Supervised Classification with Deep Graph CNN

Mathias Tiberghien
Paris 8 University
Paragraphe Laboratory
Paris, France
Mathias.tiberghien@gmail.com

Rakia JAZIRI
Paris 8 University
Paragraphe Laboratory
Paris, France
Rjaziri@univ-paris8.fr

***Abstract*—Convolution neural networks (CNNs) have performed remarkably well in recent decades and become essential for classification tasks based on images or voice. However, this paper addresses some of their limitations in terms of generalization and examines some well-known CNNs architectures that try to create hierarchical structures based on graph databases. The contribution of this work is to present a structure where the convolution blocks are distributed in nodes, the relations between each node being articulated using a data partitioner. This exponentially multiplies the number of models depending on the depth of the graph and the number of partitions, but it keeps track of the hierarchical relationships between each node.**

Keywords-convolution neural network; image processing; classification; neuronal network architecture; deep learning; incremental learning

I. Introduction

In recent decades, Convolutional Neural Networks (CNNs) have demonstrated remarkable performance and have become indispensable for image and voice classification tasks. However, as the field of deep learning continues to advance, it has become increasingly important to address the limitations of CNNs in terms of generalization and explore alternative approaches to overcome these challenges.
This paper aims to shed light on some of the inherent limitations of CNN architectures and examines existing CNN models that attempt to create hierarchical structures using graph databases. While these models have shown promise in capturing hierarchical relationships, they still face certain limitations in terms of scalability and flexibility. The paper is composed of a state of the art, contribution and experimentation.

II. Related works

The section on related work provides a comprehensive review of existing literature and research efforts focusing on CNN architectures, incremental learning, hierarchical classification, and approaches for improving explainability in deep learning models.

A. Symbolic representation vs Texture

Convolution neural networks (CNNs) [1] [2] can be thought of as filters whose role is to reveal the parts of an image that best identify the category it belongs to. One way to visualize how these filters operate is the GRAD-CAM technique [3], which draws a heatmap representing the areas of images that contribute most to their classification.

Figure 1 is a typical example of GRAD-CAM taken from [4]. This picture suggests that the model used ears and skin

Fig. 1. Example of GRAD-CAM

to correctly classify the picture as African elephants.

Figure 2 represents a set of cat images submitted to a VGG19 [5] architecture trained on ImageNet [5] allowing classification of images among 1000 categories. The pictures were enhanced using GRAD-CAM in order to highlight the most discriminating parts.

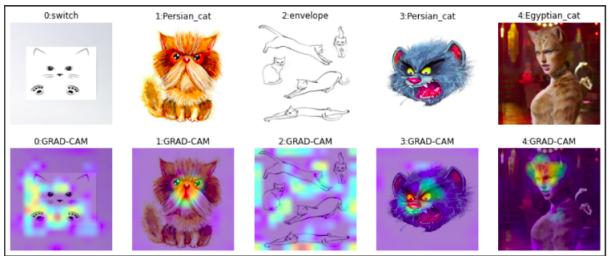

Fig. 2. Classification of cats

While some cats were correctly classified (the second, fourth and fifth image), others were not, and the GRAD-CAM helps to understand why. The model is myopic: it focuses on the hair texture of the cat which seems to be a better discriminant than the shape of an ear, eye, or tail. In the first

 ISBN: 978-1-68558-051-3

and third image, we can see that the model has revealed what we as humans perceive to be background elements. The model relies on details, completely missing the main features likely captured in the first layers of the model. The accuracy of a prediction depends on the density of details in the pictures that are related to a specific category and not on the strength of the symbolic representation of the category, which is more related to the shape or the specific parts of the object to be classified. Intriguingly, the helicopter is not part of the 1000 categories while there are many detailed means of transport. Is this an oversight, or is it because the helicopter is a perfect example of an object with strong symbolic representation but a poor density of discriminatory details—having a propeller and a body with heterogeneous textures.

B. Prioritization and relationships

Figure 2 shows the result of the classification of pictures of humans.

