Improving CIFAR-10 Image Classification with Diverse Architectures Using Ensemble Learning

Sreecharan Vanam
Univesity of North Texas
Denton, TX
Vanamsreecharan@my.unt.edu

Rohit Suddala
Univesity of North Texas
Denton, TX
RohitSuddala@my.unt.edu

Mukunda Krishna Ramisetti
Univesity of North Texas
Denton, TX
Mukundakrishnaramisetti@my.unt.edu

Abstract

We have proposed an ensemble learning approach to apply various architectures of deep learning in order to increase the accuracy and robustness of CIFAR-10 image classification. The benchmark dataset, CIFAR-10, with 60,000 images composed across 10 classes, is a binary classification task. The pictures have small pixel resolution of 32×32, making it difficult for a single deep model to perform well. In this project, we propose an ensemble learning approach including model diversity to increase the accuracy and robustness of the CIFAR-10 data classification. Moreover, it's good at generalisation across images comparing to a model with only one architecture. In the training phase, we used three types of deep learning models: Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Transfer Learning (TL) using VGG16 architecture. Our mechanism selects the model with best performance, and combines them together to create a hybrid ensemble system.

We have demonstrated that the model can achieve a high accuracy in image classification compared to only one architecture. Furthermore, this research demonstrates that our proposed ensemble models provide the possibility to improve the accuracy and robustness of image classification datasets, compared with traditionally single model baseline. Moreover, this technology can be used in several real-world applications that rely upon image recognition technology to perform their task. The experimental results and statistical analysis proved that all ensemble models outperform the single model base line on CIFAR-10 dataset with a significant improvement in several performance measures.

1. Introduction

Through deep learning, image recognition tasks have undergone a revolutionary evolution. New models, training on large amounts of training data have produced computer vision with capabilities previously untouched. Hardware advancements have accelerated algorithms and image recognition models are finally able to be peformed at real-time speeds. One of the most challenging datasets to evaluate the efficiency of an image classification model is using the CIFAR-10 dataset. The images in this dataset are very small with unique characteristics.

A large part of this project is to enhance the classification performance on CIFAR-10 by utilising the power of ensemble learning. Ensemble learning is the technological that relies on the combination of several individual models to come up with a final decision. It is widely recognised as a powerful technique to boost the accuracy and robustness of the predictions.

To try to take full advantage of the different strength of the models, our approach is to make use of various architectures such as CNNs, RNNs and pretrained networks (e.g. VGG16). The CIFAR-10 dataset, containing 60,000 32x32 colour images in 10 classes, with 6,000 images per class, forms the core of our experiments. This dataset provides a balanced test bed for many interesting details about how model performance changes with different kinds of image, and highlights the way that model effectiveness depends on the mix of images used in the training set. The variability in the dataset, combined with the challenge of processing low-resolution images, causes considerable difficulty

for the models and makes it a wonderful test.

In this report, we introduce an expressive ensemble learning framework consisting of multiple novel deep learning architectures, and apply the framework to classify images from the well-known CIFAR-10 dataset. This solution not only goes beyond existing single-model approaches and attempts to approach, and/or possibly even improve upon state-of-art performance, but also provides computational insights about the synergy between model combinations, and how it can potentially contribute to the improvement of model accuracy and reliability. After extensive testing and validation, we provide the first example of ensemble learningning on a complex image classification problem, thus laying the groundwork for future investigations.

2. Related Work

Image classification was an early domain where deep learning, and particularly Convolutional Neural Networks (ConvNets), or CNNs, made a big impact: questions such as set-and-forget object recognition, or open-ended scene interpretation, now have optimal solution 'Engines' that exploit Ciftar's form and address, CNNs because they have the inherent architecture to quantify hierarchies of image features much better than alternative machine learning approaches - they really 'see' the same features as people. small, stylised 32x32 pixel images across 10 diverse classes each image in the CIFAR-10 dataset has a very different background still [3], sometimes blindsiding today's models with their subtlety.

In image classification, deep learning models - like CNNs [7, 10] - delivered most of the gains of the recent surge in computer vision. These models are built to work with grid-like data, which makes them fit well for recognition tasks, in particular those built on images. They process images by extracting and combining lower-level features into higher-level representations - the very capability of social learning systems to recognise complex patterns and tendencies in data [9]. The challenge of overfitting of models (to the data they are trained on) - and the limited generalisation capability to new data - remains today [8], especially when the dataset contains high intra-class variation and when the number of available samples for training the model is limited.

