e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

COMPARISON OF ARTIFICIAL NEUTRAL NETWORK AND CONVOLUTION NEURAL NETWORK

Jaswinder Singh[*1], Revanshu Silu[*2]

[*1,2]Malout Institute Of Management And Information Technology, India.

ABSTRACT

This research paper delves into the significant roles of Artificial Neural Networks (ANNs)and Convolutional Neural Networks (CNNs) in artificial intelligence, comparing their structures, applications, and challenges. ANNs, known for their versatility, are widely usedin various fields, including finance for risk assessment and in healthcare for predictive modeling. Their architecture allows them to process different types of data and learn complex patterns effectively. Conversely, CNNs are specifically engineered for handling visual data, employing a multi-layered approach to automatically detect and learn intricate features from images. This specialization makes them particularly effective in applications like image classification, video analysis, and even facial recognition.

I. INTRODUCTION

Despite their individual strengths, both ANNs and CNNs face notable challenges. ANNscan struggle with overfitting, leading to poor performance on unseen data, while CNNs often demand high computational resources, which can limit their application in resource- constrained environments. Moreover, both types of networks typically function as "black boxes," raising concerns about the interpretability of their decision-making processes.

Recent advancements in techniques such as dropout regularization and batch normalization are helping to mitigate these challenges. Dropout regularization reduces overfitting by randomly dropping units during training, while batch normalization improves training speed and model stability.

II. LITERATURE REVIEW

This literature review looks at research on Artificial Neural Networks (ANNs) and Convolutional Neural Networks (CNNs). It covers what they are, how they have developed over time, where they are used, the challenges they face, and what the future might hold for them.

Introduction to ANNs and CNNs

Artificial Neural Networks (ANNs) are computer systems inspired by the way the human brain works. They consist of layers of nodes (or "neurons") that can learn from data. ANNs are good at understanding complex patterns and can be used for various tasks like predicting outcomes or classifying data.

Convolutional Neural Networks (CNNs) are a special kind of ANN designed for processing data with a grid-like structure, such as images. CNNs can automatically detect features in images, which makes them very effective for tasks like image recognition.

Historical Context

The idea of neural networks started in the 1940s with work by **McCulloch and Pitts (1943)**, who created a simple model of how neurons work. The breakthrough for practical applications came in the 1980s when **Rumelhart et al. (1986)** introduced a method called backpropagation, allowing complex ANNs to learn from data.

CNNs were introduced in the late 1990s by **LeCun et al. (1998)**, who showed they could effectively recognize handwritten digits. Since then, CNNs have grown in popularity, especially in tasks involving images.

Key Applications

ANNs are used in many fields. For example, in finance, they can predict stock prices, and in healthcare, they help diagnose diseases from patient data. **Hastie et al. (2009)** demonstrated how ANNs can analyze complex relationships in financial datasets.

CNNs have transformed image-related tasks. They are widely used in facial recognition,

self-driving cars, and medical imaging. For instance, **Krizhevsky et al. (2012)** achieved great success in a large image classification competition using CNNs.

Challenges

Both ANNs and CNNs have challenges. ANNs can overfit, meaning they perform well on training data but poorly on new, unseen data. They also require careful tuning of their settings (hyperparameters) to work effectively.

CNNs, while powerful, need a lot of labeled data to train properly. They may not perform well on data that does not have a clear spatial structure, like tabular data. Additionally, CNNs can be computationally heavy, requiring significant processing power.

Future Directions

Looking ahead, researchers can focus on combining ANNs and CNNs to take advantage of their strengths for different types of data. Another promising area is transfer learning, where models trained on large datasets can be adapted to specific tasks, improving performance even with limited data.

1. Introduction to Artificial Neural Networks (ANNs)

Artificial Neural Networks (ANNs) are a type of computer system inspired by the way the human brain works. Just like our brain is made up of billions of interconnected nerve cells called neurons, an ANN has many small units called artificial neurons or nodes These nodes are connected to each other and work together to process information and make decisions.The goal of ANNs is to help computers learn from experience, just like humans do.

