DOI : https://doi.org/10.56452/6-3-664

ISSN: 0974-5823 Vol. 6 No. 3 December, 2021

International Journal of Mechanical Engineering

Exploring the Transfer Learning: A Comparative Study of Pre-Trained Models and Fine-tuning Techniques on image dataset

Nidhi Mehra

Asst. Professor, School of Computing, Graphic Era Hill University,

Dehradun, Uttarakhand India 248002

ABSTRACT

In the modern world, RAM for computers is affordable and widely accessible. For a small investment or rent, we can get the hundreds of GBs of RAM we need to run a super complex supervised machine learning problem. However, getting access to GPUs is not free. We need access to GPUs with 100 GB of VRAM, but getting there won't be easy and will cost a lot of money. Now, in the future, that might change. However, for the time being, it means that we must use our resources more wisely when tackling Deep Learning issues. Particularly when we attempt to address challenging real-world issues in fields like speech and picture recognition. Once your model has a few hidden layers, adding more would require an enormous amount of resources. When the high-level attributes that the lower layers have learnt are insufficient to distinguish the classes in your issue, transfer learning will not be effective. For instance, a pre-trained model could be excellent at recognizing doors but not at determining whether they are open or closed. Instead of high-level features in this situation, utilize the low-level features from the pre-trained network. In this situation, you will either need to leverage features from older levels or retrain additional model layers. Features transfer poorly across datasets that are not comparable. In further depth, this research examines dataset similarity. Nevertheless, the paper demonstrates that using pre-trained weights as the network's initialization improves performance over using random weights. In such a pre-trained event, transfer learning is unlikely to be effective. Because removing layers decreases the number of trainable parameters, overfitting may occur. Furthermore, it takes a lot of time and effort to figure out how many layers to remove without overfitting.

1. INTRODUCTION

A model developed for one job can be used as the basis for another task that is similar using the machine learning approach known as transfer learning. Transfer learning has been effectively used in the context of computer vision for a variety of picture classification, object recognition, and semantic segmentation tasks. There are various well-liked options for pre-trained models for transfer learning, including VGG, Reset, Inception, and Efficient Net. These models are fine-tuned on smaller datasets for particular purposes after being pre-trained on big datasets like ImageNet. The size and variety of the dataset, the difficulty of the job, and the available computing resources should all be taken into account when comparing pre-trained models versus fine-tuning strategies on an image dataset. Starting with a pre-trained

DOI : https://doi.org/10.56452/6-3-664

model and fine-tuning it using the target dataset is one method. The pre-trained model is fine-tuned by training it on the fresh dataset while maintaining the first layer's pre-trained weights and changing the weights of the deeper layers. Another strategy is to build a classifier on top of the extracted features using a pre-trained model as a feature extractor. When the target dataset is compact and comparable to the pre-training dataset, this approach is advantageous.

The particular job and dataset will determine the pre-trained model and fine-tuning method to use. Smaller datasets benefit from using simpler models and feature extraction approaches, but more complicated tasks and bigger datasets often call for deeper and more potent pre-trained models. It is crucial to conduct in-depth experiments and assess how various pre-trained models and fine-tuning methods perform on the target dataset. In order to do this, measures like accuracy, precision, recall, and F1 score are measured and the results are compared to the most advanced methods for the same job.

A machine learning approach called fine-tuning is used to raise a trained model's performance on a given job. Here are some typical methods for fine-tuning:

1. Learning Rate Schedule: Changing the learning rate while a model is being trained will assist and avoid overfitting and increase model accuracy.

2. Data Augmentation: Rotations, flips, and zooms are examples of modifications that can be added to the training data to help the model generalize more effectively to new data.

3. Dropout: To avoid overfitting, this regularization strategy randomly removes certain neurons during training.

4. Weight Decay: Another regularization method that penalizes the loss function to induce the model to use lower weights is weight decay.

5. Early Stopping: This strategy halts the training procedure as soon as the model begins to overfit the validation set.

6. Transfer Learning: Improving a pre-trained model's performance on a similar task will have a major impact on how well it performs on a new task.

7. Gradient Clipping: To avoid gradients that explode during training, gradient clipping is a method that restricts the gradients' amplitude.

8. Batch Normalization: To assist the model converge more quickly and avoid overfitting, batch normalization normalizes the inputs to each layer.

These methods can be utilized with a variety of models and tasks and are often employed in deep learning. A machine learning model that has been pre-trained on a significant quantity of data before being used in a particular job. These models learn complicated patterns and correlations that would be difficult or time-consuming to learn from scratch since they are trained utilizing enormous quantities of data and computer resources.