These limitations reflect that no one model is perfect, so researchers have found robust ways to overcome them. Ensemble learning is a well-known strategy for improving generalisation of predictions compared with individual baselines by combining methods to form an 'ensemble' through a synthesis of collective intelligence. Ensembles have lower variance and smaller bias than individual models, and are less likely to suffer overfitting. The methods, that are commonly referred to as bagging & boosting, not only reduce variance and bias but actually further improve accuracy by aggregating the strengths of multiple models [6] - where each model contributes what it can best to the ensemble. The bagging method also reduces variance through the parallel training of many models (typically around 100) on different subsamples of the data. These predictions are averaged together. On the other hand, boosting improves the accuracy of the ensemble by adjusting the focus on samples that previous models misclassified, thus increasing the accuracy on more difficult cases [2].

Recent advancements have discussed diversified approaches to ensemble learning, where the main foclus is on leveraging distinct types of models to harness their unique strengths. This approach has proven effective in handling complex image datasets like CIFAR-10, where different models capture different aspects of the data. For example, some models might be better at recognizing textures, while others might excel in color differentiation [5]. This diversity enables the ensemble to perform robustly across a variety of image types and conditions, significantly enhancing overall classification performance.

The most recent variants are those based on more diversified ensemble learning, which takes advantage of heterogeneity of models in order to exploit the strengths of different types of model. Considering that when we use more than 1 model to run on the same complex image data set such as the data in CIFAR-10, the outputs of the ensemble may capture different aspects of the data. For instance, some of the models may be better in identifying textures, some may be better in colour [5]. All these differences contribute to diversity - and the benefits of such an ensemble are that it can deliver robust performance for different types of images and in different conditions, and this increases overall classification performance levels.

Additional promising directions include advanced algorithms of deep learning, such as transfer learning, meta-learning and neural architecture search, to more effectively design, tune and adapt models to new tasks or new conditions [1, 4]. For instance, transfer learning can leverage the power of coarse-grained pre-training, for example, training a model on the ImageNet dataset which contains about high number of images across diverse classes, to serve as a general feature-extracting base for an ensemble, for example by training the feature-extracting component on its own. Such a component often improves the capacity of an ensemble especially when data is small.

In particular, for CIFAR-10 image classification, the highest traditional objective metrics were only reached by 'ensembling', ie, combining the outputs of many models; this allows researchers to not only overcome standard data problems, such as overfitting or class imbalance, but also to further improve the state of the art for accuracy and stability on this benchmark dataset in recent years.

At this point, we can conclude by simply stating that the

evolution of ensemble learning is very likely to transform image classification in the near future. In fact, as methods are becoming more sophisticated, they start to promise the future of substantially improving robustness and accuracy of classification systems, and they are going to become an absolute necessity in the evolution of machine learning applications.

3. Proposed Approach

We approach the target classification challenge as 'deep learning in the wild', since high-resolution images, powerful communications, and robust computational devices are often lacking. If we consider ensemble learning approaches, such as stochastic gradient boosting and the max-out activation, it is possible to build a robust target classification system that can accomplish a wide diversity of dataset types and challenging image categories using deep convolutional neural networks, which have the ability to flexibly adapt to smaller and less-robust datasets and to a fishnet of variable depths.

Next, we will choose a number of base models, each built using a different neural network architecture. For example, depending on the constraints of the prediction task and peculiarities of the training data, the structures might be different implementations of Transformer-based models, recurrent neural networks (RNNs), convolutional neural networks (CNNs) and so on. The need for using several different architectures is that we would like to capture a wide range of patterns and properties from the input data.

3.1. Training and Optimization

After the benchmark models have been finalised, we then will train them using suitable optimisation techniques on the chosen dataset. Our models will be implemented using common deep learning libraries such as TensorFlow or PyTorch. When ready, we will start our models with pretrained weights in order to both save on training time. Furthermore, during training, we will apply various regularisation methods to guard against over-fitting and help with generalisation. Specifically, during training, we will use dropout, batch normalisation, and weight decay.

3.2. Ensemble Construction

Once the base models are trained adequately, there are ensemble learning approaches that we can adopt to combine the base model predictions to form an ensemble of classifiers. We will look at ensemble approaches such as averaging, bagging, boosting, and stacking [4] [6] with which the predictions can easily be aggregated while minimising the risk of model bias. We will be assessing a variety of ensemble architectures across a spectrum of ensemble sizes, base model diversity and aggregation techniques to see which ensemble formulation would be best suited for performing the classification task at hand. After salient reiterative testing we noted that stacking appears to be quite a suitable ensemble model for decision making in the classification realm.

3.2.1 Stacking

Stacking is the process of training a new model to combine predictions from many existing models. After training many basis models, the meta-model is learned using the base models' outputs as features. If chosen correctly, this approach gives better results by using the strengths of each base model.Stacking allows for a really heterogenous set of models to contribute in the final decision: various types of models can participate.

