Low-Overhead Resilience for Recommendation System Training Using Erasure Codes

Kaige Liu and Anlun Xu

Carnegie Mellon University

Abstract

Deep-learning based recommendation systems are becoming popular and are widely adopted by the industry. Embedding & MLP paradigm is the most popular model that handles sparsity in training data. Due to the large model size of embedding tables, training of such models is usually split onto multiple machines in cluster settings and thus are prone to slowdowns and failures. Current checkpointing approach prevents retraining from the very beginning at failure yet introduces delay in training as well as rollback at recovery.
Erasure coding is a popular technique for achieving resource-efficient resilience to data unavailability in storage and communication systems. We designed an efficient erasure coded embedding table for basic SGD based on linear parity functions that is adaptable to any (n, k) coding scheme. We implemented the erasure coded embedding table into Alibaba's XDL and replace it's current checkpoint based fault tolerance with the erasure code based approach. Our experimental results demonstrate feasibility of this model, with 24% overhead on training throughput, as well as ability to successfully recover using redundant table entries and resume training with a failed server.

Keywords

Erasure codes, Deep learning, High-dimensional sparse data, XDL

1 Introduction

1.1 MLP-based recommendation systems

Content filtering was the most common technique used in early recommendation systems. A set of experts classified products into categories, while users selected their preferred categories and were matched based on their preferences. Later on, collaborative filtering is introduced in the recommendation system, where recommendations are based on past user behaviors, such as prior ratings given to products. Neighborhood methods that provide recommendations by grouping users and products together and latent factor methods that characterize users and products by certain implicit factors via matrix factorization techniques were later deployed with success.

Deep learning is one of the most exciting breakthroughs of artificial intelligence and is extensively applied to solve real-world problems in many areas such as speech recognition, computer vision, natural language processing, and medical diagnosis. Although existing deep learning frameworks have achieved great successes, they are not friendly designed for applications involving high-dimensional sparse data, which widely exists in recommendation systems. For example, in current recommendation systems, petabytes of log data of user behavior are generated every day. Training samples typically contain billions to trillions of features, while only a few of these dimensions are non-zero for each sample.

In order to deal with this issue of sparse data processing, current recommendation systems such as Alibaba's XDL[1] and Facebook's DLRM[2, 3] use a statistical model that incorporates deep network to process dense features and use embedding tables to process sparse features that represent categorical data.

Current recommendation systems utilize embedding tables for mapping categorical features to dense representations. Embedding tables map each category to a dense representation in an abstract space. In particular, each embedding lookup may be interpreted as using a one-hot vector to obtain the corresponding column vector of the embedding table. In more complex