UVA CS 6316: Machine Learning : 2019 Fall
Course Project: Deep2Reproduce @
https://github.com/qiyanjun/deep2reproduce/tree/master/2019Fall

Deep Asymmetric Multi-task Feature Learning

Reference:
Hae Beom Lee, Eunho Yang, and Sung Ju Hwang. Deep asymmetric multi-task feature learning. In Proceedings of the International Conference on Machine Learning (ICML), pages 2956–2964, 2018.

Reproduced by: Aobo Yang, Yujia Mu, David Yao, Qi Liu

2019/12/05

Motivation

- **Multi-task Learning(original)**
  - Learn deep representation
  - Problem of negative transfer
- **Asymmetric Multi-task Feature Learning(advanced)**
  - Learn deep representation
  - Prevent negative transfer
  - Unscalable and inefficient to deep learning
- **Deep Asymmetric Multi-task Feature Learning(more advanced)**
  - Learn deep representation
  - Prevent negative transfer
  - Less noisy representations
  - Scalable and efficient

Background

- **Multi-task learning:**
  - Definition:
    - Jointly train multiple task predictors
    - Allow knowledge transferring
  - Drawbacks:
    - Existence of negative transfer
- **Asymmetric Multi-task Feature Learning**
  - Definition:
    - Allow asymmetric knowledge transfer through inter-task regulation
    - Proposed to solve the above negative transfer
  - Drawbacks:
    - Fails to reconstructed from the combination of parameters for tasks
    - Poorly scalable

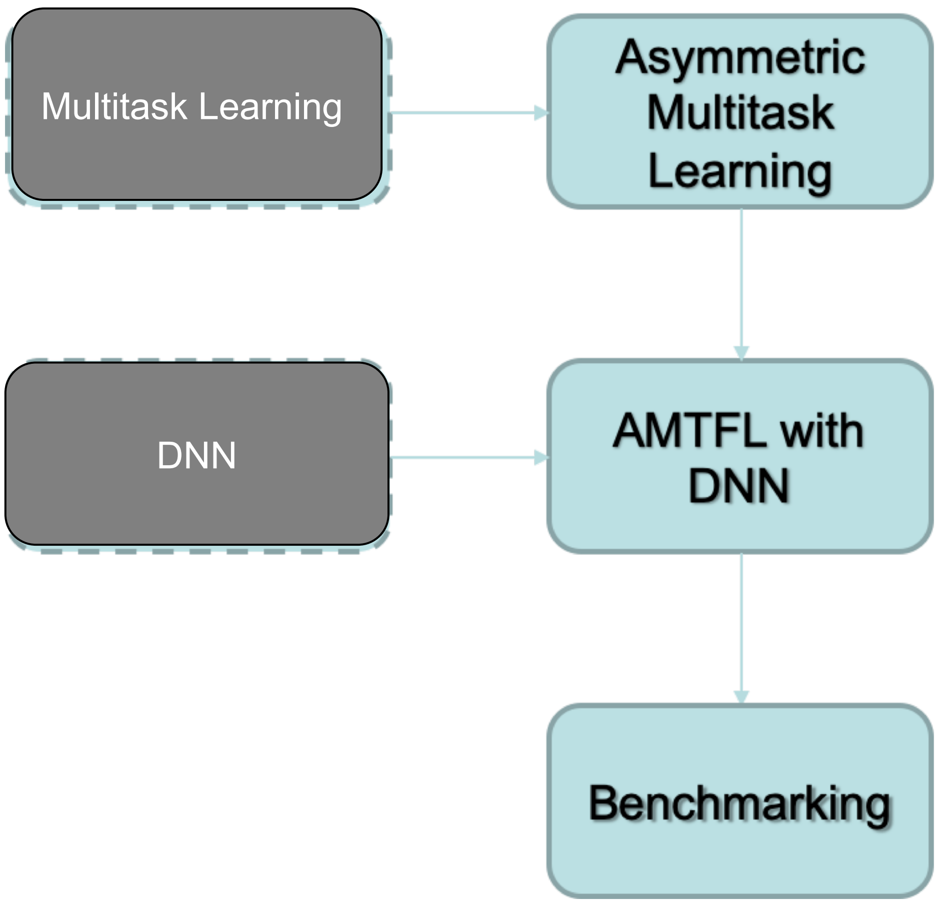
Related Work

- **Multitask Learning**
  - Definition: Jointly train a set of task predictors
  - Learning process allows knowledge transfer between predictors
  - Main limitation: cannot prevent negative transfer
- **Asymmetric Multitask Learning**
  - Definition: Break the symmetry in the knowledge transfer direction
  - Proposed in order to solve the problem of negative transfer
  - Main limitation: not scalable and hard to transfer to deep learning
- **Autoencoders**
  - Definition: transform input features and decode back to the original
  - Use a sparse nonlinear autoencoder term
  - Purpose: denoise of the latent features

