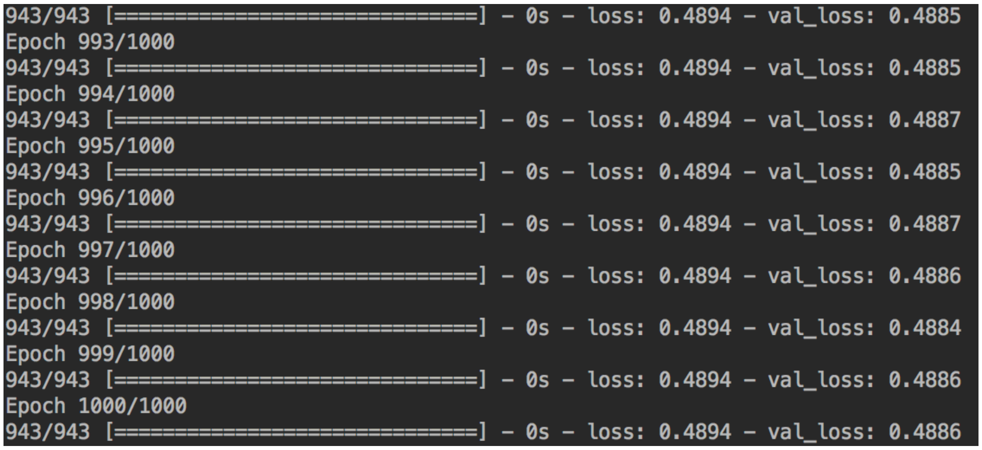
Documents

In this code repository, we have established a Collaborative Denoising Auto Encoder Neural Network. This neural net is designed and implemented to give user preference prediction of Ml-100k/ Ml-1M/ Ml-10M/ Yelp Dataset.

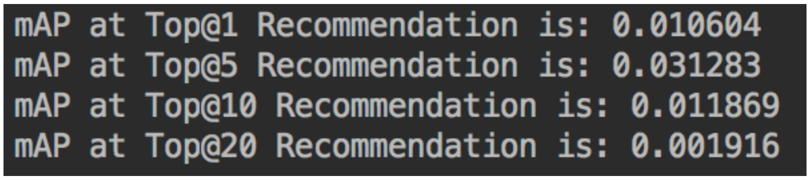
We use keras library to establish a Collaborative Denoising Auto Encoder Neural Network with one hidden layer as in net.py. The data is fed into the network using load_data() function in data.py. We code our metric evaluation function in metric.py as apk(). We train our model in train.py and print the final mean average precision of case Top@1, Top@5 and Top@10.

During Training, as in Figure 1, our model gives the output of Mean Average Precision, as has been adopted as the evaluation metrics of our recommender systems. The results are recorded as in Figure 2. (the screenshot of the results) and draft as bar figure later to visually compare model performance between each of M1 M2 M3 M4 and CDAE/DAE.



```
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4885
Epoch 993/1000
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4885
Epoch 994/1000
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4887
Epoch 995/1000
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4885
Epoch 996/1000
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4887
Epoch 997/1000
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4886
Epoch 998/1000
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4884
Epoch 999/1000
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4886
Epoch 1000/1000
943/943 [==============================] - 0s - loss: 0.4894 - val_loss: 0.4886
```

Figure 1. Training Detail



```
mAP at Top@1 Recommendation is: 0.010604
mAP at Top@5 Recommendation is: 0.031283
mAP at Top@10 Recommendation is: 0.011869
mAP at Top@20 Recommendation is: 0.001916
```

Figure 2. Mean average precision output

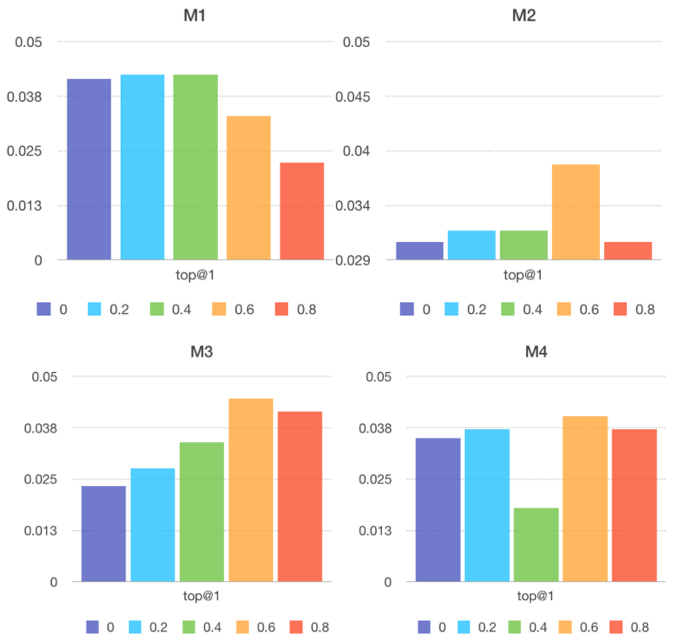


Fig.3 Model Performance Comparison at Top@1.