In transfer learning, when a pre-trained model is modified or improved for a particular task using a smaller dataset, pre-trained models are often utilized. It may not be possible to train a model from start for tasks like voice recognition, natural language processing, and picture classification due to a lack of data. Popular pre-trained models include ResNet for image recognition, BERT for text classification, and GPT-3 for natural language processing. Developers and data scientists utilize these models for a number of applications since they are often made available via machine learning frameworks and libraries like TensorFlow, PyTorch, and scikit-learn.

Depending on the job they were trained on and the architecture they used, there are many kinds of pre-trained models. Here are a few illustrations:

DOI : https://doi.org/10.56452/6-3-664

1. Language models: These models are taught to foretell the word that will come after one in a phrase or a string of words. Examples are ELMO, GPT-3, and BERT.

Models for recognizing objects, persons, or other aspects in photographs are known as image recognition models. Examples include ResNet, VGG, and Inception.

3. Speech recognition models: Trained to convert spoken words into text, these models recognize speech. Kaldi and Deep Speech are two examples.

4. Text translation models: These software programs have been taught to translate text across different languages. 5. Recommendation models: These models are trained to suggest goods or information to consumers based on their prior behaviour or preferences. Examples include Google Translate and OpenNMT. Collaborative filtering and content-based filtering approaches are two examples.

Pre-trained models are an excellent resource for many applications since they can be adjusted or customized for certain tasks with a little amount of data.

2. LITERATURE SURVEY

[1] In "A deep learning-based framework for automatic brain tumours classification using transfer learning," Rehman et al. present a system for the automatic categorization of brain tumours using transfer learning. The authors test their suggested framework using a dataset that is accessible to the public and show that it outperforms other cutting-edge techniques in terms of classification accuracy.

Pan and Yang's article [2] , "A survey on transfer learning," offers a thorough overview of this method for transferring information from one area to another. The writers present an overview of several transfer learning techniques as well as a discussion of the major ideas, applications, and difficulties in transfer learning.

Chauhan et al. [3] provide research on managing unknown pictures and transfer learning in image categorization using deep neural networks. On two different datasets, the authors compare the effectiveness of various transfer learning approaches and offer insights into the trade-offs between them.

In "Accurate Underwater ATR in Forward-Looking Sonar Imagery Using Deep Convolutional Neural Networks," Jin et al. [4] provide a deep learning-based method for automated target identification in forward-looking sonar data. On a publicly accessible dataset, the authors test their suggested method and show that it performs better than other cutting-edge approaches.

In "Landslide Susceptibility Prediction Based on Remote Sensing Images and GIS: Comparisons of Supervised and Unsupervised Machine Learning Models," Chang et al. [5] compare supervised and unsupervised machine learning models for predicting landslide susceptibility based on remote sensing images and GIS. On a dataset of landslide occurrences, the authors compare the performance of various machine learning algorithms and offer insights into the trade-offs between supervised and unsupervised machine learning techniques.

In “Side Scan Sonar Image Segmentation and Synthesis Based on Extreme Learning Machine," Song et al. [6] provide a technique for side scan sonar image segmentation and synthesis using extreme learning machine. On a publicly accessible dataset, the authors test their suggested method and show that it performs better than other cutting-edge approaches.

In "Image retrieval scheme using quantized bins of colour image components and adaptive tetrolate transform," Varish et al. [7] suggest an image retrieval strategy using these two techniques. The authors compare the performance of their suggested approach to other state-of-the-art ones using a publicly accessible dataset and show that it performs better.

DOI : https://doi.org/10.56452/6-3-664

The authors of "Real-time Computerized Annotation of Pictures," Li and Wang [8], provide a technique for real-time computerized annotation of photos. The authors compare their suggested strategy to other state-of-the-art ones and show that it performs better on a dataset of picture data.

In "Using deep convolutional neural network architectures for object classification and detection within x-ray baggage security imagery," Akcay et al. [9] offer a deep learning-based method for object classification and detection within X-ray baggage security imagery. On a publicly accessible dataset, the authors test their suggested method and show that it performs better than other cutting-edge approaches.

A research on CNN transfer learning for image classification is presented by Hussain et al. [10] in their article "A study on CNN transfer learning for image classification." On three different datasets, the authors compare the effectiveness of various transfer learning approaches, and they also discuss the advantages and disadvantages of each approach.