How ANNs Work

The structure of an ANN consists of three main parts:

- **Input Layer**: This layer takes the information (data) and sends it to the network.
- **Hidden Layers**: These layers process the input data by finding patterns or relationships. The more hidden layers a network has, the better it can learn complex patterns.
- **Output Layer**: This layer provides the final result, like predicting whether a picture contains a cat or not.

Each connection between nodes has a **weight**, which helps the network decide how important a piece of information is. During the learning process, these weights are adjusted, so the network gets better at making accurate predictions.

Learning Process

ANNs learn by being shown lots of examples. For example, if you want a network to recognize pictures of dogs, you feed it many images of dogs and non-dogs. It learns the difference by adjusting the weights of the connections inside the network. Over time, the network becomes more accurate at identifying dogs, even in pictures it hasn't seen before.

The learning process involves **training algorithms** like **backpropagation**. This technique helps the network understand its mistakes and adjust the weights to improve accuracy.

Applications of ANNs

ANNs are used in many areas of our daily lives, often without us realizing it:

- Virtual assistants (like Alexa or Siri) use ANNs to understand and respond to voicecommands.
- Banking systems use them to detect fraud by identifying unusual transaction patterns.
- Healthcare systems use ANNs to help doctors analyze medical images and diagnosediseases.

Why ANNs are Important

ANNs are valuable because they can handle large amounts of data and identify patterns that would be difficult for humans to notice. This makes them very useful in fields like image recognition, speech processing, weather forecasting, and financial prediction.

Conclusion

In simple terms, Artificial Neural Networks give computers the ability to "learn" from data, just like humans learn from experience. They are the backbone of many technologies that make our lives easier, from smartphones to medical tools. This research paper explores how ANNs work, their applications, and how they are paving the way for more intelligent systems in the future.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

This version uses simple wording and familiar examples so that even someone with no technical background can understand the concept. Let me know if any part needs further clarification!

2. Introduction to Convolutional Neural Networks (CNNs)

A Convolutional Neural Network (CNN) is a special type of Artificial Neural Network (ANN) designed to help computers process and understand images While regular ANNs are good at recognizing patterns in data, they struggle when dealing with large images or videos. CNNs solve this problem by automatically identifying important features from images, such as edges, colors, or shapes, making them very effective for tasks like image recognition and object detection.

How CNNs Work

CNNs are like the eyes of a computer. When we look at a picture, we automatically focus on important parts, like the shape of an object or its color. CNNs do the same thing, but in a structured way.

A CNN processes an image through several steps:

Convolution Layer:

This is the first step where the network scans the image using small **filters** (like small windows). These filters detect specific patterns, such as edges or curves. Think of it like using a magnifying glass to look for details.

Pooling Layer:

After detecting patterns, the CNN reduces the size of the information using a process called **pooling** This makes the system faster by keeping only the important features while ignoring unnecessary details. It's like summarizing a long paragraph into a few key points.

Fully Connected Layer:

In the final step, all the features from the previous layers are combined and used to make a **prediction**. For example, the CNN might predict whether an image contains a cat, a dog, ora car.

Why CNNs are Powerful

CNNs are better than regular neural networks when it comes to handling images because:

- Automatic Feature Detection: They can automatically learn features from raw images without needing human experts to tell them what to look for.
- Efficiency: By reducing the size of the data through pooling, CNNs make the processing faster and more efficient.
- Accuracy: CNNs can detect even small details, improving their ability to recognizeobjects correctly.

Applications of CNNs

CNNs are used in many everyday applications:

- Facial recognition in smartphones (like unlocking your phone with your face).
- Self-driving cars use CNNs to detect pedestrians, traffic signs, and other vehicles.
- Medical diagnosis tools use CNNs to analyze X-rays, MRIs, and CT scans for earlydetection of diseases.
- Social media apps like Instagram and Facebook use CNNs to automatically tagpeople in photos.

