# Pneumonia Diagnosis Using Deep Neural Networks and Fuzzy Logic with Dynamic Symptom Adjustment

Sulav Baral1, Rabindra Bista1, Sanjog Sigdel1 and João C Ferreira2

1 Department of Health Informatics, Kathmandu University School of Engineering, Dhulikhel, Kavre, Nepal 2 ISTAR-IUL, Instituto Universitário de Lisboa (ISCTE-IUL), Lisbon, Portugal

# Abstract

Pneumonia remains a leading infectious disease with high mortality rates, particularly among children under five and adults over 65 (WHO). Early and accurate diagnosis is crucial for reducing its global burden. Chest X-ray (CXR) imaging is the primary radiological tool in emergency settings for pneumonia detection. This study proposes a novel Artificial Intelligence (AI)-driven approach by integrating a Convolutional Neural Network (CNN) with Capsule Network (CapsNet) and Fuzzy Logic. Using TensorFlow’s DenseNet201 and CapsNet models, our system classifies CXR images as either Normal or Pneumonia with 97.96% accuracy. A key contribution of this work is the dynamic fuzzy membership adjustment, enabling real-time adaptation to evolving symptom severity, achieving a stabilization accuracy of 97.08%. Stratified K-Fold, Bootstrap, and Monte Carlo cross-validation further validate generalizability across diverse datasets. These findings highlight the system’s potential for clinically interpretable, automated pneumonia severity classification, bridging the gap between high diagnostic accuracy and real-world applicability.

# 1. Introduction

As per World Health Organization (WHO), Pneumonia continues to pose a global threat, particularly between the age group of children under the age of 5 and adults above 65 (1). Early diagnosis of pneumonia is critical to ensure proper treatment and increase survival rates, particularly in vulnerable populations such as Children and the elderly (2). There is a need for an Artificial Intelligence (AI) based diagnosis that provides transparency in decision-making and can track major pneumonia symptoms that are responsible for the high mortality rate (3,4) since the manual diagnosis of Pneumonia from chest radiography has been tedious due to overlapping symptoms with other respiratory conditions and the inter-observer variability among radiologists (5,6). The study explores hybrid models combining deep learning with dynamic symptom analysis in chest X-ray imaging. It aims to improve diagnostic accuracy and transparency for radiologists by combining deep learning-based feature extraction with fuzzy logic, thereby bridging the gap between static classification and evolving symptom severity over time (7). Significant contributions of our research are mentioned below:

AI-Driven Pneumonia Diagnosis: Our hybrid model integrates CNNs, CapsNet, and fuzzy logic for real-time symptom tracking, improving interpretability and diagnostic accuracy.

Overcoming Overfitting with Transfer Learning: Using DenseNet201 and CapsNet on 17,229 CXRs, we tackled data limitations, achieving 97.65% accuracy.

Dynamic Fuzzy Logic in Symptom Modeling: CNN outputs adjust fuzzy membership dynamically, ensuring 97.08% stabilization for accurate pneumonia severity estimation.

# 2. Literature Review

## 2.1 Chest X-ray Imaging and Pneumonia Disease

The main imaging method for detecting pneumonia is a chest X-ray (CXR) (8), which is accessible and reasonably priced. However, because CXR pictures have characteristics with other lung disorders, it can be difficult to interpret them, and radiologists may diagnose different conditions (9). Aforementioned, the necessity of automated diagnostic systems has consequently drawn a lot of attention. Artificial intelligence (AI)-driven methods and computer-aided detection (CAD) systems have been created to increase the accuracy and decrease diagnostic inconsistencies of CXR-based pneumonia diagnosis.

## 2.2 Machine Learning and AI in Pneumonia Diagnosis

Artificial intelligence and machine learning have revolutionized pneumonia identification by automating chest X-ray processing and minimizing human interpretation (10–12). Deep learning, particularly convolutional neural networks, has improved pneumonia categorization by extracting complex patterns from massive datasets. AI-driven diagnostic solutions reduce human error and expedite procedures, benefiting radiologists and improving the overall quality of care (13).

## 2.3 CNN Models for CXR Image Processing and Classification

Convolution Neural Network (CNN) (14–16) has been widely preferred for Chest X-ray (CXR) to perform image feature extraction. Varshni et al. (17) proposed a work that used CNN’s DenseNet-169 model for feature extraction and SVG as a classifier, achieving an AUC of 0.8002 . Similarly, the work of Rahman et al. (18) utilized four pre-trained CNN models (AlexNet, ResNet18, DenseNet201, SqueezeNet) to classify 5247 chest X-ray images of normal, bacterial pneumonia, and viral pneumonia proved DenseNet201 being the most effective CNN model for CXR for achieving 98% accuracy . Chutia et al. (19) utilized DenseNet201 for COVID-19 and Pneumonia detection which achieved an accuracy of 95.34%. Sanghvi et al. (20) achieved 99.1% accuracy for detection of COVID and Pneumonia by utilizing DenseNet201 for CXR images.Table 1 discusses the previous works in DenseNet architecture using Chest X-ray images and the results achieved.

**Table 1. Previous studies Utilizing Dense Architecture for Disease Diagnosis**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Study** | **Preferred Model** | **Target Disease** | **Accuracy** | **Remarks** |
| Kundu et al. (21) | DenseNet-121 | Pneumonia | 98.81% | Achieved high accuracy; used data augmentation to improve model generalization. |
| Jaiswal et al. (22) | DenseNet-201 | COVID-19 | 96.25% | Achieved highest accuracy on DenseNet201 |
| Rochmawanti & Utaminingrum (23) | DenseNet-121 | Tuberculosis, Pneumonia, Cardiomegaly, COVID-19 | Tuberculosis: 89.2%, Pneumonia: 90.4%, Cardiomegaly: 89.8%, COVID-19: 98.6% | Best accuracy achieved with 224x224 image resolution; Global Average Pooling and dropout used to mitigate overfitting; batch normalization accelerates training |
| Anakha et al. (24) | DenseNet201 | COVID-19 | 96.54% | DenseNet201 outperformed other models with the best accuracy for COVID-19 detection from X-rays |
| Chutia et al. (25) | DenseNet201 | Lung diseases: Pneumothorax, and Atelectasis | 95.34% | DenseNet201 proved efficient compared to other models |

DenseNet architectures have shown remarkable metrics (26,27), particularly for extracting image features. However in a clinical setting, Deep learning isn’t standalone enough due to variability in Pneumonia’s symptom pattern, as Xin KZ et al (28) found that deep convolutional neural networks performed well on internal datasets with an AUC of 0.95, but their accuracy fell to 0.54 on external datasets, indicating limited generalizability in different clinical settings.

