SymBa: Symmetric Backpropagation-Free Contrastive Learning with Forward-Forward Algorithm for Optimizing Convergence

Heung-Chang Lee[*†], **Jeonggeun Song**[*†]
Kakao Enterprise
Seongnam-si, Republic of Korea
andrew.com@kakaoenterprise.com; po.ai@kakaoenterprise.com

Abstract

The paper proposes a new algorithm called SymBa that aims to achieve more biologically plausible learning than Back-Propagation (BP). The algorithm is based on the Forward-Forward (FF) algorithm, which is a BP-free method for training neural networks. SymBa improves the FF algorithm's convergence behavior by addressing the problem of asymmetric gradients caused by conflicting converging directions for positive and negative samples. The algorithm balances positive and negative losses to enhance performance and convergence speed. Furthermore, it modifies the FF algorithm by adding Intrinsic Class Pattern (ICP) containing class information to prevent the loss of class information during training. The proposed algorithm has the potential to improve our understanding of how the brain learns and processes information and to develop more effective and efficient artificial intelligence systems. The paper presents experimental results that demonstrate the effectiveness of SymBa algorithm compared to the FF algorithm and BP.

1 Introduction

In recent years, deep learning has made remarkable strides in various research domains. By utilizing stochastic gradient descent with a large number of parameters and abundant data, deep learning has achieved state-of-the-art results. This success has sparked interest in investigating the learning mechanisms employed by biological systems, particularly the brain. Researchers are curious to explore whether the mechanisms used in deep learning have any resemblance to those employed in the brain. However, despite significant efforts, the Back-Propagation (BP) algorithm [14], which is a commonly used technique in deep learning, is still not considered a plausible model for how the cortex learns.

This has motivated researchers to explore alternative theories of how the brain learns and processes information. Recent research has focused on developing biologically plausible neural networks that can learn in a manner similar to the brain. These models have the potential to improve our understanding of how the brain works and to develop more effective and efficient artificial intelligence systems.

Our paper introduces SymBa algorithm, which achieves more human-like training by avoiding BP. Based on the preprint of the Forward-Forward (FF) algorithm [9], SymBa algorithm stabilizes converging behaviors and improves overall performance.

[*] Equal contribution
[†] Corresponding authors

Preprint. Under review.

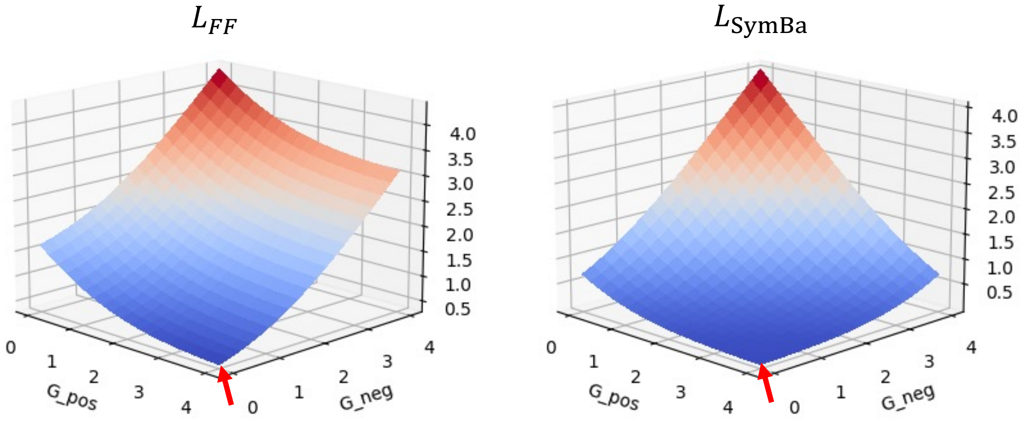

Figure 1: **Comparison of L_{FF} and L_{SymBa} over G_{pos} and G_{neg}.** The direction of convergence is that G_{pos} increases continuously, while G_{neg} decreases to 0. In order to clarify, the red arrows in the figure represent the points of convergence. In the case of L_{FF}, there is a significant gap of gradient scales depending on the choice of initial points. Conversely, L_{SymBa} exhibits precise symmetry across all the values of G_{pos} and G_{neg}. In contrast to L_{FF}, L_{SymBa} converges along the same slopes in most cases regardless of the choice of initial point.