3. RELATED WORK

Transfer learning is a well-liked machine learning approach that entails using the information acquired by a trained model to address a separate but connected issue. It might be a good way to cut down on the data and training time required for a new task. Transfer learning generally entails employing a pre-trained model—typically trained on a large dataset—and its acquired characteristics to complete a new job. By changing the weights of the final layers of the pre-trained model or by stacking new layers on top of the existing ones, the model is generally adjusted for the new job. Using transfer learning for pre-trained models has a number of advantages. First of all, it avoids the requirement to train a model from beginning, saving time and money. Furthermore, pre-trained models are frequently trained on substantial and varied datasets, which can enhance the model's generalizability for the new task. Last but not least, transfer learning might be helpful for jobs when the quantity of accessible labelled data is restricted.

VGG, ResNet, and Inception are a few well-known pre-trained models that are often utilized for transfer learning. For the purposes of picture classification, these models were first trained on huge datasets like ImageNet. Since then, however, they have been improved for a wide range of additional tasks, including text classification and sentiment analysis as well as object identification and segmentation. A potent method for enhancing the effectiveness and performance of machine learning models for a number of applications is transfer learning for pre-trained models. Computer vision often and successfully employs pre-trained machine learning models for image identification. Pre-trained models are very accurate in identifying objects, patterns, and characteristics within photos because they are trained on massive datasets, often containing millions of photographs, using complicated neural network designs.

The Convolutional Neural Network (CNN), a well-liked pre-trained model for image recognition, has produced state-of-the-art outcomes in several image identification challenges. To further increase the recognition accuracy, a particular dataset can be used to fine-tune the pre-trained CNN models. ResNet (Residual Network), a well-liked pre-trained model, was created to address the issue of disappearing gradients in very deep neural networks. In many image identification tasks, it has been shown that the ResNet architecture performs better than other pre-trained models. An image is sent into a pre-trained model for image recognition, which will then provide a forecast of what objects or characteristics it identifies in the image. Pre-trained models for image recognition are supported by a number of machine

DOI : https://doi.org/10.56452/6-3-664

learning frameworks and libraries, including TensorFlow, PyTorch, and Keras. Overall, utilizing machine learning models that have already been trained to detect common characteristics in images will save time and resources.

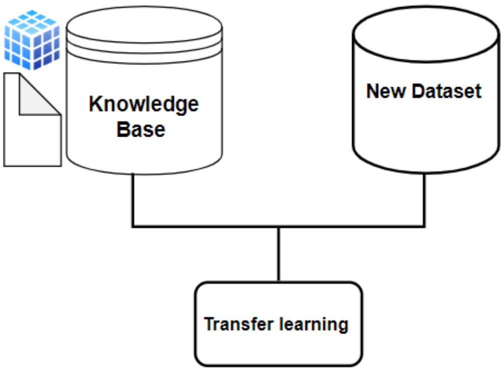

Figure 1. Transfer Learning model derived from Knowledge Base & New Dataset

The characteristics that are first employed in machine learning are created manually by researchers. Deep learning networks can automatically extract features. Nevertheless, you still have to decide which features to add to your network, so feature engineering and expertise are still crucial. Nevertheless, with the aid of a representation learning algorithm, neural networks can learn which features are critical and which are not. Even for difficult jobs that need a lot of human effort, it can swiftly discover a favourable combination of attributes. The representation acquired subsequently be used for further tasks. To decide on the appropriate representation of characteristics for that, utilize the initial layers. In order to utilize the output layer as an intermediate layer, we only input data into the network. The raw data will subsequently be represented by this intermediary layer.

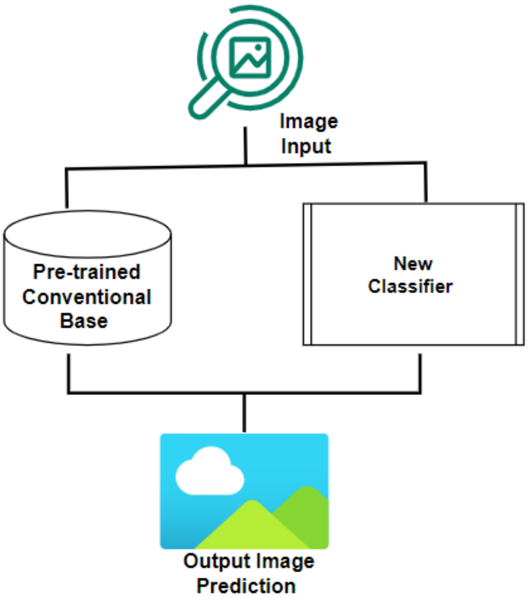

Figure 2. Building blocks presentation for Pre-defined ML based Turing techniques for transfer learning

