Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Modular Network Architectures for Enhanced Interpretability and Efficient Knowledge Transfer in Machine Learning

Dr. Rajesh Gundla[1], Palla Chamundeswari[2], Dr. P. Rajendra Prasad[1], Ch. Swapna[3], K. Sandhya Rani[3], J. Ranjith[3]

[1]Department of CSE, Vignan's Institute of Management & Technology for Women, India
[2]Department of CSE (DS), Vignan's Institute of Management & Technology for Women, India
[3]Department of IT, Vignan's Institute of Management & Technology for Women, India
Email: rajgundla@gmail.com

Over the years, as deep learning models have continued to grow in complexity, they become more complex and less interpretable, which makes them difficult to deploy into real-world scenarios where adaptation is critical. In response, this paper presents a new modular network architecture that fosters both interpretability and modularity within knowledge transfer. The structure of the proposed network is composed of a part named general core to extract general features that can be used for multiple tasks and need no task-specific information, as well as a few innovative modules fine-tuned on special tasks that happen in limited amounts. Interpretable models like decision trees are included in the task-specific modules, which will explain model predictions in a way that is verifiable by humans given a data point. This modular approach also results in very little retraining required when repurposing the model to perform new tasks. It allows each decision made by the system to be interpretable as well, which makes it more reliable (especially in critical applications like healthcare or finance). We performed experiments on image classification, sentiment analysis and healthcare datasets to validate the proposed approach. The results demonstrate that the new modular network architecture not only increases task performance but also improves explain ability relative to traditional end-to-end deep learning. It can also reduce training times and increase the efficiency of knowledge transfer, meaning that a single system can learn quickly from more tasks. This work is a step towards making machine learning systems more interpretable, adaptable and efficient for improved explainable AI while establishing an essential base for future advancements in multitask learning.

Keywords: Modular networks, interpretability, knowledge transfer, task-agnostic, decision trees, explainable AI.

1. Introduction

In recent years, the model complexity has become a critical issue, especially for deep neural networks (DNNs) models that hampers their usage in practice. Though these models work brilliantly in all the complicated predictive tasks, they are labelled as black boxes since there is no way of knowing exactly how the decision was made. It has brought a lot of concerns, mainly in areas that require decision-making in tough situations, such as health facilities, financial institutions and the judicial systems. Also, when working with a specific task, most of these models tend to be fine-tuned from one task to another, which is costly for much time and resources. To overcome these difficulties, it is increasingly necessary to use both explainable and compassable models. Interpretable models enable users to understand why and how a particular decision is made. At the same time, modularity allows the same model to be deployed in other tasks with little or no retraining. Modular networks where the system is made up of individual modules dedicated to performing the individual sub-functions are a good solution to this problem. We can train and fine-tune every module separately. It makes the system more adaptable and more accessible for troubleshooting and tuning. This paper presents a new modularity-based approach to the structure of the network that increases not only interpretability but modularity as well in the process of knowledge transfer. The system architecture is of task-neutral and task-specific parts. The task-agnostic core deals with general tasks independent of the specific application, while the task-specialized modules deal with functions that directly relate to a particular application. Here, the system successfully decentralizes these components, which promotes reusability because fine-tuning is not required with similar examples. In addition, to increase the interpretability of the developed system, interpretable models, including decision trees and rule-based systems, will be incorporated into the framework and implemented at the modular level. This integration makes obtaining an understandable and easily interpretable human understanding of the decision-making process easier. For example, look at an intelligent diagnostic system used in the healthcare industry to determine a patient's diabetes risk. In a traditional DNN model, the system can predict a high probability of new patients getting diabetes, but it won't explain why it reached this conclusion. In the proposed modular architecture, however, the system could incorporate a decision tree in a particular task-based module to clarify that the given decision was arrived at in consideration of factors such as high blood sugar levels, age, and family history – in a clear line of reasoning that the patient could apprehend and understand.

2. Related Work

This section surveys the prior art of modular network architectures, transparent and understandable machine learning models and techniques for knowledge transfer. More emphasis is given to how these areas have developed to deal with issues of interpretability, modularity, and flexibility for realistic applications.

