrid similarity models. As shown, while prior models offer strong domain-specific performance, they often introduce computational complexity, sensitivity to parameter tuning or limited adaptability to binary-featured data. This study addresses these gaps by proposing *NHSim*, a computationally efficient hybrid similarity measure that combines Euclidean, Cosine and Jaccard

similarities using normalization and a simple average. *NHSim* is specifically designed to support binary-encoded cases in CBR systems, reducing retrieval errors while maintaining interpretability and scalability.

3.0 Methodology

3.1 Research Design

The development of the improved hybrid geometric similarity model is aimed at enhancing the case retrieval process in CBR systems. The proposed model integrates three geometric similarity measures; Euclidean distance, Cosine similarity and the Jaccard index to mitigate the individual limitations of each technique and produce a more robust and accurate similarity measurement.

As depicted in Figure 2, the methodology mainly comprised the following stages:

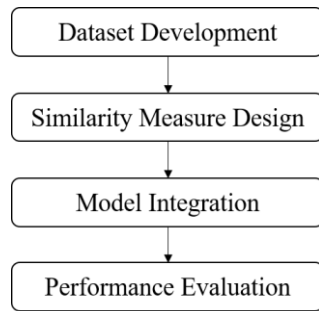


Figure 2: A framework of the methodology for developing and evaluating the retrieval model

- (i) Development of a domain-specific dataset on catfish diseases;
- (ii) Design and implementation of the hybrid similarity measure using Python;
- (iii) Integration of the similarity model into a k-Nearest Neighbour (k-NN) retrieval framework;
- (iv) Evaluation of retrieval performance using established metrics, including accuracy, precision, recall, F1-score and area under the ROC curve (AUC).

3.2 Case Base Design

Each case is represented as a tuple consisting of a problem description and a corresponding solution. The problem description includes a set

of binary-valued symptoms encoded as attribute-value pairs, while the solution denotes the diagnosed catfish disease.

The case base was constructed using a structured CSV file containing 45 real-world cases with 33 symptom attributes. Sixty percent (60%) of the cases were stored in the case base and the remaining forty percent (40%) represents the query set/new problems. Each case includes a unique case ID, a list of observed symptoms, optional aliases and a solution (disease label) as represented in table 2.

Table 2: Case Base Structure

	Field Name	Data Type	Remark
Problem	Case ID	Varchar	Unique
	Symptoms (33)	Boolean	
Description	Case Alias	Text	
Solution	Diseases/Diagnosis	Text	

3.3 Hybrid Similarity Computation

The proposed similarity model computes both local and global similarity scores between a query case and each base case.

3.3.1 Local Similarity

The local similarity assesses the correspondence between individual attributes of the query and base cases. The three measures used are applied as follows:

- (i) Euclidean Distance captures absolute differences [14];

$$Dist(x, y) = \sqrt{(x - y)^2} \quad (1)$$

Where;

$Dist()$ is the distance that exists between the two compared vectors.

x and y represent the base and query cases respectively.

- (ii) Cosine Similarity measures angular orientation;

$$Cos(\theta) = Cos(x, y) = \frac{x \cdot y}{||x|| ||y||} \quad (2)$$

Where;

$\cdot$ represents the dot product of the vectors.

$||x||$ and $||y||$ represents the lengths of x and y respectively.

θ is the angle between the vectors

Thus, $\|x\|$ and $\|y\|$ represent the Euclidean form of vector x and y

- (iii) Jaccard Similarity evaluates set-based overlap.

$$JS(x, y) = \frac{|x \cap y|}{|x \cup y|} \quad (3)$$

Where;

$x \cap y$ represents the number of attributes that are shared between sets x and y (the intersection).

$x \cup y$ is the total number of unique attributes in both sets x and y (the union).

$|\dots|$ represents the cardinality of the set of attributes.

To hybridize the Jaccard similarity, Cosine similarity and Euclidean distance into one formula, there is need to ensure consistency in scale. While the Jaccard similarity ($JS(x, y)$) measures similarity based on the ratio of the intersection to the union of two sets and ranges between $[0, 1]$ (1 being most similar and 0 least), the Cosine similarity ($Cos(x, y)$) measures the cosine of the

angle between two vectors, ranging between $[-1, 1]$, but typically $[0, 1]$ for non-negative vectors. Euclidean distance ($Dist(x, y)$) measures the straight-line distance between two points in space, with values ranging between $[0, \infty]$ (where smaller is more similar). Scaling these similarity measures will help to improve their interpretability and usability in CBR systems.

Euclidean distance was normalized using min-max scaling;

Since Euclidean distance ($Dist(x, y)$) typically ranges between 0 and a maximum value depending on the data, the Min-Max scaling was used to range it to values $[0, 1]$.

The formula for Min-Max scaling is:

$$Scaled_Dist(x, y) = \frac{value - min}{max - min}$$

Where:

min is the minimum possible value (0 in this case).

max is the maximum possible value (6 in this case).

