Hybrid Travel Recommender Systems

Nilufa Yeasmin

Education:

- MSc. Data Science and Analytics - Ryerson University
- B.Sc. Computing Science and Eng.

Table of Contents

- **Introduction and Motivation**
- **Dataset**
- **Data processing and EDA**
- **Models**
  - **Collaborative Filtering (Baseline)**
  - **Wide & Deep**
  - **Deep Factorization Machine**
- **Evaluation**
- **Technology Stack**
- **Resources**

Introduction and Motivation

- ✓One of the first things to do while planning a trip is to book a good place to stay.

- ✓Booking a hotel online can be an overwhelming task with thousands of hotels to choose from, for every destination. Therefore, the recommendation system comes into play.

- ✓ For the goals of the project, we have used 4 different recommendation models – 2 Collaborative filtering techniques and 2 Hybrid technique.

- ✓The motivation behind this was to explore different models in the field of recommender systems and come up with an efficient solution.

The Expedia dataset

Data Source

- We have used the Expedia Hotel Recommendation dataset from Kaggle competition.

- The dataset, which had been collected in the 2013-2014 time-frame, consists of a variety of features that could provide us great insights into the process user go through for choosing hotels.

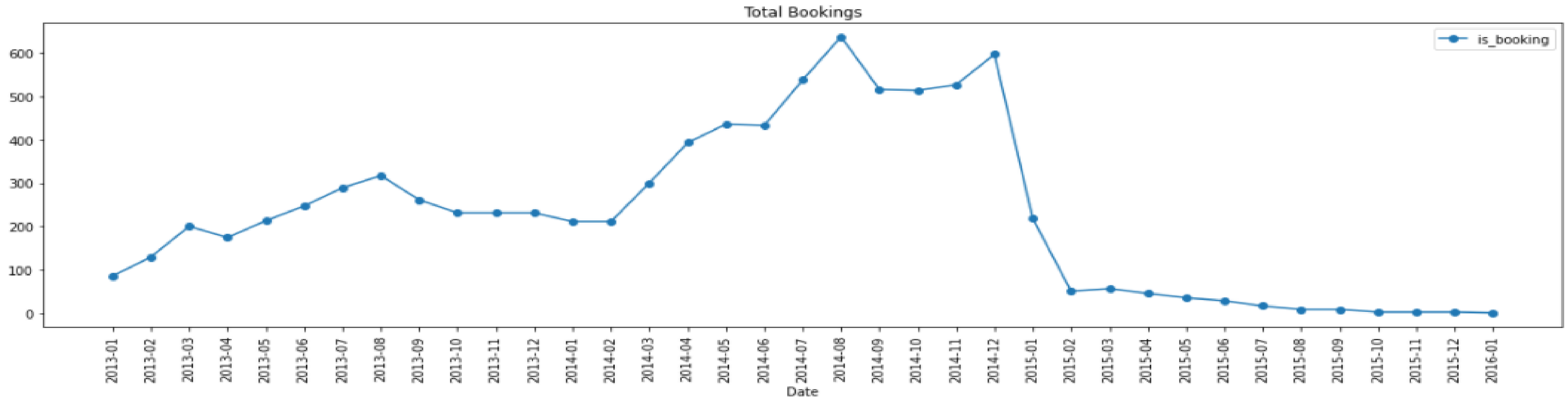
The Expedia dataset

- **train.csv**
  - The training set consists of 37,670,293 entries and the test set contains 2,528,243 entries.

- **destination.csv**
  - Consists of 150 latent features for each of the destinations recorded and the top 10 most correlated features were chosen to be used.

EDA & Data pre-processing

Our first step exploratory analysis was performed to get extract interesting insights into the process of choosing a hotel.



The above figure represents Total Number of Bookings over Date between the years 2013 to 2014.

The next step was to clean and pre-process the data.

- For the users who ranked an hotel twice, only the maximum rating was kept.
- PCA was applied 149 latent features for each destination to extract the most relevant dimensions

Feature Engineering

- We can not use date time, check-in date and check-out date columns directly.

