**Infusing Inferred Personality to Improve Performance in Recommendation Systems**Project Report  
ISE 244 AI Tools and System Engineering

Uttej Kumar Reddy Gade  
016065543  
[uttejkumarreddy.gade@sjsu.edu](mailto:uttejkumarreddy.gade@sjsu.edu)

Code: <https://github.com/uttejkumarreddy/ISE-244-Final-Project>

1. **PROBLEM DEFINITION**

The Oxford dictionary defines personality as the combination of characteristics or qualities that form an individual’s distinctive characters. From this, it can be inferred instinctively that similar personalities tend to have similar likes. There are studies which have proven this to be true [2]. Hence, this can be assumed to have a major impact in collaborative filtering, which is a type of recommendation system which predicts a user’s preferences based on preferences of other similar users. There is a possibility that incorporating personality might increase the performance of the system. It might also be a solution to the cold start problem that recommendation systems often face as there are multiple ways in which personality information can be derived apart from explicitly recorded user activity in regards to the system, such as browsing history or online reviews. This is the basis of the paper, [Improving Recommendation Systems with User Personality Inferred from Product Reviews](https://arxiv.org/pdf/2303.05039.pdf) [1] which is what my final project is built upon.

The major challenges in experimenting with this idea are the availability of datasets with personality incorporated in them. Such datasets are scarce since there are a lot of privacy and sensitivity concerns connected to them. Hence the authors constructed their own datasets from the Amazon dataset [3] and also used the existing Personality 2018 dataset [4]. The authors inferred personalities from the review text [5] in the Amazon dataset in the OCEAN model format [6] which is the personality model in which the scores are stored in Personality 2018 dataset. They applied Neural Collaborative Filtering [7] model which is the first deep learning model used in recommendation systems and got an increase of 3-28% on the recommendation system performance.

1. **PROJECT OBJECTIVES**

As mentioned in the former paragraph, the author applied NCF on personality datasets for their experiments. However, the code they used for this is not made public. So the objective of this project is multi-fold.

1. Implement the algorithms as outlined in the paper on a single dataset.
2. Modify the dataset and incorporate a different type of personality model, MBTI and run the algorithms on the new dataset to check for performance improvement.
3. **ANALYSIS AND EXPERIMENTS**

***Dataset***

The authors of the paper constructed two datasets, Amazon music and Amazon beauty which are subdomains of the Amazon dataset. However, these two datasets are not made public. The method used to construct these datasets is extracting the review texts from the 2 domains and send them to software which would return the personality scores. This software however is paid and so the reconstruction of that dataset for this project was put on hold.

Next, I looked at the [Personality 2018](https://grouplens.org/datasets/personality-2018/) dataset which was publicly available. It is a dataset based on MovieLens and hence has user\_id, movie\_id, rating and the OCEAN scores for each user. The dataset was clean and with some little pre-processing could be applied to the algorithms. Hence, this was chosen to proceed with.

The dataset consists of 2 CSV files. The ratings.csv file has user\_id, movie\_id, rating. The personality-data.csv has user\_id, openness, conscientiousness, extraversion, agreeableness, neuroticism.

The major preprocessing involved was mapping the user\_id and movie\_id to natural numbers to make them suitable to be passed into Tensorflow neural networks. Next both the csv files were split into train and test files so that all the models we construct can be trained and evaluated on the same dataset hence giving us a fair evaluation. These steps can he seen [here](https://github.com/uttejkumarreddy/ISE-244-Final-Project/blob/master/01-preprocess-data.ipynb).

***Neural Collaborative Filtering***

Next, I implemented the NCF model on the Personality dataset without actually including the personality data. This is to establish a baseline to compare models with personalities infused and see if they perform better. The authors specify that they use a 4-layer MLP and a 16-dimensional user and item embedding in the NCF. There are no other details as to the optimizer or loss. Hence, for all the models, I default to ADAM optimizer and mean squared error loss. The user and movie ids are embedded using the 16 dimensions embedding layer. These embeddings are then passed through 4 densely connected layers with relu activation. The hidden layers are 64, 32, 16 and 1 in that order. The model is compiled and the training data is passed. It is run for 10 epochs with a batch size of 64. The resultant trained model is stored to be evaluated together with the rest of the models later. This can be seen [here](https://github.com/uttejkumarreddy/ISE-244-Final-Project/blob/master/02-ncf.ipynb).