A maximum Euclidean distance of 6 was used in the Min-Max normalization formula, corresponding to the highest observed distance between cases in the dataset, derived from the symptom vectors with the most non-overlapping features.

Hence, if the Euclidean distance is 0, the scaled value will be 0, but if the distance is 6, then the scaled value will be 1. Intermediate distances will be proportionally scaled between 0 and 1.

Cosine similarity was also scaled to the [0, 1] interval using linear transformation approach;

So also, since Cosine similarity (**Cos**(*x*, *y*)) values typically fall between -1 (completely dissimilar) and 1 (completely similar), linear transformation was used to map these values to the range [0, 1].

The formula for linear transformation is:

$$\text{Scaled_Cos}(x, y) = \frac{\text{Cos}(x, y) + 1}{2}$$

Where:

Cos(*x*, *y*) is the cosine similarity value.

This implies that, when **Cos**(*x*, *y*) = -1, the scaled value will be 0 and if **Cos**(*x*, *y*) = 1, the scaled value will be 1. While intermediate values will be linearly mapped to the [0, 1] range.

Jaccard similarity was used in its native [0, 1] range.

The Jaccard similarity index naturally lies in the range [0, 1] (where 0 means no overlap and 1 means complete overlap). Since it is already bounded within this range, no additional transformation or scaling is needed.

Rather than assigning explicit weights (α , β , γ), a simple arithmetic mean was used to aggregate the normalized Euclidean, Cosine and Jaccard similarity scores and to provide a [0,1] range/bound for the *hsim*(*x*, *y*). This

approach avoids the subjectivity of manual weight assignment and ensures balanced contribution from each metric as represented in equation 4.

$$nhsim(x, y) = \left(JS(x, y) + \frac{Cos(x, y) + 1}{2} + \frac{Dist(x, y)}{6} \right) / 3 \quad (4)$$

where:

nhsim(x, y) is the normalized hybridized local similarity score.

Equation 4 was used to achieve the result of the feature matching for each pair of attributes (x, y) which is referred here as the normalized hybridized local similarity function score.

Normalization will ensure that the similarity components produce interpretable scores, particularly when varying measuring components are mixed. Normalization will prevent any one metric from unduly influencing the similarity score

but provides a balanced consideration to all metrics and resulting in a more accurate case retrieval and also brings similarity scores into a standardized range, typically [0, 1].

But then, equation 4 only computes a hybrid score that can only be used to measure one attribute. That is, for effective case retrieval, the hybrid score, equation 4 is used to define similarity measures for each attribute. In this case, it was used on only the Symptoms attributes as this is the key retrieval attribute. Other attributes are non-retrieval determinant.

3.3.2 Global Similarity

Global similarity was obtained by aggregating local similarities across all relevant attributes. Hence, combining the local measures at the global level requires scaling up the distance measures from the single dimensional approach which only entails dealing with one attribute,

$nhsim(x(a_j), y(a_j)) \equiv nhsim(x, y)$ to the multidimensional approach which combines all the case features, $NHSim(X, Y)$.

To achieve this, the features of each case in the case base was matched against the features of the queried case.

Hence, the geometric similarity measures of Euclidean distance, Cosine and Jaccard similarity measures as represented in equation 1, 2 and 3 respectively will become:

$$Dist(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (5)$$

Where;

$Dist()$ is the distance that exist between the two compared vectors.

i is the attribute for each case.

n is the number of attributes for each case.

x_i, y_i represent the i th attribute in the vector x and y respectively.

$$Cos(\theta) = Cos(X, Y) = \frac{x \cdot y}{\|x\| \|y\|} \quad (6)$$

Where;

$$x \cdot y = \sum_{i=1}^n x_i * y_i$$

$\|x\|$ is length of the vector x and represented as:

$$\|x\| = \sqrt{\sum_{i=1}^n x_i^2}$$

While $\|y\|$ is the length of the vector y and represented by:

$$\|y\| = \sqrt{\sum_{i=1}^n y_i^2}$$

θ is the angle between the vectors

x represents values from base case where $x_i = x_1, x_2, x_3, \dots, x_n$

y represents values from target case where $y_i = y_1, y_2, y_3, \dots, y_n$

n is the number of features in each vector.

x_i and y_i are the i -th attribute values of the feature in vectors x and y , respectively.

Thus, $\|x\|$ and $\|y\|$ represent the Euclidean form of vector x and y

$$JS(X, Y) = \frac{\sum_{i=1}^n x_i \cap y_i}{\sum_{i=1}^n x_i \cup y_i} \quad (7)$$

Where;

$x_i \cap y_i$ represents the number of attributes that are shared between sets X and Y (the intersection).

$x_i \cup y_i$ is the total number of unique attributes in both sets X and Y (the union).

