Memory-based Parameter Adaptation

Sanket Vaibhav Mehta (SVM)

LLL Reading Group @ CMU

Motivation

- General scheme for continual, one-shot, incremental or life-long learning problems
- Challenges:
    - Sequential distributional shift (task)
    - Shifts in task label distributions (class/ label)
    - Domain shifts (domain)
- Common attributes (for models):
    - Negate the effects of catastrophic forgetting (**avoid negative backward transfer**)
    - Rapid acquisition of knowledge (**positive forward transfer**)
    - Unbalanced/ scarce data and good generalization

Method

Three components:

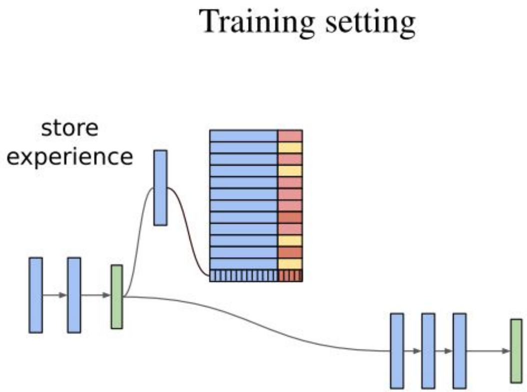
1. **Embedding** network f_γ - FNN or RNN
2. **Memory**, $M = \{(h_i, v_i)\}$ (circular buffer /FIFO)
    - keys $h_j \leftarrow f_\gamma(x_j)$
    - values $v_j \leftarrow y_j$
    - retrieval: KNN search on the keys with Euclidean distance
3. **Output** network g_θ

Training

- MLE for parameters (γ,θ)

$$p_{\text{train}}(y|x,\gamma,\theta) = g_\theta(f_\gamma(x))$$

- Classification: g_θ => softmax layer



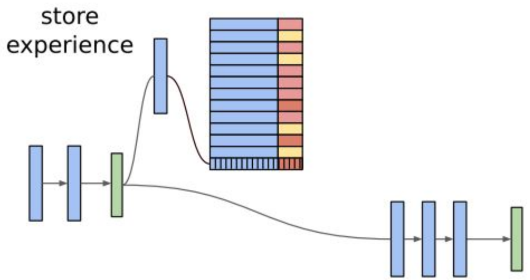
Training

- MLE for parameters (γ,θ)

$$p_{\text{train}}(y|x,\gamma,\theta) = g_\theta(f_\gamma(x))$$

- Classification: g_θ=> softmax layer

Training setting



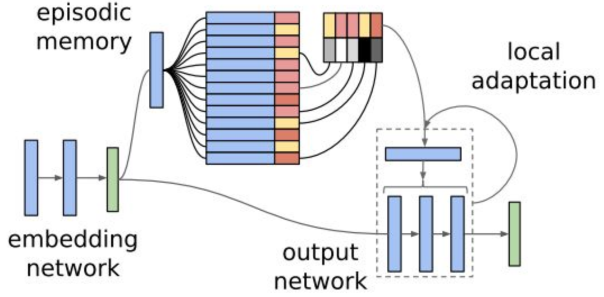
procedure MBPA-TRAIN
Sample mini-batch of training examples $B = \{(x_b, y_b)\}_b$ from training data.
Calculate the embedded mini-batch $B' = \{(f_\gamma(x_b), y_b) : x_b, y_b \in B\}$.
Update θ, γ by maximising the likelihood (1) of θ and γ with respect to mini-batch B
Add the embedded mini-batch examples B' to memory M: $M \leftarrow M \cup B'$.

