Masked Relational Knowledge Distillation For Self-Supervised Learning

Cristopher McIntyre Garcia
Department of Computer Science
University of Ottawa
Ottawa, ON K1N 6N5
cmcin019@uottawa.ca

Abstract

This report presents a new self-supervised learning (SSL) framework for computer vision. It draws inspiration mainly from the iBOT framework [20] and the concept of relational knowledge distillation (RKD) [16]. iBOT is the current SoTA SSL framework that uses self-distillation [2], and masked image modeling (MiM) [1] as a pretext task, to pre-train a backbone architecture. RKD is a type of knowledge distillation that has been shown to outperform its vanilla counterpart in certain cases by distilling information via structural representations of similar images rather than instance-to-instance [16]. The idea of this project is to combine SSL and RKD by incorporating the latter into the iBOT framework as self-RKD, and in doing so, perform MiM on the structural representations of similar images; thus increasing the underlying comprehension of the images that can later be beneficial in downstream tasks. This report show that, when using a smaller dataset with smaller images, the combination of RKD with a distance potential metric and iBOT out performs vanilla iBOT, which uses regular knowledge distillation, on downstream tasks such as supervised linear classification and unsupervised class token classification. Specifically this report demonstrates the results for fine-tuning models trained on different configurations of iBOT on the CIFAR10 dataset (see 1). This report also show that using this method does not jeopardize the quality of the feature map representations, demonstrated in figure 3. Though this report does not present state of the art results, we believe that with better performing hardware, a larger dataset, and more fine-tuning of the implementation, that it may be possible to train a backbone model that could outperform the current SoTA.

1 Introduction

Self-supervised learning has seen great advances in the field of computer vision [2], [1], [20]. Based on the successful methodologies of masked language models in the field of natural language processing [5], these SSL frameworks aim to pre-train backbone architectures through knowledge distillation and pretext tasks involving the recovery of missing information. To make up for the lack of labels, SSL models use image augmentations such as cropping, resizing, color jittering, and flipping to produce same class images that are then fed into the framework as positive pairs [12].

The most popular method of processing the data to generate image representations is through a co-distillation framework, meaning that a teacher backbone and a student backbone of the same architecture are used to perform knowledge distillation [2]. The SSL framework is then put in charge of performing a pre-text task that will ensure the student network is trained to create good image representations that can be used on downstream tasks. In most cases, the teacher network is also trained via an exponential moving average, making the framework a self-distillation framework [2]. In the case of iBOT [20], the student receives a version of the augmented image that has undergone a

level of masking, and the teacher receives the regular image. The student is then tasked to predicted the missing information so as to match the teacher network's output representation. This method has seen great success and has become the current SoTA SSL framework, besting many supervised learning methods [20].

This report discusses a potential improvement to the iBOT framework via a different knowledge distillation method. Many forms of knowledge distillation exist, one in particular being relational knowledge distillation (RKD) [16]. Most methods of distillation distill information by performing instance-to-instance loss [8]. In RKD, a potential function is used to compute the structure between same-class data in the batch to then perform structure-to-structure loss and back-propagate through the student network. Because RKD uses the same architecture as other knowledge distillation methods (teacher-student networks), swapping them is possible. The difference mainly lies in the manipulation of the output representations prior to computing the loss. Replacing regular knowledge distillation with RKD could improve the generalization capabilities of a network and help generate better representations [16].

The results shown in this report demonstrate that replacing regular knowledge distillation with RKD in the iBOT framework yields improved results in short-term training. Due to limitations in computational power and time restrictions, the models were not trained on as much data or for as long as the original iBOT models. In fact, the iBOT models were trained for twenty times more epochs and on ten times more data. Having said that, we demonstrate that there is potential in further analyzing different types of distillation in the SSL framework despite the limited resources at our disposal. More specifically, with our additions, we see improvements over the original corresponding iBOT trained model (ViT-S/8) on CIFAR10 downstream tasks such as top-one accuracy on unsupervised class token classification (+8.17%) and top-one accuracy on supervised linear classification (+2.05%) (see table 1).

2 Related Work

The MoCo framework [12] was one of the first to introduce the concept of contrastive learning to the SSL framework, which to this day still performs fairly well, as seen in [4]. By matching an encoded augmented image with many other momentum-encoded augmented images, and through contrastive loss [10], they are able to train an encoder to create visual representations. The SimCLR framework [3] has shown that the type of image augmentation chosen heavily impacts the performance of the model, and using a non-linear transformation between representations and contrastive loss improves the quality of the representations. These methods need careful treatment of negative pairs, mandate the use of large batch sizes, and rely heavily on augmentation.

