ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

Temperature as a Regularizer for Semantic Segmentation

Chanho Kim[1] and Won-Sook Lee[1]

University of Ottawa - School of Electrical Engineering and Computer Science (EECS)
800 King Edward Ave, Ottawa, ON - CANADA

Abstract. A data-oriented approach including all deep learning methods is usually suffered by overfitting. A regularizer has been, from the beginning, introduced to resolve this problem. Inspired by Generative Adversarial Network (GAN), our framework generates the adversarial loss to penalize a segmentation model like a regularizer. We introduce temperature as a regularizer when calculating Least-Square losses. Temperature affects losses in both a discriminator and a generator in our DCGAN framework. Our experiment suggests L2 losses on top of the original LSGAN losses for optimization. This new regularizer using temperature improves semantic Segmentation accuracy both in Pixel accuracy and mean Intersection-of Union.

1 Introduction

The Convolutional Neural Network [1] model complexity and the performance in many tasks are proportional, resulting in more computation power and memory. There have been trials for reducing workload or generating efficient networks. MobileNet models [2] have been developed to reduce the number of parameters and memory usages. Inside well-known structures such as [3][4], there have been trials to improve partial structures [5] without increasing the number of parameters greatly.

Our work starts with the desire to improve given models for semantic segmentation. One of most popular approaches in semantic segmentation is using GAN [6] having 2 parties and they are in a dispute; generator and discriminator. Our observation shows that a discriminator gets a chance to penalize a generator by measuring fake samples against real ones. Temperature provides a modified probability map, which requires only an additional softmax calculation without any network structure change. Therefore, it works as a regularizer and our experiments with different temperatures in both the segmentation loss and the adversarial loss show it avoids overfitting and therefore improves the segmentation accuracy.

2 Related Works

Generative Adversarial Network: Started from the first GAN paper [6] which introduces the adversarial network design between a generator and a discriminator, various GAN loss functions are introduced such as minimax loss strategy used in DCGAN [7], Least-Square loss from LSGAN [8], or Wasserstein

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

Loss from WGAN [9][10]. GAN is used in several applications including Semantic Segmentation where cross-entropy losses and adversarial losses are typically used. DCGAN loss converges to the specific value and WGAN loss diverges to the high negative values, while LSGAN loss does not have this problem. All discriminator losses from GAN systems are easily saturated, blocking a generator having time to be trained. So we are motivated to find the best loss function and how to slow down the discriminator training.

Temperature: Temperature generates different softmax probabilities from logits used in knowledge distillation [11][12]. Temperature was used to control the probability map transferred to another probability map by training with two or multiple models also in semantic segmentation [13][14]. Rather than methods using multiple models, we use temperatures in softmax layers with a small discriminator for reducing the computational power.

Regularization: Dropout [15] and Dropconnect [16] are popular regularizers Loss penalizing is also another common way such as weight decay [17] or L_1 and L_2 Regularization [18] widely used in supervised learning tasks. Those methods produce versatile model structures in each training step, resulting in preventing overfitting.

3 Adversarial Framework in Semantic Segmentation

In semantic segmentation, a model processes RGB images for generating the corresponding probability maps. Each channel has probabilities that each pixel indicates whether its label is the specific class or not. With feature maps, the model calculates the categorical cross-entropy loss for updating its weights by backpropagation. Eq. (1) is how to calculate the loss in a general way.

$$\mathcal{L}_{seg} = -\frac{1}{N}\sum_{i=0}^{N}\sum_{h,w}\sum_{c=1}^{C} y_{h,w,c}\log p_{h,w,c}^{(i)} + \frac{\lambda}{2}\sum_{i} w_i^2 \tag{1}$$

The feature map has 4 dimensions: Batch size (N), Height (H), Weight (W), Channel (C). We obtain $y_{h,w,c} = 0$ or 1 by converting the groundtruth labels to one-hot labels. L2 (Least-Square) Regularization Loss can be applied to prevent the model from overfitting. λ is L2 weight decay rate and w_i should be a trainable weight.

$$\frac{1}{N}\sum_{i=1}^{N}\left[E\Big(\log D(\mathbb{X}^i)\Big) + E\bigg(\log\Big(1 - D(G(\mathbb{Z}^i))\Big)\bigg)\right] \tag{2}$$

Eq. (2) is the objective loss function of DCGAN [7] suitable for the only one sigmoid output of a discriminator for classification. $\mathbb{X}$ is the one-hot encoded groundtruth label map, and $\mathbb{Z}$ is the RGB minibatch image pixels. However, the DCGAN discriminator loss saturates and cannot penalize the generator (segmentation network) properly. Thus, we instead decide to use LSGAN [8] loss, keeping the DCGAN structure because LSGAN discriminator is more appropriate for single-class or multi-class classification tasks.