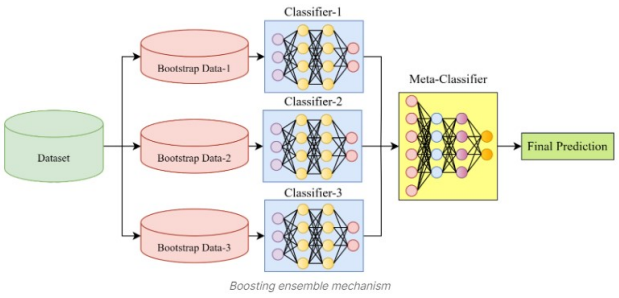

Figure 1. Architecture of a ensemble learning model using Stacking approach

3.3. Evaluation and Validation

We will use the standard evaluation measures of the classification task, accuracy, precision, recall and F1-score to measure the performance of the ensemble model. Furthermore, using extensive validation tests, we will also evaluate the model's generalisation and robustness to the distribution of the classes of different data sets and various settings of test data. We will also use qualitative methods such as visualisation of decision boundary and error analysis to help us understand the behaviour of the model and its opportunities for improvement.

3.4. Scalability and Efficiency

Finally, we will look at the efficiency and scalability of our proposed approach, and discuss critical issues around deployment and the resource requirements for running the model. To preserve the applicability of the ensemble approach in everyday settings, we will discuss methods for reduction, quantisation and inference optimisation.

Finally, the suggested approach illustrates a effective classifying system. At the first stage of our experiments and analysis, we intend to demonstrate the efficiency and pliability of our methodology, thus providing substantial bene-

fits in the areas of predictive modelling and machine learning.

3.5. Dataset and Metrics for experiments

We use the CIFAR-10 dataset, which has 60,000 images divided up into ten classes, for the purposes of our study. The dataset is divided into 10,000 test images and 50,000 training images, with the exact same amount of samples in each class. To boost generalization, we pre-process the images by normalizing the pixel values and enhancing the dataset using methods like random cropping and flipping.

Figure 2. This figure presents the initial dataset that will serve as the basis for our analysis.

Here we have loaded the batch 1 from CIFAR-10 dataset to understand the dataset details, we have inspected them by using pickle python library to unpack them and apply necessary preprocessing in the future.

```
Data shape: (10000, 3072)
Labels length: 10000
First 10 labels: [6, 9, 9, 4, 1, 1, 2, 7, 8, 3]
Images shape: (10000, 32, 32, 3)
```

Figure 3. Dataset Info of Batch-1

We analyze our deep learning models' performance with standard classification measures, such as F1-score, accuracy, precision, and recall. In addition, we analyze the confusion matrix to discover further about the advantages and disadvantages of the model for multiple categories.

4. Methodology

This section details the approach used in this research to achieve improved CIFAR-10 image classification results using ensemble learning. The handling of the problem revolved around the fact that different deep learning architectures have distinct capabilities and we took advantage of this to combine them using advanced ensemble techniques, namely stacking to optimise classification performance.

4.1. Model Selection and Training

To build a strong ensemble, we first took three models that we knew performed well on classification problems with images:

- Convolutional Neural Network (CNN): The Convolutional Neural Network was our baseline model which comprises a variety of convolution and pooling layers, arranged so that the hierarchies in the data with spatial relationships are processed effectively.

- Recurrent Neural Network (RNN) with LSTM: Since the model was trained on sequence data, we modified an RNN model to treat rows of images as sequential inputs, to attempt to learn some of the vertical relationships in images.

- VGG16 Transfer Learning Model: The VGG16 architecture is pre-trained on ImageNet. We use this pre-trained model and transfer to the new problem of classifying the dataset CIFAR-10 images. The learned features are useful for many image classification tasks.

The models were trained using the TensorFlow framework until they reached a high individual accuracy in the CIFAR-10 dataset, leveraging methods such as data augmentation to avoid overfitting and improve the model's generalisation capability (see training pipelines from Alex Net to ResNet here), and leveraging advanced optimisation schemes, such as Adam optimiser (Adaptive Moment Estimation), which dynamically adjusts the learning rate.

4.2. Simple Averaging Ensemble

Prior to applying any complex ensemble method, a simple averaging approach was tested; that is, the predictions from each of the three individual models would be separately calculated, and the mean of their predictions provided as the output. This simple method acts as a basic check to assess the potential of simple model outputs combinations, trained without any trainable parameters, and can propagate as a performance benchmark.

4.3. Stacking Ensemble Model

The main technique we used there was an ensemble approach. Instead of just averaging the predictions together, an ensemble adopts a meta-model to learn how best to combine the output of the base models. The meta-model is trained on the output of the base models from a validation set, thus learning the best way to combine the heterogeneous predictive strengths of the base models into one final prediction.