## 2.4 Fuzzy Membership for Symptom-Based Pneumonia Diagnosis

Fuzzy logic (29,30) has been utilized to handle imprecise information by combining it with Neural Networks (31). Fuzzy logic systems are good at managing uncertainty and gradual changes (32). For instance, Rakshitha et al. (33) proposed a hybrid pneumonia detection system integrating CNNs (VGG16, ResNet50V2) with a fuzzy expert system, achieving higher reliability than standalone CNNs by combining clinical parameters and X-ray features, producing percentage-based pneumonia likelihood for improved diagnosis. Similarly, Arani et al. (34) created a system to diagnose pneumonia using Mamdani-style fuzzy logic which uses 17 input variables, like body temperature and cough severity, to assess the likelihood of pneumonia but only relies on rule-based fuzzy inference.

**Table 2. Comparison of Static and Dynamic Fuzzy Logic Approaches in Pneumonia Diagnosis.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | **Fuzzy Logic Adaptability** | **Handling of Borderline Cases** | **Clinical Relevance & Usability** |
| Rakshitha et al. | Static fuzzy rules applied to clinical symptoms. | No specific method for uncertain cases. | Integrates clinical parameters but lacks adaptability. |
| Arani et al. | Mamdani fuzzy logic with 17 input variables, but rules are fixed. | Uses multiple fuzzy variables but lacks CNN-based refinement. | Handles multiple respiratory diseases but lacks real-time updates. |
| This particular work | Dynamic fuzzy membership updates based on symptom trends. | CNN confidence refines fuzzy severity for borderline cases. | Tracks evolving symptoms, improving real-time severity estimation. |

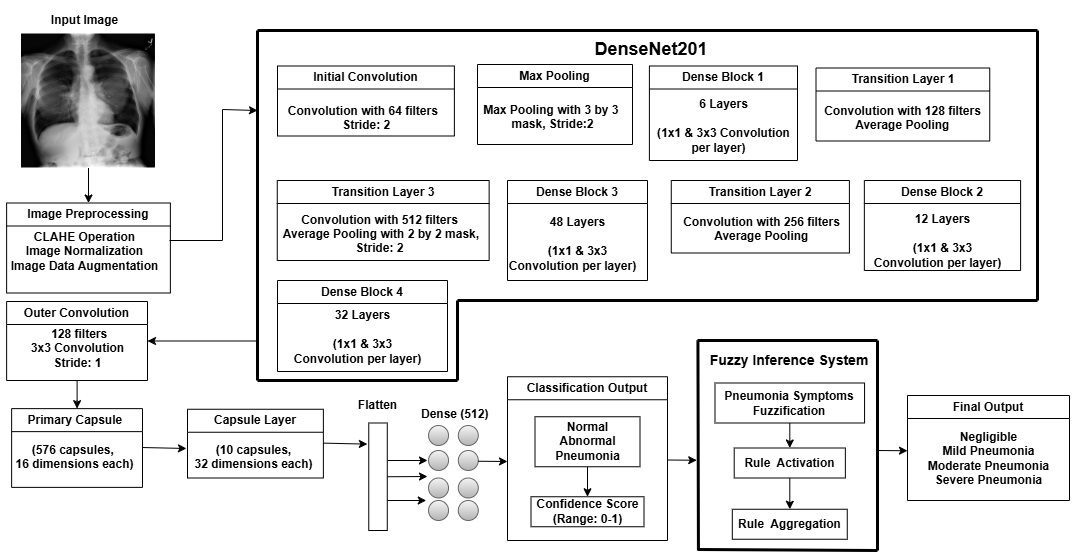
# 3. Methods

## 3.1 Data Collection

For this study, we acquired 17,229 CXR image datasets from Kaggle.com (35). A systematic approach was employed to train and validate the model using datasets generated through multiple folds. The standard anatomy of the chest X-ray for Pneumonia Diagnosis are taken in different position. The dataset consists of varying orientation of Chest X-rays, including both male and female of different age. There are three classes of images: “Abnormal”, “Normal” and “Pneumonia” split into a specific Train to Validation to Test dataset as per the cross-validation method utilized in this work.

## 3.2 Proposed Method

We propose a novel method for Pneumonia diagnosis by combining a DenseNet201-based feature extraction process with a fuzzy inference system to classify pneumonia severity level. This process consists of three main parts: image preprocessing, feature extraction using DenseNet201, and severity classification via a fuzzy inference system. The work flow is illustrated in **Figure 1**.



**Figure 1. Flowchart of CNN-Based Fuzzy Inference System for Pneumonia Severity Classification and Symptom Analysis**

### 3.2.1 Input Image Preprocessing

The preprocessing phase enhances input images for feature extraction. This phase ensures uniformity and enhances the quality of images, addressing challenges related to varying imaging conditions, noise, and limited datasets. The following techniques are applied.

**Contrast Limited Adaptive Histogram Equalization (CLAHE):** CLAHE was applied to improve CXR image quality, especially in low-contrast areas of the CXR, utilizing Clip Limit of 2.0 and the image split into an 8 by 8 tile, to prevent over-amplification in the image region and the tile grid size to divide the image for applying CLAHE operation. The clip limit was calculated as:

|  |  |  |
| --- | --- | --- |
|  |  | () |

After calculating Clip limit via Equation 1, the mapping function calculates new intensity for each pixel in the tile, enhancing local contrast as:

|  |  |  |
| --- | --- | --- |
|  |  | () |

Once each tile has been processed by Equation 2, the tiles are blended together to form the final equalized image.

**Image Normalization:** Normalization adjusts the pixel values in images to a similar range, typically [0,1]. For an 8-bit image whose maximum pixel value is 255, the Normalization (Equation 3) is performed as:

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

**Data Augmentation:** To simulate real-world imaging variances, especially for chest X-rays, TensorFlow-based data augmentation was utilized to increase model resilience. Spatial diversity was created by the use of techniques such as zooming, shearing, random rotation, and horizontal and vertical shifts. Better pneumonia diagnostic accuracy is achieved by flipping to account for patient orientations, adjusting brightness to simulate changing lighting conditions, and using nearest-neighbor filling to retain anatomical features.

### 3.2.2 Image Feature Extraction

Though this work also experimented on DenseNet121 and DenseNet169, the DenseNet201 architecture has been leveraged to extract robust features from the preprocessed images, and the reason is provided in the Experiment section of this report. Key components of this architecture include:

**Initial Convolution and Max Pooling:** The input image undergoes convolution with 64 filters and max pooling to reduce spatial dimensions.

**Dense Blocks and Transition Layers:** DenseNet201 employs four dense blocks interconnected by transition layers. These layers use batch normalization, ReLU activation, and convolutional filters for feature extraction. Outputs are progressively refined, culminating in high-dimensional feature maps.

**Feature Bottleneck:** The final dense block produces a feature tensor, which undergoes further processing through custom convolutional layers and capsule layers.