In the BYOL framework [9], negative pairs are completely omitted. Instead, they iteratively bootstrap the weights of an encoder to serve as the weights of a target network. By doing so, they improve performance robustness to different image augmentation and lower the necessity of using large batch sizes. DINO [2] incorporates the Vision Transformer (ViT) [6] into this approach, and presents it as self-distillation. They are able to create segmentation masks from images without using predefined ground truth labels, something that has been difficult to do, even with CNNs. Many variations of this model have been studied, such as the Seed [7], EsViT [14] and CLIP [18] frameworks.

Very recently, the Beit framework [1] proposes a MiM task to pre-train ViTs, very reminiscent of the pre-training task of BERT [5]. In this method, random portions of an image are masked and fed into a backbone ViT. The pre-training task aims to reconstruct the visual information that was masked from the original image. The iBOT framework [20] combines the ideas of MiM from Beit and self-distillation from DINO to create a new current SoTA that is able to outperform certain SoTA supervised methods. Many papers have incorporated MiM, such as the MAE [11], HOG [17] and Simmim [19] frameworks.

Though the use of instance-to-instance knowledge distillation as the proposed concept for self-distillation has been shown to work well, RKD [16] demonstrated that knowledge distillation is able to be outperformed by RKD in generalizing. That said, no work has been done on incorporating this concept into SSL. the dBOT framework [15] use a multi-stage distillation framework, by freezing the teacher and re-initializing the student after every step. This is different from RKD because they still perform instance-to-instance knowledge distillation. We theorize that incorporating RKD may help improve the results of current SSL frameworks.

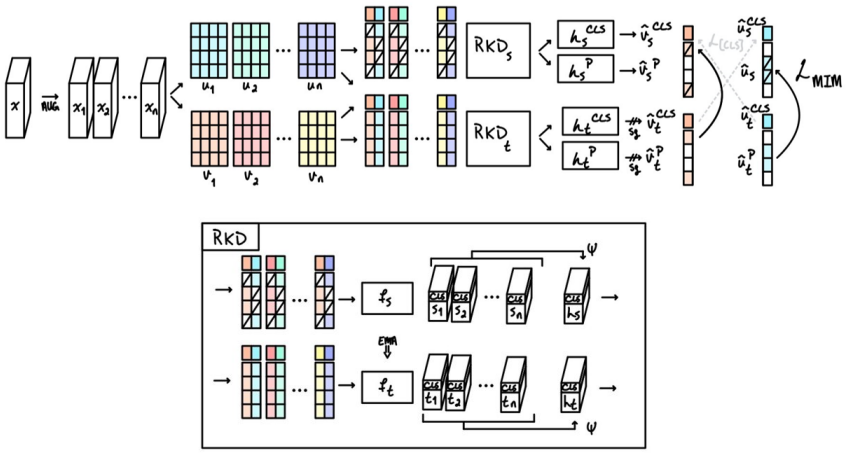

Figure 1: Relation Knowledge Distillation iBOT.

3 Method

The proposed idea was to combine the ViT and MiM from iBOT with the concept of RKD (Figure 1). To achieve this, we replace the knowledge distillation component in iBOT with a variation of RKD. This is done by providing the networks with additional image augmentations and computing the structure of their output representations. The reason why more augmentations are needed is because no labels are provided, and so there is no way of knowing which images are of the same class. Using multiple augmentations enables the model to generate structural representations that accurately represent the image without requiring knowledge of its class. In doing this, the student network will learn to create similar representation for similar images. Also, having multiple augmentations has been shown to improve the generalization and accuracy of models [3]. In performing RKD, the loss that is computed afterwards is no longer instance-to-instance but rather structure-to-structure. The loss used can also be modified to a Huber loss [13], which is what was used in [16], or the loss can stay the same as what was used by the iBOT model, which is the cross-entropy between the class tokens and the patch tokens. RKD aims to teach a student network the distribution of structural output representations of similar images generated by a teacher network. The way to achieve this is by using potential functions such as those proposed in [16]. Afterwards, the loss between the structural components is computed and used to train the student network. This project aimed to evaluate the properties and performance of each potential function, and their impact on the SSL framework.

