

Deep Autoencoders for eco recommendations

1) Introduce

In order to create an eco-friendly activity recommendation system using the data of users who will be launching the application, we developed a recommendation algorithm using Movielens 100k data.

2) Observations

Rank	Model	RMSE ↓ (u1 Splits)
1	Bayesian timeSVD++ flipped + Feat w/ Ordered Probit Regression	0.884
2	Bayesian timeSVD++ flipped + Feat	0.886
3	Bayesian timeSVD++ flipped	0.886
4	MG-GAT	0.890
5	GraphRec + Feat	0.897
6	GraphRec	0.904
7	IGMC	0.905
8	GC-MC + feat	0.905
9	GC-MC	0.910
10	Self-Supervised Exchangeable Model	0.91
11	GRAEM	0.9174
12	Factorized EAE	0.920
13	sRGCNN	0.929
14	GRALS	0.945
15	GMC	0.996

The above data shows the model, rank and RMSE loss values of the movieens 100k from the paperswithcode website. But Most of the above models for recommendations only consider the user and the ratings given by the user. Such recommendations do not take into account other attributes.

3) Data structure

First of all, the dataset of movielens is composed as follows.

<i>Users</i>					<i>Ratings</i>			
UserID	Gender	Age	Job	Zip	UserID	MovieID	Rating	Timestemp
1	F	1	10	48067	1	1193	1	978300760
2	M	56	16	70072	1	661	56	978302109
3	M	25	15	55117	1	914	25	978301968
4	M	45	7	02460	1	3408	4	978300275
5	M	25	20	55455	1	2355	5	978824291

<i>Movies</i>		
MovieID	Title	Genres
1	Toy Story	Animation Children's Comedy
2	Jumanji	Adventure Children's Fantasy

It contains various attributes like gender and age of the user. The movies also contain the genre information.

4) Data preprocessing

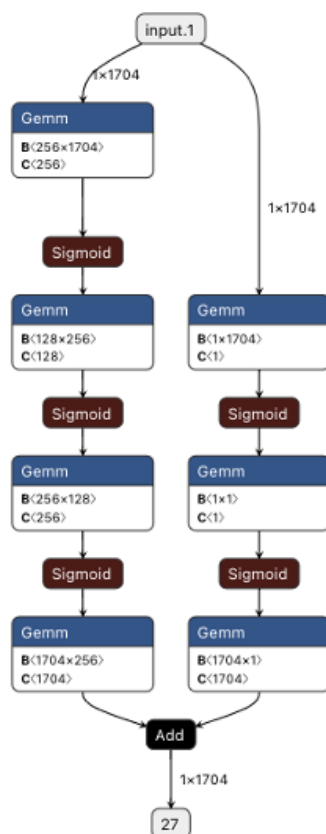
For AutoEncoder input, the data should represent a two-dimensional array where each row represents a user. The first n columns refer to n movies where the user has rated the movie. In case the user has not rated the movie, it will contain a value of 0. The next n columns will represent the number of genres. The last few rows indicate the user attributes like user gender, age and job.

Train_df

User	Movie 1	Movie 2	Movie 3	---	Movie n	Genre 1	Genre 2	---	Genre n	Male	Female	Age	Job
User 1	3	2	5	---	4	10	5	---	0	1	0	21	2
User 2	2	1	4	---	4	8	3	---	2	1	0	1	4
User 3	4	3	4	---	3	4	1	---	5	0	1	31	17
---	---	---	---	---	---	---	---	---	---	---	---	---	---
User n	3	2	3	---	5	14	23	---	1	0	1	51	6

5) AutoEncoder Model

Model structure



Our model basically shows the AutoEncoder structure. We have tried deep neural networks with 3, 5 and 7 layers. We have chosen 3 layers because computation exponentially increases with the increase in layers and hence neurons. In our case the results provided by 5 and 7 layered neural networks perform only marginally better than 3 layered network. We also propose a wide and deep model by adding a simple linear wide model.

The input of the model is a tensor of the form (1, 1704) by connecting 1682 movies, 18 genres, and 4 user information. The deep model goes through [1704, 256, 128, 256, 1704] layers, and the wide model goes through [1704, 1, 1704] layers. All layers use the sigmoid activation function, and the out of the model is the sum of the deep model and the wide model.

6) Train