Testing

- **Retrieval** $C = \{(h_k^{(x)}, v_k^{(x)}, w_k^{(x)})\}_{k=1}^{K}$

  $w_k^{(x)} \propto \mathrm{kern}(h_k^{(x)}, q) \quad (h, q) = \frac{1}{\epsilon + \|h - q\|_2^2}$

Testing setting



Testing

- **Retrieval** $C = \{(h_k^{(x)}, v_k^{(x)}, w_k^{(x)})\}_{k=1}^{K}$

  $$w_k^{(x)} \propto \text{kern}(h_k^{(x)}, q) \quad (h, q) = \frac{1}{\epsilon + \|h - q\|_2^2}$$

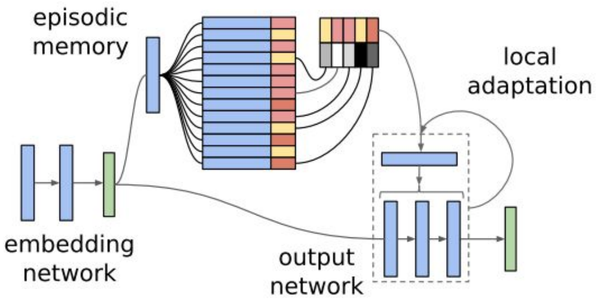
- **MbPA** adaptation

  correction

  $$\theta^x = \theta + \Delta_M(x, \theta)$$

  $$p(y|x, \theta^x) = p(y|x, \theta^x, C) = g_{\theta^x}(f_\gamma(x))$$

Testing setting



Testing

- **Retrieval** $C = \{(h_k^{(x)}, v_k^{(x)}, w_k^{(x)})\}_{k=1}^{K}$

  $w_k^{(x)} \propto \mathrm{kern}(h_k^{(x)}, q) \quad (h, q) = \frac{1}{\epsilon + \|h-q\|_2^2}$

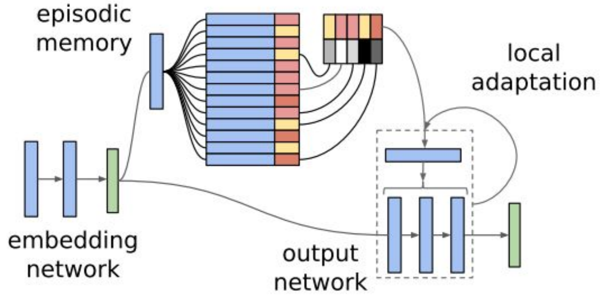
- **MbPA** adaptation

correction

$$\theta^x = \theta + \Delta_M(x, \theta)$$

$$p(y|x, \theta^x) = p(y|x, \theta^x, C) = g_{\theta^x}(f_\gamma(x))$$

Testing setting



procedure MBPA-TEST(test input: x, output prediction: $\hat{y}$)
  Calculate embedding $q = f_\gamma(x)$, and $\Delta_{\text{total}} \leftarrow 0$.
  Retrieve K-nearest neighbours to q and producing context, $C = \{(h_k^{(x)}, v_k^{(x)}, w_k^{(x)})\}_{k=1}^{K}$
  for each step of MbPA **do**
    Calculate $\Delta_M(x, \theta + \Delta_{\text{total}})$ according to (4)
    $\Delta_{\text{total}} \leftarrow \Delta_{\text{total}} + \Delta_M(x)$.
  Output prediction $\hat{y} = g_{\theta + \Delta_{\text{total}}}(h)$

MAP Interpretation of MbPA

- Posterior: $p(\theta^x|\theta, x_c, v_c, x) = \dfrac{p(v_c|x_c, \theta^x, x)p(\theta^x|\theta)}{p(v_c|\theta, x_c, x)}$
- Maximise the posterior over the context C w.r.t θ^x

$$\begin{aligned}\arg\max_{\theta^x} \mathbb{E}_C \{\log p(\theta^x|\theta, x_c, v_c, x)\} &= \arg\max_{\theta^x}\ \log p(\theta^x|\theta) + \mathbb{E}_C \{\log p(v_c|x_c, \theta^x, x)\} \\ &= \arg\max_{\theta^x}\ \log p(\theta^x|\theta) + \sum_{k=1}^{K} w_k^{(x)} \log p(v_k^{(x)}|h_k^{(x)}, \theta^x, x)\end{aligned}$$

$\log p(\theta^x|\theta) \propto -\frac{||\theta^x - \theta||_2^2}{2\alpha_M}$ (Gaussian prior on θ^x centered at θ)

