Learning pixel objectness with Positive and Unlabelled examples

First Author
Institution1
Institution1 address
firstauthor@i1.org

Second Author
Institution2
First line of institution2 address
secondauthor@i2.org

July 13, 2017

Abstract

The ABSTRACT HERE

1 Introduction

The lack of "gold standard" annotations becomes the bottleneck for semantic segmentation.

The state-of-art deep learning algorithms for semantic segmentation [3] usually suffered from the lack of large-scale annotated dataset. Most of these methods assume the existence of precise, consistent and exhaustive annotations. However, collecting such perfect segmentations manually on a large scale is expensive and time-consuming. Millions of images are available through Flikr, ImageNet and many other sources on the Internet but only a few of them [1, 4, 2] were well-annotated for semantic segmentation tasks.

A typical method to overcome the lack of training samples is to use the pre-trained weights of the convolutional layers from the classification models because collecting the classification annotations is easier to scale up compared to segmentation annotations. However, the neural network architecture design for image semantic segmentation does not necessarily follow the design of image classification architectures. For instance, the segmentation models require less subsample pooling layers than the classification models do to keep more local information to locate the object. The difficulty for training comparable representation of the images in the context of semantic segmentation arises from the lack of "gold standard" on a large scale. If the "gold standard" annotation assumption can be released, collecting segmentation annotation would be much easier to scale. That leads us to explore methods to learn image representation in the presence of annotation noise.

This paragraph should explain the difficulties for collecting segmentation annotation, including what could be the main errors in annotation.

The benchmark datasets usually provide perfect segmentations with all instance annotated (exhaustive) and no misannotated instance (precise). However, it is natural for human beings to make mistakes while annotating due to the lack of expertise, the inherent ambiguity of tasks or unconscious bias. Enormous efforts are required to correct the mistakes made, including double-checking the annotations over and over again and ensembling opinions from multiple annotators. Otherwise, these errors may lead to annotations that contain misannotated instances, misclassification of the instances and unannotated instances.

This paragraph should reason the idea obtain pre-trained features by learning "objectness" with Positive and Unlabeled examples.

For end-to-end models contained convolutional neural networks, the purpose of a learning objective is not only to train a classifier with minimum errors but also to learn satisfying intermediate convolutional filters. It is clear that training with the noisy annotations would lead to higher segmenting errors than training with the correct annotations if the validation samples had correct annotations. It is nevertheless unclear how the annotation noises influence the learned image representation.

That leads to our research questions:

1. How to compensate the classification bias introduced by misannotation and inexhaustive annotations with the appropriate prior knowledge.

2. How do the label noises influence the learned representation.

Briefly formulate the problem.
The Semantic Segmentation problem can be considered as per-pixel classification. Each of the pixels is assigned a label of either 0, indicating a background pixel, or $k \in 1, \ldots, K$ denoting a foreground pixel corresponding to an instance from one of the K categories. The aforementioned errors can be interpreted by the pixel label flipping: *misannotation* is label flipped from 0 to k, *inexaustive annotation* flipped from k to 0, and *misclassification* flipped from i to j, where $i, j, k \in 1, \ldots, K$.
Misannotating
One hypothesis made in this work is that the *misannotated* instances can still possibly provide information to learn visual representation for objects, assuming the "objectness" is shared in the low-level features. Supposing a dog toy instance is wrongly annotated as a dog, given the "toy" is not one of the pre-defined categories while the "dog" is, the misannotation error would introduce bias to the classification layer but not necessarily to the convolutional layers, especially the bottom layers. The bottom-level features are believed to be shared among different categories and thus should be more robust to the misannotation error than the top-level features if the misannotated instance is still visually distinguishable and semantically meaningful.
Inexaustive annotating
The exhaustive annotations, on the other hand, can introduce bias to both the decoding layer and the encoding layers because they negatively contribute to the activations in all the layers. Therefore, the inexhaustive annotations need to be properly handled given the prior knowledge modeling the missing pattern of the annotations. Given that we believe any annotated instance provide information, all the foreground pixels that correspond to the annotated instances become reliable and the background pixels may contain both the true background pixels and object pixels unannotated. That satisfies a Positive and Unlabeled learning setup where the training dataset contains only the positive examples and unlabeled examples that are the mixed of the positive samples and negative samples.
Table of contents
Related works are summarized in the next section. In Section 3 we judge the possibility of learning convolutional representation with misannotations by learning to predict the pixel objectness. Section 4 explored the methods to compensate the inexaustive annotations in a Positive and Unlabeled Learning setup. Features learned by predicting the pixel objectness with inexaustive annotations were then validated with experiments described in Section 5.