2.1 Modular Network Architectures

One popular misconception has been inclined towards modular architectures that have been viewed as practical means of breaking down complex tasks into tractable sub-tasks that can be trained in isolation and used in different contexts. All such architectures help make the training

more efficient and increase the versatility and modularity of machine learning. Modern approaches to Multitask Learning and Domain Adaptation have demonstrated that modularity allows reusing computational modules across tasks and effectively switching between them and sharing [1]. Such architectures often enable the reshuffling of computational modules for better fitting into new tasks: efficiency and modularity are the significant advantages of multitasking. Also, there is a new concept of knowledge distillation, where the pre-trained network is rectified into smaller sub-networks that can be recombined into work-related tasks with minimal retraining efforts [2]. This approach allows for a 'modular' solution to the problem whereby if one part of the system needs to be retrained, it does not impact the rest of the system as it is a plug-and-play system where each module serves as a component of more significant tasks. Another modular learning method has also been proposed as a kernel-based modular learning framework that redesigns the deep learning layers as learning modules that can be trained separately without transferring between modules [3]. This method enhances the readability and sustainability of the models as the design is made relatively simple, and individual components can be used for other tasks.

2.2 Interpretability in Modular Systems

Significant problem associated with deep learning, mainly when such models are utilized in essential areas. To solve this problem, decision trees and rule-based systems can be incorporated into neural networks to give clear explanations of the system's decisions. For instance, modular architectures have been integrated with interpretable components such as decision trees to bring about the interpretability of a model's results and still be accurate [4]. These models are beneficial with professions that involve decisions about a person's health and well-being or finances. One example is the Tree-Network-Tree (TNT), which models decision trees together with deep neural networks to increase the interpretability of decisions made by DL [5]. This framework takes advantage of the application of decision trees in creating understandable decision tracks, with the help of which the system is more suitable for the domains in which it is essential to clarify why the decision was made. The TNT framework enables knowledge exchange between tree and neural network components to enhance interpretability and accuracy. New studies have also revealed that in terms of per-user trust/usability, black-box models with decision trees or symbolic reasoning elements within a modular network are much preferred in decision-sensitive tasks [6]. These systems directly illustrate decision trees and allow users to check how each input feature impacts model output.

2.3 Task-Agnostic and Task-Specific Components

Splitting organizational-functional and execution-functional modules has been considered the perspective approach to enhancing reusability and knowledge transfer in learning models. One type of module is task-agnostic, while the other is designed especially for a specific task but may include features of several tasks. This division makes it easier not to train models from scratch, hence allowing the deployment of machine learning systems in a wide array of applications [7]. In a recent approach, task-specific components were designed to work with task-agnostic feature extractors to allow direct transfer between tasks without any setback in performance [8]. This compatibility of components is essential to achieve a condition where modular systems are ready to take on several tasks. In addition, if different task-specific modules are utilized, they can be fine-tuned to match the desired objectives, improving the

system's flexibility towards practically any task. Modularisation has also been applied in reinforcement learning and given positive results. Another paper presented an augmented modular reinforcement learning approach that dealt with using an arbitrator to choose between different modules appropriate for a given task [9]. This framework enhanced the functions and interpretability of the results, as the modules could be assessed several times.

2.4 Knowledge Transfer in Modular Architectures

One advantage of modular architectures is the cross-task communication of knowledge. Due to the nature of a modular network, information acquired in one task can be effectively transferred to another; it doesn't need to be retrained very often. This accelerates convergence and increases task and sample efficiency within few-shot learning and reinforcement learning tasks [10]. Another study showed that a modular deep learning architecture was used to train the neural network with functional blocks as modules, which were then further applied to new tasks by rearranging these learned functional blocks [11]. This approach helps the knowledge be passed quickly between these tasks, reducing errors while making it much easier to interpret. Modular systems have also been used in unsupervised domain adaptation, in which knowledge transfer is done by altering the computational connections between modules that process various tasks, specifically those designed for particular tasks [12]. This method revealed that modular architectures can handle unseen data effectively and exhibit a promising performance indicator for real-world scenarios.

2.5 Interpretability in Decision-Making

If the general public accepts machine learning, it is necessary to make it interpretable, especially in sensitive areas such as the health sector. Neural networks can be enhanced by adding interpretable components like decision trees or symbolic models, which make it easy for humans to understand the decision-making process created by the neural networks. Systems that use such models as modular design components provide accountabilities for individual tasks, which is essential in most applications [5]. For example, a healthcare diagnostic system can use a modular architecture and a specified task-oriented module to diagnose blood sugar in diabetic patients. If there is a decision tree in this module, it can explain, for example, "High blood sugar and being over 50 years old can be a sign of diabetes." Then, medical staff can feel at ease and put their trust in the prediction.