### 3.2.3 Capsule Network for Classification

The feature tensor is passed through a capsule network (36), enabling dynamic routing and hierarchical feature representation:

**Primary Capsule Layer:** These transform feature maps into capsule vectors, preserving spatial hierarchies.

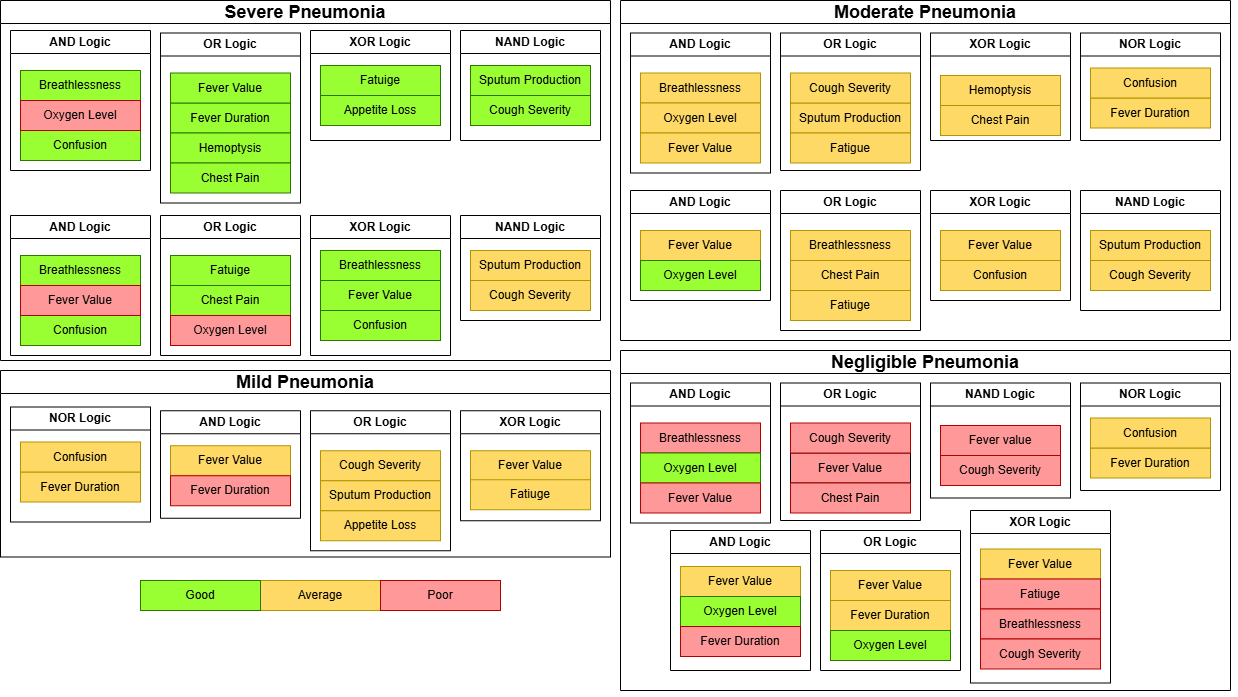
**Capsule Layer:** Capsules with higher dimensions encode spatial relationships between features.

**Dense and Dropout Layers:** The capsule output is flattened and passed through dense layers with dropout regularization. The final classification layer outputs three confidence scores: *Normal,* *Abnormal,* and *Pneumonia*.

### 3.2.4 Fuzzy Inference System for Severity Classification and Membership Adjustment Function

A collection of language expressions known as fuzzy rules explain how to use a fuzzy inference system to classify inputs or regulate outputs. The fuzzy inference system translates classification confidence scores and pneumonia symptoms into severity levels:

**Fuzzification:** The Mamdani-type fuzzy Inference System maps symptoms and CNN classification confidence scores into Fuzzy sets using Fuzzy Membership Functions (type-1 fuzzy), categorizing them into linguistic labels such as "Poor," "Average," and "Good." These membership categories are crucial for transforming crisp input values into fuzzy representations, forming the basis for rule-based reasoning. This implementation uses piecewise linear membership functions to determine fuzzy grades based on symptom severity. A fuzzy rule-based method then evaluates multiple patient symptoms to determine the pneumonia severity score. Logical operators AND, OR, XOR, and NAND are used to establish the final severity level **Figure 2**, ensuring robust classification.



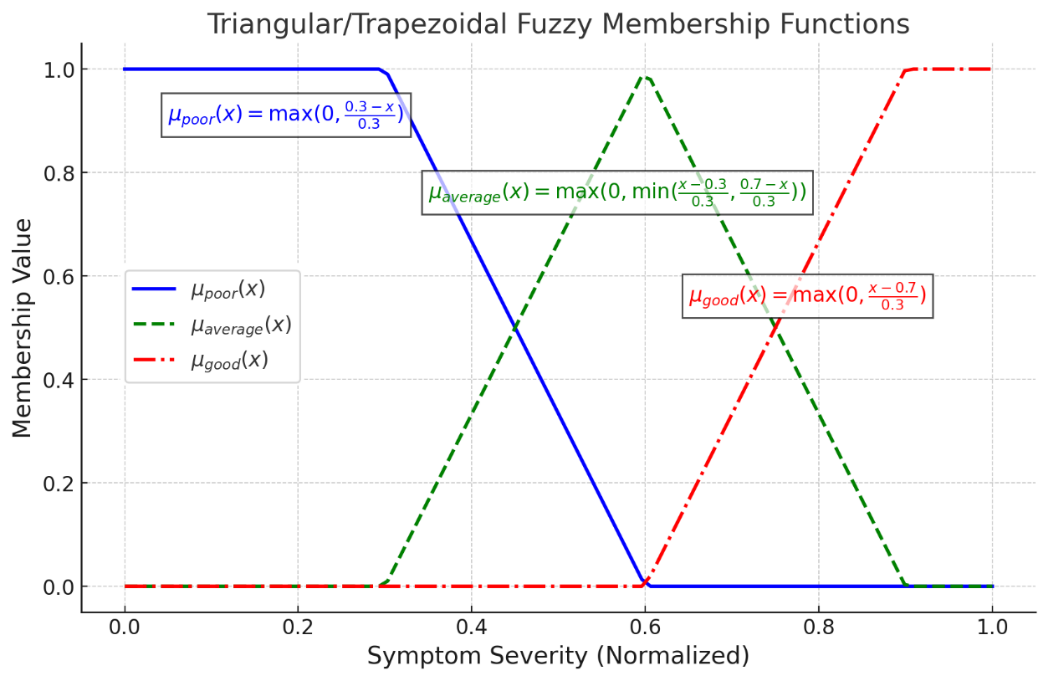
**Figure 2. Fuzzy Logic-Based Pneumonia Severity Classification Using Symptom Contribution**

Each input symptom x is transformed into its corresponding fuzzy membership grade using the piecewise linear functions where x represents the crisp symptom value, min and max denote the range of possible symptom values. **Figure 3** illustrates how pneumonia symptoms are mapped into fuzzy categories—Poor, Average, and Good—using triangular/trapezoidal membership functions. Each symptom is assigned a membership value based on its normalized severity, allowing for a smooth transition between categories.

