# Scientific Background

Deep learning has witnessed unprecedented advancements in recent years, propelling artificial intelligence (AI) to new heights and enabling a wide range of applications, from computer vision and natural language processing to autonomous vehicles and healthcare diagnostics. Understanding deep learning calls the following questions to be addressed: (i) optimization—the effectiveness of gradient-based algorithms in solving neural network training programs that are non-convex and thus seemingly difficult—and (ii) generalization—the phenomenon of deep learning models that do not overfit the data, even when they have many more parameters than examples to learn from. Existing analyses of generalization typically adopt the language of classical learning theory, abstracting away many details from the considered setting. Despite the remarkable strides in the field of deep learning, a striking gap exists in our understanding of how to consistently achieve robust and effective deep model generalization. This gap in the current literature underscores the critical need for further research, development, and innovation in this area. In this grant, I will study model generalization and its relationship to the deep learning key blocks of normalization, optimization, and regularization. This endeavor promises not only to enhance our understanding of deep learning generalization but also to empower the deployment of more resilient, adaptable, and capable deep learning systems across diverse domains.

## Deep Learning Generalization

Deep model generalization is the ability of neural networks to make accurate predictions on new, unseen data. It is a crucial test for practical utility in real-world applications, requiring models to understand underlying patterns rather than just memorizing training data. Challenges include data-efficient generalization, model robustness, and domain adaptation, especially as models grow in complexity. Generalization techniques in deep learning can be divided into two categories: innovative loss functions that minimize domain-specific influences [1-4] and robust network architectures that promote generalization [5-7].

The research proposed in this grant falls into the latter category. In particular, our research delves into and introduces novel adaptive models for three key deep-learning components: data normalization, deep optimization and network regularization. These components play pivotal roles in the training process and the overall performance of deep neural networks.

**These adaptive models enhance both performance and generalizability, striking a delicate balance between model simplicity and complexity. In other words, our adaptive models will empower us to use simple models to achieve the task accuracy obtained by complex models while mitigating the risk of overfitting, which is common in complex models.**

## Data Normalization

Data normalization, which scales and centers input data, is closely linked to deep model generalization in deep neural network training in several ways:

1. **Training Stabilization**: Deep learning models, especially those with many layers, are sensitive to input feature scales and distributions. When features have differing scales, slow weight updates, convergence issues, and longer training times can occur. Data normalization equalizes input feature ranges and stabilizing training.
2. **Faster Convergence**: Normalizing data often increases convergence speed during training because optimization algorithms navigate the loss landscape more effectively when features are on a similar scale. Faster convergence enhances generalization by reducing overfitting to training noise.
3. **Overfitting Mitigation**: Data normalization helps prevent models from becoming overly complex and fitting to training noise rather than underlying patterns. Normalization reduces this overfitting by allowing models to focus on relevant patterns.
4. **Improved Gradient Flow**: Normalizing data improves gradient flow during backpropagation, reducing the risk of exploding or vanishing gradients. Improved gradient flow aids generalization by enabling better learning from training data.
5. **Transferability**: Deep models trained on normalized data often perform better on new, unseen datasets. These models learn patterns independent of input feature scale and distribution, enhancing generalization.

Various normalization techniques exist, including Batch Normalization [8], Layer Normalization [9], Instance Normalization [10], Weight Normalization [11], and Group Normalization [12]. While each has its merits and drawbacks, none are universally optimal. **In this research grant, our aim is to delve deeper into the normalization–generalization relationship, studying the impact of hybrid normalization on deep network generalization. We will then introduce a novel adaptive strategy to overcome the constraints of existing normalization methods.**

## Deep Optimization

Deep learning architectures rely heavily on adaptive and non-adaptive optimization algorithms, which improve convergence to the global minimum through techniques such as momentum, individual learning rates, and Exponential Moving Averages (EMAs). Example optimizers are Stochastic Gradient Descent (SGD), RMSprop, Adam, AdaGrad, AdamW, AdaHessian, and others that combine SGD and Adam [13–14]. These optimizers mostly depend on EMAs, which improve generalization for the following reasons:

1. **Training Stabilization**: EMA-based optimizers introduce stability into the training process by maintaining a smoothed version of the model's weights that reduces rapid fluctuations, preventing overfitting to noise and helping the identification of real data patterns.
2. **Noise Reduction**: During training, the optimization process can encounter noisy updates when batch sizes are small or the loss landscape is complex. Deep EMA-based optimizers, average parameter values over time, reducing the impact of such noise. This leads to more consistent weight updates and helps generalization by focusing on underlying data patterns.
3. **Regularization**: EMA-based optimizers act as a form of regularization. By maintaining an exponential moving average of the model's weights, they encourage the exploration of more stable regions of parameter space, which often correspond to better generalization.
4. **Memory of Past Information**: EMA-based optimizers remember past parameter values. This benefits generalization, especially on large, noisy datasets, because the model can employ useful information from earlier stages of training even as it adapts to current data.
5. **Improved Exploration-Exploitation Balance**: EMA-based optimizers help strike a better balance between exploration (trying out new weight configurations) and exploitation (refining the current best-known configuration). This can improve generalization by ensuring that the model explores a wider range of solutions before settling on a final one.

These benefits make EMA-based optimizers a valuable tool for improving the ability of deep neural networks to generalize to unseen data. However, existing EMA-based optimizers have several limitations and often struggle to quickly adapt to changing data trends during optimization. **Hence, our financial-inspired approach integrates a robust technical indicator into adaptive methods. The indicator in our method is active, affecting the weights of the network directly, in contrast to the passive role of a technical indicator in finance.**

## Network Regularization

The relationship between regularization and deep model generalization is a cornerstone in deep learning. Regularization aims to ensure accurate predictions without overreliance on specific neurons. This enhances a model's generalization performance through various mechanisms:

1. **Overfitting Prevention**: Methods like L1 and L2 regularization, dropout, and early stopping combat overfitting by imposing constraints or penalties on model parameters, discouraging excessive complexity. This focus on relevant patterns over noise enhances generalization.
2. **Smoother Decision Boundaries**: Regularization encourages models to learn smoother decision boundaries, relying on broader data patterns and trends rather than fitting data points exactly. This results in more stable and generalizable predictions on new data.
3. **Weight Pruning**: Techniques like L1 regularization induce sparsity in model parameters, reducing overfitting by eliminating unnecessary connections and enhancing generalization.
4. **Dropout as Noise Injection**: Dropout deactivates a portion of neurons during training, injecting noise and variability into the process. This makes models more robust and less prone to overfitting, as they do not overly rely on any single feature or neuron.

Regularization balances training data fitting and capturing underlying patterns, ensuring deep models perform well on unseen data. To optimize dropout regularization, some adaptive approaches have been based on prior network knowledge. For example, [15] uses SGD to guide dropout, and [16] normalizes dropout probabilities at each layer and training batch for consistent effectiveness. However, most existing dropout methods remove individual activations independently, ignoring relationships between feature maps. Adaptive dropouts like group-wise dropout [17] and spatial dropout [18] addresses this by considering feature map activation patterns. Other methods include cross-map dropout [19], region dropout [20], Gaussian approximation to adaptive dropout [21], and variational dropout [22]. Dropout training as adaptive regularization is introduced in [23], and elementwise probability for mask matrix generation is updated in [24]. The authors in [25] extended this by learning adaptive dropout rates for different neurons or neuron groups. **Nonetheless, the primary hurdle faced by these methods stems from their departure from randomness. Adaptive knowledge-based dropout aims to use various strategies to thwart co-adaptations between neurons. However, if these efforts fail, overfitting becomes likely because of the absence of randomness.**

# Research Objectives and Expected Significance

**This research aims to comprehensively study the generalization abilities of deep learning architectures. Its primary objective is to understand the underlying factors driving generalization in deep networks, enabling high accuracy in simple models without the risk of overfitting common in complex models.** To this end, we will delve into the practical implications of three core elements: data normalization, optimization, and regularization. A systematic analysis will enhance our comprehension of deep network generalization.