$\sum_{i=1}^n |x_i \cap y_i|$ is the sum of values that are present in both sets (the intersection).

The Jaccard Similarity, like the traditional Jaccard Similarity, focuses on the proportion of shared items while considering their relative importance.

From equation 5, 6 and 7, the hybrid similarity measure is thus given as:

$$NHSim(X, Y) = \sum_{i=1}^n nhsim(x_i, y_i) \quad (8)$$

where $hsim$ is the normalized hybrid similarity score for attribute i and n is the number of attributes considered. This holistic similarity score reflects both

structural and contextual alignment between the cases.

Equation 8 gives a global representation of the equation. The measure reduces the risk of retrieving incorrect cases by providing a holistic similarity assessment that accounts for binary overlap, pattern alignment and magnitude difference. This balanced, normalized approach enables CBR systems to retrieve cases that are contextually relevant and reduces reliance on any single metric that could otherwise lead to inaccurate results.

Ensuring that a balance exist amongst the three measures such that the effect of the orientation (Cosine similarity), magnitude distance (Euclidean distance) and commonalities (Jaccard similarity) between new case and base cases are considered.

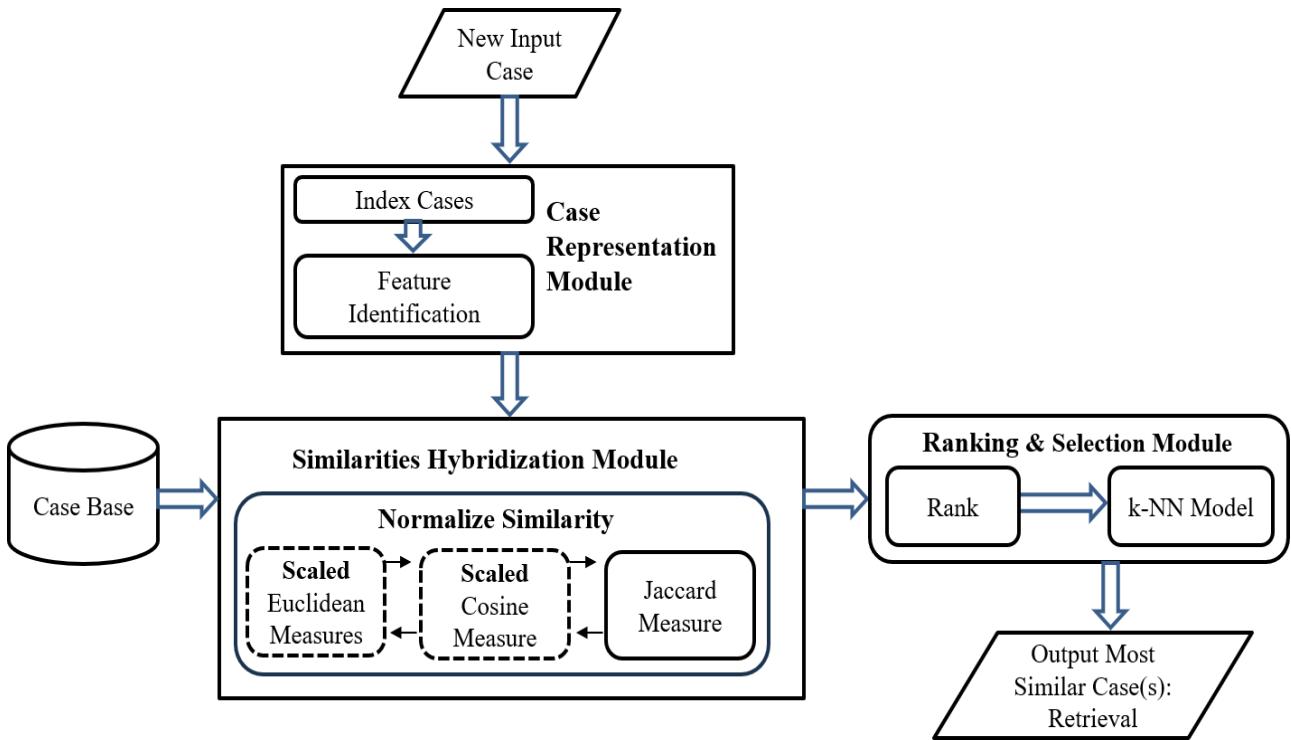


Figure 3: The Case Retrieval Conceptual Diagram

From the conceptual diagram in figure 3, already indexed cases whose features have been properly represented are stored in the case base in the form of problem description along with the corresponding solution. Such that, a case is stored as;

$$\text{Case} = \{\text{Problem Description}, \text{Solution}\}$$

Where *Problem Description* refer to a set of features which describes the problem and *Solution* represent a set of features which describe the solution.

When a new case is presented to the model, the features of the new case (which are the attributes) are identified and compared with the features of already existing cases in the case base by

using equation 8. The description of the features of the new problem forms a list of attributes and their corresponding value pair but with no solution features. A description of a new case is given in figure 4.

