Expert Gate: Lifelong Learning with a Network of Experts

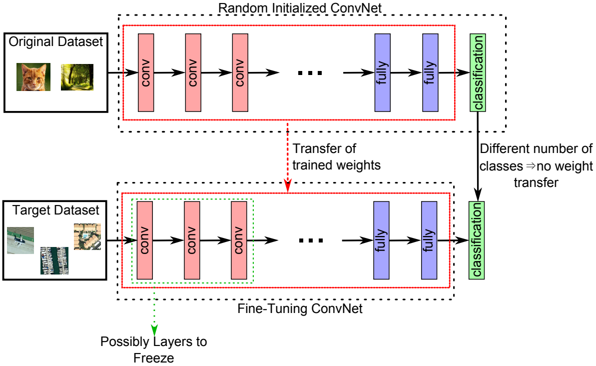
Rahaf Aljundi
Punarjay Chakravarty
Tinne Tuytelaars

July 14, 2017

DCC
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
UFMG
PATREO
PATTERN RECOGNITION FOR EARTH OBSERVATION
DCC - UFMG

Introduction

- Different models for different tasks trained on different datasets
- Expert on its own domain, but not on others
- Fine-tuning pre-trained model ⇒ performs well on the new task, but has a degraded performance on the old ones
- Catastrophic forgetting



Introduction

- Ideal ⇒ a system should be able to operate on different tasks and domains and give the best performance on each of them
- Simple solution: retrain a new model for each new task or domain
- Problems:
  - negative inductive bias ⇒ adversarial or not related tasks
  - fail to capture specialized information ⇒ hidden representation beneficial
  - re-train a network every time

Introduction

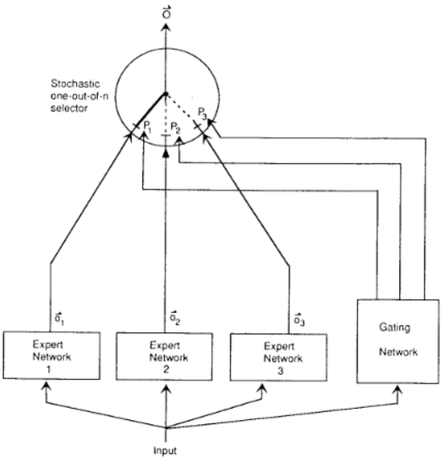
- Another solution: retrain the network using the current configuration as virtual labels

LEARNINGWITHOUTFORGETTING:
Start with:
θ_s: shared parameters
θ_o: task specific parameters for each old task
X_n, Y_n: training data and ground truth on the new task
Initialize:
$Y_o \leftarrow \text{CNN}(X_n, \theta_s, \theta_o)$ *// compute output of old tasks for new data*
$\theta_n \leftarrow \text{RANDINIT}(|\theta_n|)$ *// randomly initialize new parameters*
Train:
Define $\hat{Y}_o \equiv \text{CNN}(X_n, \hat{\theta}_s, \hat{\theta}_o)$ *// old task output*
Define $\hat{Y}_n \equiv \text{CNN}(X_n, \hat{\theta}_s, \hat{\theta}_n)$ *// new task output*
$\theta_s^*, \theta_o^*, \theta_n^* \leftarrow \underset{\hat{\theta}_s, \hat{\theta}_o, \hat{\theta}_n}{\operatorname{argmin}} \left(\lambda_o \mathcal{L}_{old}(Y_o, \hat{Y}_o) + \mathcal{L}_{new}(Y_n, \hat{Y}_n) + \mathcal{R}(\hat{\theta}_s, \hat{\theta}_o, \hat{\theta}_n) \right)$

Introduction

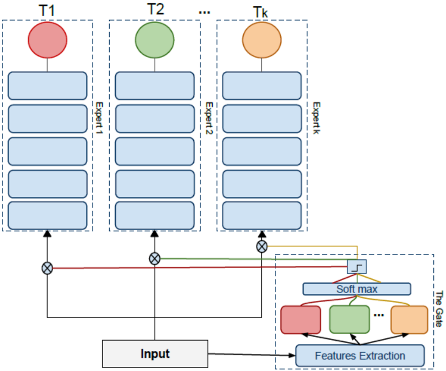
- Network of Experts: different specialist or expert models for different tasks
- New expert model is added whenever a new task arrives and knowledge is transferred from previous models

Adaptive Mixtures of Local Experts



Introduction

- Gate used to define which model to load and use
- Task recognizer that can tell the relevance of its associated task model for a given test sample