The expected significance of this work is multifaceted and extends across various domains, with the potential to bring about transformative advancements in the field of machine learning and AI. An expanded perspective on the key points is as follows:

1. **Improved Model Performance**: This research pioneers strategies for data normalization, optimization, and regularization to enhance deep learning model performance, increasing accuracy, reducing errors, and boosting reliability for real-world applications.
2. **Enhanced Generalization**: Research insights have the potential to revolutionize how models adapt to unseen data, making them more versatile and proficient in handling diverse datasets and cross-domain applications.
3. **Expansion of Knowledge about Generalization**: The study introduces groundbreaking concepts, unravels model implications, and explores underlying factors, demystifying deep learning and guiding the development of more interpretable and robust models.

In summary, the expected significance of this work is profound, with the potential to push the boundaries of machine learning, impacting everything from model performance and generalization to the expansion of our knowledge in this field and its widespread real-world applications. This research has the power to reshape how we approach and utilize AI, unlocking new possibilities for innovation and problem-solving. The specific research goals are as follows:

* ***AIM 1: Investigating Adaptive Strategy for Hybrid Data Normalization***

Our first objective explores the potential of **hybrid adaptive data normalization techniques integrating knowledge-based and data-driven approaches**. Diverging from conventional fixed-grouping Group Normalization (GN), our method employs the concept of channel similarity to **dynamically re-order the data**, elevating both the adaptability and generalizability of normalization. Moreover, our framework allows flexible group sizes (in contrast to conventional GN), enabling normalization to be better tailored to specific data. **In this research, we will study the impact, strengths, and potential limitations of this hybrid model on data normalization procedures.** **We will also explore the impact of our adaptive normalization on the generalization capabilities of learning models.**

* ***AIM 2: Exploring High-Order EMA Optimizers***

Building on our recent successful work introducing the Triple EMA (TEMA) optimizer [26], we strive to further **explore higher-order EMA-based optimization techniques to enhance the performance and stability of deep optimization**. This endeavor will thoroughly examine higher-order EMAs and their potential effect on the overall performance of deep learning models, with a particular emphasis on model generalizability and accuracy. Our study encompasses an evaluation of the stability of these higher-order frameworks across a diverse spectrum of datasets, architectural configurations, and training scenarios. **Key research questions include whether higher-order EMAs improve generalization because of their enhanced ability to accurately identify and track gradient trends with minimal lagging, whether there exists an optimal order beyond which generalization diminishes, and whether adaptation of the optimizer order to network characteristics (such as depth) affect generalizability.**

* ***AIM 3: Studying Multi-level Dropout for Enhanced Regularization***

Conventional dropout regularization techniques and their adaptations are routinely employed to handle feature co-adaptation within individual hidden units. Inspired by concepts like group dropout, we propose a novel approach for multi-level dropout regularization. This technique aggregates neurons into distinct groups, guided by spatial and semantic correlations among neighboring neurons, which is at knowledge-based level. We then introduce a randomness-based level by selecting a random neuron from each group. In this way, we enjoy both worlds—we improve dropout by incorporating knowledge into the procedure while mitigating overfitting by randomly selecting neurons from each group. **Key research questions include whether our multi-level regularization outperforms traditional complete random dropout and whether it enhances model generalizability.** By addressing these issues, we aim to contribute significantly to the advancement of deep learning, ultimately empowering the development of more reliable, generalizable, and adaptable deep learning models with broad applicability across various domains. We evaluate our proposed models with deep architectures whose number of parameters that is much higher than the number of training examples.

# Detailed Research Plan

## AIM 1: Investigating Adaptive Strategy for Hybrid Data Normalization

Our research introduces a novel hybrid normalization strategy that integrates domain knowledge into data-driven learning. Hybrid models are extensively employed across various domains. **However, our innovation lies in introducing a hybrid model for data normalization. Furthermore, our model employs a distinctive approach featuring a non-trivial blend of online and offline phases.** Our method is based on conventional GN, but instead of relying on fixed channel-grouping, we **leverage channel similarity to** **dynamically re-organize the channel data**. This dynamic organization both augments the efficiency of the normalization process and enhances its adaptability to diverse datasets, increasing generalizability.

### Preliminary work

Our proposed Similarity-based GN (SGN) employs **online and offline phases**. The whole channel re-clustering occurs offline, while the core of the learning procedure is performed online.

* **Offline K-means clustering and outlier exclusion:** We calculate the mean and standard deviation for each deep channel. The isolation forest algorithm is used to exclude outlier channels (using feature means and variances), followed by K-means clustering on the inlier channels. Figure *1* illustrates channel clustering in SGN, and Figure 2 presents its effects.

