基于深度神经网络的迁移学习系列二

ZZL

2017 年 12 月 2 日

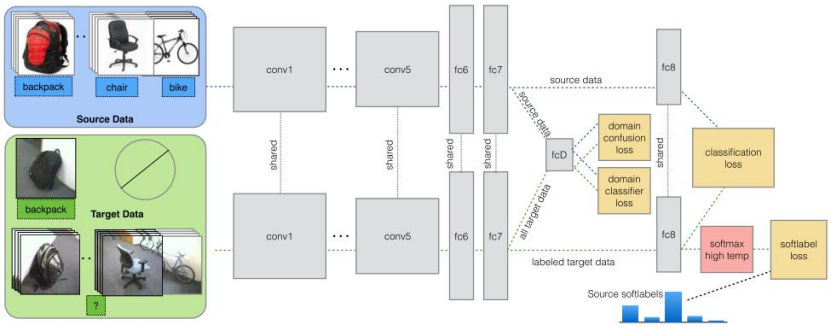
Introduction

近些年对抗生成网络（GAN）的出现，给 AI 届打了一针鸡血，各种基于对抗网络的文章层出不穷，在迁移学习领域也有一些文章用到了对抗的思想。对抗网络的优化目标是使得生成的数据分布与真实的数据相同（实际中很难收敛到该最优值），而在迁移学习中有一个优化目标是使得源域与目标域之间的数据分布尽可能的接近，这两个优化目标很相似，所以在迁移学习中可以借鉴对抗的方法。关于 GAN 大家可以看《Generative Adversarial Nets》。本文介绍以下两篇文章《Simultaneous Deep Transfer Across Domains and Tasks》，《Adversarial Discriminative Domain Adaptation》。关于文章中的一些损失的涉及，是跟 GAN 优化时候的特性有关，本人目前没有调试过 GAN 网络，因此不能给出具体的解释。

一． Simultaneous Deep Transfer Across Domains and Tasks

这篇文章与《Deep Transfer Network: Unsupervised Domain Adaptation》很相似，也可以看作是 JDA 的深度网络实现版，但他采用了对抗的思想，使得在适配边缘分布的时候表现的更好。在适配边缘分布时候，该文在网络对 source 进行训练的时候，把 source 的每一个样本处于每一个类的概率都记下来，然后，对于所有样本，属于每一个类的概率就可以通过求和再平均得到。这样的目的是根据 source 中的类别分布关系，来对 target 做相应的约束。该文的假设是目标域有少量的标签，这个模型稍作改变也能处理没有标签的情况。

网络的结构如下：



网络的损失如下：

$$L(x_S, y_S, x_T, y_T, \theta_D; \theta_{repr}, \theta_C) = L_C(x_S, y_S, x_T, y_T; \theta_{repr}, \theta_C) + \lambda L_{conf}(x_S, x_T, \theta_D; \theta_{repr}) + vL_{soft}(x_T, y_T; \theta_{repr}, \theta_C)$$

L 是网络总的损失，L_C是网络的分类损失，L_{conf}是两域边缘密度的损失，L_{soft}是标签分布的损失，也就是适配条件概率的损失

1、L_C

$\mathcal{L}_C(x, y; \theta_{\text{repr}}, \theta_C) = -\sum_k \mathbb{1}[y = k] \log p_k$，网络的输出是输入数据属于每个类别的概率，这里是用交叉熵损失$p = \text{softmax}(\theta_C^T f(x; \theta_{\text{repr}}))$。$\theta_{repr}$表示对分类器的输入（前面是特征提取层，后面是分类层），θ_C是分类器的权重信息。

2、L_{conf}

适配边缘分布采用了对抗的思想。“生成器”使得源域和目标域的数据分布尽可能的接近，判别器判断数据是来自源域还是目标域，两者迭代优化。

$\mathcal{L}_D(x_S, x_T, \theta_{\text{repr}}; \theta_D) = -\sum_d \mathbb{1}[y_D = d] \log q_d$是判别器，判断数据来自哪个域。它希望源域和目标域之间的差异越大越好。

$\mathcal{L}_{\text{conf}}(x_S, x_T, \theta_D; \theta_{\text{repr}}) = -\sum_d \frac{1}{D} \log q_d$，是“生成器”，把判别器得出的结果与均匀分布进行比较，进而得到源域和目标域分布之间的差异。它希望判别器不能分辨数据来源，这样判别器得到的结果也是均匀分布，也就是说源域和目标域几乎相同。

$$\min_{\theta_D} \mathcal{L}_D(x_S, x_T, \theta_{\text{repr}}; \theta_D)$$
$$\min_{\theta_{\text{repr}}} \mathcal{L}_{\text{conf}}(x_S, x_T, \theta_D; \theta_{\text{repr}}).$$

