Online Incremental Feature Learning with Denoising Autoencoders

Guanyu Zhou, Kihyuk Sohn, Honglak Lee

Presenter : Seonghyeon Kim

2018.9.21

Notation

$x \in [0,1]^D$: Input
$\widetilde{x} \in [0,1]^D$: Corrupted input
y : Label

$f_{\mathcal{O}}(\widetilde{x}) = h_{\mathcal{O}} = \sigma(W_{\mathcal{O}}\widetilde{x} + b_{\mathcal{O}})$: Old features
$f_{\mathcal{N}}(\widetilde{x}) = h_{\mathcal{N}} = \sigma(W_{\mathcal{N}}\widetilde{x} + b_{\mathcal{N}})$: New features

$\hat{x} = \sigma(W_{\mathcal{O}}^T h_{\mathcal{O}} + W_{\mathcal{N}}^T h_{\mathcal{N}} + c)$: Reconstructed input
$\hat{y} = softmax(\Gamma_{\mathcal{O}} h_{\mathcal{O}} + \Gamma_{\mathcal{N}} h_{\mathcal{N}} + \nu)$: Predicted label

$\theta_{\mathcal{N}} = \{W_{\mathcal{N}}, b_{\mathcal{N}}, c, \Gamma_{\mathcal{N}}, \nu\}$

Algorithm

Algorithm 1 Incremental feature learning

repeat

Compute the objective $\mathcal{L}(\mathbf{x})$ for input $\mathbf{x}$.

Collect hard examples into a subset B (i.e., $B \leftarrow B \cup \{\mathbf{x}\}$ if $\mathcal{L}(\mathbf{x}) > \mu$).

if $|B| > \tau$ **then**

Select $2\Delta M$ candidate features and merge them into ΔM features (Section 3.3).

Add ΔN new features by greedily optimizing with respect to the subset B (Section 3.2).

Set $B = \emptyset$ and update ΔN and ΔM.

end if

Fine-tune all the features jointly with in current batch of data (i.e., optimize via gradient descent with respect to all parameters).

until convergence

Adding features

- Generative training

$$\min_{W_{\mathcal{N}}, b_{\mathcal{N}}} \frac{1}{|B|} \sum_{i \in B} \mathcal{L}_{gen}(x^{(i)}), \quad \mathcal{L}_{gen}(x) = \mathcal{H}(x, \hat{x})$$

- Discriminative training

$$\min_{W_{\mathcal{N}}, b_{\mathcal{N}}, \Gamma_{\mathcal{N}}} \frac{1}{|B|} \sum_{i \in B} \mathcal{L}_{disc}(x^{(i)}, y^{(i)}), \quad \mathcal{L}_{disc}(x, y) = \mathcal{H}(y, \hat{y}(x))$$

- Hybrid training

$$\min_{W_{\mathcal{N}}, b_{\mathcal{N}}, \Gamma_{\mathcal{N}}} \frac{1}{|B|} \sum_{i \in B} \mathcal{L}_{hybrid}(x^{(i)}, y^{(i)}),$$

$$\mathcal{L}_{hybrid}(x, y) = \mathcal{L}_{disc}(x, y) + \lambda \mathcal{L}_{gen}(x)$$

Merging features

- Find a pair of features whose distance is minimal and replace $f_{\mathcal{O}}$ by $f_{\mathcal{O}\setminus\hat{\mathcal{M}}}$

$$\hat{\mathcal{M}} = \arg\min_{\{m_1,m_2\}} d(W_{m_1}, W_{m_2})$$

- Initialize the new feature parameters as a weighted average of two candidate feature parameters

