Supplementary Material

Proof of Theorem

We introduce the derivation of theorem of the main paper. The ideal joint hypothesis is defined as $h^* = \arg\min_{h \in H} \big(R_{\mathcal{S}}(h^*) + R_{\mathcal{T}}(h^*)\big)$, and its corresponding error is $C = R_{\mathcal{S}}(h^*) + R_{\mathcal{T}}(h^*)$, where R denotes the expected error on each hypothesis.

We consider the pseudo-labeled target samples set $T_l = \big\{(x_i, \hat{y}_i)\big\}_{i=1}^{m_t}$ given false labels at the ratio of ρ. The distribution of the source samples is denoted as $\mathcal{S}$; that of the target samples, as $\mathcal{T}$; and that of the pseudo-labeled target samples, as $\mathcal{T}_l$. The minimum shared error on $\mathcal{S}, \mathcal{T}_l$ is denoted as C'. Then, the following inequality holds:

$$\forall h \in H, R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \tfrac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_{\mathbf{X}}, \mathcal{T}_{\mathbf{X}}) + C$$

$$\leq R_{\mathcal{S}}(h) + \tfrac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}_{\mathbf{X}}, \mathcal{T}_{\mathbf{X}}) + C' + \rho$$

Proof. The probabiliy of false labels in the pseudo-labeled set T_l is ρ. When we consider 0-1 loss function for l, the difference between the error based on the true labeled set and pseudo-labeled set is

$$|l(h(x_i), y_i) - l(h(x_i), \hat{y}_i)| = \begin{cases} 1 & y_i \neq \hat{y}_i \\ 0 & y_i = \hat{y}_i \end{cases}$$

Then, the difference in the expected error is,

$$\mathbf{E}[|l(h(x_i), y_i) - l(h(x_i), \hat{y}_i)|] \leq |R_{\mathcal{T}_l}(h) - R_{\mathcal{T}}(h)| \leq \rho$$

From the characteritic of the loss function, the triangle inequality will hold, then

$$\begin{aligned} R_{\mathcal{S}}(h) + R_{\mathcal{T}}(h) &= R_{\mathcal{S}}(h) + R_{\mathcal{T}}(h) - R_{\mathcal{T}_l}(h) + R_{\mathcal{T}_l}(h) \\ &\leq R_{\mathcal{S}}(h) + R_{\mathcal{T}_l}(h) + |R_{\mathcal{T}_l}(h) - R_{\mathcal{T}}(h)| \\ &\leq R_{\mathcal{S}}(h) + R_{\mathcal{T}_l}(h) + \rho \end{aligned}$$

From this result, the main inequality holds. □

CNN Architectures and training detail

Four types of architectures are used for our method, which is based on (Ganin & Lempitsky, 2014). The network topology is shown in Figs 2, 3 and 4. The other hyperparameters are decided on the validation splits. In the all scenarios, the learning rate is set to 0.01. In the initial training step, The batchsize is set as 128. After the initial step, the batchsize for training F_t, F is set as 128, the batchsize for training F_1, F_2, F is set as 64 in all scenarios.

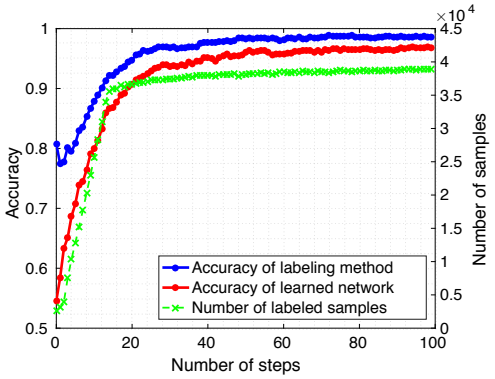

Figure 1. The behavior of our model when increasing the number of steps up to 100. Our model achieves accuracy of about 97%.

In MNIST→MNIST-M, the dropout rate used in the experiment is 0.2 for training F_t, 0.5 for training F_1, F_2. The number of iteraions per one step is set 2000. In MNIST→SVHN, we did not use dropout. We decreased learning rate to 0.001 after step 10. The number of iteraions per one step is set 3000. In SVHN→MNIST, the dropout rate used in the experiment is 0.5. The number of iteraions per one step is set 3000. In SYNDIGITS→SVHN, the dropout rate used in the experiment is 0.5. The number of iteraions per one step is set 5000. In SYNSIGNS→GTSRB, the dropout rate used in the experiment is 0.5. The number of iteraions per one step is set 5000.

Semi-supervised domain adaptation experiments

In semi-supervised domain adaptation in MNIST→SVHN, we used the same architecture we used in the unsupervised setting. For the first step of training, we trained all networks solely on source samples. We add randomly selected labeled target samples into pseudo-labeled target training sets. Other hyperparameters are the same as the ones used in unsupervised settings.

Supplementary experiments on MNIST→MNIST-M

We observe the behavior of our model when increasing the number of steps up to one hundred. We show the result in Fig. 1. Our model's accuracy gets about 97%. In our main experiments, we set the number of steps thirty, but from this experiment, further improvements can be expected when the number of steps is increased.

References

Ganin, Yaroslav and Lempitsky, Victor. Unsupervised domain adaptation by backpropagation. In *ICML*, 2014.