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

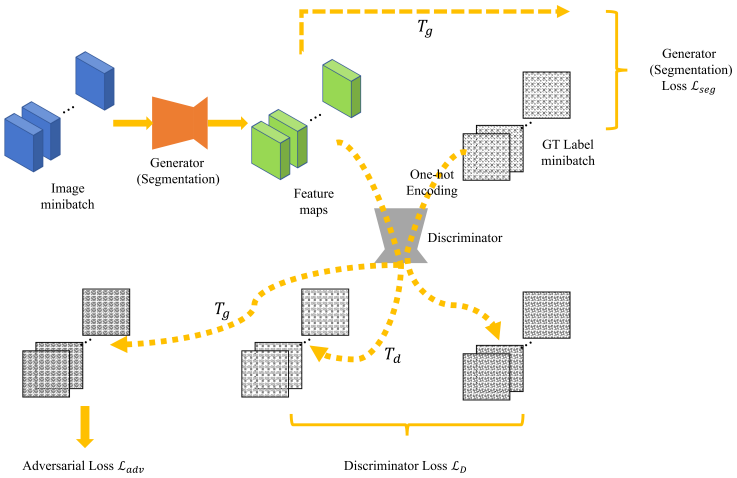

Fig. 1: The Adversarial Framework Structure. We apply both temperatures T_G and T_D on the feature maps generated from the segmentation network and calculate losses. (T_G to $\mathcal{L}_{seg}$ and $\mathcal{L}_{adv}$, T_D to $\mathcal{L}_D$)

Fig. 1 shows our framework for adversarial loss for regulating and penalizing the segmentation network inspired by [14]. We calculate the least-square losses in Eq. (3) and Eq. (4). We aim to backpropagate $\mathcal{L}_D$ in discriminator training. $\mathcal{L}_{adv}$ is added to $\mathcal{L}_{seg}$ with a weight α. In addition to $\mathcal{L}_{adv}$, we want to find whether L2 regularization properly optimizes the segmentation network. Thus, we have three primary parameters in Section 4: λ, T_G, and T_d.

$$\mathcal{L}_D = \frac{1}{2}\sum\left[E\left(\left(D(\mathbb{X})-1\right)^2\right)+E\left(\left(D\left(G(\mathbb{Z})\right)_{T_D}\right)^2\right)\right] \tag{3}$$

$$\mathcal{L}_{adv} = \frac{1}{2}\sum E\left(\left(D\left(G(\mathbb{Z})\right)_{T_G}-1\right)^2\right) \tag{4}$$

$$\mathcal{L}_G = \mathcal{L}_{seg} + \alpha\mathcal{L}_{adv} \tag{5}$$

4 Evaluation

4.1 Training details

We select the segmentation network as FCN [19] and ResNet-50 [4]. The backbone network of FCN-8s is VGG-16 [3]. Also, we utilize ImageNet [20] pretrained weights in training with VOC2012 augmentation dataset [21][22] and CityScapes dataset [23]. We use a DCGAN discriminator with Batch Normalization layers.

We apply the same learning rate $lr = 5{\cdot}10^{-3}$ for all cases with Cosine learning rate with Warmup [24] epoch 5, and Mixed Precision which enables larger batch size and higher resolution by training in 16-bit Floating-Point (FP16). For the

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

discriminator, we apply the learning rate $lr_d = 10^{-7}$ in FCN cases and $lr_d = 5 \cdot 10^{-8}$ in ResNet-50 cases, with $\alpha = 0.05$ in most cases except for *FCN-8s-C* ($\alpha = 0.1$). Also, we use Momentum=0.9 in Batch Normalization layers. Due to the limit of 8GB VRAM from RTX 2070 Super GPU, we set the crop size 416×416 for training with batch size 16.

4.2 Results

L2 regularization is commonly used in training many CNN models as a regularizer. However, we want to explore whether it also properly works in training from pretrained weights. In the following tables, '-**V**' and '-**C**' mean that given networks are trained with **V**OC2012 and **C**ityScapes dataset, respectively.

Model	$\lambda = 0$	$\lambda = 10^{-4}$	$\lambda = 2 \cdot 10^{-4}$	$\lambda = 5 \cdot 10^{-4}$
ResNet-50-V	**90.62/61.97**	90.60/61.88	90.59/61.71	90.69/61.94
ResNet-50-C	92.50/58.57	92.47/58.57	92.74/59.44	**92.85/60.35**
FCN-8s-V	**92.82/69.26**	92.72/69.02	92.78/68.94	92.51/68.14
FCN-8s-C	**94.31/67.03**	94.30/66.94	94.31/66.88	94.10/65.52

Table 1: The L2 Regularization Effect (pAcc/mIoU)

Except for the ResNet-50-C case, L2 regularization does not improve the accuracy in Semantic Segmentation. We set the most highest accuracy case to the baseline in the Table 1.