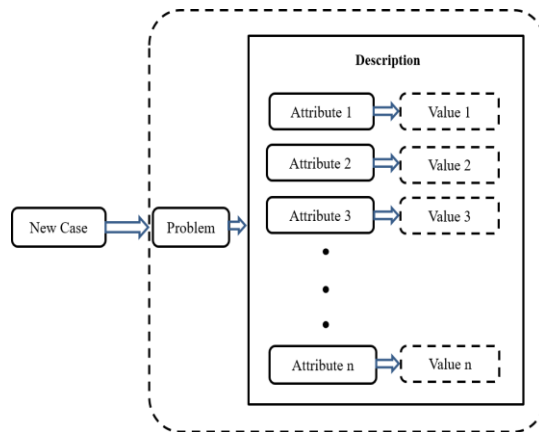


Figure 4: Structure of a New Case

For each case in the case base, a pair-to-pair attribute comparison is carried out using $nhsim(x, y)$ signifying the local similarity measure and cumulated to the Hybridized Similarity scores ($NHSim$) which is calculated by using the proposed geometric similarity measure signifying the global similarity measure.

3.4 Case Retrieval Strategy

The system employs the k-NN retrieval algorithm using the $NHSim$ scores. The retrieval process includes:

- (1) Rank the Retrieved Cases Ranking cases based on their similarity scores;

$NHSim$ will return set of cases with different similarity degrees (values) to the query case. As a result, cases are ranked according to their preference relation which was induced by their utility for solving the case. This creates a ranked list where the most similar cases to the query case appears at the top. Hence, all retrieved case sets were sorted in descending order of their $NHSim$ scores.

- (2) Selecting the top k cases that exceed a predefined threshold τ ;

Top k cases whose similarity values is higher than a set benchmark (threshold, τ) to form the k-nearest neighbours value will be considered for the next task.

- (3) If no cases exceed τ , the top-ranked cases are reviewed by an expert for solution adaptation.

To find the k-NN:

- a. Define a value for the threshold, τ .
- b. Define a value for “ k ”, the desired number of nearest neighbours.
- c. Sort the cases in descending order of their similarity scores (SS).

$$SS_i = S(x_i, q) \text{ for each } x_i \text{ in } X$$

- d. Select the top “ k ” cases from the sorted list as the k-NN:

$$SS(q) = \{x_i \mid i \in \text{argsort}(SS_i) [0 : k]\}.$$

From the foregoing, the system uses a CBR approach embedded with the proposed hybridized geometric similarity measurement method to determine how similar base cases and new cases are. The user enters the problem description of the new problem into the system, this contains the features of the problem. The system uses the defined local similarity measures for

each pair of attributes. Using the proposed *NHSim*, the system uses the local-global approach and integrate the k-NN algorithm to retrieve the most similar base case to the new case. The case which has the highest similarity value (which should be more than the threshold) will be used as the solution to the new case. If the computed similarity score does not surpass the set threshold value, then the case solution will be updated by an expert and will be stored in the case base in case of future episode. The system outputs the solution of the query entered using the proposed algorithm. This approach ensures both automated and expert-assisted retrieval when appropriate.

3.5 Data Collection

This study employed data on catfish diseases, which are well suited for feature-value representation in CBR systems. Secondary data were sourced

from peer-reviewed journal articles [38], [39] and diagnostic manuals on catfish health [40], [41]. Interviews with local aquaculture experts helped validate ambiguous cases, confirm symptom-disease mappings and ensure the dataset's practical relevance.

A total of forty-five (45) disease cases and thirty-three (33) symptoms were documented. Of these, thirty-two (32) cases (60%) were used to populate the case base, while the remaining thirteen (13) cases (40%) were reserved for testing the retrieval model. Cases were

structured into problem descriptions and corresponding solutions to support effective retrieval using the proposed hybrid similarity model.

Snippet of these datasets are presented in Table 3 while Table 4 presents an example of the coded diseases and each symptom was assigned a unique feature code as exemplified by the snippet in Table 5 for clarity and consistency.

Table 3: Snippet of Table Showing the Dataset Description of Common Catfish Diseases and Symptoms

Disease	Symptoms
Dropsy	Swollen abdomen (ascites)
	Lethargic behaviour
	Clamped fins
	Breathing difficulty
	Gasping at the water surface

		Swollen anus
		Curved spine
		Loss of appetite
		Pale gills
		Hemorrhagic margin/signs
		Discolouration of the skin
		Exophthalmia (pop eyes: either or both eyes)
Viral Hemorrhagic Septicemia (VHS)		Hemorrhagic margin/signs
		Exophthalmia (pop eyes: either or both eyes)
		Pale gills
		Lethargic behaviour
		Erratic/abnormal swimming
Epizootic hematopoietic necrosis virus (EHNV)		Hemorrhagic margin/signs
		Swollen abdomen (ascites)
		Discolouration of the skin
		Loss of appetite
Dermatomycosis		Coloured spot or nodules on the skin, fins or gills
		Cotton-like growth on affected areas
		Loss of appetite
		Skin ulcers/lesions
Nitrite Toxicity		Pale gills
		Rapid gill movement

Breathing difficulty
Lethargic behaviour
Loss of appetite
Clamped fins

The simplified case structure enabled the hybrid geometric similarity model to process, compare and retrieve relevant past cases effectively.