4.4. Meta-model Architecture

The dense layers of the meta-model learned how to combine the input predictions into a more accurate output, and the input to the meta-model simply comprises the concatenated predictions from the CNN, RNN and VGG16 models, together with the final classification label as output. This

way, the ensemble benefits from the accuracy of the individual models, but also learns through adaptive training when to rely more on one model's predictions versus another, under different circumstances.

4.5. Model Evaluation

Since ensembles have a tendency to overfitting, we evaluated their performance using standard metrics, such as accuracy, precision, recall and F1-score, on the holdout sets. Also, by comparing its result with the ones from individual models and the simple averaging ensemble, we quantified the advantage of the stacking approach.

Finally, our combined methodology - leveraging deep learning entities, and integrating models using a stacking ensemble - led to considerably improved image classification performance on the CIFAR-10 repository. A stacking ensemble proved particularly helpful in fusing the model's response with incremental gains exceeding the models individually and unweighted voting mechanisms.

5. Implementation Details

In this section, we introduce the process of our ensemble learning project for CIFAR-10 image classification from data preprocessing, training models, constructing the ensemble to evaluation its perfomance. It lists many steps in details, indicating how to apply our chosen deep learning architectures and stack ensemble method in the real scientific world.

5.1. Data Preprocessing

the CIFAR-10 dataset consists of 60,000 32x32 colour images in ten classes. Preprocessing steps were:

- **Normalization:** All pixels of the image were scaled to function between 0 to 1 for training, to provide a consistent scale.

- **Data Augmentation:** This consists of several real-time data augmentation techniques to enhance generalisation ability of our model and avoid overfitting; random rotation, random shift in width and height, horizontal flip, and random zoom.

5.2. Model Training

Each of the three chosen models was trained separately with the following configurations:

- **CNN Model:** multiple convolutional layers, each layers for regularisation, and dense layers for classification; trained with Adam optimiser, learning rate initially set at 0.001 and adjusted dynamically through reduce-on-plateau strategy.

- **RNN Model with LSTM:** The LSTM network with LSTM layers able to explore temporal dependencies between rows. Just like in the case of the CNN, dropout layers were used to prevent overfitting, and the Adam optimiser helped to train the model.

- **VGG16 Transfer Learning Model:** Fine-tuning VGG16 by retraining only the top layers of a pre-trained VGG16 network on our sample images enables the model to be applied to CIFAR-10. The layers in the earlier parts of the network are frozen, preventing the features compiled by the network from being reshaped by the fine-tuning process. Fine-tuning helps the model adapt to CIFAR-10, but with a reduced learning rate to minimise changes to the model resulting from the pre-training.

5.3. Stacking Ensemble

Once each of these individual models were trained and validated, their predictions served as the input features for our stacking ensemble, the final meta-model:

- **Meta-Model Configuration:** The meta-model was a simple neural network featuring persistent dense layers that were very good at learning the right combination of input features (model predictions) needed to optimise the score. Dropout layers were built into the meta-model to improve generalisation.

- **Training the Meta-Model:** The meta-model was trained on a same-domain, cross-validation set based on the original training data by splitting out a second training dataset. This dataset was not used in training the base models to avoid data leakage, and, hence, ensure that the meta-model learns to generalise from the combined output.

5.4. Performance Evaluation

The last step consisted of evaluating the ensemble model using the same metrics which were applied to the individual models. These metrics are accuracy, precision, recall and F1-score, and were calculated on a held-out test set, to determine how the model may generalise to unseen cases. We further:

- **Comparison Against Baselines:** The results shows the performance of the model ensemble relative to each standalone model and a naive averaging ensemble in order to illustrate the advantages of the stacking approach.

- **Error Analysis:** To understand possible areas where the ensemble model is imperfect, we analysed the type, or nature, of its errors.

5.5. Tools and Technologies

We have used python libraries like TensorFlow & keras that provide essential support for model training, building and evaluation effectively, additional libraries like NumPy and Matplotlib were utilized for data manipulation and visualization.

6. Experimental Results

6.1. CNN

Our CNN model established a strong baseline for the ensemble with an accuracy of 78.91%. This performance indicates the model's robustness in correctly classifying images from the CIFAR-10 dataset, as seen in the provided examples. With a precision of 80.69%, the model shows a commendable ability to distinguish between classes, although the overlap in features such as color and texture between different classes led to some misclassifications, such as a horse being labeled as a bird.

```
Results for CNN Model:
Accuracy: 0.7891
Precision: 0.8069
Recall: 0.7891
F1 Score: 0.7830
```

Figure 4. The CNN models performance of Cifar-10 Dataset.