**Table 3. Symptom Classification by Fuzzy Membership Ranges (Poor, Average, and Good).**

|  |  |
| --- | --- |
| **Membership Category** | **Range (Normalized Values)** |
| Poor | x ≤ 0.3 |
| Average | 0.3 < x <0.7 |
| Good | x ≥ 0.7 |

The "Poor" function decreases as severity increases, "Good" increases with severity, while "Average" peaks in between, covering intermediate values. These functions are critical for the fuzzy logic-based pneumonia diagnosis as the membership category is determined (as in Table 3) for each symptom.



**Figure 3. Fuzzy Membership Functions for Symptom Severity**

**Rule Activation and Aggregation:** Expert-defined fuzzy rules are activated based on input variables, in the form of IF-THEN statements. All conditions must be satisfied simultaneously in a rule for it to be triggered. Since multiple rules exist, the final aggregated membership for pneumonia severity is taken as the maximum of all activated rules where m represents the total number of fuzzy rules and the represents a particular rule.

|  |  |
| --- | --- |
|  | (4) |

is the weight assigned to rule k based on logic type AND (+10 points), OR (+8 Points), XOR (+9 Points) and NAND (+7 Points). is the membership activation of the kth rule. To ensure the stability and accuracy of severity classification, CNN confidence scores are integrated with the fuzzy rule-based severity estimator. CNN confidence is applied as a proportional weighting factor, adjusting the fuzzy severity score dynamically. The CNN confidence contribution is computed as shown by Equation 5.

|  |  |
| --- | --- |
|  | (5) |

This method ensures that higher fuzzy severity scores receive greater CNN confidence contribution, reinforcing severe classifications. When total severity scores are low, CNN confidence has minimal influence, preventing overestimation. The final severity classification is determined by the highest adjusted severity score, balancing contributions from fuzzy rules and CNN confidence (**Table 4**).

**Final Output:** The system classifies pneumonia into four severity levels: Negligible Pneumonia, Mild Pneumonia, Moderate Pneumonia, and Severe Pneumonia.

**Table 4. Final Pneumonia Severity Classification Based on Rule-Based Scoring**

|  |  |
| --- | --- |
| **Severity Level** | **Rule-Based Score Range** |
| Negligible | Total severity score < 10 |
| Mild | 10 ≤ Score < 25 |
| Moderate | 25 ≤ Score < 40 |
| Severe | Score ≥ 40 |

Negligible Pneumonia cases have few activated rules, leading to a low severity score. Mild Pneumonia falls within a score range of 10 to 25, indicating symptoms that are present but not severe. Moderate Pneumonia scores 25 to 40, reflecting multiple symptoms without extreme severity, while Severe Pneumonia exceeds 40, with strong fuzzy rules activated. CNN confidence enhances classification consistency, reinforcing severity determination. Key symptoms like breathlessness, low oxygen levels, and high fever play a critical role in diagnosis. The final severity classification is based on the highest adjusted score after applying CNN confidence, ensuring a robust and adaptable clinical assessment.

## 3.3 Dynamic Membership Adjustment Algorithm

This work has integrated Dynamic Fuzzy membership adjustment to cover the problem of biased Diagnosis form the previous static models.

**Step 1:** Check the Data Frame for Trending Symptoms, then perform Data cleaning by Filtering out values which deviates by more than three standard deviations from the mean using Z-Score (37).

|  |  |
| --- | --- |
|  | (6) |

**Step 2:** Calculate recent variance σ2 and Mean μ, for the last 10 data points (n=10)

If data points < 10; use variance and mean of all available data

|  |  |
| --- | --- |
|  | (7) |

**Step 3:** Calculate the smoothing factor S with initial base smoothing B defined as 0.3

|  |  |
| --- | --- |
|  | (8) |

**Step 4:** Apply the exponential smoothing (38)

**4.1** Initialize first value of the weighted symptom data

**4.2** For each symptom data point xi calculate smoothed value

|  |  |
| --- | --- |
|  | (9) |

**Step 5:** Set percentile thresholds values with respect to recent mean value μ

**5.1** If μ < 0.5; set lower percentile = 10% and upper percentile = 90%

**5.2** If μ > 0.5; set lower percentile = 15% and upper percentile = 85%

**Step 6:** Calculate adjusted minimum and maximum range as per the percentile thresholds values, then generate Adjusted range Ar of values from the adjusted minimum to the adjusted maximum with an increment of 0.1

|  |  |
| --- | --- |
|  | (10) |

**Step 7:** Limit Adjusted range Ar within the specified minimum and maximum bounds and return the clipped range

# 4. Results

## 4.1 Simulation Environment

The system, powered by an Intel Core i7 10700k CPU, 16GB RAM, and an NVIDIA RTX 3070 GPU, was designed for efficient deep learning tasks. It was developed using Windows 10 with Visual Studio Code, Sublime Text Editor, and Jupyter Notebook, and Python as the primary programming language.

## 4.2 Libraries Used

TensorFlow was used to define and load deep learning models, such as Capsule Network and CNN models. This library is the core of Data augmentation and image preprocessing methods used for model training, and were trained efficiently to classify the CXR images (39,40). It was also used to build custom layers that implemented the Capsule Network function. NumPy (41) was used for array based operations like flattening model features, Image Preprocessing operation (CLAHE), image normalization and calculating fuzzy input factors for fuzzy logic classification. The fuzzy membership values assigned to symptoms were calculated dynamically using NumPy operations, ensuring smooth integration between deep learning outputs and the rule-based fuzzy inference system. Matplotlib was used for plotting training accuracy and Validation loss plots (42), and specifically for plotting Grad-CAM heatmaps. The skfuzzy library was integrated for fuzzy logic operations, specifically for defining fuzzy membership functions and handling uncertainty in symptom classification.