Table 4: Snippet of Table Containing Common Catfish Diseases List and Disease Codes

Disease Code	Disease Name
D1	Motile Aeromonad Septicemia (MAS)
D2	Pseudomonas Infection
D3	Enteric Septicemia of Catfish (ESC)
D4	Dropsy
D5	Vibriosis
D6	Streptococcosis
D7	Epitheliocystis
D8	Bacterial Gill Disease (BGD)
D9	Columnaris Disease
.	
.	
.	

D32	Nitrite Toxicity
------------	------------------

Table 5: Snippet of List of Catfish Disease Symptoms and Disease Feature Codes

Feature Code	Symptoms/Features Description
S1	Lethargic behaviour
S2	Exophthalmia (Pop eyes: either or both eyes)
S3	Loss of appetite
S4	Erratic/abnormal swimming
S5	Rapid gill movement
S6	Breathing difficulty
S7	Frayed/eroded gill filament
S8	Skin ulcers/lesions
S9	Cotton-like growth on affected areas
S10	Clamped fins
S11	Excessive mucus production on the gills and/or skin
S12	Ragged/frayed fins and/or tail
S13	Scratching/rubbing body against hard object (Flashing)
S14	Swollen abdomen (Ascites)
S15	Visible flukes under migration
S16	Discolouration of the skin
.	
.	

S33

Swollen gills

3.6 Evaluation Metrics

The performance of the proposed model was assessed using standard metrics: Accuracy, Precision, Recall, F1-score and AUC. These metrics enabled a comprehensive assessment of the model's effectiveness when benchmarked against baseline retrieval methods such as; Euclidean distance, Cosine similarity and Jaccard similarity independently.

The entire implementation was carried out using Python, leveraging libraries such as NumPy, scikit-learn and Pandas for similarity computation, metric evaluation and result visualization.

4.0 Results and Discussion

The model was evaluated using a domain-specific dataset comprising 45 real-world catfish disease cases described by 33 binary symptom attributes.

The dataset was partitioned into a split of 60/40. Where 32 cases were stored in the case base and 13 cases were used as the query cases which represents the new problems or queries used to attempt to solve the cases in the case base. A k-value of 4 and a threshold $\tau = 6.5$ were used for retrieval. While this dataset is sufficient for initial feasibility testing, its relatively small size may limit generalizability to broader applications and should be considered when interpreting results.

4.1 Comparative Performance of the Similarity Measures

The results across all four similarity models; Euclidean, Cosine, Jaccard and the proposed *NHSim* was

assessed using standard evaluation metrics; accuracy, precision, recall, F1-score and AUC-ROC as summarized in table 6 and figure 5.

Table 6: Performance of Similarity Measures

Metric	Euclidean	Cosine	Jaccard	NHSim
Accuracy	0.77	0.77	0.77	0.85
Precision	0.68	0.65	0.65	0.77
Recall	0.77	0.77	0.77	0.85
F1-Score	0.71	0.69	0.69	0.80
AUC	0.88	0.88	0.88	0.92

The results show that while the three traditional measures performed identically in terms of accuracy (0.77), their low precision (0.65-0.68) indicates low case retrieval accuracy.