Fig. 3. Classification of humans

Humans examining these pictures would immediately recognize members of the human race, while the model focuses on their clothing. Clothing might be an interesting way to classify humans, but there are many use cases when it is important to obtain some information on the person wearing them. This leads to a major concern about the model and how it was trained—the horizontality of the classification seems to have a serious limit. The model tries to differentiate objects, species of animals, and types of clothes on a unique level when classification is actually mostly hierarchical, as shown in Figure 4.

A tree is a good structure to represent a hierarchical classification but a graph will complete this one by defining the type of relation existing between categories as shown in Figure 5.

C. Stability on retraining

One of the recurring challenges in image classification, especially in medical classification tasks such as providing a diagnosis using medical imagery, is the stability of a model after successive training sessions [6]; the model is first trained for a task without having the whole spectrum of data and loses efficiency when images whose type is a little different from the initial domain are added. In medical fields, these differences, known as domain shift, may be due to the evolution of imaging techniques, the difference between brands, hospital practices, but also because of the difficulties in obtaining a dataset from

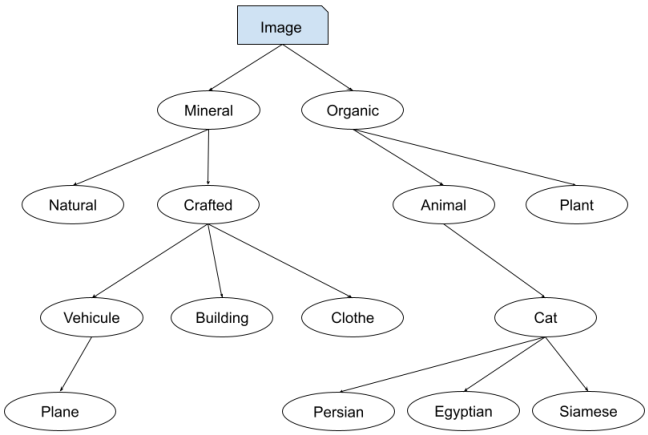

Fig. 4. Hierarchical classification

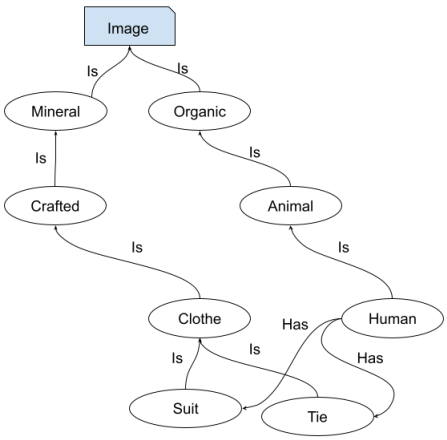

Fig. 5. Graph classification

several hospitals due to the private nature of the images. Another cause of the domain shift, is the medical condition (e.g., standing and conscious or lying down and unconscious) of a patient at the moment an image is taken, affecting the images quality and interpretability. Medical models tend to specialize in a specific dataset, and it is often difficult to increase the diversity of image sources without degrading the quality of predictions. Some techniques [10] and [11] try to attenuate the problem of stability, but this also raises the question of the pertinence of having one unique model handle a specific task. While humans can adapt their judgment according to context (an expert can recognize the specificity of a machine or a medical condition), why should a CNN model find a middle way in order to globally reduce its loss function?

D. Desconstructing a CNN

The examples above show that even if CNN can extract image features to an extraordinary extent, stacking convolutions blocks may focus on detail and texture, losing the broader

 ISBN: 978-1-68558-051-3

picture. The use of a horizontal, flat categorization hides the hierarchical relationships between classes. Finally, a dataset can be composed of different contextual information even if the purpose (a diagnostic by example) is unique. Incorporating this diversity into a single model can cause its performance to drop as it tries to average the best solution among different cases. We believe that an evolution of classification tasks based on CNNs, should organize convolutional blocks into more complex structures, such as graph databases.