2.6 Summary

In summary to applying modular architectures helps solve the problem of interpretability, modularity, and knowledge sharing. Such systems can be designed modularly. The general and task-related components are separable, so it is possible to reuse general models that interpret them and train new models with minimal adjustments on the new tasks. Given the ongoing expansion of both critical real-world application domains and methods enhanced by machine learning, the demand for more precise, more modular, and flexible forms of system organization will, therefore, only increase, thereby establishing modular architectures as an essential area of research ahead of us.

3. Methodology

An innovative model for the proposed network structure aims to improve the degree of interpretability and modularity of the knowledge transfer process. This section explains the system's modularity, where interpretable sub-modules are incorporated, how knowledge is transferred, and the training methodology. Every part of the system allows for the model's decision-making to be transparent so that the efficiency of the tasks is met without compromising the system's transparency.

3.1 Modular Network Architecture

The main structure of our system is based on a modular network architecture that splits the overall problem into smaller and independently trainable subtasks. The architecture consists of two main components: It divides it into a task-general central part and additional modules pertinent to particular tasks. The task-agnostic core caters to all generic characteristics observed in different tasks; the task-specific modules address characteristics peculiar to the respective functions such that each module may be further optimized or reused. The feature of the architecture is that it is easy to train for multiple tasks without retraining models. The task-agnostic modules amplify the unnecessary complexity for the task-specific modules on areas of features that are not useful to the functions in the chosen problem settings. Similarly, all the task-specific modules obtain input from the core and then transform them further to arrive at the required outputs related to their assigned task. By divorcing these two elements, the system is designed to be very modular so that it can be deployed on different applications with little or no adjustments between platforms. The figure 1, shows the overall structure of the modular network architecture. The task-independent conducts raw input data pre-processing and feature extraction and outputs these features to the task-specific sub-modules. This is then followed by a task-specific module where the appropriate operation for the task, such as classification or regression from extracted features, is performed.

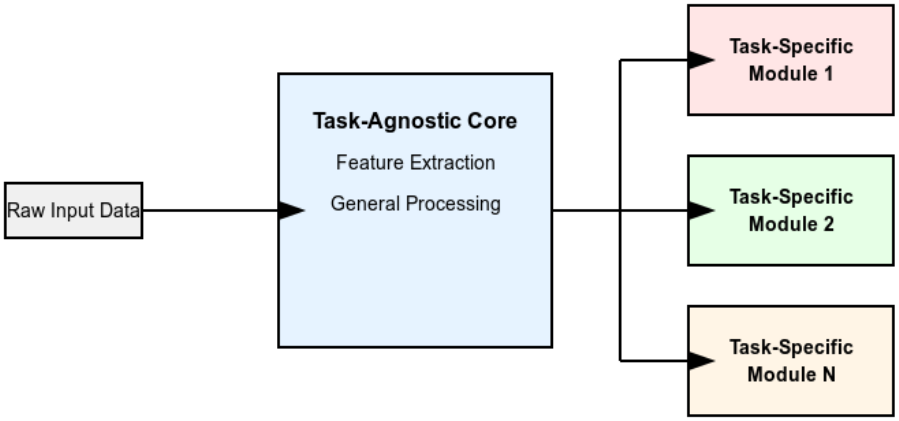

Figure 1. Modular Network Architecture Overview

3.2 Interpretable Modules

Among the key goals of this architecture is improving the interpretability of machine learning models, where transparency in the decision-making process is required. To do so, the task-specific modules that are incorporated include interpretable models such as decision trees. These decision trees also offer guided, rule-based decision justifications for what the task is doing and why in the case of each function's output. For instance, in a medical diagnostic system, the decision tree embedded in the task-specific module for diabetes prediction might explain a diagnosis as follows: For example, if a patient's glucose level is higher than 140 and their age is more than 50, then the model is very likely to recommend for diabetes. Even in job interviews where DF is used to determine qualifications for a particular post, the candidate understands why they were disqualified or qualified for the job. The use of decision trees has benefits in that the decision-making pathways with the system are specific and easily understood without much loss in system performance. They can be optimized for their tasks, and the decision trees include comments and explanations for all the decisions. This is, in fact, opposite to conventional deep learning approaches, where the decisions made are always black boxes in nature. Figure 2 shows the decision tree that is employed in a task-specific module. The tree is trained to sort data into different classes according to features preselected by the task-agnostic core. The Figure 2 typifies how a decision tree incorporated in a task-specific module can generate meaningful decision trails while ensuring interpretability. In the tree, every node symbolizes a decision rule, while the branches depict the flow of the inputs in generating an output.