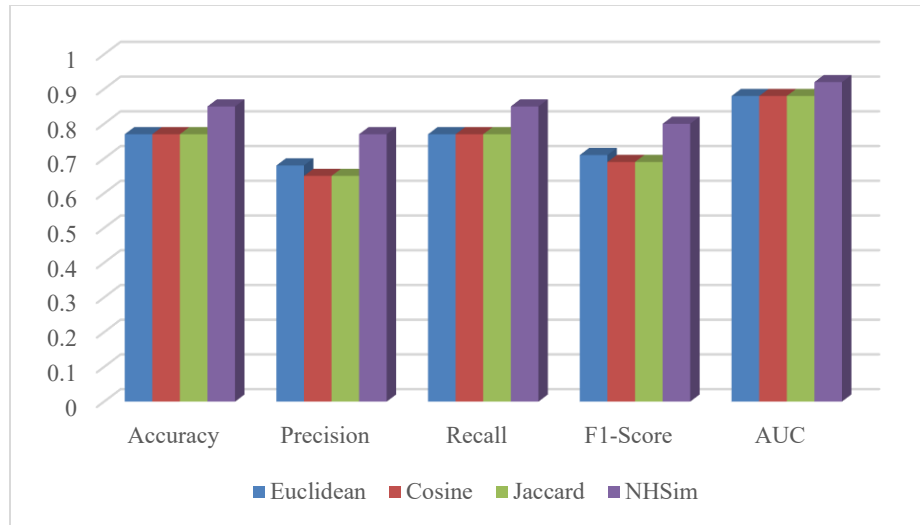


Figure 5: Comparative Performance of Similarity Measures

In contrast, *NHSim* consistently outperformed all baselines, achieving 0.85 accuracy, 0.77 precision, and 0.80 F1-score, with the highest AUC (0.92). The superior performance of *NHSim* across all evaluation metrics as shown in figure 9, can be attributed to its ability to capture multiple dimensions of similarity; magnitude, orientation and binary feature overlap in a single, unified score. This multidimensional perspective enables *NHSim* to distinguish between superficially similar cases and truly relevant ones, thereby reducing false positives and

improving retrieval accuracy. The improved precision and F1-score indicate that *NHSim* retrieves more correct cases while avoiding incorrect matches.

4.2 Patterns in Misclassifications

Analysis of errors revealed that misclassifications were most common in diseases with overlapping symptoms (e.g., grasping at the water surface, lethargy, breathing difficulty) thereby indicating false case retrieval. Traditional models struggled to distinguish such cases because they rely on a single similarity dimension (magnitude,

orientation or binary overlap). *NHSim* reduced but did not completely eliminate these errors, as binary encoding cannot fully capture severity or contextual differences.

4.3 Practical Implications of *NHSim*

The combination of Euclidean (magnitude), Cosine (orientation) and Jaccard (binary overlap) provides a hybrid design (*NHSim*) which provides a multidimensional perspective that balances retrieval accuracy and efficiency. Such balanced retrieval performance ensures that users, such as clinicians, engineers or support staff, receive recommendations that are not only accurate but contextually meaningful. This increases system reliability, fosters user trust and enhances decision-making quality.

Its computationally lightweight nature makes it scalable and well-suited for real-time or resource-limited environments. These are factors critical to decision reliability in diagnosis systems. *NHSim* effectively

balances these metrics, suggesting a well-rounded retrieval performance.

4.2 Discussion

The capturing of multiple dimensions of similarity in a single score introduced by *NHSim* emphasizes shared feature presence in cases. By normalizing and averaging these metrics, *NHSim* creates a balanced and interpretable similarity score that is particularly suited for binary-encoded case structures. The simple approach improves retrieval effectiveness while remaining computationally efficient.

The evaluation confirmed that *NHSim* outperforms Euclidean, Cosine and Jaccard measures in accuracy, relevance and discrimination. The F1-score of 0.80 and AUC of 0.92 indicate that *NHSim* achieves a better trade-off between sensitivity and specificity. This balance is essential in real-world decision-making scenarios where false positives (e.g., misdiagnosis) must be minimized.

Although, despite its improved precision, *NHSim* occasionally wrongly retrieved cases when different diseases shared identical symptom combinations but varied in severity or clinical relevance. This highlights a limitation in binary symptom encoding and suggests that incorporating weighted features or domain-specific symptom hierarchies could further reduce ambiguity in future models.

In practice, *NHSim* offers an efficient and scalable solution for CBR applications particularly in domains like veterinary diagnostics where data are often binary, overlapping and limited in size. Its hybrid design allows it to capture nuanced similarities missed by traditional approaches, thereby improving the accuracy, interpretability and practical relevance of case retrieval in decision support systems

5.0 Conclusion

The challenge of improving case retrieval accuracy in CBR systems, by addressing the

individual limitations of traditional similarity measures and avoiding the complexity of existing hybrid models, motivated the development of a normalized hybrid geometric similarity model (*NHSim*). By integrating Euclidean, Cosine and Jaccard similarities into a single normalized score within the $[0, 1]$ range, application of simple average approach and embedding it in a k-NN framework, *NHSim* delivers a balanced, interpretable and computationally efficient similarity measure that outperformed individual metrics on a veterinary dataset of 45 catfish disease cases and achieving 85% accuracy, 77% precision, 85% recall, 80% F1-score and 92% AUC.

However, while the *NHSim* model shows promising results, the model has limitations such as; fixed equal weighting of similarity components, domain specificity and the small dataset size used. Future work should explore dynamic weight tuning using machine learning, test the model on larger

and multiclass datasets, support mixed data types and improve interpretability through visual explanations.

Nonetheless, the *NHSim* model demonstrates strong potential for building more accurate, reliable and intelligent CBR systems, with enhanced retrieval accuracy, robustness and real-world applicability.

Declarations

Ethics approval and consent to participate

Not Applicable

Consent for publication

Not Applicable

Availability of data and materials

The datasets used in the study are available from the corresponding author on reasonable request.