Model	Baseline	Adversarial	$T_d = 0.5$	$T_d = 2$
ResNet-50-V	90.62/61.97	90.68/62.03	90.68/61.86	**90.74/62.37**
ResNet-50-C	**92.85/60.35**	92.83/60.16	92.64/59.35	92.83/60.17
FCN-8s-V	92.82/69.26	92.83/69.33	92.79/69.10	**92.78/69.36**
FCN-8s-C	94.31/67.03	94.32/67.30	**94.29/67.36**	94.30/67.10

Table 2: T_D Test ($T_G = 1$)

Model	Baseline	Adversarial	$\mathbf{T_G = 0.5}$	$T_G = 2$
ResNet-50-V	90.62/61.97	90.68/62.03	**90.88/62.54**	90.52/61.72
ResNet-50-C	92.85/60.35	92.83/60.16	**93.02/60.75**	92.70/60.22
FCN-8s-V	92.82/69.26	92.83/69.33	**92.90/69.82**	92.72/68.96
FCN-8s-C	94.31/67.03	94.32/67.30	**94.36/67.42**	94.28/67.38

Table 3: T_G Test ($T_D = 1$)

Table 2 and Table 3 indicate the accuracies with using our adversarial framework with T_D and T_G. In Table 2, we can observe that the accuracy increase is inconsistent in terms of discriminator temperatures. Moreover, *ResNet-50-C* case shows the training without the discriminator overwhelms other trials. In

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

contrast, $T_G = 0.5$ cases in Table 3 are the best result in terms of pAcc and mIoU, among the all trials in both tables above. To sum up, the $T_G = 0.5/T_D = 1$ case strengthens the adversarial framework with the optimal λ values.

Model	Train Params	D params	T_{train}	T_{Adv}
ResNet-50-V	23.6M	2.79M	2.6mins	4.1mins
ResNet-50-C	23.6M	2.78M	2.6mins	3.0mins
FCN-8s-V	134.5M	2.79M	6.8mins	9.2mins
FCN-8s-C	134.4M	2.78M	3.5mins	4.5mins

Table 4: Training Information of Two Models

Table 4 shows the number of parameters and training times. With the smaller number of discriminator parameters, the training time is increasing no more than 3 minutes per epoch. The time increment comes from the factors such as passing both the groundtruth feature map and the predicted feature map into the discriminator and the backpropagation in the segmentation model.

5 Conclusion

In this paper, we introduced the adversarial framework with temperatures. Our experiments concatenate a discriminator to a segmentation model and apply temperatures to both the discriminator loss and the adversarial loss, resulting in the increased model accuracies on top of L2 regularization. Our results show that penalizing the segmentation model with the adversarial loss prevents models from overfitting. Even though our backbone models are not state-of-the-art, our approach is consistent and efficient, in terms of improvement and time consumption, respectively. In future work, we will evaluate our methods in various environments and investigate more efficient feature-based regularization methods.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

[2] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324, 2019.

[3] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2019.

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

[6] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

[7] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[8] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017.

[9] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[10] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

[11] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[12] Rafael Müller, Simon Kornblith, and Geoffrey Hinton. When does label smoothing help? *arXiv preprint arXiv:1906.02629*, 2019.

[13] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang. Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2604–2613, 2019.

[14] Wei-Chih Hung, Yi-Hsuan Tsai, Yan-Ting Liou, Yen-Yu Lin, and Ming-Hsuan Yang. Adversarial learning for semi-supervised semantic segmentation. *arXiv preprint arXiv:1802.07934*, 2018.

[15] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[16] Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, and Rob Fergus. Regularization of neural networks using dropconnect. In *International conference on machine learning*, pages 1058–1066. PMLR, 2013.

[17] John Moody, Stephen Hanson, Anders Krogh, and John A Hertz. A simple weight decay can improve generalization. *Advances in neural information processing systems*, 4(1995):950–957, 1995.

[18] Andrew Y Ng. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, page 78, 2004.

[19] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[20] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[21] Mark Everingham and John Winn. The pascal visual object classes challenge 2012 (voc2012) development kit. *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep*, 8, 2011.

[22] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 International Conference on Computer Vision*, pages 991–998. IEEE, 2011.

[23] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[24] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.