- Contextual update:

$$\Delta_M(x, \theta) = -\alpha_M\ \nabla_\theta \sum_{k=1}^{K} w_k^{(x)} \log p(v_k^{(x)}|h_k^{(x)}, \theta^x, x)\Big|_\theta - \beta(\theta - \theta^x),$$

Attention to Local Fitting: MbPA for a regression

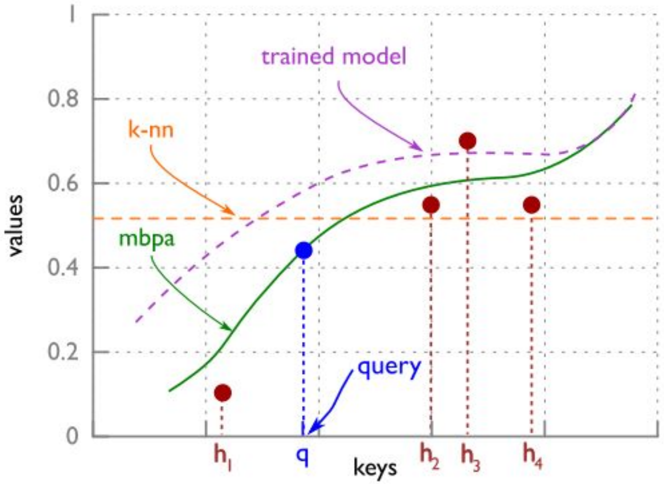


Figure 2: Illustrative diagram of the local fitting on a regression task. Given a query (blue), we retrieve the context from memory showed in red.

CL: Sequential Distributional Shift (Permuted MNIST)

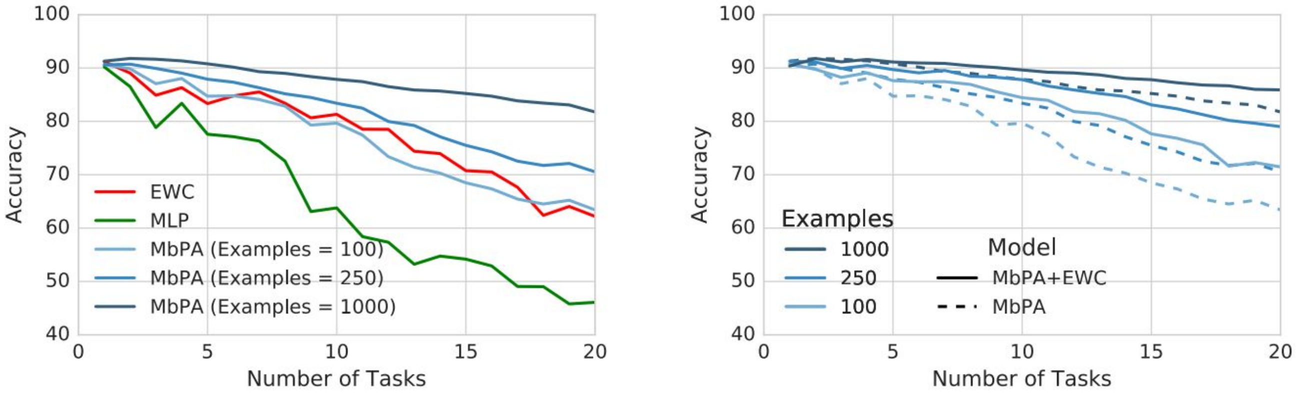


Figure 3: (Left) Results on Permuted MNIST comparing baselines with MbPA using different memory sizes. (Right) Results augmenting MbPA with EWC, showing the flexibility and complementarity of MbPA.

Takeaways: Few gradient steps on carefully selected data from memory are sufficient to recover performance. Flexibility and complementarity of MbPA.