**Figure 1:** Proposed similarity-based group normalization approach

* **Online integration of the re-ordered channels into learning:** After establishing the ***offline*** *channel clusters*, all channels are grouped into individual clusters and re-integrated into the deep network. Training then continues until the next re-clustering procedure.

(a)

(b)

(c)

**Figure 2:** Each point indicates the mean (x-axis) and variance (y-axis) of individual channels.

(a) Original GN: channel groups are highly mixed. (b) SGN-V1. (c) SGN-V2.

Two grouping strategies are explored: *SGN-V1* employs a single mini-batch re-clustering where the channels of different images in the whole minibatch are clustered. This offers flexibility by clustering channels from different images but requires identical batch sizes in training and inference. *SGN-V2* uses single-image re-clustering, where K-means is performed on each image, enabling testing with various batch sizes and single-image applications, albeit at the potential cost of optimal channel group statistical alignment. Our experiments show SGN-V2 slightly underperforms SGN-V1 in downstream tasks. We re-cluster every 10 epochs because the weights evolve and affect image features differently as training progresses. This adaptation prevents overfitting, which can occur if the static groupings of the initial training statistics are kept.

### Preliminary results

To evaluate our technique, we first tested it on the CIFAR-100 benchmark. We employed the common ResNet50 architecture, which incorporates convolutional layers and pooling layers to mitigate overfitting. The training used 100 epochs for all models and a learning rate of 0.001. Mini-batches of 8–16 images were used. Figure 3 presents the experimental results for three normalization methods: GN, SGN-V1, and SGN-V2. SGN-V1 achieves a substantial increase in accuracy (3.15%) with respect to GN. Furthermore, SGN-V2 obtains a further improvement of 1.32% over SGN-V1. These findings underscore the advantages of SGN over conventional GN.

**Figure 3:** Comparison of test accuracy on the CIFAR-100 dataset versus number of training epochs obtained using SGN-V1, SGN-V2, and GN in the ResNet50 architecture.

### Ongoing and Future Work

In our ongoing SGN analysis research, we plan to further introduce two key innovations:

1. Variable Grouping: Unlike traditional methods with fixed group sizes, we leverage K-means to create diverse group sizes for data channels. This approach enables a more comprehensive exploration of channel grouping's impact and significance, offering valuable insights into our data dynamics by accommodating varying group sizes.
2. Dynamic Standard Deviation Recalculation: Prior to each re-clustering iteration, we calculate the standard deviation of the mean values within each group of channels. Groups with standard deviations below a predefined threshold are automatically excluded from subsequent channel re-clustering, which focuses our efforts on groups with standard deviations higher than the threshold. This strategic approach optimizes computational resources, enhancing re-clustering efficiency and reducing computational overhead.

These enhancements will bolster the generalizability and efficiency of our SGN analysis, enabling us to extract more meaningful patterns and insights from our data. To comprehensively assess the strengths and limitations of our approach, we will integrate it into various deep neural networks. We will rigorously analyze its performance across a range of publicly available benchmarks spanning both computer vision and natural language processing domains. Evaluation metrics will be tailored to specific downstream tasks, which may include but are not limited to object detection, image classification, sentiment analysis, and language translation. Following the methodology evaluations, our research endeavor raises intriguing questions:

1. Domain Knowledge and Normalization: One of the core elements we will explore is the infusion of domain knowledge into the normalization process. We are driven to understand how this infusion impacts efficiency and adaptability. Does the contextual wisdom of domain knowledge lead to more effective and context-aware normalization, potentially improving model generalization?
2. Adaptive Normalization and Data Imbalance: Our venture into adaptive normalization, marked by dynamic data re-organization and adjustable group sizes, opens the door to addressing common challenges tied to data imbalance and heterogeneous datasets. Can this approach address these challenges, thus leveling the field for machine learning algorithms?
3. Trade-offs in Normalization Strategies: As we tread into this uncharted territory, we must also consider the trade-offs involved. Are there scenarios where traditional normalization techniques, such as group normalization, still outperform our hybrid approach? In understanding these nuances, we aim to refine our strategy for the most optimal results.
4. Measuring Improved Generalization: The ultimate goal of our research is to enhance the generalization capabilities of deep learning models. We must hence assess if and to what extent the adaptability introduced by our normalization strategy improves generalization. We are keen to quantify and measure this improvement across a wide spectrum of tasks and domains, revealing the path towards more robust and versatile AI solutions.

