Exploring Few-Shot Performance of Self-Supervised Visual Representations

Samir Char, Shivani Modi, Saloni Ajay Gupta

Columbia University

Learning representations of the world as a pre-training task before a supervised or reinforcement learning phase is a common application for self-supervised learning. In theory, if the learnt representations are good and broad enough, a new model based on them should require fewer instances to learn a specific job. It is generally expected that good self-supervised algorithms are also good few-shot learners. This research examines the few-shot capabilities of two well-known self-supervised learning algorithms for visual representations, SimCLR [1] and SimSiam [2], and compares them against a supervised counterpart. A common evaluation protocol is to train a linear classifier on top of (frozen) representations learnt by self-supervised methods. This protocol is taken a step further by evaluating supervised and self-supervised methods on a few-shot image classification task using frozen representations. The experiments find that, as expected, the supervised method has higher performance than self-supervised methods SimCLR [1] and SimSiam [2]. However, SimCLR [1] gives a consistently better performance and is proposed as a potentially good self-supervised approach to learn image representations for few-shot image tasks when supervised learning is not feasible. Furthermore, this opens up a new area of focus for future research - evaluating the efficacy of other SOTA self-supervised methods like BYOL [3] and Barlow Twins [4] for a variety of few-shot tasks across domains.

Self-supervised | Few shot | SimCLR | SimSiam

Link to Code

Introduction

Animals and humans seem to learn about the world primarily through observation, with minimal initial interaction, rather than by focusing on specific tasks. Self-supervised learning is a paradigm that enables machines to learn about the world through observation. More specifically, self-supervised learning aims to understand dependencies within data by predicting one part of the input from the rest. Training a model in this manner compels it to learn something about the world to perform well.

A key application of self-supervised learning is developing hierarchical representations of the world, which can be used as pre-training before a supervised or reinforcement learning phase. Theoretically, if these learned representations are sufficiently robust and generic, a new model using them should need fewer examples to master a particular task.

This theory suggests that effective self-supervised algorithms should excel in few-shot learning, meaning a good representation should require minimal examples to understand a concept. This paper examines two prominent self-supervised learning algorithms for visual representations, SimCLR [1] and SimSiam [2], comparing their few-shot capabilities against a supervised counterpart. The hypothesis is that representations learned through supervised pretraining will outperform those from self-supervised methods in few-shot learning. Additionally, it is expected that SimSiam and SimCLR [1] will perform equivalently based on related experiments.

Related Work

Previous work has evaluated the accuracy gap between a selection of self-supervised and supervised methods when used to produce feature representations to be used for few-shot Image Classification [5]. Additionally, a study was performed in which they test when does self-supervision improve few-shot learning, and one of the key takeaways was that self supervision alone was not enough because they reported a significant under performance of these methods against supervised ones. However, these works only cover Jigsaw, Colorization, and Rotation pretext tasks for self-supervision; and these methods were published in 2016 and 2018. Since then, new selfsupervised methods such as SimCLR [1], SimSiam [2], and many more came out of research papers focused on developing better self-supervised learning algorithms for image related tasks, and more. There has also been work done to come up with specific self-supervised learning algorithms that work well on Few-Shot Natural Language Classification Tasks [6]. However our focus is different in two ways - first, that we are not developing new self-supervised algorithms but rather evaluating the performance of some of the latest ones on a few-shot classification task, and second, we look at few-shot Image Classification rather than Few-Shot Natural Language Classification Tasks.

These algorithms and strategies that have been proposed for self-supervised learning in recent years, share the main goal of learning pre-trained features that represent not only how images relate to one another but also, robustness to "nuisance factors" (i.e. invariant to different types of perturbations or changes). To achieve this, a popular principle has been to train algorithms that are robust to data augmentation, meaning that the representation of an image shouldn't change for augmented views of itself. One natural way to tackle this challenge has been through Siamese Networks and variants of it.

Regular Siamese networks by themselves are not capable of solving this task because they quickly learn trivial solutions. In particular, the network learns a solution in which all outputs "collapse" to a constant, no matter the image. Because of this, most of the recent self-supervised methods can be categorized in how they deal with collapsing solutions: Contrastive Learning (e.g. SimCLR [1]), Clustering (e.g. SwAV [4]), Distillation (e.g. SimSiam), Redundancy Reduction (e.g. Barlow Twins [7]). Typically, the performance of these models is tested by training a linear classifier on the self-supervised representation or by fine-tuning the model. Nevertheless, little has been done to assess the few shot capabilities of these representations.