Competing interests

The authors affirm that they have no conflicting interests.

Funding

No funding was received from any agency for this research.

Authors' contributions

OO conceived the idea. In addition to designing and implementing the model, OO, SO, DB carried out the research. AO validated the data. AO and DB harmonized and ensured the data are well represented. SO structured the manuscript. All authors contributed to the writing process and gave their approval to the final manuscript.

Acknowledgements

We thank the staff and management of Kineth Consult Nigeria Limited, Ibadan for their assistance towards the collection of the datasets used in this research.

References

- [1] Kaurav, S., Devi, G., Deepak, P. and Sutanu, C. (2021). Thinking fast and slow: A CBR perspective. The International FLAIRS Conference Proceedings, 34, 2021.
- [2] Leake, D., Ye, X. and Crandall, D. (2021). Supporting case-based reasoning with neural networks: An illustration for case

- adaptation. Proceedings of the AAAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering (AAAI-MAKE 2021).
- [3] Aamodt, A. and Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations and system approaches. *AI Communications*, 7(1):39–59.
- [4] Sourati, Z., Ilievski, F., Sandlin, H. Â. and Mermoud, A. (2023). Case-based reasoning with language models for classification of logical fallacies. In Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI '23), 576:5188–5196.
- [5] Wang, H. Q., Sun, B. B. and Shen, X. F. (2018). Hybrid similarity measure for retrieval in case-based reasoning systems and its applications for computer numerical control turret design. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 232(5):918–927.
- [6] Wang, D., Liu, J., Lin, Q. and Yu, H. (2024). A decision-making system based on case-based reasoning for predicting stroke rehabilitation demands in heterogeneous information environment. *Applied Soft Computing*, 154, 111358.
- [7] Feuillâtre, H., Auffret, V., Castro, M., Lalys, F., Le Breton, H., Garreau, M. and Haignon, P. (2020). Similarity measures and attribute selection for case-based reasoning in transcatheter aortic valve implantation. *PLoS ONE*, 15(9):1–21.
- [8] Bannour, W., Maalel, A. and Ghezala, H. H. B. (2020). Case-based reasoning for crisis response: Case representation and case retrieval. Proceedings of the 24th International Conference on

- Knowledge-Based and Intelligent Information & Engineering Systems, 176:1063–1072.
- [9] Zhai, Z., Ortega, J. M., Castillejo, P. and Beltran, V. (2019). A triangular similarity measure for case retrieval in CBR and its application to an agricultural decision support system. *Sensors*, 19(21):1–23.
- [10] Mihajlovic, M. and Xiong, N. (2019). Finding the most similar textual documents using case-based reasoning. *ArXiv Preprint*, 1–9.
- [11] Oyelade, O. N. and Ezugwu, A. E. (2020). A case-based reasoning framework for early detection and diagnosis of novel coronavirus. *Informatics in Medicine Unlocked*, 20:1–22.
- [12] Naqvi, S. M. R., Ghufraan, M., Meraghni, S., Varnier, C., Nicod, J. and Zerhouni, N. (2022). CBR-based decision support system for maintenance text using NLP for an aviation case study. In 2022 Prognostics and Health Management Conference (PHM-2022 London), 344–349.
- [13] Heidarian, A. and Dinneen, M. J. (2016). A hybrid geometric approach for measuring similarity level among documents and document clustering. In 2016 IEEE International Conference on Big Data Computing Service and Applications (Big Data Service), 142–151.
- [14] Diallo, B., Hu, J., Li, T., Khan, G. A. and Hussein, A. S. (2021). Multi-view document clustering based on geometrical similarity measurement. *International Journal of Machine Learning and Cybernetics*, 13:663–675.
- [15] Benmessaoud, N., Adla, A. and Bella, A. B. (2019). A CBR decision support system for loan evaluation. *Journal of Digital Information Management*, 17(2):75–86.

- [16] Zhang, K., Wang, Y. and Zheng, J. (2021). Regret theory-based case-retrieval method with multiple heterogeneous attributes and incomplete weight information. *International Journal of Computational Intelligence Systems*, 14(1):1022–1033.
- [17] Bebart, D. K., Das, T. K., Chowdhary, C. L. and Gao, X. (2021). An intelligent hybrid system for forecasting stock and forex trading signals using optimized recurrent FLANN and case-based reasoning. *International Journal of Computational Intelligence Systems*, 14(1):1763–1772.
- [18] Fan, Z. P., Li, Y. H., Wang, X. and Liu, Y. (2014). Hybrid similarity measure for case retrieval in CBR and its application to emergency response towards gas explosion. *Expert Systems with Applications*, 41(5):2526–2534.
- [19] Thota, S. R. and Arora, S. (2024). Neurosymbolic AI for explainable recommendations in frontend UI design—Bridging the gap between data-driven and rule-based approaches. *International Research Journal of Engineering and Technology (IRJET)*, 11(5):766–775.
- [20] Hyung, W., Kim, S. and Jo, J. (2020). Improved similarity measure in case-based reasoning: A case study of construction cost estimation. *Engineering, Construction and Architectural Management*, 27(2):561–578.
- [21] Yan, A. and Cheng, Z. (2023). A review of the development and future challenges of case-based reasoning. *Applied Science*, 14(16):1–22.
- [22] Das, R., Zaheer, M., Thai, D., Godbole, A., Perez, E., Lee, J. Y., Tan, L., Polymenakos, L. and McCallum, A. (2021). Case-based reasoning for natural language queries over