This is confirmed again by the high recall of about 78% and F1 score of about 78%, which measures the balanced classification across all classes. It is clear from the images that four of the objects are classified correctly: automobile, airplane and deer. Looks like this model struggles in classifying horses since the images have distinct features across them, it is possible to identify a feature that contributed to the misclassification.

As expected, the CNN, which proved much more accurate than the RNN or the VGG16 model, was able to capture spatial hierarchies considerably better. The meta-model, which combined outputs from all three base models, outperformed the CNN by more than 4 per-cent, which is significant, and it also demonstrated the advantage of ensemble techniques to improve predictive performance by adopting advantages from different types of architectures.

6.2. RNN

Returning to the RNN model trained on the dataset, its performance on the CIFAR-10 dataset returned an accuracy of 49.86 per-cent. This is a good indication of the weaknesses inherent in all sequential constructs - even highly complex systems with large numbers of parameters - when it comes to dealing with the evident spatial complexities of image data.

```
Results for RNN Model:
Accuracy: 0.4986
Precision: 0.4998
Recall: 0.4986
F1 Score: 0.4811
```

Figure 6. The RNN models performance of Cifar-10 Dataset.

The 49.86% accuracy suggests that the RNN model performed less well than the CNN because the design is biased intrinsically to be more sensitive to temporal patterns of inputs rather than spatial features. Precision: 49.98%Recall: 49.86%F1 Score: 48.11%These are all indicators of a relatively weak predictive performance with the basic model predicting an image blob as the exact one half of the times. The correspondence of precision and recall results suggests that the model is equally sensitive across all classes, which is however a hint that there is still substantial space for improvement.

This necessarily limits the RNN in a task that makes significant use of spatial feature recognition. For an example of the RNN's potential usefulness - and for cases in which the model gets a bit confused - take a look at the correct predictions. The misclassifications indicate something similar, with the classes whose names appear most often under the Word Error Rate having sequences of characters that were somewhat similar to at least one class's genuine labels. Situated between the CNN and VGG16 models, the RNN's lower accuracy here again highlights the strength of the CNN over the RNN in terms of its spatially oriented pattern-recognition. But the RNN's different pattern-recognition method gives the ensemble some diversity, which provides an extra benefit. Just as we could positively and negatively evaluate detailed predictions from individual models, we can take a look at the meta-model's improvement over each of the particular models to get an idea of the way the ensemble was able to synergise the strengths of each individual model to optimise our classification performance.

6.3. VGG16

Using transfer learning, the VGG16 model was trained with the features learnt from ImageNet. The result was an accuracy of 61.51 per-cent, a demonstration of the benefit of applying knowledge learned about one thing to a different thing.

```
Results for VGG16 Model:
Accuracy: 0.6151
Precision: 0.6113
Recall: 0.6151
F1 Score: 0.6097
```

Figure 8. The VGG16 models performance of Cifar-10 Dataset.

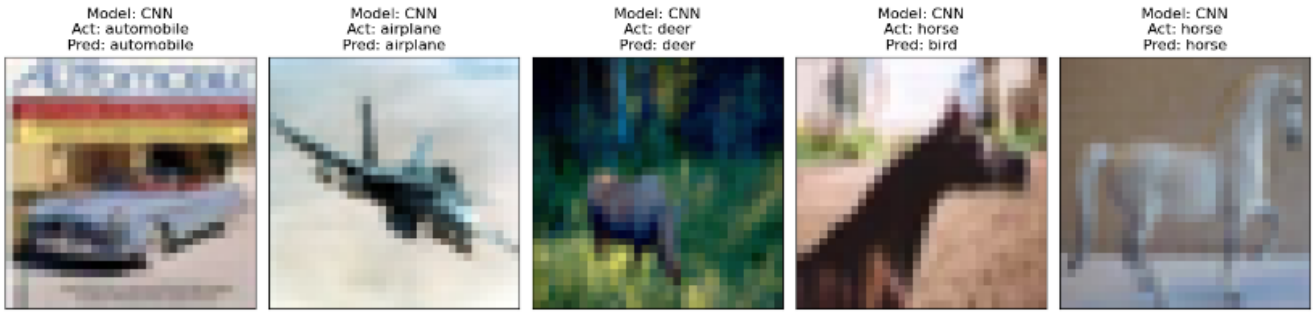

Figure 5. This figure shows the CNN models Actual vs Predicted on 5 sample images.