IL: Shift in Task Label Distributions (ImageNet)

Subset	Model	Top 1 (at epochs)			AUC (at epochs)		
		0.1	1	3	0.1	1	3
Novel	MbPA	**46.2 %**	**64.5 %**	**65.7 %**	27.4 %	**57.7 %**	**63.0 %**
	Non-Parametric	40.0 %	53.3 %	52.9 %	**28.3 %**	47.9 %	51.8 %
	Mixture	31.6 %	56.0 %	59.1 %	18.6 %	47.4 %	54.7 %
	Parametric	16.2 %	53.6 %	57.9 %	5.7 %	41.7 %	51.9 %
Pre Trained	MbPA	68.5 %	**70.9 %**	**70.9 %**	71.4 %	70.3 %	**70.3 %**
	Non-Parametric	62.7 %	69.4 %	70.0 %	45.9 %	65.8 %	68.7 %
	Mixture	**71.9 %**	70.3 %	70.2 %	74.8 %	**70.6 %**	70.1 %
	Parametric	71.4 %	68.1 %	68.8 %	**76.0 %**	68.6 %	68.3 %

Table 1: Quantitative evaluation of the learning dynamics for the Imagenet experiment. We compare a parametric model, non-parametric model (prediction based on memory only (9)), a mixture model and MbPA. We report the top 1 accuracy as well as the area under the curve (AUC) at different points in training.

Takeaways: MbPA outperforms both parametric and mixture model in speed and performance. MbPA acquires knowledge from very few examples.

IL: Shift in Task Label Distributions (ImageNet)

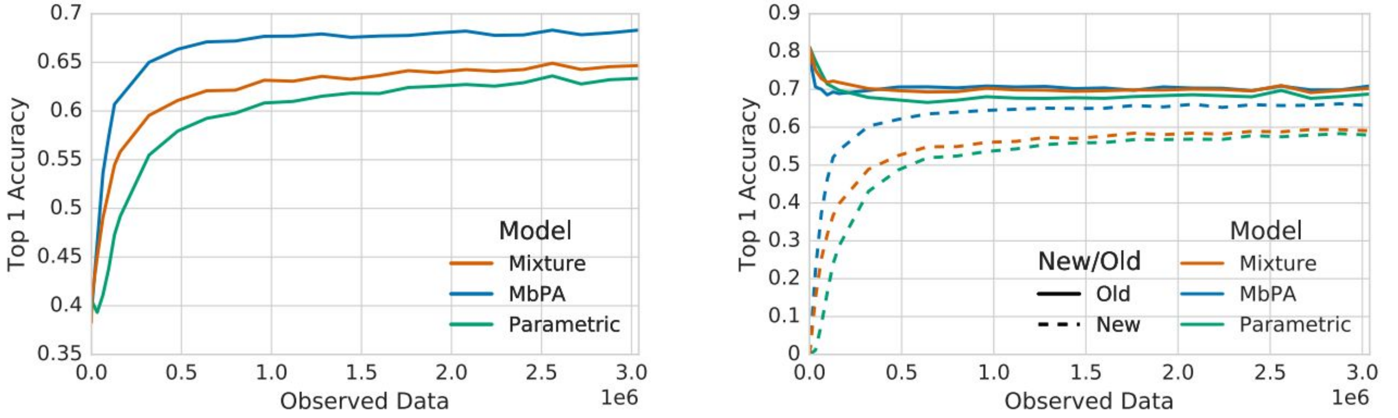


Figure 4: The figure compares the performance of MbPA (blue) against two baselines: the parametric model (green) and the mixture of experts (red). (Left) Aggregated performance (Right) disentangled performance evaluated on new (dashed) and old (solid) classes.

Takeaways: MbPA outperforms both parametric and mixture model in speed and performance. MbPA acquires knowledge from very few examples.

IL: Unbalanced Dataset (ImageNet)

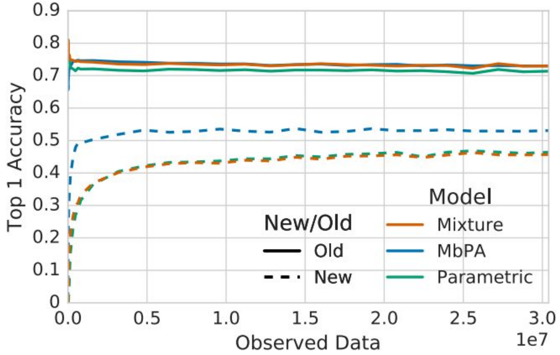


Figure 5: (Left) MbPA outperformed both parametric and memory-based mixture baselines, in the presence of unbalanced data on previously unseen classes (dashed lines). (Right) Example of MbPA. Query (shown larger in the top-right corner) of class "TV" and neighbourhood (all other images) for a specific case. Mixture and parametric models fail to classify the image while MbPA succeeds. 8 different classes in the closest 20 neighbours (e.g. "desktop computer", "monitor", "CRT screen"). Accuracy went from 25% to 75% after local adaptation.