In this architecture, the input picture is fed into a pre-trained convolutional base, which is a model built on a large dataset for feature extraction from images. This model is used above visualization for transfer learning utilizing pre-trained ML models for image recognition. A pre-trained model like VGG, ResNet, Inception, etc. serve as this basis. Then, features from the input image are extracted using the pre-trained base. A new classifier receiving these attributes is trained on a more focused dataset for the job at hand. The new classifier can be as simple as a linear classifier or as intricate as a neural network. The weights of the pre-trained base are frozen throughout training and only the weights of the new classifier are changed. In this manner, the pre-trained base makes the use of its understanding of images characteristics to raise the new classifier's accuracy on the given task. The new classifier, which uses the feature representation derived from the pre-trained base as input, then produces the output prediction.

DOI : https://doi.org/10.56452/6-3-664

A. Pre-trained ML model algorithm for transfer learning

1. Start by loading the trained model.

2. Freeze every layer that has been taught

3. Include a brand-new trainable output layer.

4. Include an appropriate optimizer and loss function when compiling the model.

5. Use your new dataset to train the output layer.

6. Assess the model's effectiveness using a validation set.

7. Adjust the model by thawing out parts of the layers that were pre-trained.

8. Use a lower learning rate to train the whole model on the new dataset.

9. Test the refined model on a set of data.

Depending on your unique use case, there are numerous variants on this method; this is just a basic idea. To better meet your objectives, you could decide to change the pre-trained model's architecture or add more layers. Furthermore, there are a wide range of optimization techniques and loss functions that could be suitable for your dataset. But this should give you a general idea of how to use transfer learning with a trained machine learning model.

B. Using a pre-defined model, present mathematical equation for transfer learning:

Using a pre-trained model on a certain task and then adjusting it on a different task is known as transfer learning. For transfer learning, there are several pre-trained models available in a variety of fields, including as computer vision and natural language processing.

The general formula for a transfer learning pre-trained model is as follows:

$y = f(x; \theta)$

where y is the model's output or forecast.

The input for the model is x.

The model's collection of parameters or weights is referred to as.

The function f uses the learnt weights to map the input to the output.

In transfer learning, we generally add new layers for the new job on top of the pre-trained model while freezing its weights. In order to optimize the model for the new job, the model is tweaked together with some of the pre-trained layers after the new layers are started randomly.

So, y = f(g(x; _pretrained), _new) where g is the pre-trained model with frozen weights can be used to represent a transfer learning model.

The weights of the pre-trained model are indicated by _pretrained.

The new layer weights that were introduced for the new job are denoted by _new.

We can get a model that performs well on both the previously taught task and the new task by training this model on the new task.

4. COMPERATIVE ANALYSIS

A. Comparison Between Traditional ML Model and Transfer Learning Model

DOI : https://doi.org/10.56452/6-3-664

Table 1. Comparison between Traditional ML model and Transfer Learning Model

Parameters	Traditional ML Model	Transfer Learning Model
Data Quantity	Need large amount of data set	Small dataset can be useful
Costing	Very expensive	Cost effective as it derives from pre trained model
Training/ knowledge Share	Independent training for each and every model related to particular task	Knowledge gets transferred from pre-defined ML model
Time	Long time requires to complete task	It creates fast output as it uses transfer learning algorithm to fetch knowledge from pre-trained ML model

B. Use case Transfer Learning Model:

Transfer learning can be used in the following circumstances by ML practitioners:

1. Inadequate data:

Poor performance would arise from using less data. A pre-trained model is used to produce models that are more precise.

2. Lack of Time:

It takes too long to train certain machine learning models. If you don't have enough time to create a new model, choose one that is comparable but already trained.

3. Limited capacity for calculation

The introduction of the pre-trained model is a tremendous aid since the large number of machine learning tasks required to train the model demand a lot of computes.

Figure: Graphical Representation of Transfer Learning use cases

C. Use Cases for traditional Model:

1. Short characteristics

Transfer learning is ineffective if the classes for a new issue cannot be distinguished by the characteristics learnt by the classification layer in the lower layer. Think about the famous "dogs and cats" illustration while attempting to determine if transfer learning is appropriate in your situation. Let's say we've categorized the first and are now trying to identify the

DOI : https://doi.org/10.56452/6-3-664

second. Low-level feature representations are still acceptable in this scenario, whereas mid-level and high-level ones are no longer viable. Therefore, rather than using the high-level features of the pre-trained model, you can use its low-level features. If even deeper layers don't provide satisfactory results, you will also need to retrain additional layers or, in the worst scenario, start again with a new model.