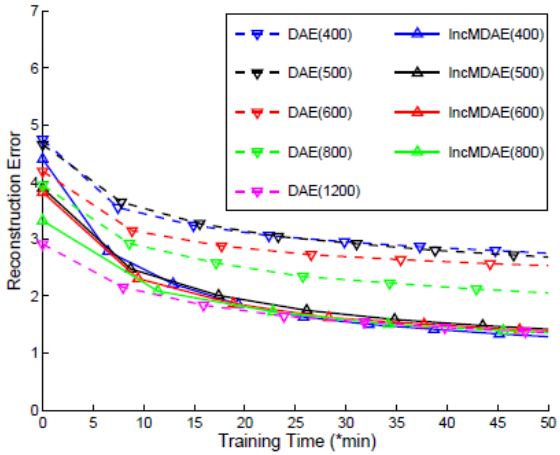
$$\theta_{\mathcal{N}} = \frac{\sum_{x\in B}\{P(h_{m_1}|x;\theta_{m_1})\theta_{m_1} + P(h_{m_2}|x;\theta_{m_2})\theta_{m_2}\}}{\sum_{x\in B}\{P(h_{m_1}|x;\theta_{m_1}) + P(h_{m_2}|x;\theta_{m_2})\}}$$

- Add new features to $\theta_{\mathcal{O}\setminus\hat{\mathcal{M}}}$ by solving

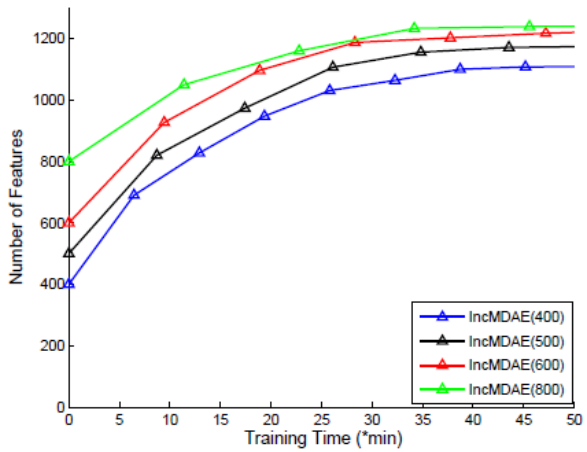
$$\hat{\theta_{\mathcal{N}}} = \arg\min_{\theta_{\mathcal{N}}} \frac{1}{|B|}\sum_{i\in B}\mathcal{L}_{hybrid}(x^{(i)}, y^{(i)})$$

Experiment

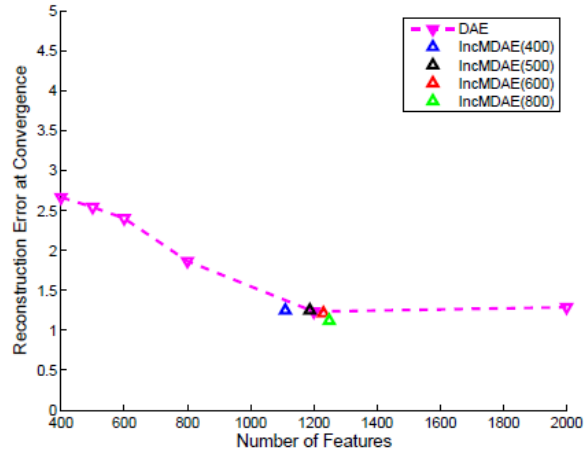
Result - Generative/Discriminative



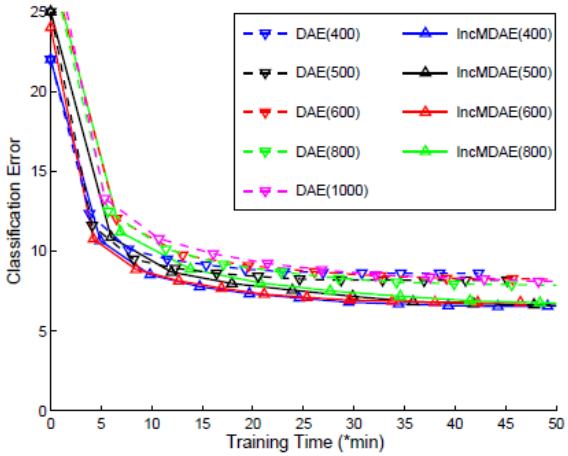
(a)