3.1 Basic Framework

The basic framework used in this experiment is based on the DINO implementation [2]. The DINO framework presented the SSL problem as a co-distillation framework, which means that the teacher and student networks share the same architecture. It also introduced the use of the ViT as the backbone architecture. In doing so, they discovered certain properties, such as the ability to generate segmentation maps of images without the need for training with segmentation labels. When training, both the teacher and student networks are provided with augmentations of the same image. The student network is provided with an augmentation containing less information than that provided to the teacher. The goal of the framework is to train the student network to predict the missing information so as to generate the same output representations as the teacher network. The model is trained via a loss function that is then back-propagated through the student network to update its parameters. Typically, the loss will involve computing the similarity of the representations via dot products, followed by cross-entropy loss. The parameters of the teacher network are updated by an exponential moving average so as to keep the representations slowly evolving. This makes sure the model does not collapse, which can happen when trained quickly. A model has collapsed when it finds trivial solutions to lower the loss without necessarily improving the performance of the model [2]. The novelty of iBOT was changing the pretext task of the DINO framework to incorporate MiM [20]. The novelty of this paper is modifying the type of distillation used to incorporate structural

representations when calculating the loss. In our framework, the backbone architecture is a ViT and the pretext task is MiM.

3.2 Pretext Task

The pretext task used in the DINO framework and performed by the student network was predicting a global view from a local view. An image augmentation that contained under fifty percent of the information via a crop was passed to the student network. An image augmentation that contained over fifty percent of the information via a crop was passed to the teacher network. The student network was tasked with predicting the same representation as the teacher network. This pretext task forces the student network to learn how to predict the proper missing information in order to have similar output representations as the teacher [2]. iBOT modified this pretext task to incorporate MiM [20]. Instead of removing information via cropping, it instead patches certain areas of the image that is provided to the student network. The student network is then tasked with predicting the missing patches so as to generate the same representation as the teacher network, which received the same image without hidden patches. In our implementation, the MiM pretext task is used to train the student network.

3.3 Loss Functions

Two loss functions were proposed in RKD: Distance-wise and Angle-wise loss [16]. Each of them uses the potential metrics 1 and 3 respectively. The purpose behind these functions is to find the fundamental structure of similar data, as opposed to only using a instance-to-instance comparison.

$$\psi_D(t_i, t_j) = \frac{1}{u} \left\| t_i - t_j \right\|_2 \tag{1}$$

$$\mathcal{L}_{RKD-D} = \sum_{(x_i, x_j) \in X^2} l_\delta \left(\psi_D(t_i, t_j), \psi_D(s_i, s_j) \right) \tag{2}$$

In Eq. 1, the euclidean distance between a pair of examples is computed from their output representation over a normalization factor. Note that t_i and t_j are the teacher output representations of augmented images x_i and x_j respectively, and s_i and s_j are the student output representations of augmented images x_i and x_j respectively. Both augmentation x_i and x_j come from the same original image. Eq. 1 is used in a Hubert loss function l for each pair of similar images in a batch and is shown in Eq. 2.

$$\psi_A(t_i, t_j, t_k) = \cos \angle t_i t_j t_k = \left\langle e^{ij}, e^{kj} \right\rangle ; where \;\; e^{ij} = \frac{t_i - t_j}{\left\| t_i - t_j \right\|_2} \tag{3}$$

$$\mathcal{L}_{RKD-A} = \sum_{(x_i, x_j, x_k) \in X^3} l_\delta \left(\psi_A(t_i, t_j, t_k), \psi_A(s_i, s_j, s_k) \right) \tag{4}$$

The angle-wise metric uses a triplet of examples to create an angle. Eq. 3 also originally uses the Huber loss function and is shown in Eq. 4. Another way of implementing these potential functions is by incorporating them into the loss of iBOT. This way, we compute the class and patch token structures and then compute the cross-entropy loss between the student and teacher outputs [20]. This method makes it simple to incorporate RKD into the iBOT framework without needing to completely modify the loss. In the implementation of this project, cross-entropy loss is used on the structural output representations.

3.4 Training

We train a ViT-small/8 model as the backbone. The 8 dictates the size of the patches that are fed into the ViT. The dataset used to pre-train the model is a version of ImageNet-1k that contains images downsized to 32x32 and contains a tenth of the images. The total number of patch tokens fed into the ViT is 16. The batch size used was 256 for pre-training while using regular iBOT, and iBOT with the RKD-D framework. When pre-training the ViT using iBOT with the RKD-A framework, the batch size needed to be lowered to 16 due to virtual memory limitations. This, as will be seen later on, impacts the results substantially. Once the backbone model was trained, we could either fine-tune the backbone to perform unsupervised class token testing or we could fine-tune the model to

Table 1: Results on CIFAR10 downstream tasks

	Sup Linear		Unsup CLS	
Pre-train	Acc@1	Acc@5	Acc@1	Acc@5
Random init	**33.900**	83.870	32.100	84.240
iBOT & KD	25.650	74.420	30.850	84.110
iBOT & RKD-D	33.820	**84.260**	**32.900**	**84.270**
iBOT & RKD-A	33.720	83.830	31.790	83.860