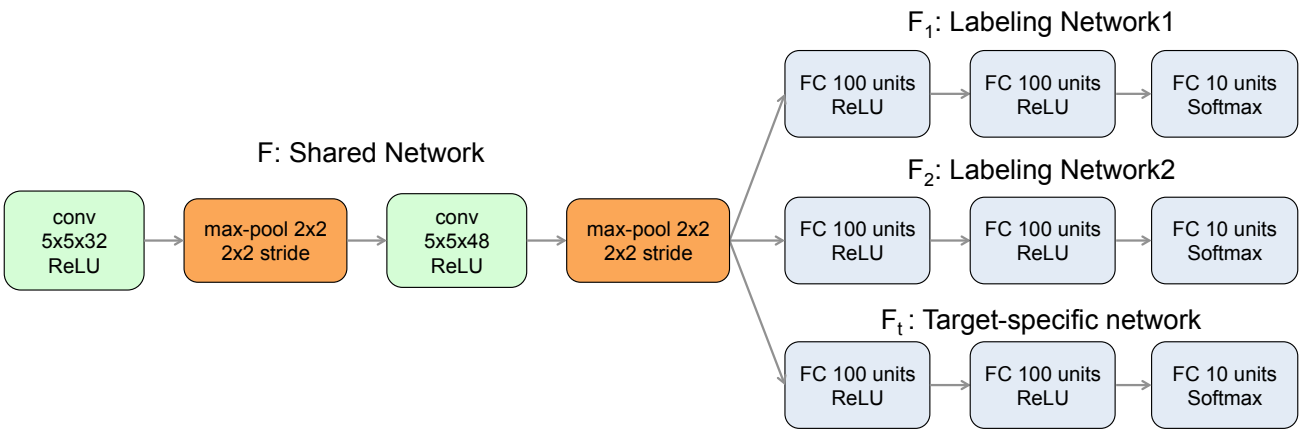

Figure 2. The architecture used for MNIST→MNIST-M. We added BN layer in the last convolution layer and FC layers in F_1, F_2. We also used dropout in our experiment.

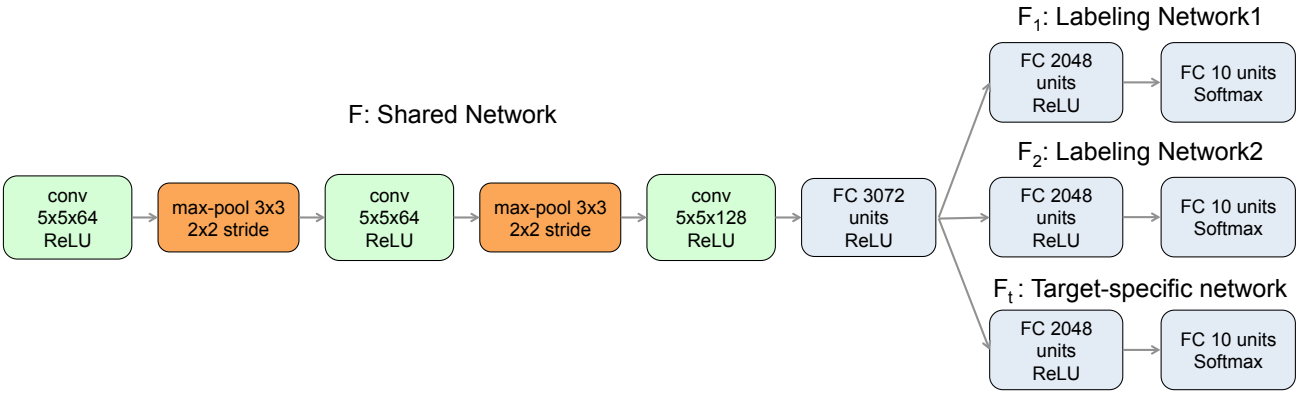

Figure 3. The architecture used for training SVHN. In MNIST→SVHN, we added a BN layer in the last FC layer in F. In SVHN→MNIST, SYN Digits↔SVHN, we added BN layer in the last convolution layer in F and FC layers in F_1,F_2 and also used dropout.

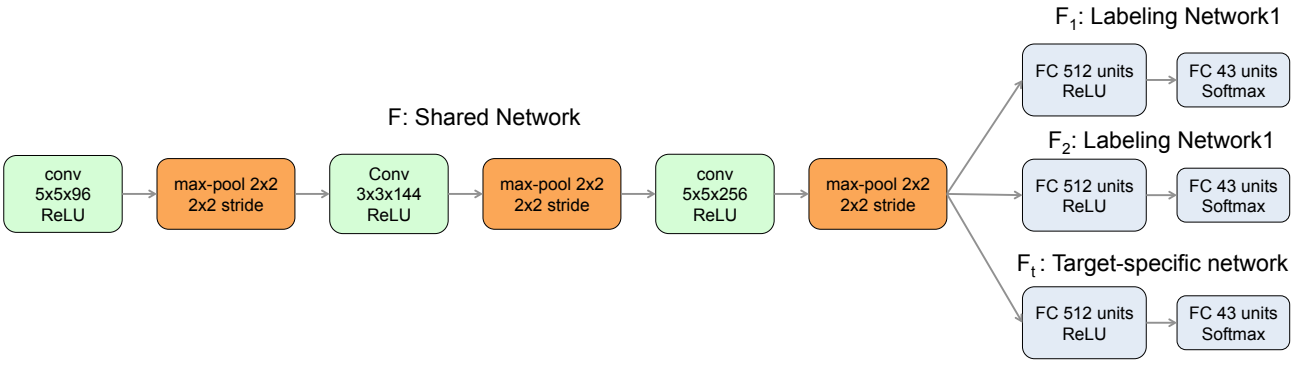

Figure 4. The architecture used in the adaptation Synthetic Signs→GTSRB. We added a BN layer after the last convolution layer in F and also used dropout.