2. Unrelated Datasets:

If the datasets are not comparable, the characteristics do not transfer well. In order to avoid overfitting, this would need limiting the parameters that may be learned and deleting certain layers. How many layers can be eliminated without overfitting is quite difficult to assess and takes a lot of effort.

3. Larger Datasets

On tasks that call for larger datasets, the effects of transfer learning might not be as anticipated. Traditional machine learning is adjusting randomly generated weights until they converge. Although larger data sets also result in more iterations, your initial weights become meaningless when using transfer learning, which starts with a pre-trained model.

Transfer learning's network layers are necessary for developers to accurately determine which AI models are the best. The number of trainable parameters decreases when the initial layers are removed, which has an impact on dense layers. Dense layers may also be a useful starting point for layer reduction, but it requires a lot of time and effort to decide how many layers and neurons to preserve to avoid model overfitting. Overfitting happens when a new model absorbs extraneous information and noise from the training set, which has an impact on the model's predictions.

5. CONCLUSION

This paper has provided a comprehensive comparative study of various pre-trained models and fine-tuning techniques for transfer learning in the context of image datasets. Our findings demonstrate that transfer learning, when effectively applied, can significantly improve the performance of deep learning models on diverse tasks, reducing both training time and computational resources. Through our extensive research, we found that the choice of pre-trained model and fine-tuning technique greatly affects the overall performance of the transfer learning process. Specifically, we observed that models with a larger number of layers and parameters tend to yield superior results, while the optimal fine-tuning technique depends on the nature of the target task and dataset. Nevertheless, the most effective approaches often involve fine-tuning only a subset of layers and employing techniques like learning rate schedules and early stopping to prevent overfitting. From our research we found that the success of transfer learning depends on a careful consideration of various factors, including the choice of pre-trained model, fine-tuning technique, and dataset similarity. This study provides valuable insights and guidance for researchers and practitioners seeking to leverage transfer learning in their work, ultimately contributing to the advancement of the field and the development of more efficient, accurate, and resource-friendly deep learning models. Future work in this area could investigate the impact of emerging models and techniques, as well as explore the application of transfer learning to other domains and modalities.

References:

[1] A. Rehman, S. Naz, M. I. Razzak, F. Akram, and M. Imran, "A deep learning-based framework for automatic brain tumors classification using transfer learning," Circuits, Systems, and Signal Processing, vol. 39, no. 2, pp. 757–775, 2020.

[2] S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Transactions on knowledge and data engineering, vol. 22, no. 10, pp. 1345– 1359, 2009.

[3] V. Chauhan, K. D. Joshi, and B. Surgenor, "Image classification using deep neural networks: Transfer learning and the handling of unknown images," in International Conference on Engineering Applications of Neural Networks. Springer, 2019, pp. 274–285.

[4] Jin, L.; Liang, H.; Yang, C. Accurate Underwater ATR in Forward-Looking Sonar Imagery Using Deep Convolutional Neural Networks. IEEE Access 2019, 7, 125522–125531.

[5] Chang, Z.; Du, Z.; Zhang, F.; Huang, F.; Chen, J.; Li, W.; Guo, Z. Landslide Susceptibility Prediction Based on Remote Sensing Images and GIS: Comparisons of Supervised and Unsupervised Machine Learning Models. Remote Sens. 2020, 12, 502.

[6] Song, Y.; He, B.; Liu, P.; Yan, T. Side Scan Sonar Image Segmentation and Synthesis Based on Extreme Learning Machine. Appl. Acoust. 2019, 146, 56–65.

[7] Varish, N., A. K. Pal, R. Hassan, M. K. Hasan, A. Khan, N. Parveen, D. Banerjee, V. Pellakuri, A. U. Haqis and I. Memon., Image retrieval scheme using quantized bins of color image components and adaptive tetrolet transform. IEEE Access, 2020. 8: p. 117639-117665.

[8] Li, J. and J.Z. Wang, Real-time computerized annotation of pictures. IEEE transactions on pattern analysis and machine intelligence, 2008. 30(6): p. 985-1002.

[9] Akcay, S., M. E. Kundegorski, C. G. Willcocks and T. P. Breckon., Using deep convolutional neural network architectures for object classification and detection within x-ray baggage security imagery. IEEE transactions on information forensics and security, 2018. 13(9): p. 2203-2215.

[10] Hussain, M., J.J. Bird, and D.R. Faria. A study on cnn transfer learning for image classification. in UK Workshop on computational Intelligence. 2018. Springer.