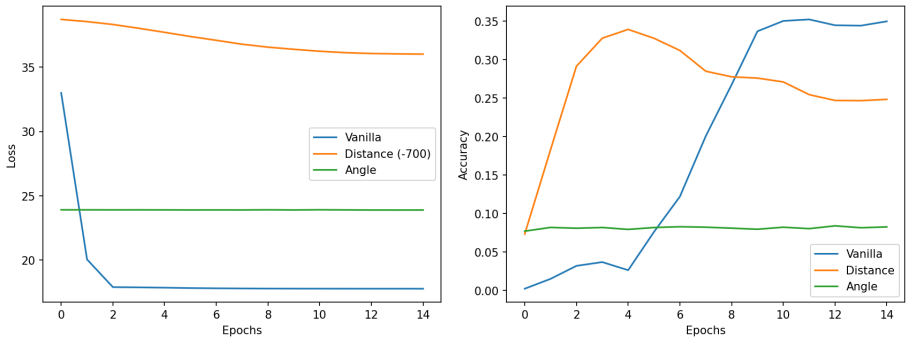

Figure 2: Loss and Accuracy over pre-training

perform supervised linear classification [20]. In both cases, the fine-tuning and testing were done on the CIFAR10 dataset, and the training underwent ten epochs. Computer vision development requires a lot of images and computational power. Every other hyper-parameter stayed the same as those in the original iBot implementation. The machine used to train the models was not as good as the one used to develop iBOT. Because of this, we were restricted to a different dataset and hyper-parameters such as batch size, patch size, and training time. We were only able to explore a portion of our ideas due to the time constraints of the project.

4 Experiments

4.1 Pre-training

Figure 2 demonstrates the loss and accuracy during training. The accuracy is computed by comparing the output of the teacher with that of the student. The main goal of the framework is to train the student network to accurately predict the same representations as the teacher network, with less information. Once this is achieved, it is assumed that the student network has learned to represent images efficiently. These representations hold malleable information that can be useful for downstream tasks.

The loss curves tend to differ largely when using different knowledge distillation frameworks. When using RKD, the loss starting point seems to be directly proportional to the batch size. Because we were unable to use a large batch size for the RKD-A framework (Eq. 4), the loss started relatively low and did not change. The same cannot be said for the model trained using the RKD-D framework (Eq. 2). The loss starting point of the original iBOT framework was also proportional to the batch size, however the impact was not as substantial. The loss of the RKD-D framework lowered at a very slow pace, and converged towards thirty five after around twelve epochs. The loss of the original iBOT framework dropped too rapidly and flat lines after two epochs.

The accuracy curves are also quite different. The best short-term accuracy is that of the model trained with the RKD-D framework. The model trained with the RKD-A framework does not change, which is representative of its loss curve. The model trained with the regular iBOT framework seems to perform better in the long term. It seems as though a larger loss can be beneficial to the accuracy of a model in the short run. However, due to the fragile nature of SSL frameworks, it is important to note that this advantage most likely would not hold for longer training with larger images. Due to a lack of data, the vanilla iBOT framework might have collapsed early, which could explain its rapid decrease in loss. However, this does not explain why its accuracy increased even after its loss flat lined.

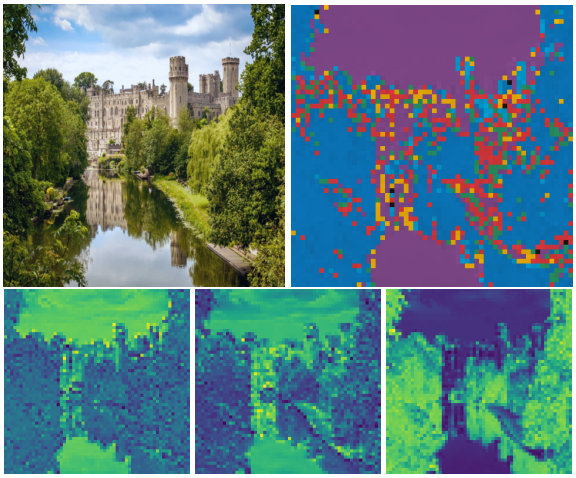

Figure 3: Attention maps.