A CNN architecture can be summarized to a feature extractor whose output is flattened to feed a classifier. The feature extractor is a sequence of convolution blocks, which are an arrangement of convolution layers followed by a pooling layer, while the classifier is a fully connected neural network, as shown in Figure 6 . There are several architectures for convolution blocks that solve different problems like ResNet and Inception .

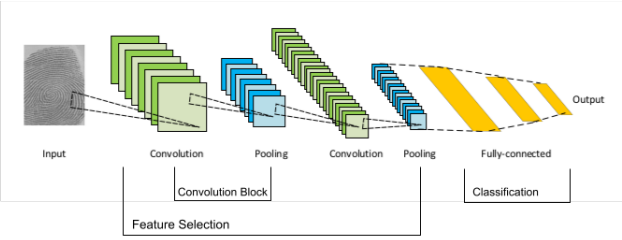

Fig. 6. CNN architecture (image taken from [7])

Related works are Tree-CNN [8], Growing Hierarchical Neural Network [9], [10] or Incrementally Growing CNN [11], which try to adapt CNN structures to the diversity of data and classified classes.

III. Contribution

The main contribution of this work is the introduction of a novel hierarchical structure where convolution blocks are distributed across nodes, with the relationships between each node orchestrated using a data partitioner. This architectural innovation exponentially increases the number of models based on the depth of the graph and the number of partitions, while still maintaining the hierarchical relationships between each node.

By distributing the convolution blocks in this manner, we create a more intricate and specialized model that is capable of capturing complex patterns and features across different levels of abstraction. The hierarchical organization allows for better representation of the underlying data structure and facilitates effective information flow throughout the hierarchical model.

A notable advantage of this distributed architecture is the sharing of loss function gradients among the common roots of the specialized models during training. This shared gradient propagation enhances the model's ability to collectively learn from the training data, leading to improved overall performance.

To provide a visual representation of this novel architecture, Figure 7 illustrates the general structure of the proposed model. It showcases the interconnected nodes, each housing specialized convolution blocks, and the data partitioner facilitating the flow of information and gradients between the nodes.

Through extensive experimentation, we evaluated the performance and efficacy of this distributed convolutional architecture on various benchmark datasets. The results demonstrate the potential of our approach to achieve enhanced accuracy and efficiency in tasks such as image recognition, natural language processing, and sensor data analysis.

In summary, our work presents a novel architectural structure where convolution blocks are distributed across nodes, connected through a data partitioner, and organized hierarchically. This approach leverages the benefits of specialization and shared gradient propagation, ultimately leading to improved performance and adaptability in deep learning tasks.

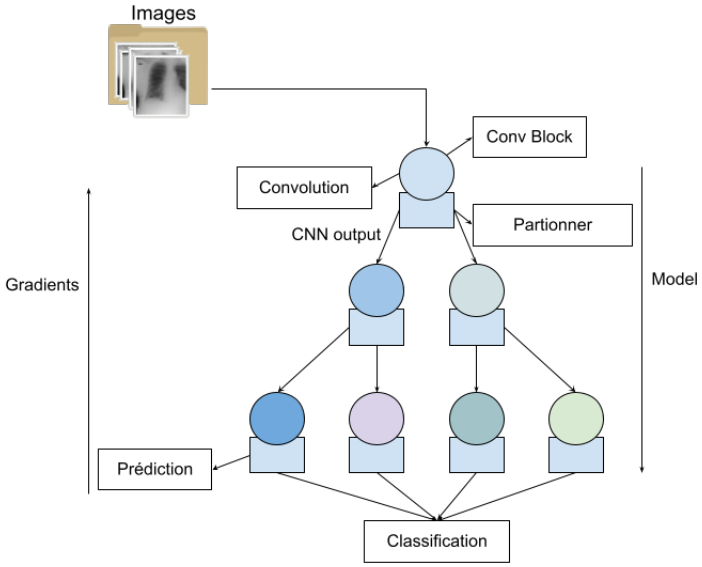

Fig. 7. Partitionned CNN architecture

IV. Experiments

To validate the effectiveness of our proposed approach, we conducted a series of experiments on various datasets spanning different domains. Our experimental setup aimed to assess the performance and robustness of the enhanced CNN architectures in real-world scenarios and on different aspects described as follows:

 ISBN: 978-1-68558-051-3

A. Dividing data

We retained two approaches to split the data: similitude and intuition. Similitude divides data between entries that are similar when intuition divides data using a result of a prediction, and also the degree of confidence to that prediction.