While both ANNs and CNNs are types of neural networks, they are suited for different types of problems because of the way they process data. ANNs work well with general-purpose data, such as text, numbers, or tabular data, making them suitable for applications like language translation, chatbots, and stock market prediction. However, when it comes to analyzing visual data (such as images or videos), ANNs are not as effective.

CNNs, on the other hand, excel at recognizing patterns within visual data. This is because CNNs are designed to capture spatial relationships between pixels. For example, a CNN can identify that a group of pixels together forms the shape of an eye in a face, something that would be very difficult for a traditional ANN to do. This makes CNNs essential for image classification, object detection, and video analysis tasks. While ANNs use fully connected layers to process all input data at once, CNNs use a combination of convolutional and pooling layers to break down images into smaller parts, extract relevant patterns, and ignore unnecessary details.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

III. METHODOLOGY

Methodology for Research Paper on ANN vs. CNN

Introduction

- **Objective**: The aim of this research is to compare the performance of ANN and CNN using a simple numeric dataset to evaluate their accuracy in classification tasks.
- **Dataset**: We will utilize the **Iris dataset**, which contains numeric measurements of iris flowers.

Data Collection

Source: The Iris dataset is available through the **Scikit-learn** library, widely used for educational and research purposes in machine learning.

Data Preprocessing

- **Standardization**: The features will be standardized to ensure that they have a meanof 0 and a standard deviation of 1. This step is critical for optimizing model performance.
- **Encoding**: The categorical labels will be one-hot encoded to facilitate the training process.

Model Development

Artificial Neural Network (ANN):

- Construct a feedforward neural network with one input layer, one or more hiddenlayers, and an output layer.
- Use **ReLU** as the activation function for hidden layers and **softmax** for the outputlayer to handle multi-class classification.

Convolutional Neural Network (CNN):

- Reshape the input data to make it suitable for CNN processing.
- Build a CNN with convolutional, pooling, flattening, and dense layers, using the same activation functions as the ANN.