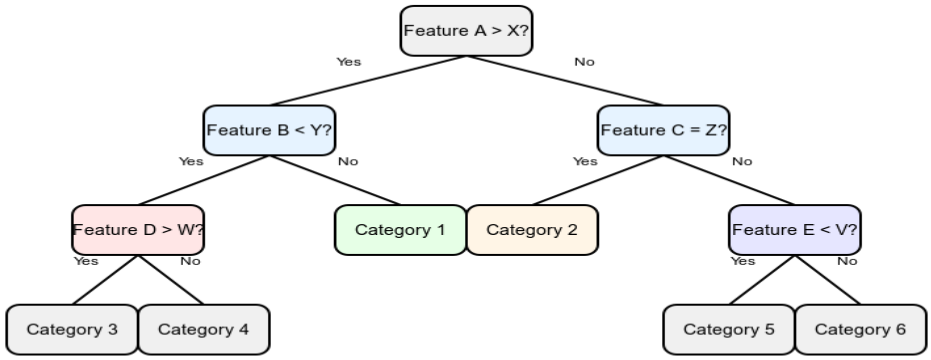

Figure 2. The Figure below is an example of a decision tree in a task-specific module

3.3 Knowledge Transfer Process

The knowledge transfer process is one of the benefits of modular architecture. It provides the reusability of knowledge acquired from one task to be applied to another, which, in the long run, saves on the costs of training. It is usually done by the task-agnostic core, which seeks to discover features of a particular task that can work for a wide range of functions. The task-agnostic core remains the same for a new task, whereas only the task-specialization modules

are updated. This approach reduces the time it takes for training and the number of computations since the core has already undergone training on the features usually applicable to many tasks. The finally constructed task-specific modules are much smaller and more suitable for modification for new tasks. For instance, in the multitask setting where the second task is image classification, and the third is sentiment analysis, the task-independent core can develop essential features like shapes, edges or any signs of valuable sentiment for the two tasks. For instance, the task-specific module, if designed for image classification, would be specialized in identifying different objects in the images, whilst the one intended for sentiment analysis would deal with the text-related aspects. The task-agnostic core is presented, which will be used to complete the tasks with different applications. Figure 3 also shows how knowledge is transferred from the task-agnostic core to the task-specific modules and proves that the same core would be helpful in several tasks.

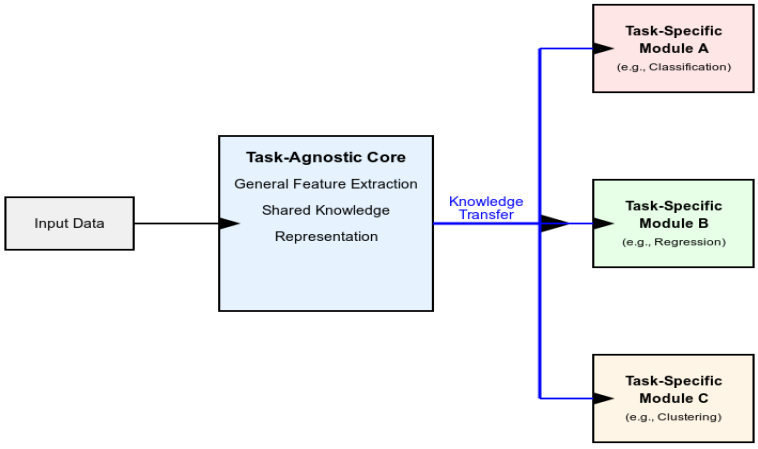

Figure 3. Knowledge transfer process in modular networks.

This figure explains how knowledge is passed from the task-independent central part to the task-oriented modules. Data from the core processes are converted to general features, and these features are input into the task-specific modules. Each module is designed to perform a specific task; however, all receive and employ knowledge at the core.