### Pitfalls and Alternative Approaches

We minimize the risks for pitfalls by presenting very promising preliminary results. Obviously, testing our proposed method on a larger and more diverse cohort may present new challenges. In the event that the similarity-based method will not deliver satisfactory results, we will explore a random-based re-grouping approach, which we have already begun to explore. In contrast to our similarity-based approach presented above, the channels will be re-grouped randomly. The motivation for this is our assumption that generating a "mess" in the channels order every *nth* epoch will enforce the learning architecture to be more robust and strengthen its abilities.

## AIM 2: Exploring High-Order EMA Optimizers

We hold a strong conviction that the implementation of a higher-order *EMA* optimization, capable of swiftly adapting to gradient variations and trends while minimizing lag, will substantially enhance model generalization. This, in turn, will diminish our reliance on diverse learning architectures and downstream tasks.

### Preliminary work

*EMA* is widely used in deep learning optimization. It applies exponentially decreasing weighting factors to past data, smoothing out short-term fluctuations; thus, it is used to denoise data during optimization. Let be a data sequence up to time ; *EMA* recursively combines the current and previous data values by

Here, is a tuning parameter that determines its responsiveness to data changes, where lower values provide faster response but less effective denoising.

*Double EMA (DEMA)* and *Triple EMA (TEMA)* are extensions of *EMA* used in finance to assess trends [27]. They incorporate lag-correcting terms to reduce noise and identify trends while maintaining smoothness. In our recent paper [26], we introduced a novel optimizer based on *TEMA*. It is designed to accurately estimate the first and second moments of the gradient to estimate the true gradient at each time step with less bias than adaptive EMA-based methods like Adam. This is achieved by *TEMA* 's ability to reduce lag in the gradient-moment estimation while still performing effective denoising (Figure 4). The main contributions of [26] are as follows:

**Figure 4:** Simulated demonstration of gradient trend estimation and lagging. Ground truth (purple), TEMA-based estimation (blue), and EMA-based estimation (red).

* **High-order EMAs**: To our best knowledge, we were the first to exploit the strong potential of high-order EMAs to overcome the limitations of first-order EMA such as lag and insufficient adaptation to data trends. In [26], only third-order EMAs were explored. Our current aim is to investigate a series of high-order EMAs
* **Active Optimization Guidance**: Our *TEMA* optimizer is a finance-inspired technical indicator. However, in finance, such indicators are used as passive indicators, merely analyzing market changes *without* directly affecting them. By contrast, our *TEMA* optimizer *actively* guides the optimization process and affects network weights, extending its usefulness in high-dimensional optimization problems.

### Preliminary results

We have compelling evidence that the *TEMA*-based optimizer outperforms conventional EMA-based optimizers in terms of stability, robustness, generalizability, and accuracy. This finding represents a significant validation of the concepts underpinning the current grant. Our rigorous evaluation of the TEMA-based optimizer encompasses a diverse set of domains and tasks. We conducted tests on six datasets (CIFAR-10, CIFAR-100, PASCAL-VOC, MS-COCO, Cityscapes and ImageNet) using 15 architectural configurations and various computer vision tasks such as object detection, image classification, and semantic understanding. We compared our optimizer with SGD with momentum, Adam, AdamW, AdaHessian, AdaBound, and AdaGrad, and the results were averaged over two different weight initializations. The results are clear—our optimizer consistently outperforms the competitors, establishing it as a state-of-the-art optimizer for neural network models across a wide range of domains and applications.