Method

Our goal was to test and compare the capabilities of selfsupervised and supervised methods on a few-shot image classification task. To achieve this, we pre-trained supervised and self-supervised networks on CIFAR 100 dataset [8]. We tested these methods for few-shot image classification on CIFAR10 [8]. The ImageNet [9] dataset is a popular largescale benchmark for training deep neural networks but the high cost of performing experiments (e.g, including hyperparameter tuning) made CIFAR100 a better choice for our pretraining dataset to learn image representations. CIFAR100 is a great alternative since it is computationally far less expensive while also being sufficiently diverse, containing images across 100 classes. Since the goal was to extract feature representations for images from the pre-trained models to use for few-shot image classification, the features must be as generalisable as possible, to perform well on new data [10]. CIFAR10 came as a natural choice for testing on few-shot image classification because there is a similarity in the classes of CIFAR100 and CIFAR10- they belong to the same domain . It has been noted in previous work [5] that choosing datasets within the same domain for pre-training and testing, provides a greater benefit. However, we verified this using a different domain dataset STL10 [11], more details of which are provided in the Experiments section.

Self-supervised methods can be broadly categorized into 3 categories based on how they are avoiding trivial solutions. The different ways in which methods maximize similarity between a given image and its augmented version are using - contrastive learning, clustering and distillation [12]. We wanted methods that differed in this similarity maximization objective, so we chose SimCLR [1] and SimSiam [2] as the self-supervised methods for this project. SimCLR [1] uses contrastive learning, and SimSiam [2] uses distillation. We did a linear evaluation on SimCLR [1] and SimSiam [2] , based on "CIFAR Experiments" section of the SimSiam paper [2] , and obtained the following results:

Method	Best Test Acc. %	Learning Rate
SimCLR	65.12 %	0.06
SimSiam	68.99 %	0.1
Supervised	77.8%	0.1

Architecture. It is important to note that we used the same encoder (or sometimes called backbone) layer in all three methods - supervised, SimCLR [1] and SimSiam [2]. The backbone used is the CIFAR variant of ResNet-18. We replaced the first 7x7 Conv of stride 2 with 3x3 Conv of stride 1, and also removed the first max pooling operation. For data augmentation, we used the same augmentation as done for Imagenet, but leaving out Gaussian blur [1]. This was essential to ensure consistency in the models because we want to test the efficacy of the self supervised strategy, rather than the capacity of the underlying model. In essence, we didn't want to give any method an unfair advantage by using an encoder that is known to generally perform better . This encoder is followed by a one layer MLP projection head. More details of SimCLR architecture in the original paper [1]. In SimSiam [2], the projection head has 3 layers is followed by a prediction MLP head with 2 layer. The SimSiam [2] architecture takes as input two randomly augmented views from a given image, passes them through the encoder and the projection MLP head. The encoder shares weights between the two views of the image. This is followed by a prediction MLP head which transforms the output of one view and matches it to the other view of the image [2]. Further details of full SimSiam architecture are in the original paper [2].

In order to test the capabilities of the chosen self-supervised and supervised networks on few-shot image classification, we extracted image features from the backbone encoder layer of the supervised and self-supervised networks and trained a linear SVM classifier on these frozen feature representations[16, 20, 22].

Motivation. Our motivation for this project comes from previous discussions about potential of self-supervised methods for few-shot learning, in recent years. Data labeling is inherently expensive and time-consuming, particularly for tasks that need dense picture labeling, such as object detection and instance segmentation [21]. Although few-shot object detection focuses on building a model on novel object classes with limited data, it nevertheless necessitates prior training on a large number of labeled base class samples. Self-supervised techniques, on the other hand, attempt to learn representations from unlabeled data that may transfer well to downstream tasks such as object detection [13]. A potential research topic is combining few-shot and self-supervised object detection.

