

Online Continual Learning with Maximally Interfered Retrieval

Seungmin Lee (profile2697@gmail.com; 2013-11420), Dept. of Computer Science and Engineering, Seoul National University

1. Problem and Motivation

This paper tackles online continual learning. Online continual learning assumes the model faces an endless data stream, and the model should be able to learn from the stream without forgetting previous knowledge (without catastrophic forgetting). Previous works related to this paper mitigate the catastrophic forgetting using replay experience, which uses a replay memory or a generative model. However, the earlier works utilize examples randomly sampled from the replay memory or the generative model, which is suboptimal. The proposed method (*Maximally Interfered Retrieval* or *MIR*) extracts examples that are expected to suffer from an increased loss when the model is updated using new data.

2. Methods

2.1. Notation

On this problem setting, the classifier f parameterized by θ faces an endless data stream that provides a set of new samples $\{X_t, Y_t\}$ from a distribution D_t at timestep t . The classifier f has to minimize task loss \mathcal{L} on the new samples without losing previous knowledge.

2.2. Maximally Interfered Sampling from a Replay Memory

In this setting, a finite-size replay memory \mathcal{M} stores the previous examples. The memory is updated as a new set of dataset arrives. The proposed method firstly calculates new parameters $\theta^v = \theta - \alpha \nabla \mathcal{L}(f_\theta(X_t), Y_t)$ without actually updating. After that, the method searches the top-k interfered examples $x \in \mathcal{M}$ using a criterion $s_{MI}(x) = l(f_{\theta^v}(x), y) - l(f_\theta(x), y)$. The selected samples are concatenated with $\{X_t, Y_t\}$. The classifier f_θ is updated using the concatenated data. The proposed method reduces search time by sampling C examples randomly first and selecting the top- \mathcal{B} interfered ones from there.

2.3. Maximally Interfered Sampling from a Generative Model

In this setting, a generative model stores the previous examples in a compressed way. The generative model consists of an encoder q_ϕ and decoder g_γ parameterized by ϕ and γ , respectively. For the classifier f , the proposed method tries to find \mathcal{K} -dim latent features $Z \in \mathbb{R}^{\mathcal{B} \times \mathcal{K}}$ that maximize the difference between the losses of before and after update parameters of classifier. The latent features Z is initialized as

$q_\phi(X_t)$. Then, it is perturbed to find the maximally interfered latent features using the following equation:

$$\begin{aligned} \max_Z \quad & \sum_{z \in Z} [D_{KL}(y_{pre} \parallel \hat{y}) + H(y_{pre})] \quad (1) \\ \text{s.t.} \quad & \|z_i - z_j\|_2^2 > \epsilon \quad \forall z_i, z_j \in Z \text{ with } z_i \neq z_j, \end{aligned}$$

where $y_{pre} = f_\theta(g_\gamma(z))$, $\hat{y} = f_{\theta^v}(g_\gamma(z))$, H is an entropy, and ϵ is a hyperparameter for controlling diversity of latent features. The entropy induces the prior classifier to be confident and prevents the decoder from generating blurry or mixed-categories images. After selecting the latent features, the classifier is updated using $\{X_t \cup X'\}$ where $X' = g_\gamma(Z)$.

The above process assumes a well-trained encoder and decoder. This paper also suggests using MIR for training the encoder and decoder using the following equation:

$$\begin{aligned} \max_{Z_{\text{gen}}} \quad & E_{z \sim q_{\phi^v}} [-\log(p_{\gamma^v}(g_{\gamma^v}(Z_{\text{gen}})|z))] - E_{z \sim q_\phi} [-\log(p_\gamma(g_\gamma(Z_{\text{gen}})|z))] \\ & + D_{KL}(q_{\phi^v}(z|g_{\gamma^v}(Z_{\text{gen}}))||p(z)) - D_{KL}(q_\phi(z|g_\gamma(Z_{\text{gen}}))||p(z)) \quad (2) \end{aligned}$$

$$\text{s.t.} \quad \|z_i - z_j\|_2^2 > \epsilon \quad \forall z_i, z_j \in Z_{\text{gen}} \text{ s.t. } z_i \neq z_j$$

2.4. A Hybrid Approach

The replay memory method needs lots of memory for storing previous examples, and the generative model method is not scalable, according to this paper. This paper proposed a hybrid approach to alleviating the problems. In this approach, an autoencoder is used instead of generative models like VAE. Moreover, the latent features of the autoencoder are stored for memory efficiency.

3. Personal Memo

The motivation of this work is similar to that of prioritized experience replay. I believe motivation is not wrong. However, as far as I understand correctly, I think 2.2 is not well designed. Selecting \mathcal{B} prioritized examples from the C random examples needs at least one forward pass for calculating the criterion s_{MI} . If we can forward whole C examples, why should we use only some of them? For the same reason, I guess the comparisons in the experiments are unfair. The previous methods' sampling budget had to be C , not \mathcal{B} .