The contributions of our algorithm can be summarized as follows.

- The equations of FF algorithm are difficult to converge towards the global minima due to conflicting converging directions for loss of positive and negative samples. Our approach addresses this problem by ensuring that both gradients converge in the same direction, resulting in improved and efficient convergence during training.
- To accomplish the classification task and apply the overlay, the previous FF Algorithm conducted one-hot encoding of class information on the input picture. However, this approach has a disadvantage in that it can cause the class information to be lost during training, resulting in poor performance. To address this issue, we modify the FF algorithm by introducing Intrinsic Class Pattern (ICP) containing class information behind each channel. This modification prevents the class information from being lost during training, improving the overall performance of the algorithm.

2 Related Works

There has been a significant amount of research on developing biologically plausible neural networks that can learn in a manner similar to the brain. One such approach is the Hebbian learning rule, which was from Donald Hebb's 1949 book [8]. Hebb proposed that when two neurons are repeatedly activated at the same time, the strength of the connection between them increases. This idea, known as Hebb's rule, has since been widely studied and extended in many different contexts. Another approach is spike-timing-dependent plasticity (STDP), which modifies the synaptic strengths between neurons based on the relative timing of their spikes [3]. STDP has been used to train spiking neural networks, which have been shown to be computationally efficient and capable of solving a wide range of tasks [16].

Additionally, the popular method for estimating the parameters of a probabilistic model is Noise Contrastive Estimation (NCE), which was introduced by Gutmann and Hyvärinen [6]. NCE has several advantages, including its simplicity and efficiency, as it only requires the computation of simple logistic regression. It has been applied in a variety of contexts, including learning binary codes for image retrieval [13], training deep neural networks for high-dimensional data modeling [2]. However, NCE also has some limitations, such as the requirement for an explicit noise distribution and sensitivity to the quality of the noise distribution.

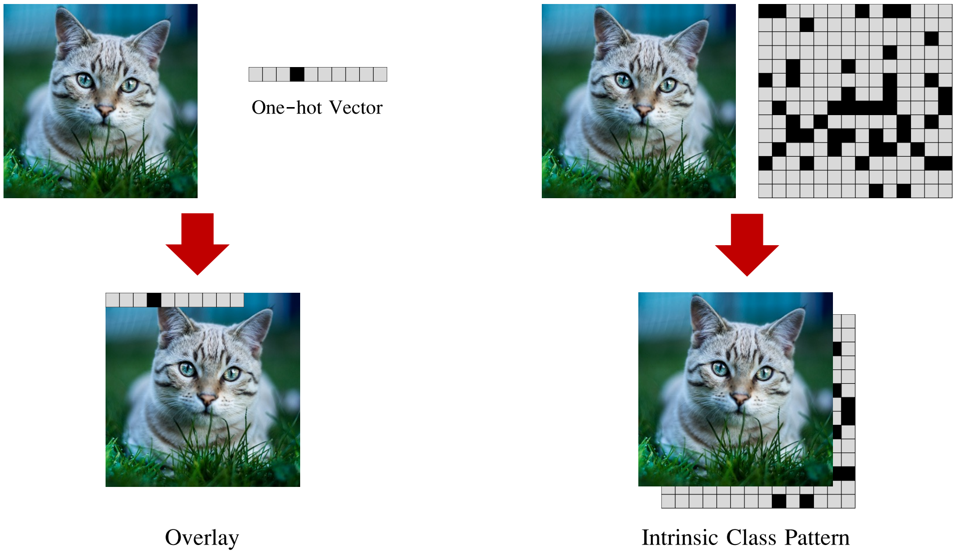

Figure 2: **Intrinsic class pattern.** When labels are overlaid directly onto input images, there is a significant loss of representations. (i.e. The edge of the cat's right ear in the above image.) In contrast, intrinsic class patterns, which are randomly generated unique fixed patterns for each class, allow the model to recognize the labels by identifying the corresponding class pattern. By avoiding the loss of input pixels, intrinsic class patterns provide more useful information about the classes without wasting the model's capacity to conjecture obscured parts of the input.