Experiments

The first stage of our experiments involved re-implementing and tuning hyperparameters for the supervised and selfsupervised methods (SimCLR [1] and SimSiam [2]). This was done to ensure that each method is being given the best possible chance to perform well in the second part of the experiment. The second part of our experiments involved extracting feature representations as frozen features to be used for performing few-shot image classification on two datasets - CIFAR10 (same domain as CIFAR100) and STL10 [11] (sampled from ImageNet [9], inspired from CIFAR10). STL10 is a more challenging benchmark for the methods in this project as the images are taken from Imagenet [9]which is a different domain.

Training and Hyperparameter Tuning of Pretrained models.

Supervised Method. We trained the supervised model using a batch size = 128, for 250 epochs with early stopping, achieving the best validation set (45k/5k train/validation split) performance of 77.98% at 193 epochs. We used a SGD with momentum = 0.9, weight_decay =5e-4, learning = 0.1 and CosineAnnealing with TMax = 200. With these settings the model achieves 77.8% accuracy on the test set. This model was trained on a NVIDIA Tesla T4 GPU,which took about 5.5 hours to train for 250 epochs.

Self-supervised methods. For training the self-supervised methods on CIFAR100 [8], we used the hyperparameters that were used in "CIFAR Experiments" section of the SimSiam paper [2] for both SimCLR [1] and SimSiam [2]. SGD with different learning rates (in Table below) has been used, a cosine decay schedule for 800 epochs, weight decay = 0.0005, momentum= 0.9, and batch size= 512. The input image size is 32×32. Since we are using a CIFAR dataset, as in the paper, we do not use blur augmentation. The results of tuning both methods at different learning rates is given in the table below. Since the given hyperparameters in the paper were for CIFAR10 and CIFAR100 is a more diverse dataset, in between Imagenet [9]and CIFAR10, we decided to try different learning weights to ensure maximum performance of our self-supervised methods before proceeding with extracting image representations for testing few-shot image classification capabilities. We used an NVIDIA Tesla T4 GPU to train SimSiam [2]. The training and linear evaluation together took approx. 24 hours to run, therefore all the runs together took around 3.5 days. We used GeForce RTX 2080 SUPER GPU to train SimCLR [1]. The training and linear evaluation together took approx. 24 hours to run, therefore all the runs together took around 3.5 days.

Model	Learning Rate	Linear evaluation Accuracy
SimCLR	0.03	60.62%
SimCLR	0.06	65.12 %
SimCLR	0.1	63.91 %
SimSiam	0.03	63.63 %
SimSiam	0.05	66.21 %
SimSiam	0.1	68.99 %

The best model for SimCLR [1] was with lr=0.06 with 65.12%, and the best model for SimSiam [2] was with lr=0.1 achieving 68.99%. The training pots are in Figure 1 and 2.

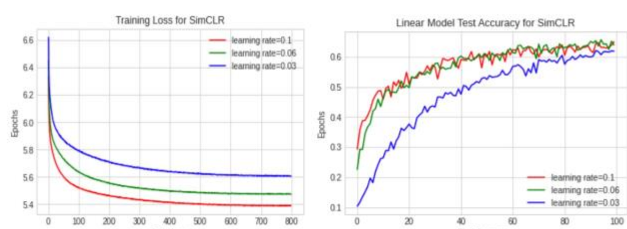

Fig. 1. Left plot: SimCLR training loss for models trained with different learning rates= 0.03, 0.06 and 0.1. Right plot: SimCLR test accuracy on linear evaluation for models trained with different learning rates= 0.03, 0.06 and 0.1

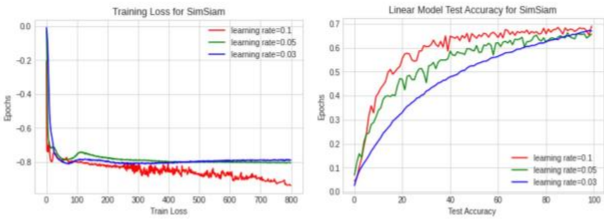

Fig. 2. Left plot: SimSiam training loss for models trained with different learning rates= 0.03, 0.05 and 0.1. Right plot: SimSiam test accuracy on linear evaluation for models trained with different learning rates= 0.03, 0.06 and 0.1

Training and Hyperparameter Tuning of Few shot Linear Classifier. We trained and tuned a linear classifier with different numbers of observations per class multiple times. For each pretrained model we extracted features from the target dataset and trained a linear classifier on frozen features using k sampled observations per class for multiple values of k. For each step we trained a multinomial logistic regression and a linear support vector machine and selected the best one. For the logistic regression only the l2 regularization was tuned by selecting values from a range of 40 logarithmically spaced values between 10^6 and 10^5 . Similarly, we tune the linear SVM by trying 26 values from the range $C \in 2^{[19,4]} \cup 10^{[7,2]}$. In all cases we use 3-fold cross validation. This entire process was repeated 5 times since we are stochastically selecting the number of observations per class[19].