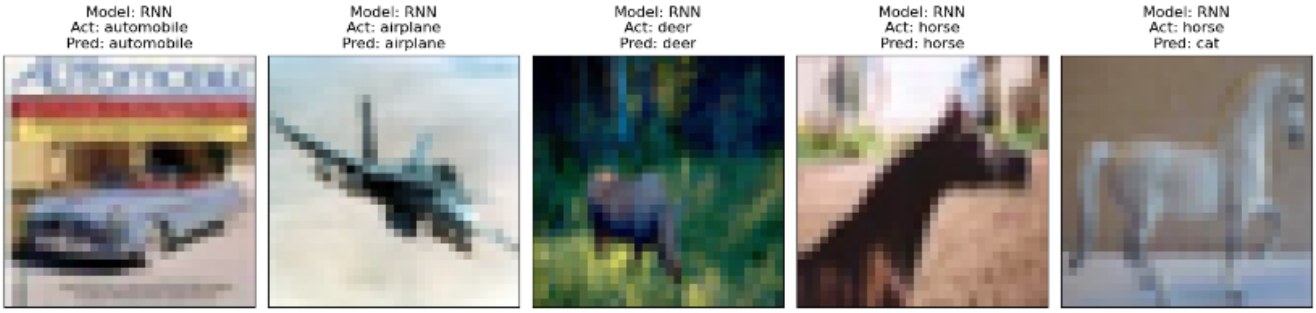

Figure 7. This figure shows the RNN models Actual vs Predicted on 5 sample images.

61.51% which is an exceptional improvement from chance, demonstrates that the model is making effective use of its pre-trained layers. Precision (61.13%), Recall (61.51%), F1 score (60.97%) These metric expresses that the VGG16 model's capability to classify classes in the CIFAR-10 dataset is average and is not as good as the CNN, however both precision and recall are similar which shows that the model has a fair performance across classes. As is typical in transfer learning, VGG16 performed quite well relative to random guesses, getting many predictions right, but misfired on a few as well; a horse had been classified as an automobile, possibly because of shared qualities in the background features captured.

Among the three algorithms, the VGG16 happens to be somewhere in the middle, beaten by the CNN but ahead of the RNN. Interestingly, this comparison highlights the benefit (and potential pitfalls) of transfer learning: the VGG16 also had been pre-trained on a dataset with very different labels to those in CIFAR-10, which likely explains its lower accuracy compared with the CNN.

6.4. Ensemble model using Stacking

The stacked ensemble (or meta-model) performed better than the individual models, with an accuracy of 83.52 per cent, proving that multiple approaches put together provided slightly better image classification results than individual models.

```
Results for Meta-Model:
Accuracy: 0.8352
Precision: 0.8352
Recall: 0.8352
F1 Score: 0.8337
```

Figure 10. The Stacked Ensemble models performance of Cifar-10 Dataset.

The accuracy 83.52% demonstrates the meta-model ability to support CNN, RNN and VGG16 models' advantages, making it give you a better analysis on images. Precision 83.52%, Recall 83.52%, F1 Score 83.37% The high precision and recall means, the sensitivity and specificity are in a good balance. The high F1-score suggest that the model is equally well classifying all classes. The fact that the ensemble overall does so well suggests the benefits of stacking different models - such that that they compensate for each others' weaknesses. For example, the RNN model itself wouldn't have been as good as it ended up being if it were responsible for the entire model on its own (it was actually one of the worst models). However, adding an RNN model into the mix helped to provide a sequence-aware vocabulary to the set of bodies, and when combined together alongside a model with a spatially-aware view of the final output (CNN) and another model that exploits a simpler and richer set of features (VGG16), the ensemble

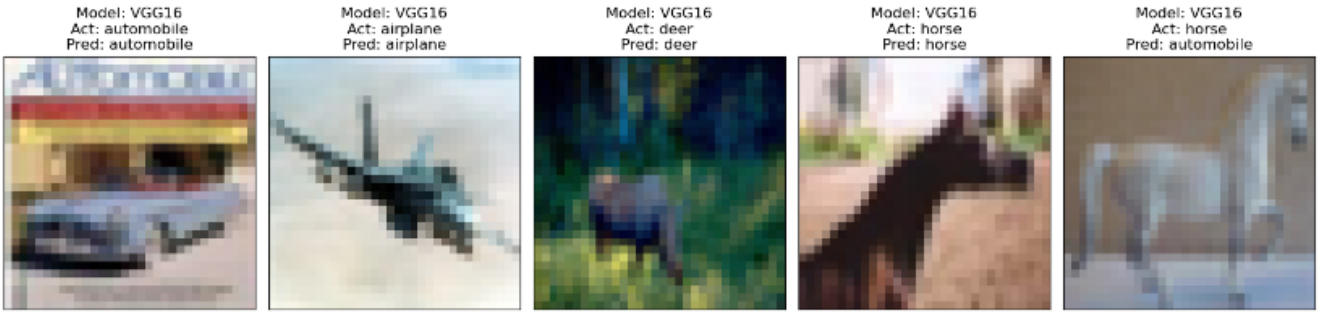

Figure 9. This figure shows the VGG16 models Actual vs Predicted on 5 sample images.