Recent work has also focused on developing biologically plausible activation functions for neural networks. One such function is the rectified linear unit (ReLU), which has been shown to be more biologically plausible than traditional sigmoid activation functions [4]. Despite the success of these biologically inspired approaches, the Back-Propagation (BP) algorithm, which is commonly used in deep learning, is still not considered a plausible model for how the cortex learns. This has motivated researchers to explore alternative theories of how the brain learns and processes information. In this context, the Forward-Forward (FF) algorithm has been proposed as a more biologically plausible alternative to BP. The FF algorithm avoids using BP by training each layer independently to maximize the goodness of positive samples and minimize that of negative samples. Our paper builds on the FF algorithm and introduces SymBa algorithm, which improves the overall performance of the FF algorithm by balancing the positive and negative losses during training. This loss of FF algorithm is similar to the contrastive loss. [1, 7, 5]

3 Method

The FF algorithm does not rely on BP to emulate the function of the human brain; instead, each layer is trained independently to maximize the goodness of positive samples and minimize that of negative samples, while the computational graph does not connect any of the layers.

Nevertheless, in the original implementation, these samples frequently cause asymmetric gradients, which result in suboptimal performance and delayed convergence. To improve the algorithm's performance, it is necessary to balance the positive and negative losses during training. It has been demonstrated that balancing the losses greatly enhances the efficiency and convergence speed of the FF algorithm.

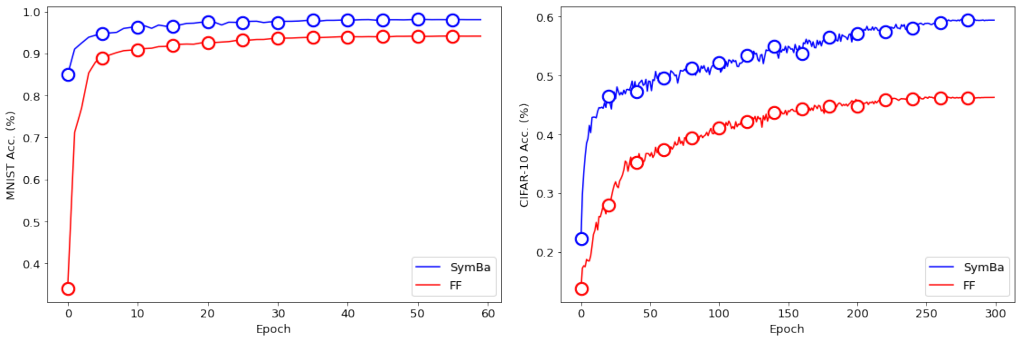

Figure 3: **The measurement of converging speed.** The graph depicts the change in the accuracy of FF and SymBa by epochs. The accuracy converges quickly in both cases, but SymBa does so considerably more rapidly than the FF. In particular, SymBa method significantly outperforms the FF algorithm in the MNIST dataset [12] 0 epoch, and it continuously surpasses the FF algorithm in the CIFAR-10 dataset [11].