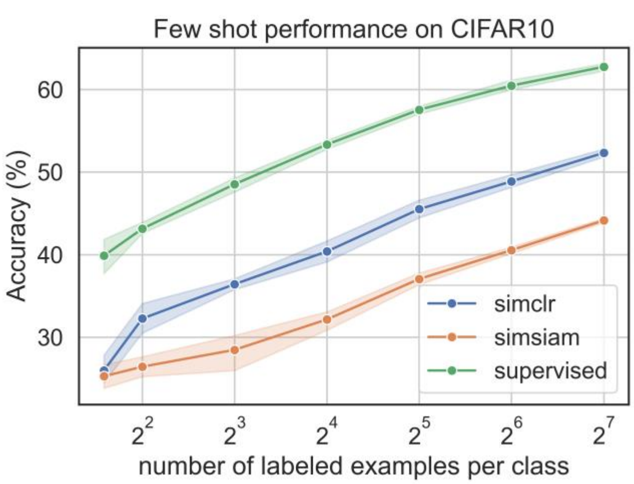

Fig. 3. Top 1 Accuracy of K-Shot Transfer via linear classifier on CIFAR10

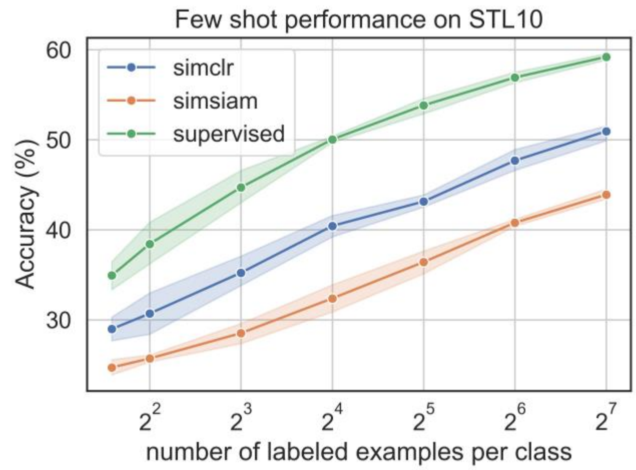

Fig. 4. Top 1 Accuracy of K-Shot Transfer via linear classifier on STL10

Result Interpretation

In Figures 3 and 4 we show the performance of few shot transfer via linear evaluation on CIFAR10 [8] and STL10 [11]. We observe a very similar behavior for both datasets. The supervised model always performs better by a large margin, followed by SimCLR [1] and then SimSiam [2]. There is a frequent 5%+ gap between the self supervised models. This is surprising, since previous experiments doing transfer learning (fine tuning with all observations, not a few shot linear classifier) the performance is very close. Moreover, in our reproduction of the algorithms SimSiam [2] performed about 4% (absolute) better than SimCLR [1], but still underperforms in this few shot setting. We hypothesize that, relative to SimCLR [1], the representations learned by SimSiam [2] might be too specific to generalize well to new classes without any fine tuning.

Even though the overall performances can be seen in the previous results, we want to know if there are any instances where one method outperforms the other. In figures 6 and 5 we show the average per class F1-score for different number of observations per class. Except mainly for class airplane in STL10, the supervised model does a better job for every class. In the very low shot range (k=3), SimSiam [2] and SimCLR [1] usually have a very close performance, but afterwards SimCLR [1] typically takes over.

Conclusions

We see a significant accuracy gap between self-supervised and supervised methods. This is expected because previous work comparing other self-supervised methods with supervised methods on Imagenet [9]showed the same trend in results [5].

Among the self-supervised methods, we see SimCLR [1] outperforming SimSiam [2] by a great margin even though

SimSiam [2] performed better during pretraining. There can be several reasons for this:

1. The contrastive nature of SimCLR[1] might produce better and more generalizable out-of-the-box representations, while distillation alone through the stopgradient operation might not be enough. It is possible that other forms of distillation (e.g. BYOL [3]) produce better results.