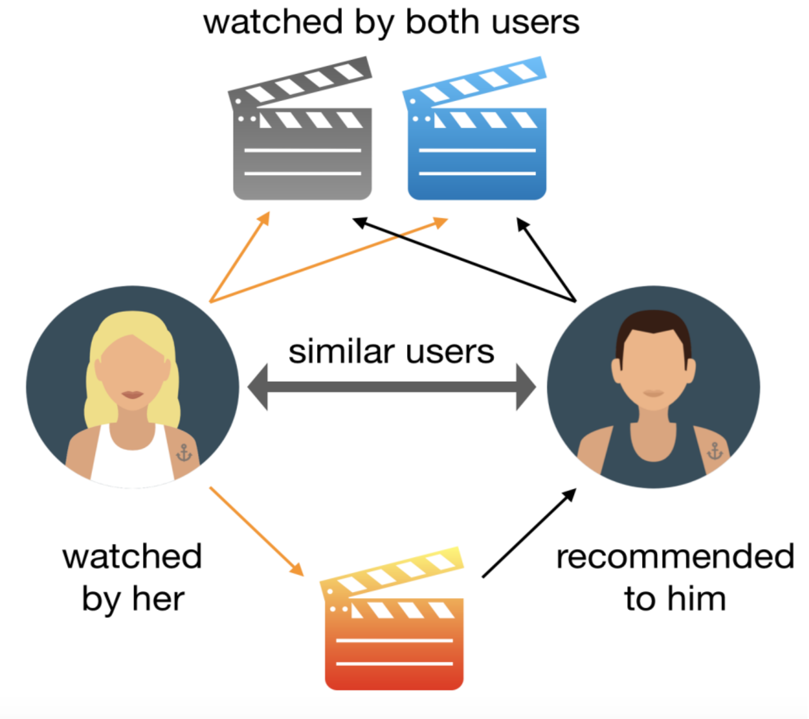
- Therefore, feature engineering was performed and many new features such as duration, date time, month, week, number of search, solo trip or family trip etc. were extracted from dataset.

- Moreover, hotel booked nights data extracted from check-in date and check-out date which represents the duration of stay each of the user.

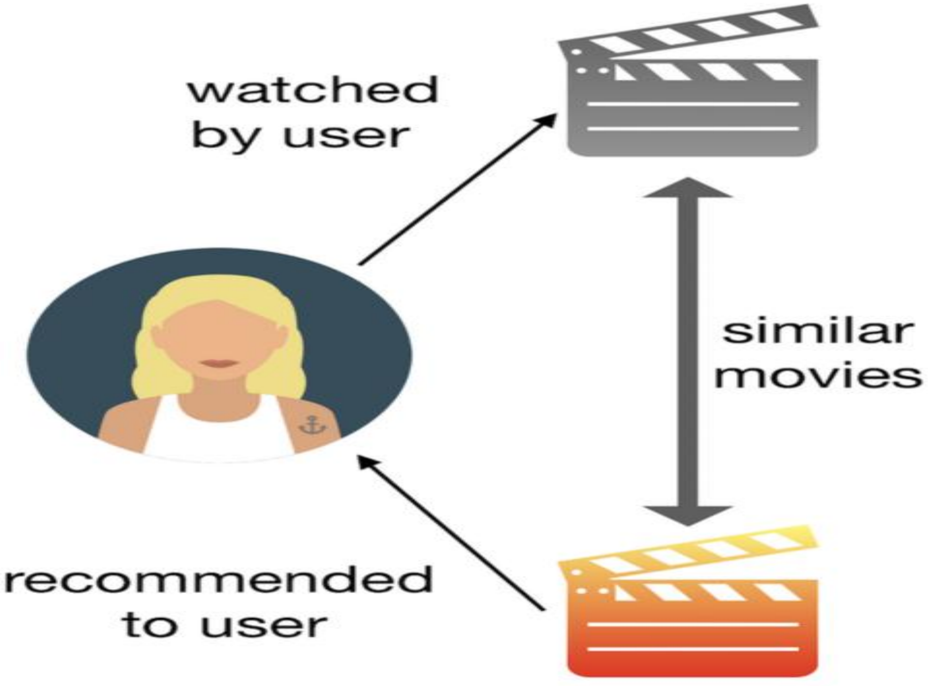
Collaborative-Filtering (Baseline)

The two main families of Recommendation system are Content-Based and Collaborative-Filtering models.