3.1 Imbalance of Positive-Negative Losses

The previous implementation of the FF algorithm utilizes the Noise Contrastive Estimate (NCE) loss function. The original loss function of the FF algorithm is as follows.

$$G_{\text{pos}} = \sum_j y^2_{\text{pos},j}, G_{\text{neg}} = \sum_j y^2_{\text{neg},j} \tag{1}$$

$$L_{\text{pos}} = \log\left(1 + e^{\theta - G_{\text{pos}}}\right), L_{\text{neg}} = \log\left(1 + e^{G_{\text{neg}} - \theta}\right) \tag{2}$$

$$L_{\text{FF}} = L_{\text{pos}} + L_{\text{neg}} \tag{3}$$

Note that θ is threshold. When the domain of the goodness G is the set of all real numbers, the above formula is symmetric about G. However, since G must be greater than 0, the positive and negative losses have different convergence behaviors, leading to imbalanced gradients during the training and slowing down the convergence.

$$\nabla L_{\text{pos}} = -\frac{2e^{\theta - G_{\text{pos}}}}{1 + e^{\theta - G_{\text{pos}}}} \sum_j y_{\text{pos},j} \nabla y_{\text{pos},j} \tag{4}$$

$$\nabla L_{\text{neg}} = \frac{2e^{G_{\text{neg}} - \theta}}{1 + e^{G_{\text{neg}} - \theta}} \sum_j y_{\text{neg},j} \nabla y_{\text{neg},j} \tag{5}$$

The gradients for positive and negative losses can be expressed as Eq (4). Due to the condition that $G > 0$, $G_{\text{pos}} \to \infty$ and $G_{\text{neg}} \to 0$ as the training progresses. Consider ϵ and Ω as sufficiently small and large numbers, respectively. Substituting them to G_{neg} and G_{pos} each, we can observe varying outcomes in the vicinity of the convergence point.

$$\nabla L_{\text{pos}} = -\frac{2e^{\theta - \Omega}}{1 + e^{\theta - \Omega}} \sum_j y_{\text{pos},j} \nabla y_{\text{pos},j} \approx 0 \tag{6}$$

$$\nabla L_{\text{neg}} = \frac{2e^{\epsilon - \theta}}{1 + e^{\epsilon - \theta}} \sum_j y_{\text{neg},j} \nabla y_{\text{neg},j} \approx \frac{2e^{-\theta}}{1 + e^{-\theta}} \sum_j y_{\text{neg},j} \nabla y_{\text{neg},j} \tag{7}$$

It shows that the gradient for positive and negative samples behave differently, toward the end of the training process. Furthermore, the final convergence values of positive and negative losses are different. The presence of this discrepancy impedes the model's ability to attain global minima.

$$G_{\text{pos}} \to \infty \Leftrightarrow L_{\text{pos}} \to 0 \tag{8}$$

$$G_{\text{neg}} \to 0 \Leftrightarrow L_{\text{neg}} \to \log\left(1 + e^{-\theta}\right) \tag{9}$$

In order to obtain the proper convergence properties, we introduce an alternative algorithm, SymBa.

Table 1: **The results on MNIST.** This table indicates the result of experiments using the BP, FF, and SymBa algorithms while varying the number of layers and channels on the MNIST dataset. The labeling method and hyper-parameters suitable for each algorithm were used. Forthermore, the best test error values are bolded, only the outcomes of our suggested SymBa algorithm are shaded

#Layers	#Channels	Algorithm	Labeling Method	Details	Test Error(%)
2	500	BP	None		1.74
		FF	Overlay	$\theta = 2.0$	6.10
		SymBa	ICP	$\alpha = 4.0$	**1.65**
	2000	BP	None		1.56
		FF	Overlay	$\theta = 2.0$	6.93
		SymBa	ICP	$\alpha = 4.0$	**1.52**
3	500	BP	None		**1.77**
		FF	Overlay	$\theta = 2.0$	5.79
		SymBa	ICP	$\alpha = 4.0$	**1.77**
	2000	BP	None		1.58
		FF	Overlay	$\theta = 2.0$	6.59
		SymBa	ICP	$\alpha = 4.0$	**1.42**

3.2 Balanced Contrastive Loss for Equilibrium of Positive-Negative Losses

To address the asymmetric nature of the original implementation, we propose an alternative loss function as follows.