Takeaways: Inductive bias in the local adaptation process is well suited to deal with data scarcity

Domain Shifts: Language Modeling (PTB/ WikiText-2)

	PTB			WikiText-2		
	Valid	Test	ΔTest	Valid	Test	ΔTest
CharCNN (Zhang et al., 2015)		78.9				
Variational LSTM (Aharoni et al., 2017)		61.7				
LSTM + cache (Grave et al., 2016)	74.6	72.1		72.1	68.9	
LSTM (Melis et al., 2017)	60.9	58.3		69.1	65.9	
AWD-LSTM (Merity et al., 2017)	60.0	57.3		68.6	65.8	
AWD-LSTM + cache (Merity et al., 2017)	53.9	52.8	- 4.5	53.8	52.0	- 13.8
AWD-LSTM *(reprod.)* (Krause et al., 2017)	59.8	57.7		68.9	66.1	
AWD-LSTM + dyn eval (Krause et al., 2017)	51.6	51.1	- 6.6	46.4	44.3	- 21.8
LSTM (ours)	61.8	59.6		69.3	65.9	
LSTM + cache (ours)	55.7	55.3	-4.3	53.2	51.3	-14.6
LSTM + MbPA	54.8	54.3	-5.3	58.4	56.0	-9.9
LSTM + MbPA + cache	54.8	54.4	-5.2	51.8	49.4	-16.5

Table 2: Table with PTB and WikiText-2 perplexities. Δ Test denotes improvement of model on the test set relative to the corresponding baseline.

Takeaways: Inductive bias in the local adaptation process is well suited to deal with data scarcity

Domain Shifts: Language Modeling (PTB/ WikiText-2)

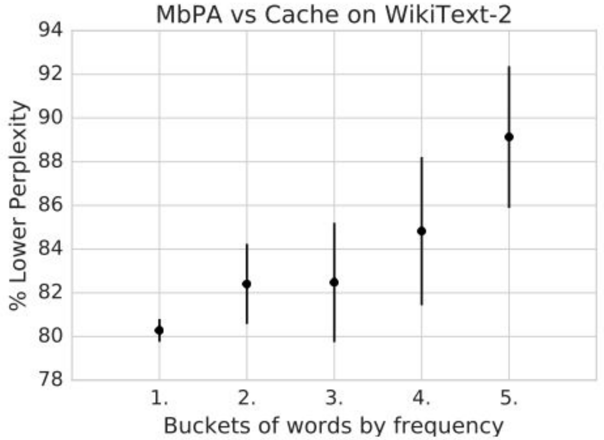


Figure 8: Percent improvement when MbPA is included with the LSTM baseline and neural cache, split by training word frequency into five equally sized buckets. The bucket 1 contains the most frequent words, and bucket 5 contains the least frequent words. The average improvement ± 1 standard deviation are shown. MbPA provides a directional improvement for less frequent words.

Takeaways: MbPA provides a directional improvement for rare words

Conclusion

- MbPA: a scheme for using an episodic memory to locally adapt the parameters
- MbPA: improvements on wide range of settings incremental, lifelong, domain shift
- MbPA: rapid adaptation to unseen classes, deal with imbalanced data, shift in word distributions in language modeling tasks

Issues:

- Keys are drifting
- **Very very very** slow

Discussion

1. Given episodic memory: Replay vs. Local adaptation
    a. Local adaptation is bound to be superior?
2. Selection strategies for experience selection are beneficial for replay
3. Random selection of experiences for local adaptation?
    a. Any strategy as long as diversity(?) is ensured
4. Rethinking: Avoiding catastrophic forgetting <=> Quickly unforgetting?

References

Sprechmann, Pablo, et al., "Memory-based parameter adaptation" ICLR 2018