***NCF + Most Salient Personality***

The authors of the paper try 3 new models over the existing NCF model in the paper. The first is the incorporation of the most salient personality or the personality that has the highest score among the 5 scores. It is included as a learnable 4-dimensional personality vector for each of the 5 types. For this change, we create a personality input and a personality embedding layer that takes in the 5 scores but considers only the highest one and embeds that in the neural network. The rest of the steps are the same as in *Neural Collaborative Filtering.* The implementation can be seen [here.](https://github.com/uttejkumarreddy/ISE-244-Final-Project/blob/master/03-ncf-most-salient-personality.ipynb)

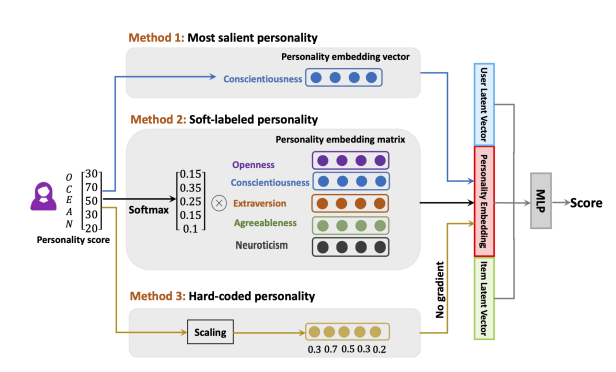
***NCF + Soft Labeled Personality***

In this method, instead of selecting the highest personality score among the 5 as in *NCF + Most Salient Personality,* a softmax is applied on the 5 personality score vector. These values are then flattened and the layer is concatenated to user and movie embeddings as a trainable layer. This makes the model predict not just on 1 personality but all 5. The implementation can be seen [here](https://github.com/uttejkumarreddy/ISE-244-Final-Project/blob/master/04-ncf-soft-labeled-personality.ipynb).

***NCF + Hard Coded Personality***

In the final variation, the authors’ hard code the personality scores and freeze the personality layer so that it is not trainable. The hard coded personality scores are normalized such that their sum equals 100 and embedded. This implementation can be seen [here.](https://github.com/uttejkumarreddy/ISE-244-Final-Project/blob/master/05-ncf-hard-coded-personality.ipynb)

All the three methods as outlined above can be seen succinctly in this diagram taken from the paper.



**Fig 1. An overview of the three methods. In the first method, the highest OCEAN score from the user’s score {30, 70, 50, 30, 20} is selected which here is 70 as the personality embedding vector. In the second method, a softmax is applied and a probability distribution of the 5 scores is taken in a learnable layer. In the third method, the personality vector is scaled and fixed in a non-trainable layer.**

***Inferring MBTI***

As my contribution to this project, I am implementing one of the future works mentioned towards the end of the paper. It was an experiment with MBTI personality scores as these scores give an 16 faceted personality score which might be more detailed then OCEAN scores and thus might result in an improved performance.

However, I was presented with the same challenge that to infer the MBTI scores, the software to do so is not free. Hence, I once again went ahead with Personality 2018 dataset. The OCEAN scores range from 1 to 7. Based on these values, I converted OCEAN scores to MBTI scores using the following algorithm.