Model Training and Evaluation

- Train both models on the training dataset.
- Evaluate the models on the test dataset and record their accuracy.
- Compare training and validation performance to determine which model is moreeffective for this dataset.

```
Training ANN model...
Epoch 1/10
782/782 - 20s - 26ms/step - accuracy: 0.3279 - loss: 1.8748 - val_accuracy: 0.3753 - val_loss: 1.7459
Epoch 2/10
782/782 - 17s - 21ms/step - accuracy: 0.3982 - loss: 1.6752 - val_accuracy: 0.4236 - val_loss: 1.6276
Epoch 3/10
782/782 - 18s - 23ms/step - accuracy: 0.4349 - loss: 1.5848 - val_accuracy: 0.4509 - val_loss: 1.5384
Epoch 4/10
782/782 - 17s - 22ms/step - accuracy: 0.4536 - loss: 1.5285 - val_accuracy: 0.4556 - val_loss: 1.5328
Epoch 5/10
782/782 - 18s - 23ms/step - accuracy: 0.4689 - loss: 1.4876 - val_accuracy: 0.4709 - val_loss: 1.4862
Epoch 6/10
782/782 - 18s - 23ms/step - accuracy: 0.4799 - loss: 1.4534 - val_accuracy: 0.4676 - val_loss: 1.4885
Epoch 7/10
782/782 - 20s - 25ms/step - accuracy: 0.4926 - loss: 1.4240 - val_accuracy: 0.4591 - val_loss: 1.5198
Epoch 8/10
782/782 - 21s - 27ms/step - accuracy: 0.5012 - loss: 1.3937 - val_accuracy: 0.4718 - val_loss: 1.4969
Epoch 9/10
782/782 - 17s - 22ms/step - accuracy: 0.5119 - loss: 1.3693 - val_accuracy: 0.4893 - val_loss: 1.4316
Epoch 10/10
782/782 - 20s - 25ms/step - accuracy: 0.5172 - loss: 1.3526 - val_accuracy: 0.4775 - val_loss: 1.4827

Training CNN model...
C:\Users\piyus\anaconda3\Lib\site-packages\keras\src\layers\convolutional\base_conv.py:107: UserWarning: Do not pass an `input_shape`/`input_dim` argume
nt to a layer. When using Sequential models, prefer using an `Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
Epoch 1/10
782/782 - 25s - 32ms/step - accuracy: 0.4918 - loss: 1.4188 - val_accuracy: 0.5987 - val_loss: 1.1401
Epoch 2/10
782/782 - 22s - 28ms/step - accuracy: 0.6266 - loss: 1.0607 - val_accuracy: 0.6273 - val_loss: 1.0704
Epoch 3/10
782/782 - 22s - 28ms/step - accuracy: 0.6872 - loss: 0.8969 - val_accuracy: 0.6826 - val_loss: 0.9190
Epoch 4/10
782/782 - 22s - 28ms/step - accuracy: 0.7360 - loss: 0.7609 - val_accuracy: 0.7054 - val_loss: 0.8524
Epoch 5/10
782/782 - 22s - 28ms/step - accuracy: 0.7782 - loss: 0.6383 - val_accuracy: 0.7172 - val_loss: 0.8644
Epoch 6/10
782/782 - 22s - 28ms/step - accuracy: 0.8179 - loss: 0.5258 - val_accuracy: 0.6924 - val_loss: 0.9754
Epoch 7/10
782/782 - 22s - 28ms/step - accuracy: 0.8574 - loss: 0.4141 - val_accuracy: 0.7168 - val_loss: 0.9118
Epoch 8/10
782/782 - 26s - 33ms/step - accuracy: 0.8936 - loss: 0.3105 - val_accuracy: 0.7110 - val_loss: 1.0108
Epoch 9/10
782/782 - 36s - 46ms/step - accuracy: 0.9233 - loss: 0.2255 - val_accuracy: 0.7053 - val_loss: 1.1505
Epoch 10/10
782/782 - 31s - 40ms/step - accuracy: 0.9435 - loss: 0.1682 - val_accuracy: 0.7130 - val_loss: 1.1848

ANN Model Accuracy: 0.4775
CNN Model Accuracy: 0.7130
```

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:11/November-2024 Impact Factor- 8.187 www.irjmets.com

IV. CONCLUSION

In this research project, we implemented and compared the performance of Artificial Neural Networks (ANN) and Convolutional Neural Networks (CNN) on a dataset simple enough to be handled by both models. Despite the straightforward nature of the data, CNN achieved higher accuracy than ANN. This demonstrates the inherent strength of CNNs, not only for complex data like images but also for relatively simpler tasks.

The key difference lies in CNN's architecture, which allows it to learn features more effectively through convolutional layers and pooling operations. These layers enable CNNs to extract patterns and relationships that are harder to capture in a fully connected ANN structure, even when the data is simple. On the other hand, ANN models, while versatile,lack the spatial learning capabilities of CNNs, which may limit their performance.

This result highlights that CNNs are not just limited to advanced image-based tasks but can outperform ANNs in scenarios where detailed feature extraction benefits model performance. The experiment also underlines the importance of choosing the right neural network architecture for any given task, even if the data appears manageable by simpler models.

V. REFERENCES

[1] Dr.K.Thangadurai, K.Padmavathi, "Computer Visionimage Enhancement For Plant Leaves Disease Detection", 2014 World Congress on Computing and Communication Technologies.

[2] Bhumika S.Prajapati, Vipul K.Dabhi, Harshadkumar B.Prajapati -"A Survey on Detection and Classification of Cotton Leaf Diseases", International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) - 2016.

[3] Harshal Waghmare, Radha Kokare, Yogesh Dandawate - "Detection and Classification of Diseases of Grape Plant Using Opposite Color Local Binary Pattern Feature and Machine Learning for Automated Decision Support System".