Target Task

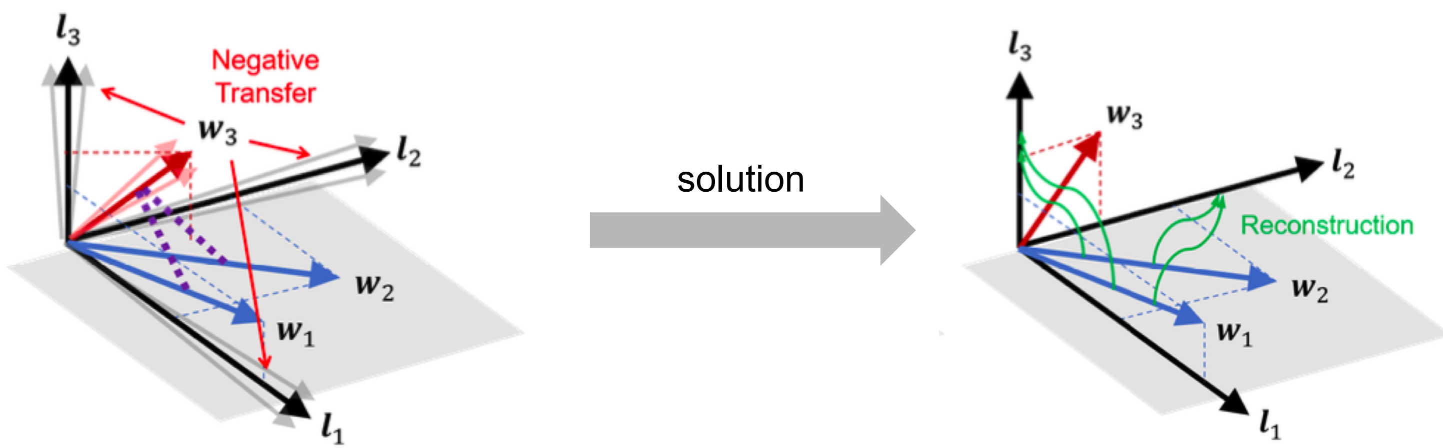
- **Asymmetric Multitask Feature Learning**
  - Learns latent features
  - Weighting up reliable task predictors; Weighting down the unpredictable ones (To prevent negative transfer)
  - Extending multitask learning to DNN with top layer feedback connections
- **Benchmarking**
  - Image classification using both the shallow and deep neural network on synthetic datasets
- **Expected Effects**
  - Better performance
  - More useful features learnt

An Intuitive Figure Showing WHY Claim



Proposed Solution

- Asymmetric multi-task feature learning (AMTFL): a completely new type of **regularization** to prevent the negative transfer from unreliable tasks to the shared latent features
    - Reconstruct latent features with task predictors' parameters
    - Enforce reconstruction to be done by reliable tasks only
    - Since task parameters are constructed by features, the reconstruction is like autoencoder



Multiple task parameters (*w*) are constructed by a set of latent features (*l*). Unreliable task (*w3*) pollutes the latent features.

Encourage asymmetric transfer by using reliable task parameters *(w1, w2)* to reconstruct the latent features *(l)*.

Implementation

The AMTFL framework is defined as

$$\min_{\boldsymbol{L},\boldsymbol{S},\boldsymbol{A}} \sum_{t=1}^{T} \left\{ (1+\alpha||\boldsymbol{a}_t^o||_1)\, \mathcal{L}(\boldsymbol{L}, \boldsymbol{s}_t; \boldsymbol{X}_t, \boldsymbol{y}_t) + \mu \left\|\boldsymbol{s}_t\right\|_1 \right\}$$
$$+ \gamma \|\boldsymbol{Z} - \sigma(\boldsymbol{ZSA})\|_F^2 + \lambda \left\|\boldsymbol{L}\right\|_F^2 . \qquad (6)$$