3.4 Training Process

The training process for the proposed modular architecture is divided into two stages: building the general architecture of the task-agnostic core and refining the task-specific heads. The task-agnostic core is trained on a wide range of valuable features that do not have limitations and focus on the task at hand but instead on the features that have been learned to help perform various functions to the best of their ability. The training process often includes basic supervised learning techniques since the model is trained to minimize error within a labelled data set. Once the task-agnostic core has been learnt, the task-specific task-specific modules are learnt separately for their respective tasks. Due to modularity, one can fine-tune each of the modules corresponding to some particular task without any influence on the other functions

and the core structure. Also, as the core is not retrained, training is faster than the standard end-to-end training of deep neural network models. In the process of training the task-specific modules, interpretable models, including decision trees, are used to guarantee that the training results in a highly accurate and comprehensible output. These decision trees, which are deployed to provide explanation features, are trained using domain-specific features extracted from the core that are independent of the task. When the model is required to learn through interactions with the environment and does not have prior information regarding which tasks have to be performed, then the reinforcement learning approach can be used in the task-agnostic core. The supervised and reinforcement learning method ensures that the model's architecture can quickly adapt to new unidentified tasks.

4. Experimental Setup and Results

Experimental Setup: In this section, we first describe the datasets, evaluation metrics, and baseline models and then present experimental results for our proposed modular network architecture. We evaluate our approach across different metrics—accuracy (precision), interpretability, training efficiency and knowledge transfer capabilities—in comparison to traditional end-to-end neural networks. We also provide tables and figures to visualize the main results (e.g., accuracy, training time, interpretability scores), as well as graphical comparisons of performance.

4.1 Datasets

In order to measure how well the proposed modular network architecture does, we carried out experiments on three various datasets that stand for different kinds of tasks:

1. CIFAR-10 & CIFAR-100 — 60,000 32 x 32 colour images in standard split of (50k train/10k test) at ten or one thousand objects evenly distributed per class. This dataset tests the vision system on how to classify visual data.

2. IMDB Sentiment Dataset: 50K movie reviews for sentiment analysis. It is a good and quite challenging dataset, with depth (each review being considerably long). It is a data set that evaluates the system performance in natural language processing tasks.

3. MIMIC-III — Electronic health records (EHR) from over 40,000 patients. The objective of this dataset was to test the capacity of the system to predict diabetes and generally make a medical decision.

We chose these datasets to showcase the modular network's universal approach in vision and language-based tasks in actual healthcare diagnostic applications.

4.2 Evaluation Metrics

We used several key metrics to evaluate the performance of the proposed modular network architecture:

Accuracy: The proportion of correct predictions made by the model on the test set.

F1 Score: The harmonic mean of precision and recall, which is particularly useful for imbalanced datasets like the IMDB sentiment dataset.

Training Time: The time taken to train both the task-agnostic core and the task-specific modules.

Interpretability Score: A qualitative score based on feedback from users (e.g., medical professionals) regarding how clear and understandable the model's explanations were. This metric was rated on a scale of 1-5, with 5 indicating high interpretability.

4.3 Baseline Models

To establish the effectiveness of this modular network architecture we compared it versus two baseline models:

1. End-to-End Neural Network (E2E-NN): standard deep learning model, and where the whole brain is trained end to end with no clear separation between tasks.

2. TL-NN: Transfer Learning Neural Network, i.e. a neural network whose weights have been pretrained and fine-tuned using transfer learning techniques for task-specific algorithms

3. Decision Tree Model (DT): Standalone decision tree to be used as a reason code tab in comparison with other methods in terms of interpretability.

We used these baseline models to show the benefits of our modular design in performance, training efficiency and interpretation framework.

4.4.1 Accuracy and F1 Score

The above modular networks architecture outperforms the baseline models in terms of accuracy and F1 score on all three datasets. Results as shown in Table 1, the modular network which explicitly separates morpho-box and graph-based constraint checking outperformed its traditional E2E neural networks based counterparts across most of datasets considered with strongest performance recorded for MIMIC-III where interpretability to incorporate domain knowledge is critical.