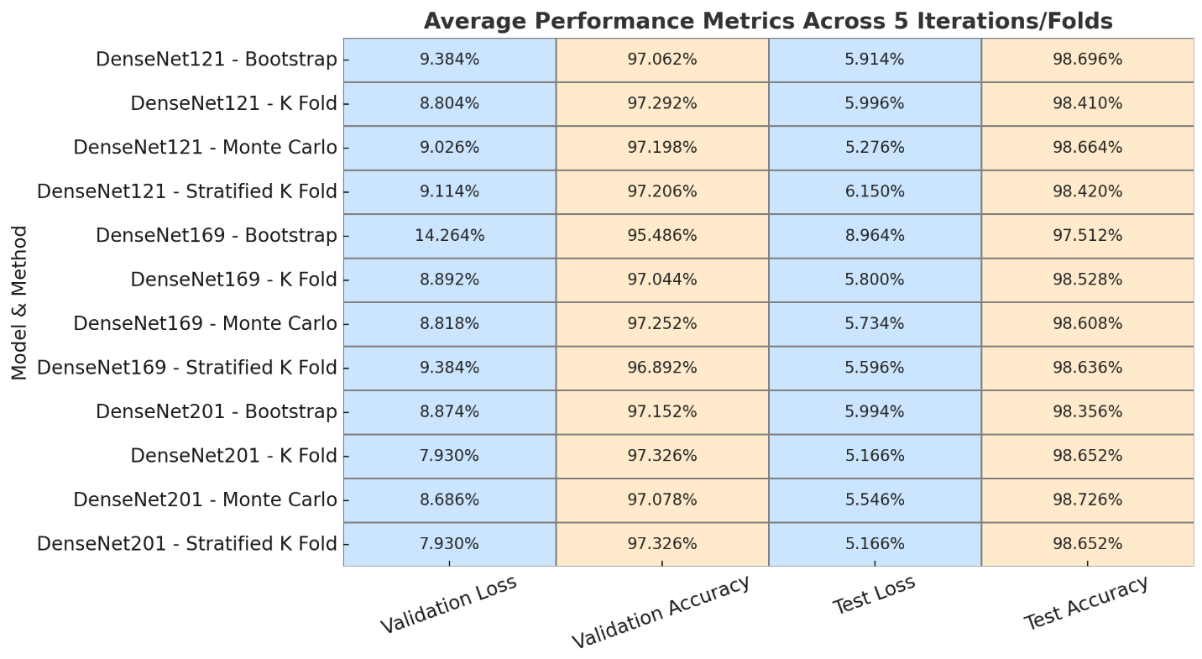
## 4.3 Cross Validation Methods and Neural Network based Classification

Several strategies were followed in data splitting to reliably test, validate, and train the model. Stratified K-Fold Cross-Validation (43) was performed in a way that classes are equally distributed within the folds to fairly represent them during the assessment of their performance. Monte Carlo Cross-Validation (44) allowed multiple different validation measures through random division of data into a training set and a test set multiple times. Multiple training datasets were produced via the resampling bootstrap (45) used to provide a view of the stability and variability and allowed research on model performance onto out-of-the-bag samples. k-Fold Cross-Validation (46) was used to group data into folds based on feature similarity, providing an alternative approach for splitting those accounts for underlying patterns in the data distribution.

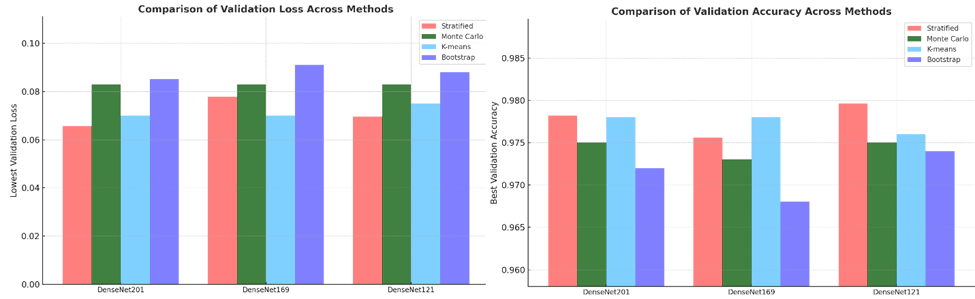
Each dataset served a distinct role: the test set evaluated model performance on unseen data, the validation set facilitated tuning and tracking, and the training set identified regularities and correlations. This comprehensive approach mitigated bias from any single partitioning strategy. Figure 4 illustrates training and validation ratios for the three categories (NORMAL, PNEUMONIA, ABNORMAL) across various methods—Stratified k-Fold, k-Means Fold, Monte Carlo, and Bootstrap—over multiple folds.

**Figure 4. Training and Validation Ratios across Five Different Folds for NORMAL, PNEUMONIA, and ABNORMAL classes.**

To identify the most suitable model for pneumonia diagnosis, we evaluated three DenseNet architectures (DenseNet121, DenseNet169, and DenseNet201) using four validation methods: Stratified K-Fold, Monte Carlo, K-Means, and Bootstrap.



**Figure 5. Average performance metrics across 5 iterations/folds for DenseNet models, comparing validation loss, validation accuracy, test loss, and test accuracy**



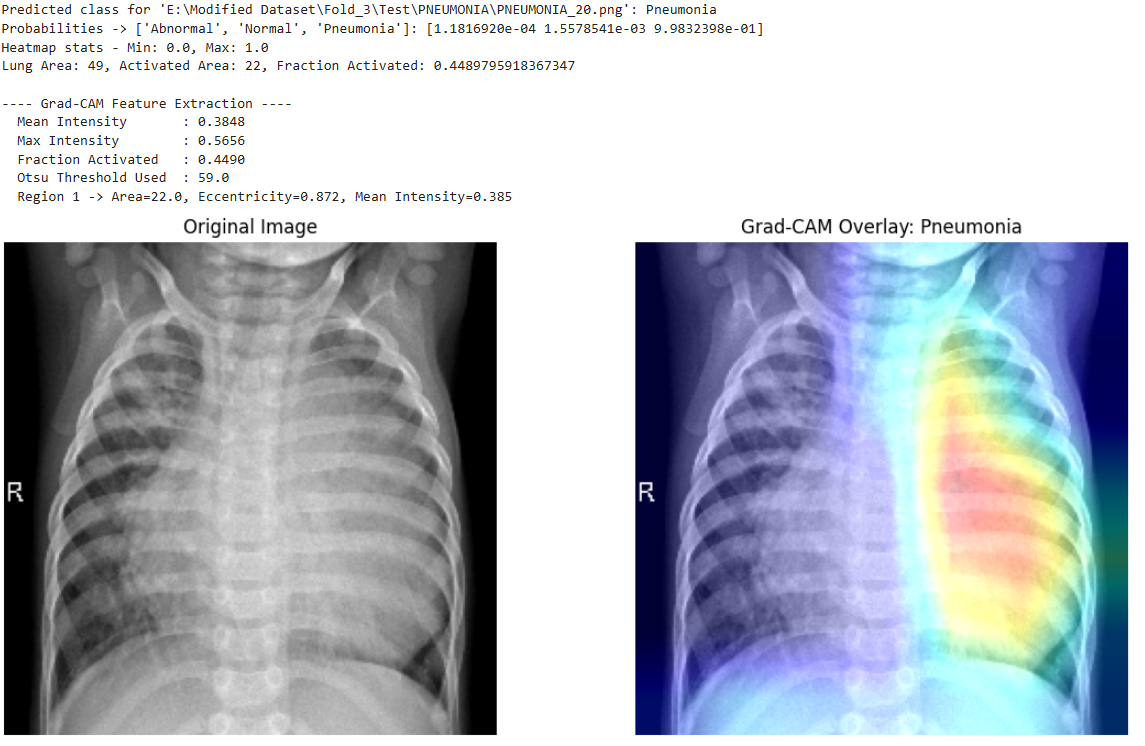
**Figure 6. Bar diagram illustrating highest validation accuracy and lowest Validation achieved from 5 different iterations/folds across different cross-validation methods.**