On CIFAR-10 and CIFAR-100, our TEMA-based optimizer consistently outperformed Adam and SGD in most architectures. It also achieved a higher accuracy than the remaining optimizers. Across the CIFAR-10/100 experiments, our optimizer increased average classification accuracy by up to 16.33% and yielded smoother test-accuracy curves than standard EMA, thanks to its lower bias. On ImageNet, our optimizer obtained average improvements of 1.9% over Adam, SGD, and AdamW. On Pascal-VOC, our optimizer excelled in YOLOv5-s and YOLOv5-m, improving mAP@0.5 by 6.56% and 7.76%, respectively. On MS-COCO, our optimizer significantly outperformed SGD, Adam, and AdamW, with average improvements of 16.9% for mAP@0.5, 16.1% for Precision, 15.7% for Recall, and 15.16% for F1-score. On the Cityscapes benchmark, our optimizer enhanced the mean IoU results by 5.3%, 1.4%, and 1.9% compared with SGD, Adam, and AdamW, respectively.

**This comprehensive evaluation underscores our optimizer's generalizability and showcases its ability to provide less noisy gradient predictions. Importantly, the accuracy achieved by our simple network is on par with, if not superior to, the accuracy attained by other optimizers using more complex models from the same family (e.g., ResNet or DenseNet).**

### Ongoing and Future Work

The work of this grant proposal involves a thorough investigation of higher-order EMAs (above third order) and their potential impact on the overall performance of deep learning models, with a specific emphasis on their influence on model generalization and accuracy. Our study will comprehensively assess the stability of these higher-order frameworks across diverse datasets, architectural configurations, and training scenarios. We are particularly interested in exploring several intriguing research questions:

* Do higher-order EMAs exhibit improved generalization abilities owing to their enhanced capacity to accurately identify and track gradient trends?
* Is there a specific order beyond which the generalization performance starts to diminish, providing insights into the optimal order for practical use?
* Should the selection of the order be tailored to the characteristics of the network, such as depth, to maintain consistent generalizability across various network architectures?

**Through this research, we aim to not only expand EMA-based optimization but also provide insights into the selection of the appropriate EMA order, ultimately advancing our understanding of optimization techniques and their impact on model generalization.**

While the simple in (1) denoises the data effectively, it also introduces lag in the estimation, which may result in inadequate gradient updates and suboptimal performance. To reduce lag, and add lag correcting terms to (see [26] for details).

In this research, we will introduce and explore the generalization of *EMA* to a general order *EMA*, called . Here, denotes recursive applications of *EMA* on . We define

 (2)

 (3)

Let , and for *k=1,2,3*…

 (4)

 can be seen as a -order lag correction operator. The calculation below shows that

(5)

Higher-order *KEMAs* sacrifice smoothness in exchange for more aggressive lag reduction. Thus, in practice, the order should be selected to balance reducing the lag and denoising the gradients. The appropriate order highly depends on the problem at hand. Evidently, a larger *k* entails larger coefficients in the expansion for , which may increase sensitivity to hyperparameter tuning. This affects the choice of order. Because the *EMA* operator is linear, we may w.l.g. expand and as the sums

By definition

(9)

Defining for convenience and plugging in the last sum into (3) gives

Matching the coefficients of in (11) to that in (7) gives the recursive equation,

Recalling that by definition and , the solution to (12) is as in (5), since

To show the expansion of in (6), we have

Matching the coefficients of to that in (10), we obtain the recursive equation,

Since and by definition , we obtain the solution in (6) since

Using the developed equations, we will study the performance of high-order optimizers while carefully considering their stability. We expect high-order EMAs to be a bit unstable because the scale of increases as the order increases*.* We will study the generalization of high-order *EMAs* and other *EMA-*based methods in terms of architecture and downstream tasks.

### Pitfalls and Alternative Approach

We have introduced promising preliminary results. Scaling up to a larger and more diverse dataset may introduce complexities; however, we stand well-prepared to tackle them directly. Should our proposed method fall short of expectations, we have contingency plans. These include the introduction of a new family of advanced high-order equations (we outlined a specific family in this proposal) tailored to better match the specific data under analysis. Moreover, our proposal aims to study the effects of higher-order optimizers, but our findings already demonstrate that *TEMA* outperforms the original *EMA*. If additional higher-orders optimizers do not further enhance *TEMA's* performance, this would be a legitimate conclusion, addressing a key research question regarding the optimal order that can be applied.