$$\Delta = G_{\text{pos}} - G_{\text{neg}} \tag{10}$$

$$L_{\text{SymBa}} = \log\left(1 + e^{-\alpha\Delta}\right) \tag{11}$$

where α is a simple scale factor. Our new loss function has various benefits for enhancing the stability of training. To begin with, it eliminates the requirement to consider the equilibrium between G_{pos} and G_{neg}. As L_{SymBa} solely depends on the discrepancy between two losses, it is inherently symmetric.

$$G_{\text{pos}} \to \infty, G_{\text{neg}} \to 0 \Leftrightarrow L_{\text{SymBa}} \to 0 \tag{12}$$

Furthermore, it utilizes the explicit relation of G_{pos} and G_{neg}. It enables the model to discern the correlation between the two quantities along the batch dimension. It facilitates the model to infer the association between two sets of samples.

For quantitative analysis, we conducted experiments on the MNIST, CIFAR-10, and CIFAR-100 datasets. The results showed that our proposed algorithm outperformed the existing FF and BP algorithms on all datasets we experimented on. Moreover, it exhibited significantly better performance in terms of convergence speed than the existing FF algorithm. While the previous algorithm converges towards the end of the training, our algorithm rapidly approached the optimal performance after only a few initial epochs.

3.3 Intrinsic Class Patterns as Labels

Since the FF algorithm employs contrastive loss, a fine-tuning process is required using the BP algorithm to evaluate its performance. In the original paper, the label information is directly injected into the input images by overlaying one-hot encoding. Rather than relying on a classifier, accuracy can be evaluated by measuring the goodness of each label and selecting the label with the highest score, as determined by $\text{argmax}_y G(x, y)$. This approach is both clever and valid, as it avoids the need to utilize BP during training.

However, the input information is compromised by overlaying the labels. The one-hot encoding representing the labels partially obscure the input images. Since the size of the one-hot encoding is determined by the number of classes, a significant portion of the images is removed as the number of classes increases. For instance, in CIFAR-100 dataset with 100 classes, each label covers approximately 9.77% of the image. It is much higher than the 1.27% coverage of labels in MNIST

Table 2: **The results on CIFAR-10 and 100.** These experiments are the results of the BP, FF, and SymBa algorithms on the CIFAR-10 and 100 datasets by adjusting the number of channels. Since CIFAR datasets contain more diverse images, three fully-connected layers are used for all cases to obtain enough model capacity. For a fair comparison, the whole experiments are conducted in the exact same environment as the MNIST experiment, and the best-performing results are bolded and the SymBa algorithm is colored to increase visibility.

Dataset	#Channels	Algorithm	Labeling Method	Details	Test Error(%)
CIFAR-10	2000	BP	None		42.66
		FF	Overlay	$\theta = 2.0$	49.32
		SymBa	ICP	$\alpha = 4.0$	**41.23**
	3072	BP	None		43.14
		FF	Overlay	$\theta = 2.0$	49.63
		SymBa	ICP	$\alpha = 4.0$	**40.91**
CIFAR-100	3072	BP	None		71.12
		FF	Overlay	$\theta = 2.0$	81.85
		SymBa	ICP	$\alpha = 4.0$	**70.72**

dataset so cannot be neglected. Furthermore, even if the number of classes is negligibly small, it cannot be guaranteed that its negative impact is negligible as well.

To address this problem, we propose an alternative method to incorporating class labels into input images, which is called **Intrinsic Class Patterns**. Rather than overlaying one-hot encoding, we generate a non-trainable random discrete pattern for each class, which is concatenated to input channels. This approach avoids directly covering images, thus preserving the original input information.

In our experiments, incorporating intrinsic class patterns in the inputs results in improved performance for both the original FF algorithm and our proposed algorithm on the CIFAR-10 and CIFAR-100 datasets.

4 Experiments

As discussed in the method session, we want to conduct experiments in order to determine whether our SymBa method can outperform BP and converge more efficiently than FF on a variety of datasets, including MNIST, CIFAR-10, and 100. Our experimental environment was based on the Classification Task for Supervised Learning. The accuracy criterion for BP was if the highest value of the final Softmax previously employed corresponded to the correct answer class, while the accuracy criterion for FF and SymBa was whether the class with the most goodness used in FF corresponds to the correct answer class.