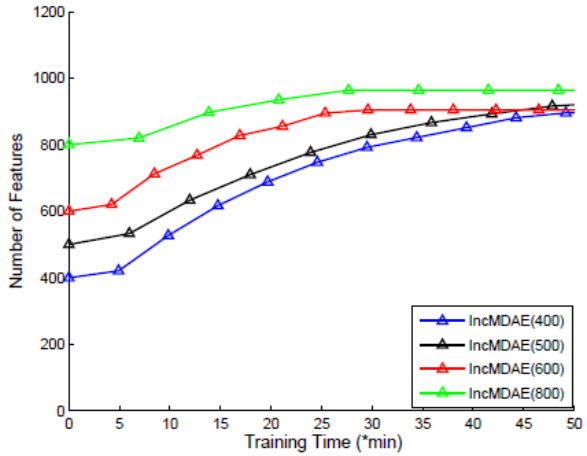
(b)



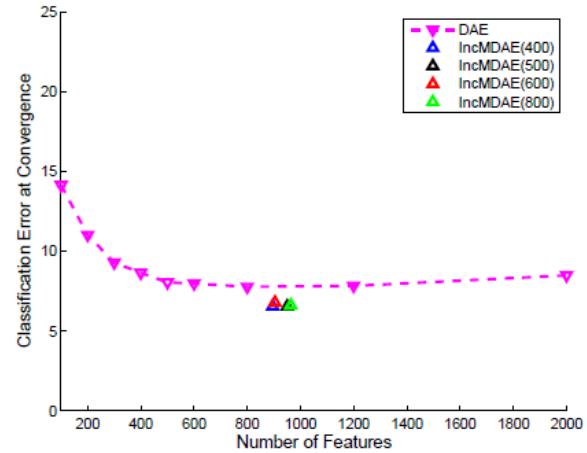
(c)



(d)

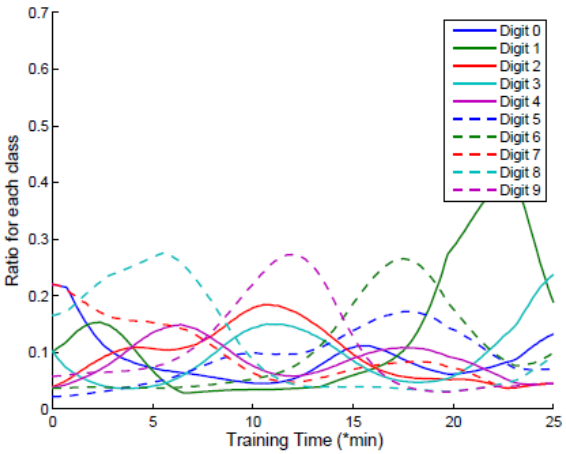


(e)

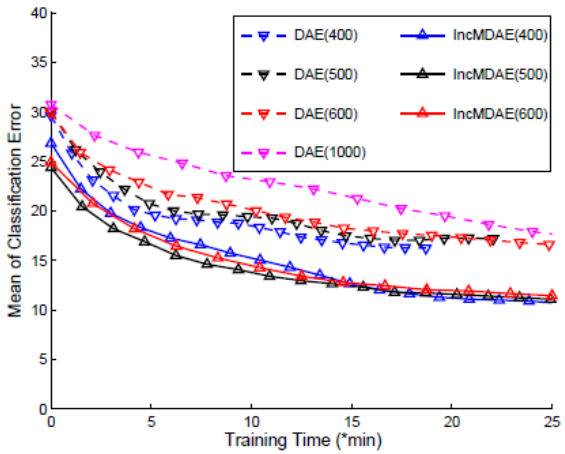


(f)

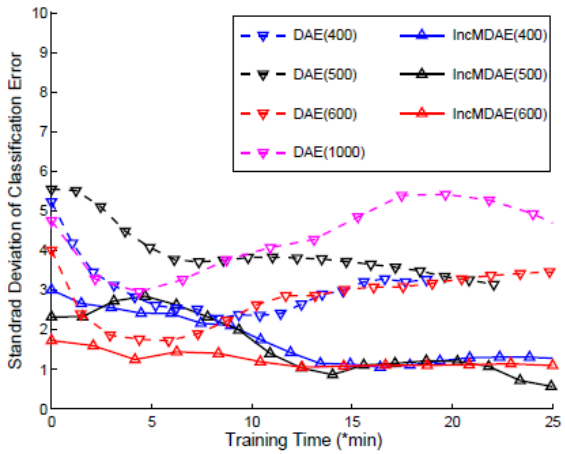
Result - Non-stationary distribution



(a)



(b)



(c)