In our research, we chose DenseNet201 due to its consistent performance in validation accuracy and loss across various validation methods. DenseNet201 achieved the lowest validation loss of 0.0656 and a strong validation accuracy of 0.9782 using the Stratified method, outperforming DenseNet121, which recorded a lowest validation loss of 0.0695 and an accuracy of 0.9796. While DenseNet121 demonstrated competitive results, DenseNet201 provided more stable and reliable outcomes with reduced losses across all validation methods, including Monte Carlo, K-means, and Bootstrap. These findings highlight DenseNet201's superior generalization and robustness, making it the more suitable architecture for our study.



**Figure 7. Training performance metrics for DenseNet201 - Fold 3: Accuracy and loss trends over epochs, learning rate adjustments, and accuracy difference indicating generalization behavior.**

As shown in **Figure 7**, DenseNet201 in 3rd fold showed efficient convergence, reducing overfitting and narrowing the accuracy gap between training and validation. Its balanced performance indicated strong generalization, while metrics like accuracy difference provided deeper insights into model behavior.

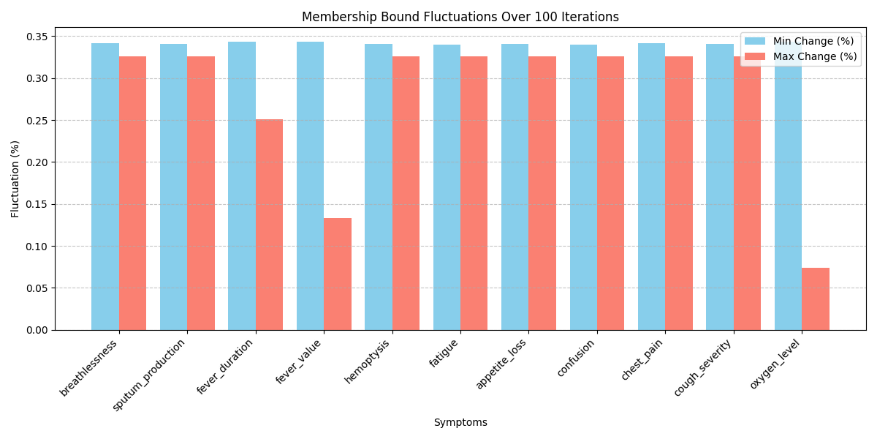


**Figure 8 Figure: Chest X-ray analysis for pneumonia diagnosis. (Left) Original image. (Right) Grad-CAM overlay highlighting key regions of interest.**

**Figure 8** shows a Grad-CAM visualization for pneumonia classification, enhancing AI transparency by highlighting key decision regions and aiding clinical interpretation, especially in resource-limited settings.

## 4.4 Result of Implementing Dynamic Membership Adjustment Algorithm

The fuzzy membership stabilization system enhances pneumonia severity classification by dynamically adjusting symptom boundaries over time. It refines limits for breathlessness, sputum production, fever duration, hemoptysis, and confusion through evaluations and smoothing methods. Initially fixed, these boundaries adapt based on stabilization tests at iterations 10, 20, 30, 50, and 100. With each iteration, new symptom values update membership functions, ensuring stable assessments.



**Figure 9. Membership Bound Fluctuations Over 100 Iterations: The chart shows percentage changes in fuzzy membership boundaries for pneumonia symptoms, with blue bars for minimum fluctuations and red bars for maximum fluctuations.**

A damping factor reduces large variations' effects, enhancing adaptability. To filter outliers, a Z-score method is used, preventing extreme values from impacting adjustments. Additionally, an exponential smoothing algorithm is applied, which adjusts the weight of recent symptom values based on variance and thresholds for gradual updates. To measure effectiveness, fluctuation percentages are calculated at different steps, and the reduction over time assessed. Stabilization accuracy is determined by comparing fluctuations from 10 to 100 iterations.

To quantify stabilization, the system computes initial and final averages of membership fluctuation, as shown in Equation 13 and 14.

|  |  |
| --- | --- |
|  | () |
|  | () |

Where Mean Initial Min Change and Mean Initial Max Change are the average minimum and maximum bound fluctuations at the 10th iteration, respectively, while Mean Final Min Change and Mean Final Max Change represent the average minimum and maximum bound fluctuations at the 100th iteration. If any symptom fluctuates more than 1% at 100 iterations, additional stabilization steps are triggered to further refine the fuzzy membership boundaries. The stabilization factor (SF), which ensures controlled adjustments, is calculated as:

|  |  |
| --- | --- |
|  | () |

The fluctuation reduction (F), which measures the percentage decrease in fluctuation over iterations, is given by:

|  |  |
| --- | --- |
| \* 100 | () |

Finally, the accuracy for each Symptom is computed as

|  |  |
| --- | --- |
|  | () |

With an overall stabilization accuracy of 97.08%, the system effectively reduces fluctuations in fuzzy membership adjustments, ensuring reliable and consistent pneumonia symptom tracking for healthcare practitioners.

## 4.5 CNN Confidence Influence on Severity Classification

The dynamic membership adjustment algorithm allows symptoms to evolve over time, but this can sometimes lead to borderline classifications. CNN confidence acts as a moderating factor, reinforcing severity determination. For example, if a patient presents with moderate breathlessness, a temperature of 38.9°C, and an oxygen saturation of 94%, fuzzy rules alone may struggle to classify the case as Moderate or Severe. High CNN confidence raises the severity score, increasing the likelihood of a Severe classification. Low CNN confidence maintains the Moderate classification, preventing unnecessary escalation. This ensures a balanced severity assessment, improving decision stability and interpretability. **Table 5** illustrates how CNN confidence refines classification, enabling smooth transitions in severity evaluation.

**Table 5. (Illustration) Impact of CNN Confidence on Fuzzy Severity Scoring**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case | Breathlessness | Fever (°C) | Oxygen Level | Fuzzy Rule-Based Severity | CNN Confidence | Final Severity After CNN Influence |
| 1 | 0.6 | 38.9 | 94 | Moderate (34.2) | 72% | Severe (36.8) |
| 2 | 0.3 | 37.5 | 96 | Mild (21.7) | 48% | Mild (22.9) |
| 3 | 0.9 | 40.1 | 89 | Severe (41.8) | 85% | Severe (44.3) |
| 4 | 0.5 | 39.2 | 93 | Moderate (29.6) | 65% | Moderate (31.8) |
| 5 | 0.2 | 36.8 | 98 | Negligible (14.1) | 40% | Negligible (14.7) |

These findings indicate that greater CNN confidence values lead to minor increases in severity scores, potentially moving borderline cases into the appropriate severity category. Simultaneously, the CNN's input stays consistent, making sure that minor variations in symptoms do not result in abrupt or significant changes in classification. This method results in diagnoses that are more stable, easier to interpret, and clinically valuable, lowering the chances of misclassification in ambiguous situations.