2 Related work

Deep Learning with Noisy Labels *Robustness analysis* Deep Learning is Robust to Massive Label Noise [7]

Entropy regularization Training deep neural networks on noisy labels with bootstrapping [6] Regularizing Neural Networks by Penalizing Confident Output Distributions [5]

Transfer Learning

Positive and Unlabeled Learning

3 Pre-train features by learning "objectness"

4 Positive and Unlabeled Learning

One sentence summary of Positive and Unlabeled Learning

Formulation *This part should explain the Positive and Unlabeled Learning setup with mathematical representation when necessary.*

Weighted Logistic Regression *This part should discuss the linear model for observing positive conditioning on true positive and its relationship to changing the class weight.*

Initial Repr.	pixel acc.	mean acc.	mean IU	f.w. IU
ImageNetModel				
CompleteCategory				
PixelObjectness				
RandomCategory				
FromScratch				

Table 1: Performances of FCN with Alexnet trained to segment 5 categories from the PASCAL VOC2011 dataset with different representation initializations. *CompleteCategory* is the model pre-trained to segment the other 15 categories from the PASCAL VOC2011 dataset; The *PixelObjectness* model was pre-trained to distinguish the instance against the background; The *RandomCategory* model was pre-trained with instances assigned random categories from the other 15 categories.

Annotation	Loss	acc.	prec.	rec.	F_1
Complete	CrossEntropyU	0.87	0.88	0.82	0.85
50%(P+N)	CrossEntropyU	0.83	0.84	0.78	0.80
50%P+U	CrossEntropyU	0.61	0.92	0.30	0.38
50%P+U	WeightedU	0.66	0.93	0.39	0.51
50%P+U	ExponentialU	0.82	0.86	0.73	0.78
50%P+U	BootstrapHard	0.74	0.81	0.60	0.67
50%P+U	DropoutRegularization				

Table 2: Image classification with positive examples partially annotated. The complete dataset contains images from CIFAR10 as the **positive** (P) set and images from CIFAR110 as the **negative** (N) set. The unannotated positive examples from P set construct the **unlabeled** (U) set together with the N set.

Exponential Loss for unlabeled examples *This part should explain why the exponential loss could perform better than the cross-entropy loss, potentially with a figure of 2D Gaussians.*

This paragraph should explain why fade-in was introduced to avoid all-positive inital prediction

This part should explain the influence of the imbalanced problem and how to overcome.

Figure 1: Varying the number of categories while pre-training the representation and the pre-trained weights were fine-tuned to segment 5 categories from the PASCAL VOC2011 dataset.

Annotation	Loss	pixel acc.	mean acc.	mean IU	f.w. IU
Complete	CrossEntU				
50%(P+N)	CrossEntU				
50%P+U	CrossEntU				
50%P+U	WeightedU				
50%P+U	ExponentialU				
50%P+U	BootstrapHard				
50%P+U	DropoutReg				

Table 3: Image semantic segmentation with images contain single instance only from the PASCAL VOC2011 segmentation dataset. The complete **positive** (P) set denotes the foreground instances and the **negative** (N) set consists of the background. The unannotated instances from P set construct the **unlabeled** (U) set together with the N set.

5 Results

6 Conclution

References

[1] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015.