Proposed Method

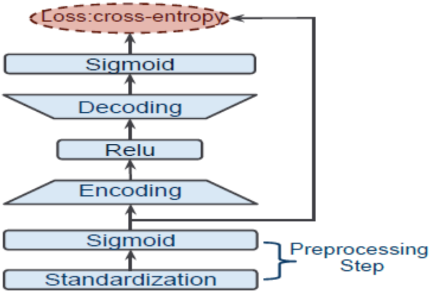
Overall

- Task: lifelong learning or sequential learning where tasks and data come one after another
- Learn a specialized model (expert) by transferring knowledge from previous tasks (most related previous task)
- Gating function that captures the characteristics of each task
    - Forwards the test data to the corresponding expert resulting in a high performance over all learned tasks

Proposed Method

The Autoencoder Gate

- Gates are undercomplete autoencoders, one for each task
- Autoencoder of one domain/task should be better at reconstructing the data of that task than the other autoencoders
- Preprocessing: subtracting mean and dividing by standard deviation (ImageNet dataset) followed by sigmoid $[0, 1]$



Proposed Method

Selecting the most relevant expert

- Softmax layer takes as input the reconstruction errors er_i from the different tasks autoencoders given a test sample x
- It gives a probability p_i for each task autoencoder indicating its confidence:
- $p_i = \frac{\exp(-er_i/t)}{\sum_j(-er_j/t)}$

Proposed Method

Measuring task relatedness

- A new task T_k associated with its data $D_k \Rightarrow$ an autoencoder for this task A_k
- Let T_a be a previous task with associated autoencoder A_a
- Objective: measure the task relatedness between task T_k and task T_a
- Using the validation set, compute average reconstruction errors Er_k and Er_a using A_k and A_a
- Relatedness between the two tasks is then computed:
- $Rel(T_k, T_a) = 1 - (\frac{Er_a - Er_k}{Er_k})$

Proposed Method

Algorithm

Algorithm 1 Expert Gate

Training Phase input: expert-models $(E_1, .., E_j)$, tasks-autoencoders $(A_1, .., A_j)$, new task (T_k), data (D_k) ; output: E_k

A_k =train-task-autoencoder (D_k)
(*rel*,*rel-val*)=select-most-related-task(D_k,A_k,$\{$A$\}$)
if *rel-val* >*rel-th* **then**
    E_k=LwF(E_{rel}, D_k)
else
    E_k=fine-tune(E_{rel}, D_k)
end if

Test Phase input: x ; output: prediction

i=select-expert$(\{A\}, x)$
prediction = activate-expert(E_i, x)

Experiments

Comparison with baselines

- Three image classification tasks:
    - MIT Scenes for scene classification
    - Caltech-UCSD Birds for finegrained bird classification
    - Oxford Flowers for finegrained flower classification

Experiments

Comparison with baselines

- Baselines:
  - Single jointly-trained model: all tasks and data **jointly** fine-tuned over an **unique** AlexNet
  - Multiple fine-tuned models: distinct AlexNet models are finetuned separately and oracle to define which one to use (best result)
  - Multiple LwF models: distinct learning-without-forgetting models using pre-trained AlexNet and an oracle gate
  - Single fine-tuned model: **one** AlexNet model **sequentially** fine-tuned on each task
  - Single LwF model: LwF **sequentially** applied to multiple tasks

Experiments

Comparison with baselines

Table 1. Classification accuracy for the sequential learning of 3 image classification tasks. Methods with * assume all previous training data is still available, while methods with ** use an oracle gate to select the proper model at test time.

Method	Scenes	Birds	Flowers	avg
Joint Training*	63.1	58.5	85.3	68.9
Multiple fine-tuned models**	63.4	56.8	85.4	68.5
Multiple LwF models**	63.9	58.0	84.4	68.7
Single fine-tuned model	63.4	-	-	-
	50.3	57.3	-	-
	46.0	43.9	84.9	58.2
Single LwF model	63.9	-	-	-
	61.8	53.9	-	-
	61.2	53.5	83.8	66.1
Expert Gate (ours)	63.5	57.6	84.8	68.6

Experiments

Gate Analysis

- Evaluate Expert Gate's ability in successfully selecting the relevant network(s) for a given test image
- Three more tasks:
  - Stanford Cars dataset for fine-grained car classification
  - FGVC-Aircraft dataset for fine-grained classification of aircraft
  - VOC Actions, the human action classification subset of VOC challenge 2012

Experiments

Gate Analysis

Table 2. Classification accuracy for the sequential learning of 6 tasks. Method with * assumes all the training data is available.