## AIM 3: Studying Multi-level Dropout for Enhanced Regularization

Expanding upon the notion of adaptive dropout, our research introduces an innovative adaptive model that leverages both knowledge and randomness to accommodate varying data characteristics. **This research aims to shed light on the nuanced interplay between regularization techniques and model generalization, exploring the inherent strengths of knowledge in contrast to full randomness or their combination.**

We employ a hybrid framework that combines Convolutional Neural Networks (CNNs) with the Neural Additive Model (NAM) [28] to better distill the contribution of each individual neuron. Our current architecture utilizes ResNet-34 as the CNN structure. We seamlessly integrate the NAM model in the middle layer of ResNet, which serves a dual purpose:

1. Individual Neuron Analysis: NAM helps dissect individual neuron contributions, offering deeper insights into model behavior.

2. Parameter Efficiency: NAMs reduce the number of network parameters, making computations more efficient for quicker training and inference.

After applying the CNN-NAM approach, we will employ a multi-level analysis: 1) a **similarity-based** clustering according to the significance of individual neurons for the given task and their prominence within a particular feature map, and 2) a **random** selection of a predetermined percentage of neurons from each cluster to undergo dropout in each training epoch. These chosen neurons (from each group) will be deactivated.

**Our first hybrid knowledge-random dropout approach ensures that dropout is not fully random but is also based on knowledge of the data.** We will explore the following aspects:

* Knowledge-Enhanced Generalization: To what extent does the knowledge-driven grouping and dropout strategy enrich the model's performance by preserving crucial feature representations? Does this approach lead to more consistent and superior generalization than conventional random dropout, and if so, in what scenarios or network architectures is this effect most pronounced?
* Stability and Overfitting Mitigation: Does the randomness carefully introduced within the adaptive group dropout framework effectively mitigate overfitting to the training set? How does this approach balance model stability and the risk of increased overfitting? Can it enhance the model's resistance to noise and perturbations in real-world data?
* Optimal Grouping Strategies for higher Generalization: What are the key factors influencing the success of different neuron grouping strategies? Are there specific spatial or semantic relationships among neurons that are particularly conducive to robust generalization? How do these strategies adapt to varying dataset characteristics and network complexities?
* Transferability across domains and tasks: To what degree can the benefits of adaptive group dropout be transferred across different domains, datasets, and tasks? Does this technique offer a transferable improvement in the generalization ability of deep learning models, and are there limitations to its applicability in specific contexts?

**By addressing these research questions, we aim to both advance the understanding of adaptive group dropout and provide actionable insights into its role as a regularization tool for enhancing model generalization.** We aspire to unravel the intricate interplay between knowledge-driven dropout strategies, model stability, overfitting mitigation, and the broader context of deep learning, contributing to the evolution of techniques that empower deep neural networks to generalize effectively across diverse datasets and real-world scenarios.

### Preliminary results

Preliminary findings reveal a noteworthy 1.5% enhancement in classification accuracy on CIFAR-100 using ResNet34. These results also indicate a substantial 92% reduction in both the number of parameters and computational load within the learning architecture. To put this into perspective, a standalone ResNet-34 configuration typically requires 21.8M parameters but integrating it with NAM results in a leaner 1.8M parameter architecture. These preliminary findings pertain to the application of the combined CNN-NAM network and were achieved without employing the multi-level approach *yet*, which involves grouping and random choice. Based on the robust performance of our similarity-based normalization techniques and findings in [17], which unequivocally demonstrate the positive impact of similarity-based knowledge, we hold the strong conviction that our proposed suggestions have the potential to further enhance performance. These aspects will be explored in this grant.