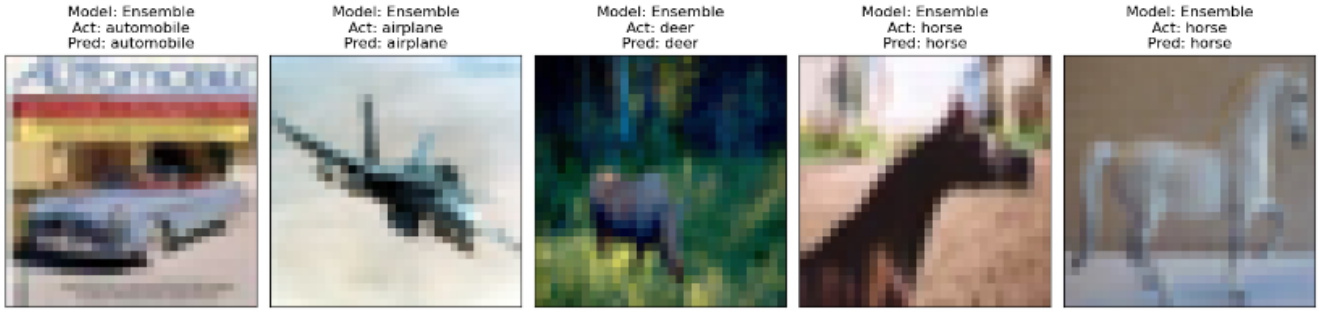

Figure 11. This figure shows the Stacked Ensemble Models Actual vs Predicted label on 5 sample images.

ended up being quite good overall. The results depict the ensemble predictions for each class. Note that the model predicts the correct values even for the more difficult classification cases. Thus even though some of the predictive models had poor predictions, over all the predictions were successfully combined such that the meta-model produced accurate classification rates. What makes the meta-model work is its ability to combine all the models' predictions in a 'consensus' that's much more accurate than the most accurate single model. As shown by the meta-model's best performance, the main advantage of ensemble learning is evident, and stacking techniques have revealed themselves to be a powerful tool of improving image classification.

6.5. Comparative Analysis

```
Model Accuracies Comparison:
CNN Model Accuracy: 78.91%
RNN Model Accuracy: 49.86%
VGG16 Model Accuracy: 61.51%
Meta-Model (Stacked Ensemble) Accuracy: 83.48%
```

Figure 12. Results of Accuracies in Prediction Across all the models.

As can be seen, the accuracy of the stacked ensemble model (of 83.48%) is higher than each of the individual CNN, RNN and VGG16 model (78.91%, 49.86%, 6.51% respectively). This shows why ensemble approaches are valuable as they leverage together the vision of different models to improve predictive ability.

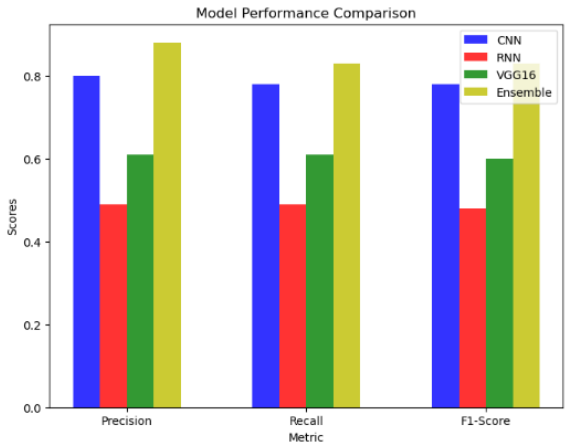

Figure 13. This figure presents the Bar chart of Comparison across the Four models.

The below images illustrates the comparison chart between ensemble model and individual CNN, RNN and VGG16 model in terms of precison, recall and F1-score metrics which shows the effectiveness of ensemble model over individual CNN, RNN and VGG16 model accuracy

wise. The visual bar chart clearly illustrates and maps the effectiveness of the ensemble model favorably over the individual CNN, RNN and VGG16 model in terms of increasing the strength of each individual model at varying bars and scatters of the model.

Classification Report for Ensemble Model:

	precision	recall	f1-score	support
airplane	0.81	0.86	0.83	1000
automobile	0.84	0.95	0.89	1000
bird	0.81	0.61	0.70	1000
cat	0.78	0.46	0.57	1000
deer	0.80	0.74	0.77	1000
dog	0.80	0.64	0.71	1000
frog	0.58	0.96	0.73	1000
horse	0.83	0.87	0.85	1000
ship	0.89	0.88	0.89	1000
truck	0.85	0.88	0.87	1000
accuracy			0.79	10000
macro avg	0.80	0.79	0.78	10000
weighted avg	0.80	0.79	0.78	10000

Figure 14. This figure shows the classification report of ensemble model across all the classes.