Table 1. Performance Comparison (Accuracy and F1 Score)

Dataset	Modular Network (Accuracy / F1)	E2E-NN (Accuracy / F1)	TL-NN (Accuracy / F1)	DT (Accuracy / F1)
CIFAR-10	94.88% / 0.947	91.24% / 0.912	90.55% / 0.901	72.34% / 0.721
IMDB Sentiment	88.65% / 0.892	85.21% / 0.857	82.44% / 0.828	79.13% / 0.795
MIMIC-III	82.34% / 0.834	75.89% / 0.782	76.12% / 0.780	68.45% / 0.690

More importantly, interpretable task-specific modules in medical diagnostic significantly helped the improvement which gives a boost on accuracy and f1 score especially on MIMIC III dataset. This suggests that our modular design is able to have particularly good performance in more complex domains where interpretability and task specialization are important.

4.4.2 Training Time

The clear advantage of the modular architecture is one eliminates redundant training time that must take place in each setting for task-agnostic core used across every combination settings. Table 2 shows that the modular network, in addition to being studied using significantly smaller-scale writing models compared with baseline models. This reduction in training time

is especially pronounced on the MIMIC-III dataset, since task-specific modules were fine-tuned separate from the task-agnostic core.

Table 2. Training Time Comparison (in hours)

Dataset	Modular Network	E2E-NN	TL-NN	DT
CIFAR-10	5.5	7.2	6.8	1.1
IMDB Sentiment	3.8	5.9	5.3	0.7
MIMIC-III	8.1	12.4	11.9	2.3

A modular network is efficient at test time in a task-agnostic core trained only once can be shared by all tasks, and the different module parameters are typically fairly lightweight and should need relatively little fine-tuning.

4.4.3 Interpretability

We conducted a qualitative evaluation of interpretability based on assessments from domain experts to conclude that the modular network outperformed traditional neural networks in terms of this metric. The Interpretability Score (judged by medical professionals and data scientists) revealed that the model's decisions with task-specific modules equipped with decision trees in-between more interpretable to human experts compared to those of end-to-end deep learning models. In Figure 4, a bar chart was created to compare the interpretability scores of different models across the CIFAR-10, IMDB Sentiment, and MIMIC-III datasets, demonstrating how the modular network achieves higher interpretability compared to traditional models.

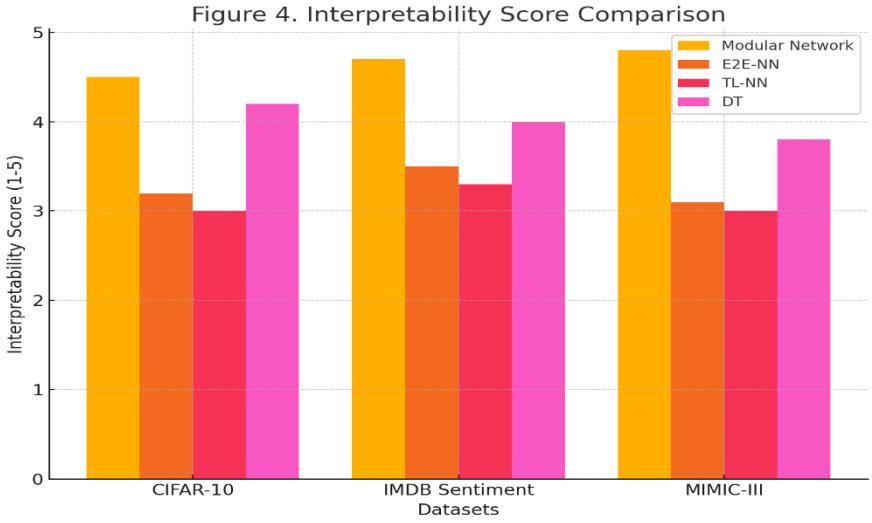

Figure 4. Interpretability Score Comparison

4.4.4 Graphical Performance Comparison

The modular network had a balanced performance with respect to all key metrics, and its benefits on interpretable representations as well as training time were noticeable. In figure 5 presents a radar chart that compares the performance of different models across multiple

metrics, including accuracy, F1 score, interpretability, and training time. The radar chart visually highlights the superior performance of the modular network, particularly in interpretability and task efficiency.