Method	Scenes	Birds	Flowers	Cars	Aircrafts	Actions	avg
Joint Training*	59.5	56.0	85.2	77.4	73.4	47.6	66.5
Most confident model	40.4	43.0	69.2	78.2	54.2	8.2	48.7
Expert Gate	60.4	57.0	84.4	80.3	72.2	49.5	67.3

Experiments

Gate Analysis

- Discriminative Task Classifier: MLP with 100 neurons, same input of Gate and number of outputs equal the number of expert networks

Table 3. Results on discriminating between the 6 tasks (classification accuracy)

Method	Scenes	Birds	Flowers	Cars	Aircrafts	Actions	avg
Discriminative Task Classifier - *using all the tasks data*	97.0	98.6	97.9	99.3	98.8	95.5	97.8
Expert Gate (ours) - *no access to the previous tasks data*	94.6	97.9	98.6	99.3	97.6	98.1	97.6

Experiments

Gate Analysis

Scenes as Flowers | Birds as Scenes | Flowers as Birds | Cars as Aircrafts | Aircrafts as Cars | Actions as Birds

Experiments

Task Relatedness Analysis

- Previous cases, the most related task was always Imagenet
  - similarity between the images of these different tasks and those of Imagenet
  - wide diversity of Imagenet classes enables it to cover a good range of these tasks
- Does this mean that Imagenet should be the only task to transfer knowledge from, regardless of the current task nature?

Experiments

Task Relatedness Analysis

- New tasks:
  - Google Street View House Numbers SVHN for digit recognition
  - Chars74K dataset for character recognition in natural images (Letters)
  - Mnist dataset for handwritten digits
- Select from previous tasks: two most related (Actions and Scenes), and the two most unrelated tasks (Cars and Flowers)

Experiments

Task Relatedness Analysis

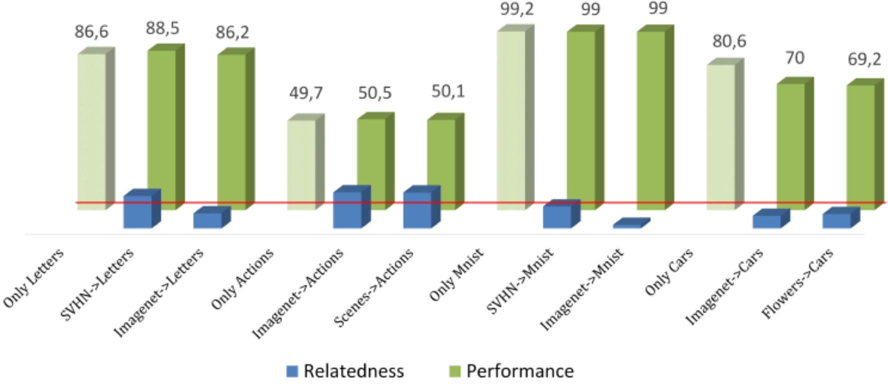


Figure 6. Relatedness analysis. The relatedness values are normalized for the sake of better visualization. The red line indicates our relatedness threshold value.

Experiments

Video Prediction for Autonomous Driving

- Task: given a sequence of 3 images, predict the next 3 images
- Prediction needs to be able to load the correct model for the current environment
- Three domains/tasks:
  - for Highway, the data from DFN (Dynamic Filter Network)
  - for Residential data, the two longest sequences from the KITTI dataset
  - for City data, the Stuttgart sequence from the CityScapes dataset

Experiments

Video Prediction for Autonomous Driving

Table 4. Video prediction results (average pixel L1 distance). For methods with * all the previous data needs to be available.

Method	Highway	Residential	City	avg
Single Fine-tuned Model	13.4	-	-	-
	25.7	45.2	-	-
	26.2	50.0	17.3	31.1
Joint Training*	14.0	40.7	16.9	23.8
Expert Gate (ours)	**13.4**	**40.3**	**16.5**	**23.4**

Experiments

Video Prediction for Autonomous Driving

Figure 7. Qualitative results for video prediction. From left to right: last ground truth image (in a sequence of 3); predicted image using sequential fine-tuning and using Expert Gate. Examining the lane markers, we see that Expert Gate is visually superior.

Conclusion

- Expert Gate's autoencoders can distinguish different tasks equally well as a discriminative classifier trained on all data
- They can be used to select the most related task and the most appropriate transfer method during training
- Proposed method outperforms not only the state-of-the-art but also joint training of all tasks simultaneously