Similitude: Similar images should be treated the same way. Evaluating similitude from an automated perspective can be achieved, through Intersection over Union, Cosine Similarity [12] but also by K-means [13]. Using K-means at a flattened output of a convolution block creates clusters of images presenting similarities. This can be used to divide the dataset, the number of clusters being determined arbitrarily or using the number of classes to be predicted.

Intuition: When a model predicts a class, it is making an assumption often using a softmax activation defining the probability of an input to belong to a specific class. The confidence in the assumption increases when the probability tends to one. Adding a fully-connected network at the output of a convolution block using a softmax activation for the last layer can be used to divide the dataset by assumption. The partition can use the degree of confidence of the model. One group contains images whose prediction has a really high degree of confidence, and other groups can be created based on classes, or just using the images with lower degree of confidence.

Tests: Using a problem of pneumonia detection realized during my internship at Delafontaine Hospital, St-Denis, based on chest x-ray, we trained a reference model based on ResNet50v2 [14] architecture trained on Imagenet coupled with a fully-connected network. The training dataset was then clusterized using K-means (K=2) at a flattened output on a ResNet50v2 (ImageNet) without any classification. Another clusterized dataset was created using the reference model dividing the dataset using intuition: one group contained images where softmax "probability" was higher than 0.98 while other contained all the other images. For each cluster, a model was trained separately with the same architecture and parameters as the reference model. Training each cluster separately produced different results in both division techniques as shown in Table I.

TABLE I
DATA CLUSTERING

Model	Training files	Testing files	Imbalance	Accuracy
Ref	6325	1581	0.29/0.71	0.86
Sim 1	5734	1432	0.23/0.77	0.80
Sim 2	593	147	0.93/0.07	0.95
Int 1	3499	874	0.16/0.84	1
Int 2	2827	706	0.46/0.54	0.74

The clustering by similitude created two groups and one of them contains almost exclusively negative cases, which is also a smaller dataset. It has isolated a specific type of image which is a clearly identifiable negative case (593*0.93 = 551). Clustering by intuition where confidence is high shows approximately the same number of images (0.16*3499 = 559) when considering negative cases. These sets of images are indeed the same which are x-rays taken in optimal condition when the patient is healthy, standing and conscious. The intuition cluster with lower confidence became more balanced. The validation scores show for that a high confidence is correlated with the accuracy and that the accuracy for images where confidence is lower reflects the capacity of the model to deal with gray areas which is an important information: a doctor want to know how the model is dealing with cases where he has doubts, not the easy cases. That correlation was confirmed by radiologists when evaluating manually the performances of our model. Splitting the data gave us the same average performance, but we gained in granularity for explainability. Future works will try to use different models, to see if it is possible to improve the average performance for the datasets containing more complicated cases.

B. Architecture

CNNs models have some limitations but they also have the great benefit of being simple, easy to maintain, obtaining great performances. At the opposite, partitioning the data and convolutions structure are complexifying the model, the training and the maintainability. Our first experiment is based on a unique model approach, trying to embrace the complexity step by step.

Unique model approach: we used the Keras library with its functional api to build a unique model having the shape of a tree. We used the VGG16 architecture as a template which is composed of 5 convolution blocks. Using a custom triage layer using a K-means model that splits the output and chaining convolutions blocks.At the end of the mode, a fully-connected network performs the prediction. Our model supports 5 levels of depths (1-5), adapting the first convolution block to recreate VGG16 architecture having K^{level} prediction models. Figure 8 presents the schema of a model with K=2 and a depth of 3 which generates 8 different VGG16 structures.