1. The 16 MBTI types are 'INTJ', 'INTP', 'ENTJ', 'ENTP', 'INFJ', 'INFP', 'ENFJ', 'ENFP', 'ISTJ', 'ISFJ', 'ESTJ', 'ESFJ', 'ISTP', 'ISFP', 'ESTP', 'ESFP' where
   1. E – Extraversion
   2. I – Introversion
   3. S – Sensing
   4. N – Intuition
   5. T – Thinking
   6. F – Feeling
   7. J – Judging
   8. P – Perceiving
2. The values in OCEAN stand for
   1. O – Openness
   2. C – Conscientousness
   3. E – Extraversion
   4. A – Agreeableness
   5. N - Neuroticsm
3. There are similarities between the two and using these similarities we convert one to another. As the OCEAN scores range from 1 to 7, we can consider 3.5 as the baseline for a personality to tend from one type to another. For example, if the Extraversion is greater than 3.5 in OCEAN score, in MBTI it can be inferred that the person is Extroverted. If not, the person is Introverted. Similarly, ‘openness’ can be used to infer Sensing/Intuition as the more open or exposed a person is, the more he can ‘sense’ based on experience instead of simply ‘guess’ (intuition). Similarly, agreeable can be used to infer Thinking/Feeling and conscientiousness can be used to infer Judging/Perceiving.
   1. **NOTE:** This is not accurate. This is just an approximation to make MBTI scores available for experiments.
4. Each user MBTI score is thus inferred, and the result is one-hot encoded which results in 16 columns (2^4) and stored wherever relevant.

See the implementation [here](https://github.com/uttejkumarreddy/ISE-244-Final-Project/blob/master/06-infer-mbti.ipynb).

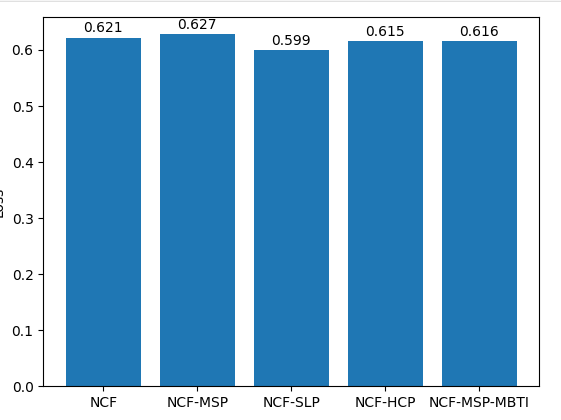
***NCF + Most Salient Personality with MBTI Scores***

In this experiment I use MBTI scores instead of personality score in the NCF + Most Salient Personality model. The personality embedding layer is removed and replaced with a MBTI score which takes in a 16 vector input (one-hot encoded MBTI personality scores). The rest is the same and when passing the data we pass the MBTI scores instead of OCEAN scores. The implementation can be seen [here](https://github.com/uttejkumarreddy/ISE-244-Final-Project/blob/master/07-ncf-most-salient-personality-mbti.ipynb).

NCF + Soft Labeled Personality and NVF + Hard Coded Personality do not make sense here as this is a one-hot encoded vector and as such applying softmax would yield no result and hard coding the vector is the same as Most Salient Personality. Hence, these two experiments were not pursued.

1. **RESULTS**

All the models from the above experiments are saved at the end of the experiment. They are all trained for 10 epochs with a batch\_size of 64 on the same dataset. They are all also evaluated on the same test dataset. Multiple metrics were considered to evaluate the models. Hit Rate and Normalized Discounted Cumulative Gain are popular metrics when it comes to evaluate the effectiveness of a recommendation system. However, since we ran the models on Personality 2018 MovieLens rating dataset, using loss is a good measure of the effectiveness. And hence the loss for each of the models is recorded and plotted as such. See the implementation [here](https://github.com/uttejkumarreddy/ISE-244-Final-Project/blob/master/08-evaluate-models.ipynb).



**Fig 2. Loss of the models experiments were conducted on**

1. **DISCUSSION**

Based on the above loss model, we can see that the loss for NCF + SLP < NCF + HCP < NCF – MSP – MBTI < NCF < NCF + MSP

We can infer the following from the above losses.