# 5. Conclusion and Future Work

This study introduces a hybrid AI model that integrates CNN-based feature extraction, Capsule Networks, and dynamic fuzzy membership adjustment to enhance pneumonia diagnosis. The system achieved 97.96% classification accuracy, outperforming traditional CNN methods. A key feature is the real-time adjustment of fuzzy membership values, ensuring stable pneumonia severity assessments. With a stabilization accuracy of 97.10%, the fuzzy inference system effectively refines severity classifications using trend evaluation, outlier removal, and smoothing techniques, minimizing drastic changes. Results confirm that these methods reduce fluctuations, ensuring consistent diagnostics. This study shows strong performance but needs development in key areas. Integrating transformer architectures could improve the capture of sequential dependencies in evolving Pneumonia symptoms, especially for follow-up cases. Future efforts should also improve the weighting of symptom significance within fuzzy rules to accurately reflect real-world diagnostic relevance. Validating the hybrid AI system in real-time clinical settings is essential for healthcare practitioners to evaluate its robustness and interpretability.

# Acknowledgement

This study is intended solely as a clinical decision-support tool, not a standalone diagnostic system. It does not endorse any commercial products, and all methods and data sources are transparently presented. Ethical standards are maintained, and further real-world testing is needed to evaluate its practical use in healthcare.

# References

1. McAllister DA, Liu L, Shi T, Chu Y, Reed C, Burrows J, et al. Global, regional, and national estimates of pneumonia morbidity and mortality in children younger than 5 years between 2000 and 2015: a systematic analysis. Lancet Glob Health. 2019 Jan 1;7(1):e47–57.

2. Meedeniya D, Kumarasinghe H, Kolonne S, Fernando C, Díez ID la T, Marques G. Chest X-ray analysis empowered with deep learning: A systematic review. Appl Soft Comput. 2022 Jul 18;126:109319.

3. Chumbita M, Cillóniz C, Puerta-Alcalde P, Moreno-García E, Sanjuan G, Garcia-Pouton N, et al. Can Artificial Intelligence Improve the Management of Pneumonia. J Clin Med. 2020 Jan;9(1):248.

4. Sheu RK, Pardeshi MS, Pai KC, Chen LC, Wu CL, Chen WC. Interpretable Classification of Pneumonia Infection Using eXplainable AI (XAI-ICP). IEEE Access. 2023;11:28896–919.

5. Elemraid MA, Muller M, Spencer DA, Rushton SP, Gorton R, Thomas MF, et al. Accuracy of the Interpretation of Chest Radiographs for the Diagnosis of Paediatric Pneumonia. PLOS ONE. 2014 Aug 22;9(8):e106051.

6. Hopstaken RM, Witbraad T, Engelshoven JMA van, Dinant GJ. Inter-observer variation in the interpretation of chest radiographs for pneumonia in community-acquired lower respiratory tract infections. Clin Radiol. 2004 Aug 1;59(8):743–52.

7. Shoaip N, El-Sappagh S, Abuhmed T, Elmogy M. A dynamic fuzzy rule-based inference system using fuzzy inference with semantic reasoning. Sci Rep. 2024 Feb 21;14:4275.

8. Kwon T, Lee SP, Kim D, Jang J, Lee M, Kang SU, et al. Diagnostic performance of artificial intelligence model for pneumonia from chest radiography. PLoS ONE. 2021 Apr 15;16(4):e0249399.

9. Gefter WB, Post BA, Hatabu H. Commonly Missed Findings on Chest Radiographs: Causes and Consequences. CHEST. 2023 Mar 1;163(3):650–61.

10. Ben-Israel D, Jacobs WB, Casha S, Lang S, Ryu WHA, de Lotbiniere-Bassett M, et al. The impact of machine learning on patient care: A systematic review. Artif Intell Med. 2020 Mar 1;103:101785.

11. Buchlak QD, Esmaili N, Leveque JC, Farrokhi F, Bennett C, Piccardi M, et al. Machine learning applications to clinical decision support in neurosurgery: an artificial intelligence augmented systematic review. Neurosurg Rev. 2020 Oct 1;43(5):1235–53.

12. Buchlak QD, Esmaili N, Leveque JC, Bennett C, Farrokhi F, Piccardi M. Machine learning applications to neuroimaging for glioma detection and classification: An artificial intelligence augmented systematic review. J Clin Neurosci. 2021 Jul 1;89:177–98.

13. Ahmad HK, Milne MR, Buchlak QD, Ektas N, Sanderson G, Chamtie H, et al. Machine Learning Augmented Interpretation of Chest X-rays: A Systematic Review. Diagnostics. 2023 Feb 15;13(4):743.

14. Sharma H, Jain JS, Bansal P, Gupta S. Feature Extraction and Classification of Chest X-Ray Images Using CNN to Detect Pneumonia. In: 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence) [Internet]. 2020 [cited 2025 Jan 8]. p. 227–31. Available from: https://ieeexplore.ieee.org/document/9057809?utm\_source=chatgpt.com

15. Sujatha B, Koujalagi A, Harika A, Kumari VS. An In-Depth Convolution Neural Network for Chest X-Ray Image Assessment Using CXRIA-Net. In: Bhateja V, Tang J, Polkowski Z, Simic M, Chakravarthy VVSSS, editors. Information System Design: AI and ML Applications. Singapore: Springer Nature; 2024. p. 485–94.

16. Yamashita R, Nishio M, Do RKG, Togashi K. Convolutional neural networks: an overview and application in radiology. Insights Imaging. 2018 Aug;9(4):611–29.

17. Varshni D, Thakral K, Agarwal L, Nijhawan R, Mittal A. Pneumonia Detection Using CNN based Feature Extraction. In: 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT) [Internet]. 2019 [cited 2024 Nov 1]. p. 1–7. Available from: https://ieeexplore.ieee.org/document/8869364

18. Rahman T, Chowdhury MEH, Khandakar A, Islam KR, Islam KF, Mahbub ZB, et al. Transfer Learning with Deep Convolutional Neural Network (CNN) for Pneumonia Detection Using Chest X-ray. Appl Sci. 2020 Jan;10(9):3233.

19. Chutia U, Tewari AS, Singh JP, Raj VK. Classification of Lung Diseases Using an Attention-Based Modified DenseNet Model. J Imaging Inform Med. 2024 Aug 1;37(4):1625–41.