Where

$$\begin{cases} \boldsymbol{W} = \boldsymbol{LS} \\ \boldsymbol{L} \in \mathbb{R}^{d\times k} \\ \boldsymbol{S} \in \mathbb{R}^{k\times T} \\ \boldsymbol{Z} = \sigma(\boldsymbol{XL}) \\ \boldsymbol{A} \in \mathbb{R}^{T\times k} \end{cases}$$

- $\boldsymbol{W} = \boldsymbol{LS}$: The model parameters W can be decomposed to *L* and *S*
- $\boldsymbol{L} \in \mathbb{R}^{d\times k}$: L is a collection of k latent base
- $\boldsymbol{S} \in \mathbb{R}^{k\times T}$: S is the coefficient matrix for linearly combining the bases
- $\boldsymbol{Z} = \sigma(\boldsymbol{XL})$: Nonnegative feature matrix with ReLU nonlinear transformation
- $\boldsymbol{A} \in \mathbb{R}^{T\times k}$: Task-to-feature transfer matrix

Implementation

L1 regularization to make S sparse. The assumption is that each task sparsely rely on the shared latent vectors

Task loss

$$\min_{\boldsymbol{L},\boldsymbol{S},\boldsymbol{A}} \sum_{t=1}^{T} \left\{ (1+\alpha||\boldsymbol{a}_t^o||_1) \boxed{\mathcal{L}(\boldsymbol{L}, \boldsymbol{s}_t; \boldsymbol{X}_t, \boldsymbol{y}_t)} + \boxed{\mu \left\| \boldsymbol{s}_t \right\|_1} \right\}$$

$$+ \gamma \|\boldsymbol{Z} - \sigma(\boldsymbol{Z}\boldsymbol{S}\boldsymbol{A})\|_F^2 + \boxed{\lambda \left\| \boldsymbol{L} \right\|_F^2}. \qquad (6)$$

L2 regularization

Implementation

Sparsity regularization. Multiplied by the amount of training loss, making the ongoing transfer from hard task more sparse than the easy ones

Any generic loss

$$\min_{\boldsymbol{L},\boldsymbol{S},\boldsymbol{A}} \sum_{t=1}^{T} \left\{ (1 + \boxed{\alpha ||\boldsymbol{a}_t^o||_1}) \boxed{\mathcal{L}(\boldsymbol{L}, \boldsymbol{s}_t; \boldsymbol{X}_t, \boldsymbol{y}_t)} + \mu \left\| \boldsymbol{s}_t \right\|_1 \right\}$$

$$+ \boxed{\gamma \| \boldsymbol{Z} - \sigma(\boldsymbol{Z}\boldsymbol{S}\boldsymbol{A}) \|_F^2} + \lambda \left\| \boldsymbol{L} \right\|_F^2 . \quad (6)$$

Reconstruction regularization. The goal of the autoencoder-like term is to reconstruct feature *Z* from model output *ZS*

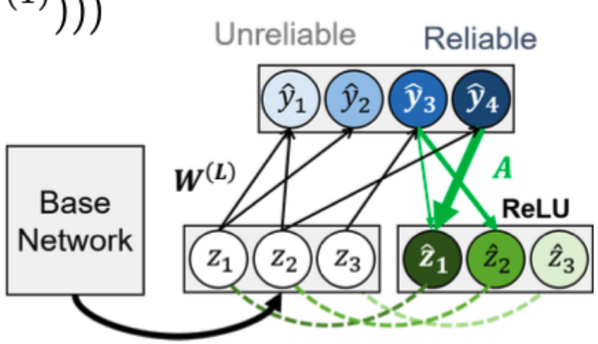
Implementation

- Since the framework considers asymmetric transfer in the feature space, it can be generalized to deep network with multiple layers
  - autoencoding regularization term Z is formulated at the second-last layer

$$\min_{\boldsymbol{A},\{\boldsymbol{W}^{(l)}\}_{l=1}^{L}} \sum_{t=1}^{T} \left\{ (1+\alpha||\boldsymbol{a}_t^o||_1)\,\mathcal{L}_t + \mu \left\| \boldsymbol{w}_t^{(L)} \right\|_1 \right\}$$
$$+ \gamma \left\| \sigma(\boldsymbol{Z}\boldsymbol{W}^{(L)}\boldsymbol{A}) - \boldsymbol{Z} \right\|_F^2 + \lambda \sum_{l=1}^{L-1} \left\| \boldsymbol{W}^{(l)} \right\|_F^2,$$

Where

$$\boldsymbol{Z} = \sigma(\boldsymbol{W}^{(L-1)}\sigma(\boldsymbol{W}^{(L-2)} \ldots \sigma(\boldsymbol{X}\boldsymbol{W}^{(1)})))$$



Data Summary

- **For shallow models:**
  - AWA-A
  - MNIST
  - School
  - Room

- **For deep models:**
  - MNIST-Imbalanced
  - CUB-200
  - AWA-C
  - ImageNet-Small

Experimental Results

- For shallow models:

Table 1. Performance of the linear and shallow baselines and our asymmetric multi-task feature learning model. We report the RMSE for regression and mean classification error(%) for classification, along with the standard error for 95% confidence interval.