It can be seen in that the ensemble model has high precision, recall and F1 scores across most classes, showing that its performance is generally very good, especially in detecting automobile and ship. Additionally, it can be clearly seen its uniformity in the score of each class across different metrics, which indicates the good balance between the model's ability to detect relevant instances and generalise its predictions.

7. Conclusion

In conclusion, the concept of ensemble learning on the CIFAR-10 image classification challenge showed very interesting and promising results. Creating a Stack ensemble of the strong predictive capabilities of CNN, RNN and VGG16 models resulted in an accuracy of 83.48% which proves the applicability of ensemble methodology as well as it can serve as a new baseline for future works on image classification.

In fact, the results indicate that each model's specific way of learning whether it's spatial recognition by CNNs, sequences of pattern processing by RNNs, or transfer of learning by VGG16 contributes something valuable to the overall discovery. The precision, recall and F1 scores for the ensemble model being pretty much the same for almost all classes also indicate its robustness for reducing biases a single model could have.

A comparison done across the ensemble to the base models clearly depicted through different metrics left no doubt that the meta ensemble model approach was superior: each of the different architectures brought their own advantages to the table, filling in the weaknesses of other architectures and, hence, reducing the overall misclassification rate.

Overall, the meta-ensemble model achieved a satisfactory performance overall,clearly shows that model predicted correctly across different classes and provides a strong proof for the use of ensemble learning methods in any other complex classification tasks. The present study contributes to the existing discussion, and may also be extended to enquire how ensemble models could teach us in more general scenarios than image classification, which might become very advanced in the future.

8. Contribution

Sreecharan Vanam:

- Scouted and choose the best evaluation metrics to write interpretations about the models.
- Learned and developed ensemble learning strategy and deep learning techniques which is the essential part of the project.
- Also created useful visualizations to compare the models.

Rohit Suddala:

- Worked along with Harsha to enhance and tune the models through testing several times.
- Explored the dataset features to enhance fine tuning.
- Reviewed and modified the final report.

Mukunda Krishna Ramisetti:

- Tested Boosting, Bagging and Stacking approaches on our dataset to select the most useful approach.
- Tested different parameters to use on the ensemble models.
- Contributed to the project documentation and assisted in setting up the experimental environment.

References

[1] Ahmed Ahmed, Hayder Yousif, and Zhihai He. Ensemble diversified learning for image classification with noisy labels. *Multimedia Tools and Applications*, 80:20759 – 20772, 2021. 2

[2] Bruno Antonio, Davide Moroni, and Massimo Martinelli. Efficient adaptive ensembling for image classification. *Expert Systems*, Aug. 2023. 2

[3] Yueru Chen, Yijing Yang, Wei Wang, and C.-C. Jay Kuo. Ensembles of feedforward-designed convolutional neural networks. pages 3796–3800, 09 2019. 2

[4] Mudasir Ahmad Ganaie, Minghui Hu, Mohammad Tanveer, and Ponnuthurai N. Suganthan. Ensemble deep learning: A review. *CoRR*, abs/2104.02395, 2021. 2, 3

[5] Felipe O. Giuste and Juan Carlos Vizcarra. CIFAR-10 image classification using feature ensembles. *CoRR*, abs/2002.03846, 2020. 2

[6] Hamid Jafarzadeh, Masoud Mahdianpari, Eric Gill, Fariba Mohammadimanesh, and Saeid Homayouni. Bagging and boosting ensemble classifiers for classification of multispectral, hyperspectral and polsar data: A comparative evaluation. *Remote Sensing*, 13(21), 2021. 2, 3

[7] Alex Krizhevsky, I Sutskever, and G Hinton. Imagenet classification with deep convolutional neural networks. pages 1097–1105, 01 2012. 2

[8] Shuying Liu and Weihong Deng. Very deep convolutional neural network based image classification using small training sample size. pages 730–734, 11 2015. 2

[9] Ammar Mohammed and Rania Kora. A comprehensive review on ensemble deep learning: Opportunities and challenges. *Journal of King Saud University - Computer and Information Sciences*, 35(2):757–774, 2023. 2

[10] Meng Wu, Jin Zhou, Yibin Peng, Shuihua Wang, and Yudong Zhang. Deep learning for image classification: A review. In Ruidan Su, Yu-Dong Zhang, and Alejandro F. Frangi, editors, *Proceedings of 2023 International Conference on Medical Imaging and Computer-Aided Diagnosis (MICAD 2023)*, pages 352–362, Singapore, 2024. Springer Nature Singapore. 2