4.2 Attention Maps

One interesting property that was discovered while using the ViT as a backbone for SSL networks was their capability of creating attention maps that look like segmented versions of the original image [2]. We attempted to keep this property with our additions to the implementation of iBOT. In Figure 3, we see that our model is still able to create these nice segmented maps. Different heads of the ViT are able to segment different objects in the image. When combined, the heads form a semantic segmentation map by separating the respective objects from the image. Though this discovery is interesting, we believe that good segmentation in the feature maps does not necessarily yield better accuracy. This belief stems from the fact that these attention maps were produced with limited data and training, and from the fact that the accuracy on the downstream tasks was not very high. That said, it is possible that with more training and more data, better segmented maps would be produced.

4.3 Downstream Tasks

The best-performing framework was the one using RKD-D. That said, the performance was not substantially better than using randomly initialized weights. We believe that due to the limited batch size and image size, the models were prone to collapsing. It is unclear whether the framework would outperform the regular iBOT framework when both are trained on larger data sets. We suspect that the framework would have to undergo more hyper-parameter tuning if more training were to be done. We also believe that our implementation has room for improvement. Despite this, the RKD-D iBOT model outperformed the vanilla iBOT model in all of our experiments (Table 1). On the supervised linear classifier downstream task, the difference in top-one accuracy was 8.17%, and the difference in top-five accuracy was 9.84%. The difference in the top-one accuracy on the unsupervised class token classifier downstream task was 2.05% and the top-five accuracy was 0.16%. We suspect that because the regular iBOT model was performing better at the end of training in terms of matching student and teacher output representations, it was able to perform better on the unsupervised downstream task. However, the representations obtained by the iBOT with the RKD-D framework appear to perform consistently better, which may be due to a better understanding of the data structure.

5 Discussion and Conclusion

The results have shown that pre-training a model using our self-supervised method can be beneficial. With more attention given to the loss functions, we believe better results can be achieved. Though our results are not SoTA, we believe that with better hardware and more fine-tuning of the hyper-parameters, we could improve on the works previously conducted. In future works, we would like to explore different distance metrics or introduce the concept of similarity as a potential metric. We also think that it would be beneficial to attempt different types of knowledge distillation techniques.

6 References

[1] Hangbo Bao, Li Dong, and Furu Wei. "Beit: Bert pre-training of image transformers". In: *arXiv preprint arXiv:2106.08254* (2021).

[2] Mathilde Caron et al. "Emerging properties in self-supervised vision transformers". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 9650–9660.

[3] Ting Chen et al. "A simple framework for contrastive learning of visual representations". In: *International conference on machine learning*. PMLR. 2020, pp. 1597–1607.

[4] Xinlei Chen, Saining Xie, and Kaiming He. "An empirical study of training self-supervised vision transformers". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 9640–9649.

[5] Jacob Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[6] Alexey Dosovitskiy et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv:2010.11929* (2020).

[7] Zhiyuan Fang et al. "Seed: Self-supervised distillation for visual representation". In: *arXiv preprint arXiv:2101.04731* (2021).

[8] Jianping Gou et al. "Knowledge distillation: A survey". In: *International Journal of Computer Vision* 129.6 (2021), pp. 1789–1819.

[9] Jean-Bastien Grill et al. "Bootstrap your own latent-a new approach to self-supervised learning". In: *Advances in neural information processing systems* 33 (2020), pp. 21271–21284.

[10] Raia Hadsell, Sumit Chopra, and Yann LeCun. "Dimensionality reduction by learning an invariant mapping". In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2. IEEE. 2006, pp. 1735–1742.

[11] Kaiming He et al. "Masked autoencoders are scalable vision learners". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 16000–16009.

[12] Kaiming He et al. "Momentum contrast for unsupervised visual representation learning". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 9729–9738.

[13] Peter J Huber. "A robust version of the probability ratio test". In: *The Annals of Mathematical Statistics* (1965), pp. 1753–1758.

[14] Chunyuan Li et al. "Efficient self-supervised vision transformers for representation learning". In: *arXiv preprint arXiv:2106.09785* (2021).

[15] Xingbin Liu et al. "Exploring target representations for masked autoencoders". In: *arXiv preprint arXiv:2209.03917* (2022).

[16] Wonpyo Park et al. "Relational knowledge distillation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 3967–3976.

[17] Chen Wei et al. "Masked feature prediction for self-supervised visual pre-training". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 14668–14678.

[18] Yixuan Wei et al. "Contrastive Learning Rivals Masked Image Modeling in Fine-tuning via Feature Distillation". In: *arXiv preprint arXiv:2205.14141* (2022).

[19] Zhenda Xie et al. "Simmim: A simple framework for masked image modeling". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 9653–9663.

[20] Jinghao Zhou et al. "ibot: Image bert pre-training with online tokenizer". In: *arXiv preprint arXiv:2111.07832* (2021).