Sudhir Kumar Suman[16D070027]
Github: SumanSudhir
Homepage: https://sumansudhir.github.io.
Email: sudhiriitb27@gmail.com
Phone Number: +91 9867315147

Neural Collaborative Filtering(NCF) Model

Swift for TensorFlow

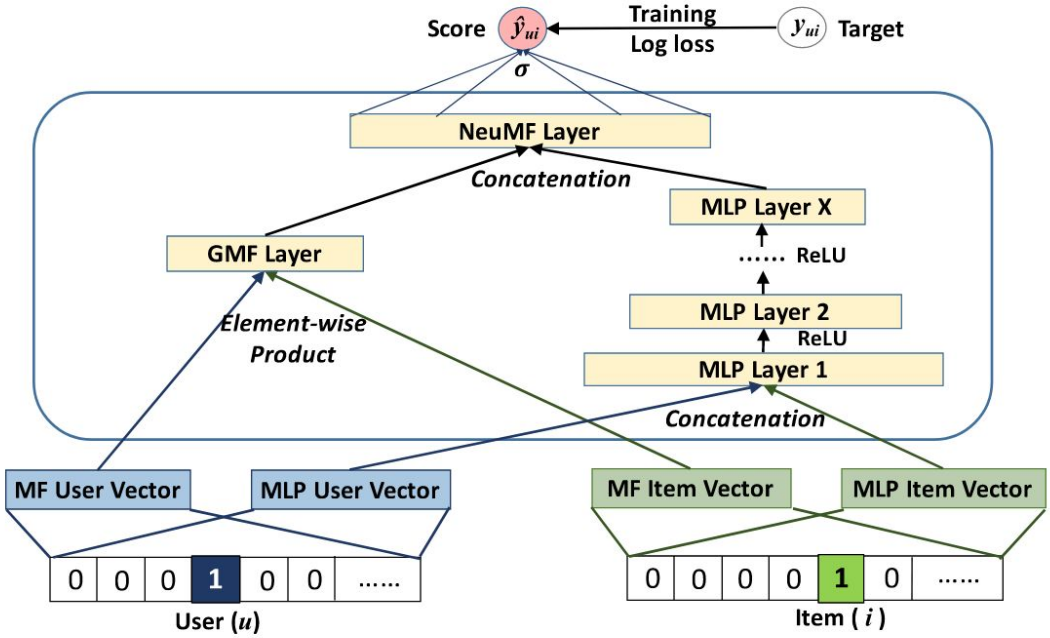
Synopsis and Motivation:

Swift for Tensorflow is a next-generation system for deep learning and differentiable computing which helps users to develop and train Machine and Deep Learning models. It maintains an open-source repository of models that are used for demonstrating and testing the functionality of Swift automatic differentiation and Machine Learning APIs. Currently, there exist few models from Computer Vision and Natural Language Processing but lack models for Recommendation, Image Segmentation, Translation, etc.

This project aims at providing missing **Recommendation Model** by implementing it in Swift for TensorFlow along with code to process relevant datasets for training it and build performance and correctness tests.

Model Description:

Neural Collaborative Filtering(NCF) is a new matrix factorization model, which ensembles Generalized Matrix Factorization (GMF) and Multi-Layer Perceptron (MLP) to unify the strengths of linearity of Matrix Factorization and non-linearity of Multi-Layer Perceptron for modeling the user-item latent structures.



This figure shows how to utilize latent vectors of items and users and then how to fuse outputs from GMF Layer and MLP Layer.

The GMF Model:

GMF introduces the neural collaborative filtering layer of standard matrix factorization. In this way, matrix factorization can easily be generalized and extended. For example, if we allow the edge weights of this output layer to be learned from data without uniform constraint, it will result in a variant of MF that allows varying importance of latent dimensions and if we use a non-linear function for activation, it will generalize MF to a non-linear setting which might be more expressive than the linear MF model.

The MLP Model:

Neural Collaborative Filtering adopts two pathways to model user and items:

1. Element-wise product of vectors
2. Concatenation of vectors

To learn interactions after concatenating of users and items features, the standard MLP model is applied. In this sense, we can endow the model a large level of flexibility and non-linearity to learn interaction between user and items

Open-source contribution in TensorFlow

The main contributions of this work in Swift for TensorFlow are as follows

- Written code for Neural Collaborative Filtering architecture in new deep learning framework Swift for TensorFlow to model latent features of users and items
- Contributed new dataset MovieLens to directly load it while working in Swift for TensorFlow
- Performed experiment on MovieLens-100K dataset for the correctness test by predicting the top K items user will interact within coming days

- **Project Pull Request**
    1. Recommendation model Neural Collaborative Filtering
- **Other Pull Request Merged**
    1. Added precondition for matmul
    2. Added precondition for moments, logSumExp, standardDeviation
    3. Added precondition for replacing, all, max, min, argmax, argmin, sum, product, mean, variance, cumulativeSum, cumulativeProduct
    4. Bugfix[query replaced with value]
    5. Replaced deprecated substring
- **Pull Request under Review**
    1. Conform Array to Module and Layer
- **Opened Issue**
    1. Session Crashes in Colab & Jupyter Lab

References:

1. https://arxiv.org/pdf/1708.05031.pdf
2. Official NCF implementation [Keras with Theano]