2. Most of the related work for transfer via linear classifiers pretrain on Imagenet [9]and are not few-shot (i.e. use all the observations to train the linear classifier) ([3] ,[2], [18], s) and show comparable results to the supervised counterpart. It is possible that the difference in performance seen in our paper comes from using a smaller, less rich dataset with lower resolution images than Imagenet. Perhaps, the self-supervised pretraining requires more examples and variety of classes to thrive in this task. Alternatively, this discrepancy could be simply due to the use of fewer labels, as reported in [14, 23].

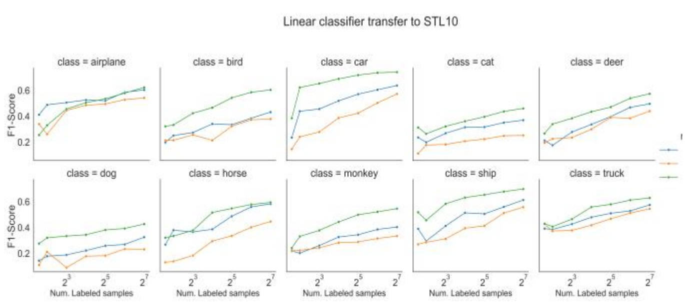

Fig. 5. Linear classifier transfer to STL10

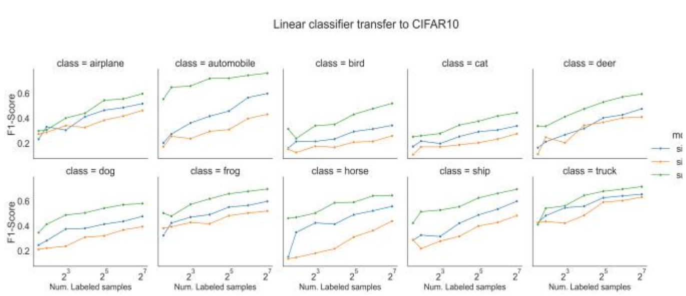

Fig. 6. Linear classifier transfer to CIFAR10

Future Work / Limitations

Our experiments were limited to just two self-supervised methods. Furthermore, we restricted hyperparameter tuning to only a relatively smaller batch size of 512 because of limitation of time and computational resources. We could expand existing analysis in multiple ways, by changing some aspect of the self-supervised pre-training. This can be done in multiple ways: First, we should repeat our experiments by transferring to other more dissimilar dataset. This is because its important to understand the generalization of representations to different domains[17]. Second, it is worth pretraining the models on ImageNet [9] to test our hypothesis that the source dataset might play an important roll on the few shot performance of self-supervised methods. Third, we should repeat our analysis on other State-of-theArt self-supervised methods such as BYOL [3], that comes under the category of Distillation. We have also noted in our Related Work section how the performance of SimCLR and BYOL on linear classification task involving all samples was similar, therefore it would be worth exploring the performance on BYOL in a few-shot setting as well. Additionally, it is worth analysing the performance of other methods of self supervision like clustering (e.g. SwAV [7])) and redundancy reduction (e.g. Barlow Twins [4]). Fourth, we can explore how we can modify the self supervised strategies to perform better in a few shot setting. One way is testing different data augmentation techniques than the ones we used. These include techniques using Reinforcement Learning like AutoAugment [15] and Generative Adverserial Networks [16]. We could then compare the performance of regular augmentation techniques versus these new techniques, for each self-supervised learning method, on the final few-shot Image Classification task[18, 20, 21].

Finally, we could also broaden the scope of our analysis to few shot tasks on Natural Language Processing tasks instead of Image Classification, like Text Classification [17, 22] to further evaluate the robustness of various self-supervised methods on a variety of few-shot tasks.

References

1. Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020.
2. Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 15750–15758. Computer Vision Foundation / IEEE, 2021.
3. Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires,

Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

4. Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Selfsupervised learning via redundancy reduction. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12310–12320. PMLR, 2021.
5. Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 6390–6399. IEEE, 2019. doi: 10.1109/ICCV.2019.00649.