### Pitfalls and alternative approaches

**Bibliography**

1. Dou, Q., Coelho de Castro, D., Kamnitsas, K., & Glocker, B. (2019). Domain generalization via model-agnostic learning of semantic features. Advances in neural information processing systems, 32.‏
2. Li, Y., Gong, M., Tian, X., Liu, T., & Tao, D. (2018, April). Domain generalization via conditional invariant representations. In Proceedings of the AAAI conference on artificial intelligence (Vol. 32, No. 1).‏
3. Motiian, S., Piccirilli, M., Adjeroh, D. A., & Doretto, G. (2017). Unified deep supervised domain adaptation and generalization. In Proceedings of the IEEE international conference on computer vision (pp. 5715-5725).‏
4. Muandet, K., Balduzzi, D., & Schölkopf, B. (2013). Domain generalization via invariant feature representation. In International conference on machine learning (pp. 10-18). PMLR.‏
5. D’Innocente, A., & Caputo, B. (2019). Domain generalization with domain-specific aggregation modules. In Pattern Recognition: 40th German Conference, GCPR 2018, Proceedings 40 (pp. 187-198). Springer International Publishing.‏
6. Li, D., Yang, Y., Song, Y. Z., & Hospedales, T. M. (2017). Deeper, broader and artier domain generalization. In Proceedings of the IEEE international conference on computer vision (pp. 5542-5550).‏
7. Matsuura, T., & Harada, T. (2020). Domain generalization using a mixture of multiple latent domains. In Proceedings of the AAAI conference on artificial intelligence (Vol. 34, No. 07, pp. 11749-11756).‏
8. Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning (pp. 448-456). PLMR.‏
9. Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. arXiv preprint arXiv:1607.06450.‏
10. Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2016). Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022.‏
11. Salimans, T., & Kingma, D. P. (2016). Weight normalization: A simple reparameterization to accelerate training of deep neural networks. Advances in neural information processing systems, 29.‏
12. Wu, Y., & He, K. (2018). Group normalization. In Proceedings of the European conference on computer vision (ECCV) (pp. 3-19).‏
13. Keskar, N. S., & Socher, R. (2017). Improving generalization performance by switching from adam to sgd. arXiv preprint arXiv:1712.07628.‏
14. Landro, N., Gallo, I., & La Grassa, R. (2020). Mixing Adam and SGD: A combined optimization method. arXiv preprint arXiv:2011.08042.‏
15. Keshari, R., Singh, R., & Vatsa, M. (2019). Guided dropout. In Proceedings of the AAAI conference on artificial intelligence (Vol. 33, No. 01, pp. 4065-4072).‏
16. Wang, S., Zhou, T., & Bilmes, J. (2019, May). Jumpout: Improved dropout for deep neural networks with ReLUs. In International conference on machine learning (pp. 6668-6676). PMLR.‏
17. Ke, Z., Wen, Z., Xie, W., Wang, Y., & Shen, L. (2020). Group-wise dynamic dropout based on latent semantic variations. In Proceedings of the AAAI conference on artificial intelligence (Vol. 34, No. 07, pp. 11229-11236).‏
18. Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C. (2015). Efficient object localization using convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 648-656).‏
19. Poernomo, A., & Kang, D. K. (2018). Biased dropout and crossmap dropout: Learning towards effective dropout regularization in convolutional neural network. Neural networks, 104, 60-67.‏
20. Choe, J., Han, D., Yun, S., Ha, J. W., Oh, S. J., & Shim, H. (2021). Region-based dropout with attention prior for weakly supervised object localization. Pattern recognition, 116, 107949.‏
21. Wang, S., & Manning, C. (2013). Fast dropout training. In international conference on machine learning (pp. 118-126). PMLR.
22. Kingma, D. P., Salimans, T., & Welling, M. (2015). Variational dropout and the local reparameterization trick. Advances in neural information processing systems, 28.‏
23. Wager, S., Wang, S., & Liang, P. S. (2013). Dropout training as adaptive regularization. Advances in neural information processing systems, 26.‏
24. Ba, J., & Frey, B. (2013). Adaptive dropout for training deep neural networks. Advances in neural information processing systems, 26.‏
25. Zhuo, J., Zhu, J., & Zhang, B. (2015, July). Adaptive Dropout Rates for Learning with Corrupted Features. In Proceedings international joint conference on artificial intelligence (pp. 4126-4133).‏
26. Peleg, R., Weiss, R., & Hoogi, A. (2023). Leveraging the triple exponential moving average for fast-adaptive moment estimation. arXiv preprint arXiv:2306.01423.‏
27. Mulloy, P. G. (1994). Smoothing data with faster moving averages. Stocks & commodities, 12, 11-19.
28. Agarwal, R., Melnick, L., Frosst, N., Zhang, X., Lengerich, B., Caruana, R., & Hinton, G. E. (2021). Neural additive models: Interpretable machine learning with neural nets. Advances in neural information processing systems, 34, 4699-4711.‏