1. Recommender systems perform the best when all the personality factors are considered (NCF – SLP has the least loss).
2. NCF + HCP outperforms NCF + MSP indicating that again utilizing all personality scores is best for recommendation systems.
3. NCF – MSP – MBTI performs better than MCF – MSP – OCEAN indicating that while the considering the most salient personality results in a higher loss, having more information of personality (16 scores in MBTI as opposed to 5 in OCEAN) is more beneficial to the recommendation system.
4. Plan NCF performs better than NCF + MSP indicating that just having a single additional personality attribute is not positively correlated with better performance.

The first three results agree with the findings in the paper. This is as per the statement in the paper, “…we further find that NCF + Soft-Labelled/Hard-coded outperforms NCF + Most salient personality in terms of NDCG. This shows that utilizing all five personality traits is better than using the most salient personality trait in NCF”.

The last observation does not exactly match with the paper though the test in the paper is a bit different than the one done in the project. As stated in the paper, “NCF with the most salient personality label outperforms NCF with the same or random personality label.” In the project though, we compare NCF with MSP with a simple NCP without added personality label. This might be the cause of the discrepancy.

1. **EVALUATION AND REFLECTION**

In this project, I attempt to implement the algorithms in the paper that the project is based on. Due to limitations in accessing software to evaluate personality results, I choose an existing personality dataset based on MovieLens dataset, Personality 2018 which contains OCEAN model personality scores. I construct 4 different models, NCF, NCF + Most Salient Personality, NCF + Soft Labelled Personality and NCF + Hard Coded Personality as per the details in the paper. The only specifications on the model are that it uses 4 layer MLP and 16 embedding vectors for movie and rating. I incorporate these and assume the default values for the rest of the hyperparameters such as 10 epochs, batch size 64, ADAM optimizer and MSE loss. I train all the models on the same training dataset and store them.

I next infer MBTI scores from the OCEAN model scores based on an approximation model. The MBTI score might differ from the actual score but due to software access limitations, I resort to the approximation to conduct the planned experiments. Next, I train the model on the dataset with MBTI scores and store the result.

Finally, I compare the losses and find that the results agree with most of the points in the paper. I also tentatively confirm an assumption in the ‘Future Work’, where the authors assume that including more personality factors such as MBTI scores which have 16 scores as opposed to OCEAN which has only 5 might increase performance in model through the comparison of NCF + MSP – MBTI and NCF + MSP – OCEAN.

This confirms that including personality information can improve the recommendation system performance. Action on this information must be taken judiciously though as collecting personality information without proper guidelines and safeguards would leave agencies to unauthorized collection, improper use thus leaving users to privacy risks. As with any AI/ML model, they must be used responsibly.

**References**

[1] X. Lu, and M.-Y. Kan, “Improving Recommendation Systems with User Personality Inferred from Product Reviews,” arXiv:2303.05039.

[2] H.-C. Yang and Z.-R. Huang, "Mining personality traits from social messages for game recommender systems," Knowl. Based Syst., vol. 165, pp. 157-168, 2019.

[3] J. Ni, J. Li, and J. J. McAuley, "Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects," in Proc. 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int. Joint Conf. on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, Nov. 3-7, 2019, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Association for Computational Linguistics, pp. 188-197.

[4] T. T. Nguyen, F. M. Harper, L. Terveen, and J. A. Konstan, "User Personality and User Satisfaction with Recommender Systems," Inf. Syst. Frontiers, vol. 20, no. 6, pp. 1173-1189, 2018.

[5] J. W. Pennebaker and L. A. King, "Linguistic styles: Language use as an individual difference," J. Pers. Soc. Psychol., vol. 77, no. 6, pp. 1296, 1999.

[6] R. R. McCrae and O. P. John, "An introduction to the five-factor model and its applications," J. Pers., vol. 60, no. 2, pp. 175-215, 1992.

[7] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural Collaborative Filtering," in Proc. 26th Int. Conf. on World Wide Web, WWW 2017, Perth, Australia, Apr. 3-7, 2017, R. Barrett, R. Cummings, E. Agichtein, and E. Gabrilovich, Eds. ACM, pp. 173-182.

,