- knowledge bases. *ArXiv Preprint*, arXiv:2104.08762, 1–18.
- [23] Lawanna, T. and Ouncharoen, R. (2023). Case-based maintenance model for handling relevant and irrelevant cases in case-based reasoning system. *Engineering Journal*, 27(11):53–71.
- [24] Zheng, J., Wang, Y. and Zhang, K. (2018). A case retrieval method combined with similarity measurement and DEA model for alternative generation. *International Journal of Computational Intelligence Systems*, 11:1123–1141.
- [25] Khosravani, M. R. and Nasiri, S. (2020). Injection molding manufacturing process: Review of case-based reasoning applications. *Journal of Intelligent Manufacturing*, 31:847–864.
- [26] Ahn, J., Park, M., Lee, H., Ahn, S. J., Ji, S., Song, K. and Son, B. (2017). Covariance effect analysis of similarity measurement methods for early construction cost estimation using case-based reasoning. *Automation in Construction*, 81:254–266.
- [27] Zhang, H., Zhang, Z., Zhou, L. and Wu, W. (2021). Case-based reasoning for hidden property analysis of judgment debtors. *Mathematics*, 9(13):1–17.
- [28] Hamrouni, B., Bourouis, A., Korichi, A. and Brahmi, M. (2021). Explainable ontology-based intelligent decision support system for business model design and sustainability. *Sustainability*, 13(17):1–28.
- [29] Zhai, Z., Martínez, J. F., Martínez, N. L. and Díaz, V. H. (2020). Applying case-based reasoning and a learning-based adaptation strategy to irrigation scheduling in grape farming. *Computers and Electronics in Agriculture*, 178:1–14.
- [30] Varshney, A. K. and Torra, V. (2023). Literature review of the recent trends and applications in various fuzzy rule-

- based systems. *International Journal of Fuzzy Systems*, 25(6):2163–2186.
- [31] Schank, R. C. (1982). *Dynamic memory: A theory of reminding and learning in computers and people*. Cambridge University Press, Cambridge.
- [32] Parsodkar, A. P., Deepak, P. and Chakraborti, S. (2023). The case for circularities in case-based reasoning. In Massie, S. & Chakraborti, S. (eds), *Case-Based Reasoning Research and Development, Lecture Notes in Computer Science*, 85–101.
- [33] Zhou, L. and Sordo, M. (2021). Expert systems in medicine. In *Artificial Intelligence in Medicine*, Academic Press, 75–100.
- [34] López, B. (2022). *Case-based reasoning: A concise introduction*. Springer Nature.
- [35] Lin, R., Kujabi, B. K., Chuang, C., Chen, Y. and Chen, C. (2022). Improving the accuracy of misclassified breast cancer data using machine learning. *Eximia*, 4(1):19–32.
- [36] Oladapo, K. A., Adedeji, F., Nzenwata, U. J., Quoc, B. P. and Dada, A. (2023). Fuzzified case-based reasoning blockchain framework for predictive maintenance in Industry 4.0. In *Data Analytics and Computational Intelligence: Novel Models, Algorithms and Applications*, 132:269–297.
- [37] Long, X., Li, H., Du, Y., Mao, E. and Tai, J. (2021). A knowledge-based automated design system for mechanical products based on a general knowledge framework. *Expert Systems with Applications*, 178:1–18.
- [38] Al-Saadi, A. S. T. and Al-Obaidi, T. S. M., Substituting Fishmeal and Fish By-Products (Internal Organs) for Imported Animal Protein, IOP Conference Series: Earth and

- Environmental Science, 1262(7), and Culture of Channel Catfish, CS Tucker
2023, 1-8. and JA Hargreaves (Editors), 2004
[39] Fryer, J. L. and Rohovec, J. S., Bacterial Elsevier BV All rights reserved, 387.
Diseases of Fish, *Pathobiology of Marine
and Estuarine Organisms*, 2021, 53-
83.
[40] Dimado, F., The Ultimate Catfish
Farming Blueprint, Kindle Edition,
[https://www.amazon.com/dp/B0BSX
SY7DM](https://www.amazon.com/dp/B0BSXSY7DM)
[41] Hawke, J. P. and Khoo, L. H., Biology

www.fjpas.fuoye.edu.ng