20. Sanghvi HA, Patel RH, Agarwal A, Gupta S, Sawhney V, Pandya AS. A deep learning approach for classification of COVID and pneumonia using DenseNet-201. Int J Imaging Syst Technol. 2023;33(1):18–38.

21. Kundu R, Das R, Geem ZW, Han GT, Sarkar R. Pneumonia detection in chest X-ray images using an ensemble of deep learning models. PLOS ONE. 2021 Sep 7;16(9):e0256630.

22. Jaiswal A, Gianchandani N, Singh D, Kumar V, Kaur M. Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning. J Biomol Struct Dyn. 2021 Sep;39(15):5682–9.

23. Rochmawanti O, Utaminingrum F. Chest X-Ray Image to Classify Lung diseases in Different Resolution Size using DenseNet-121 Architectures. In: Proceedings of the 6th International Conference on Sustainable Information Engineering and Technology [Internet]. New York, NY, USA: Association for Computing Machinery; 2021 [cited 2024 Nov 3]. p. 327–31. (SIET ’21). Available from: https://doi.org/10.1145/3479645.3479667

24. Anakha BM, Shaji G, Geetha S. Detecting COVID-19 from Chest X-Ray Images using Deep Learning. In: 2021 5th International Conference on Information Systems and Computer Networks (ISCON) [Internet]. 2021 [cited 2024 Nov 4]. p. 1–4. Available from: https://ieeexplore.ieee.org/document/9702491

25. Chutia U, Tewari AS, Singh JP, Raj VK. Classification of Lung Diseases Using an Attention-Based Modified DenseNet Model. J Imaging Inform Med. 2024 Aug 1;37(4):1625–41.

26. Shrimali S. Optimizing Chest X-ray Analysis for Pneumonia Detection Through Comparative Evaluation of Transfer Learning CNN Architectures. In: 2024 IEEE 9th International Conference for Convergence in Technology (I2CT) [Internet]. 2024 [cited 2024 Nov 5]. p. 1–5. Available from: https://ieeexplore.ieee.org/document/10543957

27. Tang J, Zhang B, Liu J, Dong Z, Zhou Y, Meng X, et al. Pneumonia Image Classification: Deep Learning and Machine Learning Fusion. In: 2024 7th International Conference on Artificial Intelligence and Big Data (ICAIBD) [Internet]. 2024 [cited 2024 Nov 5]. p. 440–7. Available from: https://ieeexplore.ieee.org/document/10604438

28. Xin KZ, Li D, Yi PH. Limited generalizability of deep learning algorithm for pediatric pneumonia classification on external data. Emerg Radiol. 2022 Feb;29(1):107–13.

29. Shi Y, Eberhart R, Chen Y. Implementation of evolutionary fuzzy systems. Trans Fuz Sys. 1999 Apr 1;7(2):109–19.

30. Konar A, Jain LC. An Introduction to Computational Intelligence Paradigms. In: Jain L, De Wilde P, editors. Practical Applications of Computational Intelligence Techniques [Internet]. Dordrecht: Springer Netherlands; 2001 [cited 2024 Nov 2]. p. 1–64. Available from: https://doi.org/10.1007/978-94-010-0678-1\_1

31. Senol C, Yildirim T. Fuzzy-neural networks for medical diagnosis. Int J Reason-Based Intell Syst. 2010 Jan 1;2:265–71.

32. (PDF) An Expert System to Diagnose Pneumonia Using Fuzzy Logic. ResearchGate [Internet]. 2024 Dec 11 [cited 2025 Mar 12]; Available from: https://www.researchgate.net/publication/333735595\_An\_Expert\_System\_to\_Diagnose\_Pneumonia\_Using\_Fuzzy\_Logic

33. B R, Rao SP, TM S, BC S, AP M. Detection of Pneumonia using Fuzzy Expert System. Int J Nov Res Dev [Internet]. 2024;9(4). Available from: https://www.ijnrd.org/papers/IJNRD2404776.pdf

34. Arani AL, Sadoughi F, Langarizadeh M. An Expert System to Diagnose Pneumonia Using Fuzzy Logic. ResearchGate. 2019 Jun;27:103.

35. Eldsouky M. chest x-ray images pneumonia dataset [Internet]. 2025 [cited 2025 Jan 13]. Available from: https://www.kaggle.com/datasets/moazeldsokyx/chest-x-ray-images-pneumonia-dataset/data

36. Sabour S, Frosst N, Hinton GE. Dynamic Routing Between Capsules. In: Advances in Neural Information Processing Systems [Internet]. Curran Associates, Inc.; 2017 [cited 2024 Nov 13]. Available from: https://proceedings.neurips.cc/paper\_files/paper/2017/hash/2cad8fa47bbef282badbb8de5374b894-Abstract.html

37. Newbold P, Carlson WL, Thorne BM. Statistics for business and economics [Internet]. Pearson; 2013 [cited 2024 Nov 18]. Available from: https://thuvienso.hoasen.edu.vn/handle/123456789/9470

38. Gardner ES. Exponential smoothing: The state of the art—Part II. Int J Forecast. 2006 Oct 1;22(4):637–66.

39. Bharati S, Podder P, Mondal MRH. Hybrid deep learning for detecting lung diseases from X-ray images. Inform Med Unlocked. 2020;20:100391.

40. Salam MA, Taha S, Ramadan M. COVID-19 detection using federated machine learning. PLoS ONE. 2021 Jun 8;16(6):e0252573.

41. Harris CR, Millman KJ, Van Der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. Nature. 2020 Sep 17;585(7825):357–62.

42. Caswell TA, Droettboom M, Lee A, Sales De Andrade E, Hunter J, Hoffmann T, et al. matplotlib/matplotlib: REL: v3.4.0. Zenodo [Internet]. 2021 Mar 26 [cited 2024 Nov 13]; Available from: https://ui.adsabs.harvard.edu/abs/2021zndo...4638398C

43. Prusty S, Patnaik S, Dash SK. SKCV: Stratified K-fold cross-validation on ML classifiers for predicting cervical cancer. Front Nanotechnol [Internet]. 2022 Aug 19 [cited 2025 Jan 14];4. Available from: https://www.frontiersin.org/journals/nanotechnology/articles/10.3389/fnano.2022.972421/full

44. Xu QS, Liang YZ. Monte Carlo cross validation. Chemom Intell Lab Syst. 2001 Apr 16;56(1):1–11.

45. Hesterberg T. Bootstrap. WIREs Comput Stat. 2011;3(6):497–526.

46. Inan O, Uzer MS. A Method of Classification Performance Improvement Via a Strategy of Clustering-Based Data Elimination Integrated with k-Fold Cross-Validation. Arab J Sci Eng. 2021 Feb 1;46(2):1199–212.