这是优化目标，迭代优化两个公式，直至满足收敛条件。

3、L_{soft}

假设一共有 k 个类别，对所有的源域数据求出其属于每个类的概率。对于所有的样本，属于每一个类的概率通过求和再平均得到。具体公式为$P_k = \sum_{i=1}^{n_s} p_{ik}/n_s$。作者认为源域和目标域每个类别的概率应该尽可能的接近。

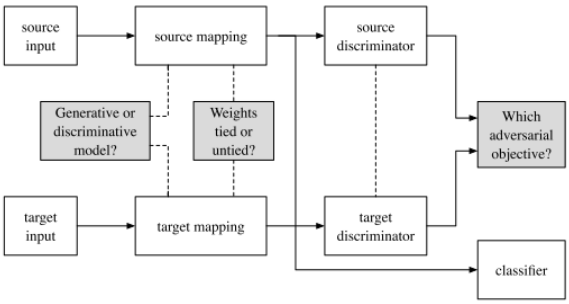
$$\mathcal{L}_{\text{soft}}(x_T, y_T; \theta_{\text{repr}}, \theta_C) = -\sum_i l_i^{(y_T)} \log p_i$$

该文的贡献在于将对抗思想融到了边缘适配中，并且适配了另外一种形式的条件概率，并且起名为 task transfer，第一眼成功的唬住了人。

二． Adversarial Discriminative Domain Adaptation

这篇文章介绍了很多基于 Discriminative Adaptation 的工作，可以作为一个小综述来阅读。由“基于深度神经网络的迁移学习系列一”，我们知道网络设计的有一类区别是权重共享的问题，本文的网络是不共享权重，作者认为用同一个网络来处理两个不同分布的数据会有缺陷，类似思想文章有《Beyond Sharing Weights for Deep Domain Adaptation》。这篇文章没有适配条件概率。

网络结构如下：



1、分类损失

$$\min_{M_s,C} \mathcal{L}_{\text{cls}}(\mathbf{X}_s, Y_t) = \mathbb{E}_{(\mathbf{x}_s,y_s)\sim(\mathbf{X}_s,Y_t)} -\sum_{k=1}^{K} \mathbb{1}_{[k=y_s]} \log C(M_s(\mathbf{x}_s))$$

这里的 M(x)是经过原始输入数据经过网络映射后的数据。这里的 C(M(x))，是判别器 C 对数据 M(x)的判别。

2、对抗损失

$$\begin{aligned}\mathcal{L}_{\text{adv}_D}(\mathbf{X}_s, \mathbf{X}_t, M_s, M_t) = \\ -\mathbb{E}_{\mathbf{x}_s\sim\mathbf{X}_s}[\log D(M_s(\mathbf{x}_s))] \\ -\mathbb{E}_{\mathbf{x}_t\sim\mathbf{X}_t}[\log(1 - D(M_t(\mathbf{x}_t)))]\end{aligned}$$

这个是分类的损失，判别数据来源于哪个域。D 是分别器，这个函数希望将源域数据分到正类，目标域数据分到父类。

$$\mathcal{L}_{\text{adv}_M}(\mathbf{X}_s, \mathbf{X}_t, D) = -\mathbb{E}_{\mathbf{x}_t\sim\mathbf{X}_t}[\log D(M_t(\mathbf{x}_t))].$$

这个是对抗损失，希望判别器将目标域数据分到正类，两个函数的优化目标正好相反。

3、网络损失

$$\min_{M_s,C} \mathcal{L}_{\text{cls}}(\mathbf{X}_s, Y_s) =$$
$$-\mathbb{E}_{(\mathbf{x}_s,y_s)\sim(\mathbf{X}_s,Y_s)} \sum_{k=1}^{K} \mathbb{1}_{[k=y_s]} \log C(M_s(\mathbf{x}_s))$$
$$\min_{D} \mathcal{L}_{\text{adv}_D}(\mathbf{X}_s, \mathbf{X}_t, M_s, M_t) =$$
$$-\mathbb{E}_{\mathbf{x}_s\sim\mathbf{X}_s}[\log D(M_s(\mathbf{x}_s))]$$
$$-\mathbb{E}_{\mathbf{x}_t\sim\mathbf{X}_t}[\log(1 - D(M_t(\mathbf{x}_t)))]$$
$$\min_{M_s,M_t} \mathcal{L}_{\text{adv}_M}(\mathbf{X}_s, \mathbf{X}_t, D) =$$
$$-\mathbb{E}_{\mathbf{x}_t\sim\mathbf{X}_t}[\log D(M_t(\mathbf{x}_t))].$$

参考文献

[1] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative Adversarial Nets[J]. Advances in Neural Information Processing Systems, 2014:2672-2680.

[2] Ganin Y, Lempitsky V. Unsupervised Domain Adaptation by Backpropagation[J]. 2014:1180-1189.

[3] Tzeng E, Hoffman J, Saenko K, et al. Adversarial Discriminative Domain Adaptation[J]. 2017.