Several hyper-parameters were left unmentioned in the paper of FF, but we made our best effort to replicate FF. Most of the hyper-parameters described in the original paper were utilized, while the unknown hyper-parameters were discovered via various experiments. Additionally, we conducted the whole experiment with essential settings which does not utilize regularization like weight decay[**?**] or the drop-out method [15] to compare fairly. Nonetheless, we employed basic augmentation methods such as random cropping and random horizontal flipping in all experiments on CIFAR-10 and 100.

4.1 Experiments with MNIST

As described in the paper, we trained architectures with 2 or 3 fully connected layers on the MNIST dataset. Since the other hyperparameters were not given, we selected them manually by conducting several experiments. All models were trained using Adam optimizer[10] for 120 epochs with the batch size of 4096, and we swept the learning rates in $\{0.001, 0.01\}$ range. The scale factor for L_{SymBa}, α, did not affect the performance significantly when it was set to any number within the

Table 3: **Ablation study between Overlay and ICP.** As mentioned in Sec. 3.3, both FF and SymBa achieved higher performance by replacing one-hot encodings with intrinsic class patterns. This substitution allowed for more effective learning, as the models could better capture the inherent characteristics of each class.

Dataset	#Channels	Algorithm	Labeling Method	Test Error(%)
CIFAR-10	3072	FF	Overlay	49.63
			ICP	**48.83**
		SymBa	Overlay	41.23
			ICP	**40.91**

range of $\{1.0, 4.0\}$. It made the difference less than 0.1% only, for the experiments on both MNIST and CIFAR datasets. Therefore, we reported the results of $\alpha = 4.0$ cases for all experiments.

In our experiments, we observed that SymBa exhibits noticeably faster convergence than FF from the first epoch onwards. (See Figure 3.) Furthermore, The performance gap between SymBa and FF is maintained until the final epoch, resulting in higher final performance. We anticipate that it is closely related to the stabilization of the converging curve discussed above. As shown in Table 1, we experimented with BP, FF, and SymBa under various setups, and SymBa consistently achieved the best performance across all setups.

4.2 Experiments with CIFAR-10 and 100

In contrast to the MNIST dataset, CIFAR datasets contain more complex features and color information. Furthermore, for the case of CIFAR-100 dataset, it has a large number of classes and fewer data per class. Since the images in CIFAR datasets contain more detailed representations, we employed models with three fully-connected layers to ensure enough model capacity. For the other experimental setups, we utilized substantially identical configurations to those used in the MNIST experiments. As shown in Table 2, our experiments show that SymBa algorithm outperforms in all experiments, while FF algorithm shows the lowest performance. It claims that SymBa can be trained better than BP on more complicated tasks.

4.3 Ablation Study

Ablation study between Overlay and ICP. We conducted an ablation study on the FF and SymBa algorithm for overlaying one-hot encoding, which was employed in the current FF, and ICP to investigate the impact of ICP. The dataset was conducted on CIFAR-10, and the channel used in the experiment session was 3072. ICP outperformed the overlay for all algorithms, as indicated in Table 3, while the SymBa algorithm outscored all FF algorithms even for the overlay.

Ablation study on ICP rate. The ablation study is conducted to find the optimal rate of ICP, which is one of the contributions of the paper. The values varied from 0.1 to 0.8, with 0.8 being a noisy input and 0.1 denoting a sparse one. The yellow dots in the upper pictures from Figure 4 are noise, and the purple dots are input. The graph in Figure 4 demonstrates that the accuracy of the CIFAR-10 improves as the noise level lowers. Nevertheless, if it is smaller than 0.1, the loss converges to NaN since it is impossible to extract class-specific information on its own. As a consequence, we established that 0.1 was the ideal rate for ICP and utilized this rate in all of our experiments.