The structure of each model being the same as the VGG19, we can still use transfer learning [15] on each convolution block. The main challenge is that we have to concatenate then sort the predictions to be able to match each batch input order (partitionning the batch shuffles it). The first convolutions blocks are shared by their children. At each level the convolution blocks are specializing to a type of image. This kind of tree has a limited complexity and will be used in future works to analyze the pros and cons of using such a model.

Multiple models approach: if the unique model architecture lets us experiment with separate training between different categories of images, developing specialization at deeper levels and keeping track of the common grounds, it has a limited perspective of evolution. Triage layers will probably generate empty datasets, and some blocks won't be activated: the model is too static. An alternative approach will consider convolution blocs and data paritionners as entities. Each convolution block is trained separately and is chained to another block to a partitionner as shown in Figure 9.

 ISBN: 978-1-68558-051-3

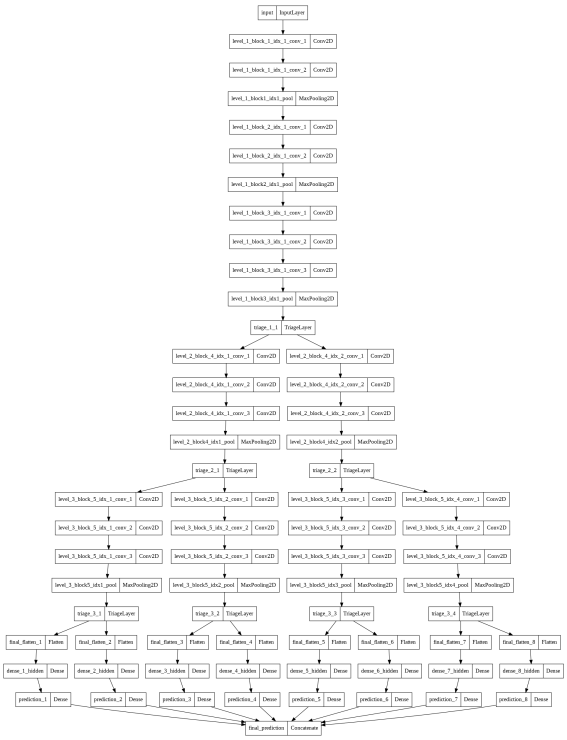

Fig. 8. Partitionned CNN architecture using Keras

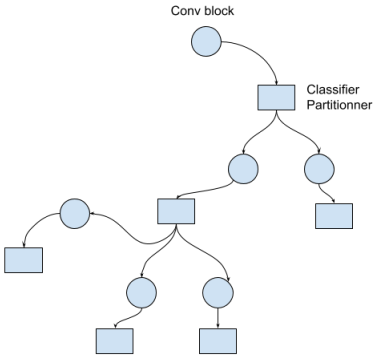

Fig. 9. CNN as as graph architecture

The structure of such an architecture can be represented as a graph and it will grow dynamically as follows: starting with a training dataset, a convolution block and a classifier, the model is trained until the loss function doesn't improve. Images proposed from the training set will have high confidence and the model will return prediction without having to change. When new images are proposed to the model and that confidence decreases, the model starts to partition and store the output data of the convolution block. When uncertain data reach a sufficient size, new convolution blocks are chained to the partitionner (according to the number of predicted classes) and trained using this new dataset.

