Adaptive L2 Regularization in Person Re-Identification

Xingyang Ni, Liang Fang, Heikki Huttunen
Tampere University, Finland

Introduction

Person re-identification involves retrieving corresponding samples from a gallery set based on the appearance of a query sample across multiple cameras. It is a challenging task since images may differ significantly due to variations in factors such as illumination, camera angle and human pose. L2 regularization imposes constraints on the parameters of neural networks and adds penalties to the objective function during optimization. It is a commonly adopted technique which can improve the model's generalization ability. Existing approaches assign constant values to regularizer factors in the training procedure, and such hyperparameters are hand-picked via hyperparameter optimization which is a tedious and time-consuming process. In this work, our major contributions are twofold:

- We introduce an adaptive L2 regularization mechanism, which optimizes each regularization factor adaptively as the training procedure progresses.
- With the proposed framework, we obtain state-of-the-art performance on MSMT17, which is the largest dataset for person re-identification.

Conventional L2 Regularization

A neural network consists of a set of \( N \) distinct parameters, \( P = \{ w_n \mid n = 1, \ldots, N \} \), with \( P \) containing all trainable parameters. Each \( w_n \) is an array which could be a vector, a matrix or a 3rd-order tensor. Conventional L2 regularization imposes an additional penalty term to the objective function, which can be formulated as follows:

\[
L_2(P) = L(P) + \sum_{n=1}^{N} \lambda_n \| w_n \|^2
\]

where \( L(P) \) and \( L_2(P) \) denote the original and updated objective functions, respectively. In addition, \( \| w_n \| \) refers to the square of the L2 norm of \( w_n \) and the constant coefficient \( \lambda \in \mathbb{R} \) defines the regularization strength.

Adaptive L2 Regularization

One may wish to add penalties in a different way. Thus, it is possible to generalize even further, i.e., defining a unique coefficient for each \( \| w_n \| \) :

\[
L_3(P) = L(P) + \sum_{n=1}^{N} \lambda_n \| w_n \| f(\| w_n \|)
\]

where each parameter \( w_n \) is associated with an individual regularization factor \( \lambda_n \in \mathbb{R} \). Obviously, it is infeasible to manually fine-tune these regularization factors \( \lambda_n \) for \( n = 1, \ldots, N \) one by one, since \( N \) is in the order of 100 for models trained with ResNet50. A straightforward extension is obtained by replacing the pre-defined constant \( \lambda_n \) with scalar variables which are trainable through backpropagation. However, such an approach without any constraints on \( \lambda_n \) will fail. Namely, setting negative values for \( \lambda_n \) allows naively increasing \( \| w_n \| \) so that \( L_3(P) \) decreases sharply.

To address the collapse problem, we apply the hard sigmoid function \( f \) which assures that the regularization factor \( \lambda_n \) would always have non-negative values. The regularization factor \( \lambda_n \) is obtained by applying the hard sigmoid on the raw parameters as

\[
\lambda_n = f(\theta_n)
\]

where \( \theta_n \in \mathbb{R} \) for \( n = 1, \ldots, N \) are the trainable scalar variables. Furthermore, we introduce a hyperparameter \( A \in \mathbb{R} \) which represents the amplitude. Hence, we get

\[
\lambda_n = A f(\theta_n)
\]

Combining Equation (3) and (5) gives

\[
L_4(P) = L(P) + \sum_{n=1}^{N} \lambda_n \| w_n \| f(\| w_n \|)
\]