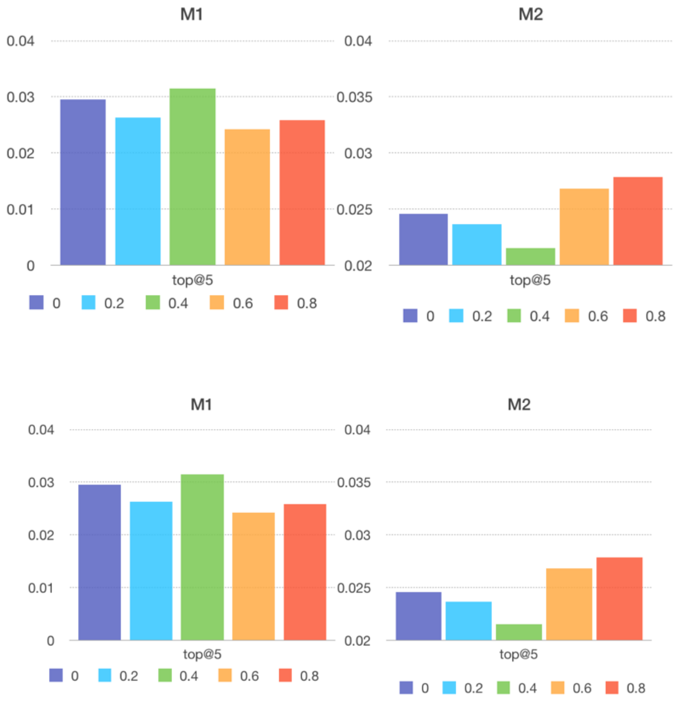


Fig.4 Model Performance Comparison at Top@5.

Adopting the result in popular research, the results given by CDAE is clearly better than classical DAE model. Furthermore, CDAE with tied weight model generally performs better than models with no tied weights.

As it is shown in the experiment results, for Top@1 recommendation, model 1, which is the combination of ReLU at hidden layer and sigmoid at output layer, and the mean squared error loss function has clearly outperformed others. Moreover, among all the corruption levels, q= {0.2, 0.4} has the highest mAP. However, for the Top@5 recommendations, the first model and the third model both performed better than the other two models, and for Top@10 recommendations, the last model has the best performance. Above all the models, the second model (ReLU, sigmoid, binary cross entropy) generally performed poorly than others.

In conclusion, for Top@1 and Top@5 recommendations, the combination of ReLU at hidden layer and sigmoid at output layer, and the mean squared error loss function has the most satisfying results; as for the Top@10 recommendation, the combination of sigmoid at hidden layer and sigmoid at output layer, and the binary cross entropy loss function is also practical. So, the choice mostly depends on the actual practice.

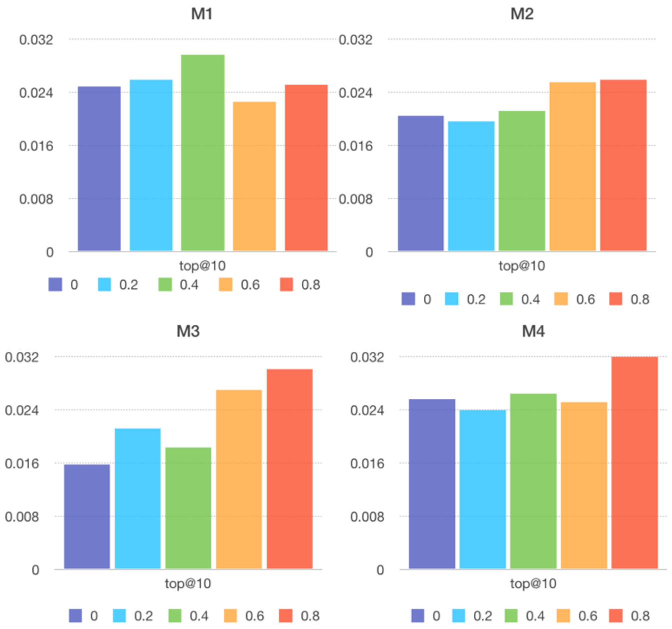


Fig.5 Model Performance Comparison at Top@10.