	AWA-A	MNIST	School	Room
STL	37.6±0.5	14.8±0.6	10.16±0.08	45.9±1.4
GO-MTL	35.6±0.2	14.4±1.3	**9.87±0.06**	47.1±1.4
AMTL	33.4±0.3	12.9±1.4	10.13±0.08	40.8±1.5
NN	26.3±0.3	8.96±0.9	9.89±0.03	44.5±2.0
MT-NN	26.2±0.3	8.76±1.0	9.91±0.04	41.7±1.7
AMTFL	**25.2±0.3**	**8.68±0.9**	9.89±0.09	**40.4±2.4**

Experimental Results

- For deep models:

Table 2. Classification performance of the deep learning baselines and Deep-AMTFL. The reported numbers for MNIST-Imbalanced and CUB datasets are averages over 5 runs.

	MNIST-Imbal.	CUB	AWA-C	Small
CNN	8.13	46.18	33.36	66.54
MT-CNN	8.72	43.92	32.80	65.69
Deep-AMTL	8.52	45.26	32.32	65.61
Deep-AMTFL	**5.82**	**43.75**	**31.88**	**64.49**

Experimental Analysis

- For shallow models:
  - AMTFL outperforms the baselines on most datasets.
  - The only exception is the School dataset, on which GO-MTL obtains the best performance, but is due to the strong homogeneity among the tasks in this particular dataset.

- For deep models:
  - Deep-AMTFL outperforms all baselines, including MT-CNN and Deep-AMTL.
  - It shows the effectiveness of our asymmetric knowledge transfer from tasks to features, and back to tasks in deep learning frameworks.

Reproduction

- In our implementation, we tried to reproduce the results for the MNIST dataset
- We use the CNN (Lenet-Conv) mentioned in the paper
- Since the paper does not include all the hyperparameters, we cannot exactly reproduce the numbers, but the gap is trivial (~1%)
- Following is the comparison between with AMTFL and without it

Model	MT-CNN	Deep-AMTFL
Accuracy	0.9026	0.9301

Test

Test and compare the accuracies

```
[9] # test normal
    print('Normal model\'s accuracy:', test_acc
    
    # test AMTFL
    print('AMTFL model\'s accuracy:', test_acc
```

```
Normal model's accuracy: 0.9026
AMTFL model's accuracy: 0.9301
```

Conclusion and Future Work

- Propose a novel deep asymmetric multi-task feature learning framework, effectively prevent negative transfer resulting from symmetric influences of each task in feature learning.

- The predictors can asymmetrically affect the learning of shared representations by introducing an asymmetric feedback connections.

- Experimental results show that our model significantly outperforms asymmetric multi-task learning for both shallow and deep frameworks.

References

[1] Lee, G., Yang, E., and Hwang, S. Asymmetric multi-task learning based on task relatedness and confidence. In ICML. ICML, 2016

[2] Caruana, R. Multitask Learning. Machine Learning, 1997

[3] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1. chapter Learning Internal Represen- tations by Error Propagation, 1986.

[4] Argyriou, A., Evgeniou, T., and Pontil, M. Convex Multi- task Feature Learning. Machine Learning, 73(3):243– 272, 2008.

[5] Kumar, A. and Daume III, H. Learning task grouping and overlap in multi-task learning. In ICML, 2012.

Division of Work

	Slide	Coding	Presentation	Other
Qi Liu	motivation, background, related work	data preprocessing PCA	√	
David Yao	target task intuitive figure of why claim	AWA dataset data preprocessing	√	
Aobo Yang	Solution Implementation	Corss entropy loss AMTFL regularization Test and analysis	√	
Yujia Mu	Data Summary; Experimental results; Experimental analysis; Conclusion and future work.	MNIST-imbalanced; CNN Lenet-Conv	√	