5 Discussion and conclusion

Back-propagation (BP) algorithm has been the de facto standard for training deep learning models for a long time. Despite its remarkable performance, several studies have suggested that it is not suitable for emulating the way the brain learns in reality. Additionally, BP requires significant computing power and vast amounts of data to learn knowledge of unknown domains, making it inefficient compared to the human brain, which can adapt flexibly to various areas with only a few samples.

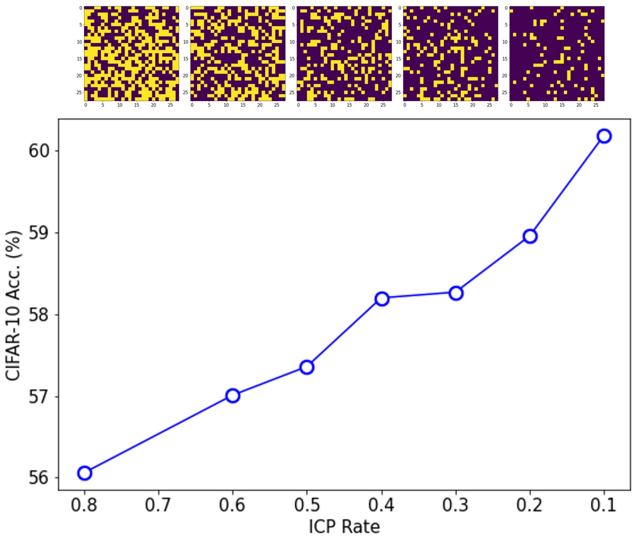

Figure 4: **Ablation study on ICP rate.** The class patterns are generated by selecting random pixels from black images and assigning them a value of 1. ICP rate refers to the sampling rate used to fill in these patterns, and our observations indicate that the class patterns with low ICP rates facilitate the models learning of class differences. As sparse patterns render the distinctions between classes more discernible, the final performance demonstrates a near-linear relationship with sparsity, as depicted in the graph.

Our proposed algorithm, SymBa, aims to mimic the learning process of the human cortex without relying on BP. It builds upon Forward-Forward (FF) algorithm, in which gradients are computed internally within each layer by estimating the contrastive loss between two forward passes of positive and negative inputs, rather than propagating information between layers. Although FF algorithm was a promising approach, issues arose, such as imbalanced losses near convergence and the loss of input information due to the overlaying of one-hot encodings on input images for evaluation without fine-tuning.

In SymBa, we address these limitations by combining the positive and negative losses into a unified loss, achieving a perfect balance while preserving the contrastive properties of the original algorithm. Additionally, we concatenate intrinsic noise patterns for each class rather than obscuring input images, thus conserving their representations. As a result, SymBa outperforms the BP algorithm on widely-used benchmark datasets, such as MNIST, CIFAR-10, and CIFAR-100.

As an extension of various approaches to reproduce the behaviors of the brain, the results of SymBa outperform those of BP with a significant margin. However, BP has been widely used in a vast number of tasks, while there exist many different tasks for SymBa to prove its ability. While SymBa is not be a perfect replacement for BP, there is potential for further development towards pioneering more human-like algorithms than BP.

References

[1] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[2] A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun. Logarithmic time memory complexity for training deep neural networks. In *Advances in Neural Information Processing Systems*, pages 2627–2635. Curran Associates, Inc., 2015.

[3] W. Gerstner and W. M. Kistler. *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press, 2002.

[4] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings, 2011.

[5] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33: 21271–21284, 2020.

[6] M. Gutmann and A. Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

[7] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[8] D. O. Hebb. The organization of behavior. 1949.

[9] G. Hinton. The forward-forward algorithm: Some preliminary investigations. *arXiv preprint arXiv:2212.13345*, 2022.

[10] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[11] A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[13] A. Mnih and R. Salakhutdinov. Learning binary codes for high-dimensional data using bregman divergences and sparse approximation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(5):1044–1056, 2012.

[14] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[15] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15 (1):1929–1958, 2014.

[16] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida. Deep learning in spiking neural networks. *Neural networks*, 111:47–63, 2019.