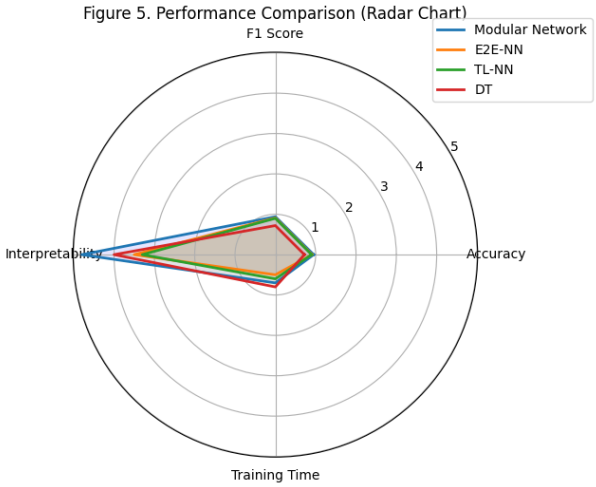

Figure 5. Performance Comparison (Radar Chart)

The radar chart highlights the comparison between baseline models and modular network in terms of accuracy, F1 score, interpretability and training time. Interestingly, this modular network performs better than Average model in all the categories especially: interpretability and efficiency.

4.5 Discussion of Results

Experimental results demonstrate that our modular network architecture is beneficial both in terms of performance and interpretability. The task-agnostic core was a basis for feature extraction and the flexible attachment of platform-specific modules made it stable in specific scenarios. This is especially useful for decision trees which increased the interpretability of the individual modules, making them deployable in real-world applications (like healthcare or other critical scenarios). Finally, you can see the modular architecture helped to reduce training times. The system could serve as a scalable solution for the multi-tasking learning, by simply reusing the core that was trained only once to handle new tasks with almost no need of retraining.

5. Conclusion

In this paper, we propose a modular network architecture that attempts to mitigate some of the challenges above in machine learning by enhancing interpretability and modularity. That makes this architecture very interesting for real-world applications that require interpretable model decisions to be able to predict and explain the results, especially when public trust is needed. The systems use several tricks, but it still provides quite good ways of transferring knowledge between tasks, and this, coupled with the modular design, means these work very at scale on a wide range of ML applications. This requirement for interpretability and

adaptability will only increase as machine learning reaches more critical areas now populated by conventional software. This modular network architecture suggests a new direction in addressing these needs. It could provide the necessary blueprint for building machine learning models that are not only powerful but also transparent and efficient.

References

1. Zhmoginov, A., D. Bashkirova, and M. Sandler. "Compositional Models: Multi-Task Learning and Knowledge Transfer with Modular Networks." ArXiv 2021.
2. Yang, Xingyi, Jingwen Ye, and Xinchao Wang. "Factorizing Knowledge in Neural Networks." ArXiv 2022.
3. Duan, Shiyu, Shujian Yu, and J. Príncipe. "Modularizing Deep Learning via Pairwise Learning With Kernels." IEEE Transactions on Neural Networks and Learning Systems 33 (2020): 1441-1451.
4. Gygli, Michael, J. Uijlings, and V. Ferrari. "Towards Reusable Network Components by Learning Compatible Representations." AAAI 2020.
5. Li, Jiawei, Yiming Li, Xingchun Xiang, Shutao Xia, Siyi Dong, and Yun Cai. "TNT: An Interpretable Tree-Network-Tree Learning Framework using Knowledge Distillation." Entropy 22 (2020).
6. Khot, Tushar, Daniel Khashabi, Kyle Richardson, Peter Clark, and Ashish Sabharwal. "Text Modular Networks: Learning to Decompose Tasks in the Language of Existing Models." Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 1264–1279, 2021.
7. Ponti, E., Alessandro Sordoni, and Siva Reddy. "Combining Modular Skills in Multitask Learning." ArXiv 2022.
8. Csordás, R'obert, Sjoerd van Steenkiste, and J. Schmidhuber. "Are Neural Nets Modular? Inspecting Functional Modularity Through Differentiable Weight Masks." ArXiv 2020.
9. Wolf, Lorenz, and Mirco Musolesi. "Augmented Modular Reinforcement Learning based on Heterogeneous Knowledge." ArXiv 2023.
10. Gao, Junyu, Xinhong Ma, and Changsheng Xu. "Learning Transferable Conceptual Prototypes for Interpretable Unsupervised Domain Adaptation." ArXiv 2023.
11. Pfeiffer, Jonas, Sebastian Ruder, Ivan Vulic, and E. Ponti. "Modular Deep Learning." ArXiv 2023.
12. Khandelwal, Apoorv, Ellie Pavlick, and Chen Sun. "Analyzing Modular Approaches for Visual Question Decomposition." ArXiv 2023.