[2] Tsung-Yi Lin, Michael Maire, Serge Belongie, James

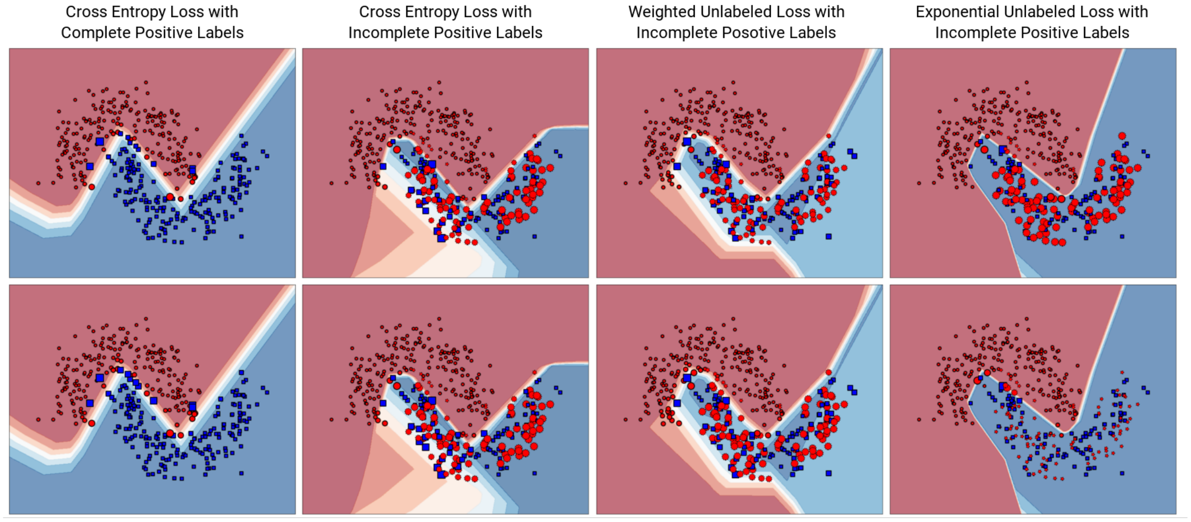

Figure 2: 2D moons dataset with non-linear separable decision boudary. Four hundreds samples per class were drawn randomly from two interleaving half circles with noises added with a minor standard deviation. A **red circle** indicates an example labelled as positive whilst a **blue square** indicates the example has a negative label. The **leftmost** figures have complete positive labels, meaning the positive and negative labels are all correct, whereas, in **the other figures** only half of the positives were correctly labelled and the rest were mixed with the negative samples. The **background colors** represent the probability for the area to be positive given by the classifier trained with the given samples and labels: **red** for high probability areas, **blue** for low probability areas and **white** for the class transition areas, i.e.decision boundaries. The **size of the markers** in the top row denotes the per-class normalized training losses and the **size of the markers** in the bottom row the per-class normalized derivatives w.r.t the output of the last layer for the trained Multilayer Perceptron (MLP) with the different losses.

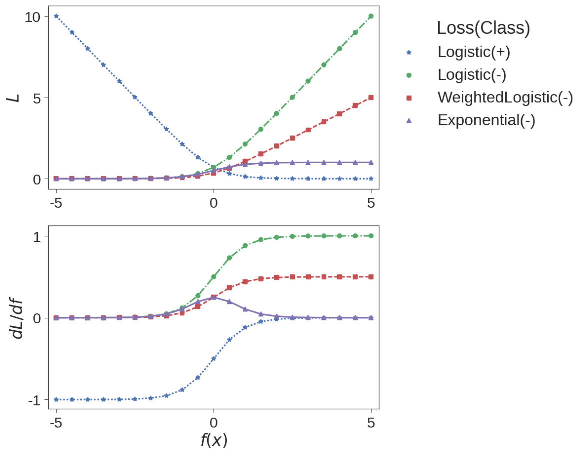

Figure 3: The Logistic Loss, Weighted Logistic Loss, Exponential Loss and their dirivatives with respect to the model output.

Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[3] 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.

[4] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 891–898, 2014.

[5] Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, 2017.

[6] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

[7] David Rolnick, Andreas Veit, Serge Belongie, and Nir Shavit. Deep learning is robust to massive label noise. *arXiv preprint arXiv:1705.10694*, 2017.

Figure 4: Varying percentage of annotated positives 10%, 20%, 50%, 80% and 100% with images from CIFAR10 as the positives and images from CIFAR110 as the negatives.