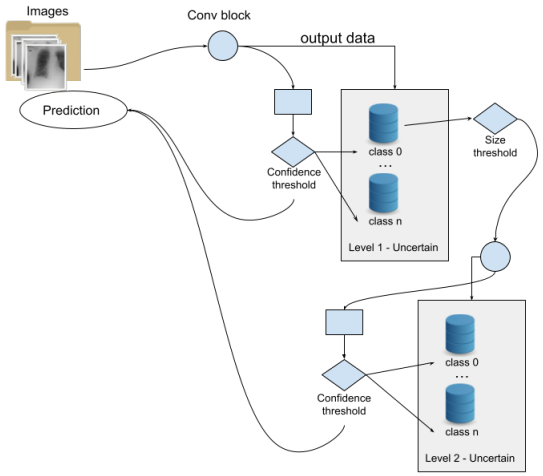

Fig. 10. CNN as as graph architecture

Figure 10 represents the data flow for training, growing and predicting with a graph CNN. The graph will grow as uncertainty samples presented to the model are increasing. New datasets are generated from output of convolution blocks only for images with low confidence. Images submitted for prediction will be treated by the first block. If the level of confidence is high enough the prediction will be returned, otherwise the output data will be forwarded to the next node if it exists or stored in the uncertain data database that will be later used to train a new child node. The uncertain data storage is necessary to be able to control data flow while training. It is a temporary data and can be deleted after training. We can consider it as a short term memory, while a trained node is considered as long term memory. Future works will try to build such a graph trying to solve different classification problems.

V. Conclusion

In this paper, our aim was to highlight certain limitations of current CNN architectures, specifically in the context of incremental learning and hierarchical classification. We put forth several ideas that revolve around data partitioning and chaining convolution blocks to address these limitations and enhance the explainability and specialization across diverse source types.

Our proposed approach, incorporating data partitioning and convolution blocks chaining, aims to overcome these challenges and improve the performance of CNNs in hierarchical classification tasks.

In summary, our paper emphasizes the limitations of current CNN architectures, particularly in the realms of incremental learning and hierarchical classification. We propose innovative ideas centered around data partitioning and convolution blocks chaining to enhance explainability and specialization across different source types. Our intention is to evaluate and refine these more complex structures through real-world use cases in future research efforts.

 ISBN: 978-1-68558-051-3

ACKNOWLEDGMENT

Thanks to Laurent Payen, head of radiology at Delafontaine Hospital, St-Denis, who helped me to understand the mechanism of interpretation of an image by a human expert.

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, vol. 86, no. 11, 1998, pp. 2278–2324.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012.

[3] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 618–626.

[4] F. Chollet, *Deep Learning with Python*. Manning, Nov. 2017.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[6] S. U. Rehman, S. Tu, O. U. Rehman, Y. Huang, C. M. S. Magurawalage, and C.-C. Chang, "Optimization of CNN through novel training strategy for visual classification problems," *Entropy (Basel)*, vol. 20, no. 4, Apr. 2018.

[7] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3523–3542, 2022.

[8] D. Roy, P. Panda, and K. Roy, "Tree-cnn: A hierarchical deep convolutional neural network for incremental learning," *Neural Networks*, vol. 121, pp. 148–160, 2020.

[9] I. Mrazova and M. Kukacka, "Image classification with growing neural networks," *International Journal of Computer Theory and Engineering*, pp. 422–427, 01 2013.

[10] J. Herrero, A. Valencia, and J. Dopazo, "A hierarchical unsupervised growing neural network for clustering gene expression patterns," *Bioinformatics*, vol. 17, no. 2, pp. 126–136, Feb. 2001.

[11] D. Sam, N. Sajjan, R. Babu, and M. Srinivasan, "Divide and grow: Capturing huge diversity in crowd images with incrementally growing cnn," 06 2018, pp. 3618–3626.

[12] G. Abosamra and H. Oqaibi, "Using residual networks and cosine distance-based k-nn algorithm to recognize on-line signatures," *IEEE Access*, vol. 9, pp. 54 962–54 977, 2021.

[13] E. Ahn, A. Kumar, D. Feng, M. J. Fulham, and J. Kim, "Unsupervised feature learning with k-means and an ensemble of deep convolutional neural networks for medical image classification," *unpublished*, vol. abs/1906.03359, 2019.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[15] M. A. Morid, A. Borjali, and G. Del Fiol, "A scoping review of transfer learning research on medical image analysis using imagenet," *Computers in biology and medicine*, vol. 128, p. 104115, 2021.

 ISBN: 978-1-68558-051-3