Trapit Bansal, Rishikesh Jha, Tsendsuren Munkhdalai, and Andrew McCallum. Selfsupervised meta-learning for few-shot natural language classification tasks. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, Novem ber 16-20, 2020*, pages 522–534. Association for Computational Linguistics, 2020.doi: 10.18653/v1/2020.emnlp-main.38.

6. Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.
7. Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.
8. Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A largescale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009. doi: 10.1109/CVPR.2009.5206848.
9. Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 8058–8067. IEEE, 2019. doi: 10.1109/ICCV.2019.00815.
10. Adam Coates, Andrew Y. Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In Geoffrey J. Gordon, David B. Dunson, and Miroslav Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, volume 15 of *JMLR Proceedings*, pages 215–223. JMLR.org, 2011.
11. Ishan Mista. Self-supervised learning in computer vision. New York University, 2021.
12. Gabriel Huang, Issam H. Laradji, David Vázquez, Simon Lacoste-Julien, and Pau Rodríguez. A survey of self-supervised and few-shot object detection. *CoRR*, abs/2110.14711, 2021.
13. Jong-Chyi Su, Subhransu Maji, and Bharath Hariharan. When does self-supervision improve few-shot learning? In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part VII*, volume 12352 of *Lecture Notes in Computer Science*, pages 645–666. Springer, 2020. doi: 10.1007/978-3-030-58571-6_38.
14. Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. Fast autoaugment. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 6662–6672, 2019.

15. Jongheon Jeong and Jinwoo Shin. Training gans with stronger augmentations via contrastive discriminator. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

16. Ved Prakash Upadhyay, Subhash Panwar, Ramchander Merugu, “Protein Sequence Structure Prediction Using Artificial Intelligent Techniques”, AICTC '16: Proceedings of the International Conference on Advances in Information Communication Technology & ComputingAugust 2016, ArticlE No.: 108Pages 1–6 https://doi.org/10.1145/2979779.2979887

17. Ved Prakash Upadhyay, Mayank Sonawat, Kalpana V.Singh, and Ramchander Merugu. (2017). “NANO ROBOTS IN MEDICINE: A REVIEW.” International Journal of Engineering Technologies and Management Research, 4(12: SE), 27-37. DOI:10.5281/zenodo.1157965.

18. Ramchander Merugu, Ved Prakash Upadhyay, Shivaranjani Manda, “Bioinformatics analysis and modelling of mycotoxin patulin induced proteins”, International Journal of Bioinformatics and Biological Science Year : 2016, Volume : 4, Issue : 1, Print ISSN : 2319-5169. Online ISSN : 2321-7111.
Article DOI : 10.5958/2321-7111.2016.00002.0

19. PB Shanthi, Shivani Modi, KS Hareesha, Sampath Kumar, “Classification and comparison of malignancy detection of cervical cells based on nucleus and textural features in microscopic images of uterine cervix”, International Journal of Medical Engineering and Informatics Vol. 13, No. 1. Published Online:November 17, 2020pp 1-13https://doi.org/10.1504/IJMEI.2021.111861

20. Ved Prakash Upadhyay, Subhash Panwar, Ramchander Merugu, and Ravindra Panchariya, “Forecasting Stock Market Movements Using Various Kernel Functions in Support Vector Machine”, AICTC '16: Proceedings of the International Conference on Advances in Information Communication Technology & ComputingAugust 2016Article No.: 107Pages 1–5https://doi.org/10.1145/2979779.2979886

21. S Comis, T Kandasamy, C Wu, SP Kumar, VP Upadhyay, A Chhikara, “Real-world evidence of CDK4/6 Inhibitor combination effects on assessing outcome with different sequence of treatment among adult patients with HR + and HER2 - metastatic breast cancer”, VOLUME 31, SUPPLEMENT 4, S365, SEPTEMBER 2020, DOI:https://doi.org/10.1016/j.annonc.2020.08.409

22. Subhash Panwar, Ved Prakash Upadhyay, SK Bishnoi, “A survey of intelligent system techniques for Indian stock market forecasting”, 39th MIPRO conference, IEEE, 2016

23. Jason Wei, Chengyu Huang, Soroush Vosoughi, Yu Cheng, and Shiqi Xu. Few-shot text classification with triplet networks, data augmentation, and curriculum learning. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 5493–5500. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.naacl-main.434.