**TEAM PROFESSIONALS**

Ritesh B

Jose Moti

Ramprasad Rakhonde

**HACKATHON – CROSS SELL AND UPSELL- REPORT**

All the codes are shared along.

Codes can also be accessed from following GitHub repo 🡪 [Github link](https://github.com/josemoti1999/abinbev_recommender)

Video of how to create an environment and run 🡪 [Video link](https://drive.google.com/file/d/1doL91-nGDQsIH_fxmUn9qmXUBQLKIg7q/view?usp=sharing)

To see modelling and pre-processing more interactively 🡪 [Link](https://colab.research.google.com/drive/1B2785sIMGp-cXrsNYivhykYdau2tfR-6?usp=sharing)

Make sure to have the following files after extracting

* app.py
* cross\_sell.py
* cs\_train.py
* other\_details.py
* readme.md – contains details on how to set up environment and run
* requirements.txt
* Data.xlsx

Make sure to have the following folders too

* templates/ - contains html templates
* static/ - contains pictures and static CSS files required
* preprocessed\_data/ - checkpoints for training

**MODELLING**

Ratings were calculated for each user-item interaction based on the amount of purchase and also the time gap between the purchases. Apart, from common modelling strategies we made our recommender model predict these exact ratings rather than predicting they were a good/bad rating.

Proper validation strategy was adopted.

* Out of 178 users, 20% of the users were made validation users and rest train users.
* Out of each validation user, 80% of their item interactions was also involved in train dataset.
* This is to make sure we have a good part of the history of the validation user before recommending the products for him.

As mentioned in the previous round we used 2 recommendation models.

* Based on Variational Autoencoders. Implemented with TensorFlow library.
* Based on Funk SVD Matrix Factorization (NMF). Implemented with surprise library.

2 evaluation metrics were used

* Ranked Normalized Cumulative gain (RNDCG) – 0.5 for random ranking and 1.0 for correct ranking
* Mean Square Error (MSE) – error between true and predicted rating
* Personalization Index (PI)– 1 for personalized recommendations and 0 means same recommendation for all

Recall @K was not used because our models were made to predict the exact rating and the training dataset involved interactions with good and bad rating as well and not just the good rating.

Before modelling, it was thought that VAE model would have high RNDCG and MSE compared to NMF model but will have low PI. But we observed the following.

For VAE based model:

* RNDCG increased from 0.5🡪0.90
* MSE decreased from 0.25🡪0.057
* PI increased from 0🡪0.155

For NMF based model:

* RNDCG increased from 0.5🡪0.88
* MSE decreased from 0.25🡪0.055
* PI increased from 0🡪0.724

So NMF performs equally or better in all the metrics. So, we took **0.9** time the predictions scores of NMF model and **0.1** times the score of VAE model.

Other predictions corresponding to popularity, similar to the products purchased and history-based recommendations were also made. Also, a feedback system regarding why the product was recommended was also given.