Collaborative-based Filter Model



Content-Based models



- Content-Based models recommend items based on how similar this item is to the other items.

- On the other hand, Collaborative-Filtering models are entirely based on past behaviours and focus on the relationship between users and items.

We have used Collaborative-Filtering model as the baseline model. First of all, We have performed some data analysis for removing duplicates data and understand better user item rating distribution. We have created user-item matrices for train and test set.

- Then, two collaborative filtering models were implemented from scratch.

  - **Memory-Based CF by computing cosine similarity:** Memory-Based Collaborative Filtering approaches can be divided into two main sections: user-item filtering and item-item filtering. In both cases, We created a rating matrix that builds from the entire dataset in order to make recommendations.

  - **Model-Based CF by using singular value decomposition (SVD) and Alternating Least Squares (ALS) method:** Model-Based CF models are developed using machine learning algorithms to predict a user's rating of unrated items.

  - As per my understanding, the algorithms in this approach can further be broken down into 3 sub-types such as Clustering based algorithm, Matrix Factorization, Deep Learning. For example, we can mention few of these algorithms SVD, NMF, KNN etc.

Problem of Baseline Our Model

- In our baseline model (collaborative-filtering recommender system), we used only three features of our large dataset.

- Therefore, we decided to build hybrid recommender systems so that we can use more features of our large dataset which could gives us more better recommendations.

- So, now the point is,
    - ✓ What is Hybrid Recommender Systems?
    - ✓ Hybrid Recommender Recommender systems that recommends items by combining two or more methods together

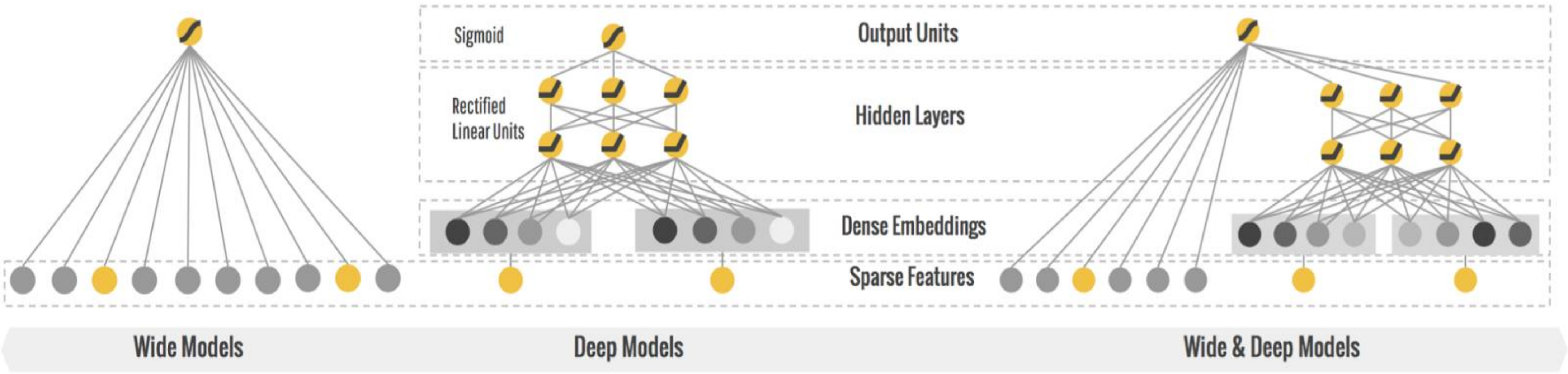
- Hybrid models will also solve the problems of
    - ✓ Memorisation : Learning the co-occurrence items
    - ✓ Generalization.: Exploring the new feature combinations that have never or rarely occurred in the past.

Wide & Deep

- Wide & Deep is a hybrid model that joins trained wide linear models and deep neural networks to combine the benefits of memorization and generalization for recommender systems.

- In this project wide and deep was implemented using DeepCTR package.

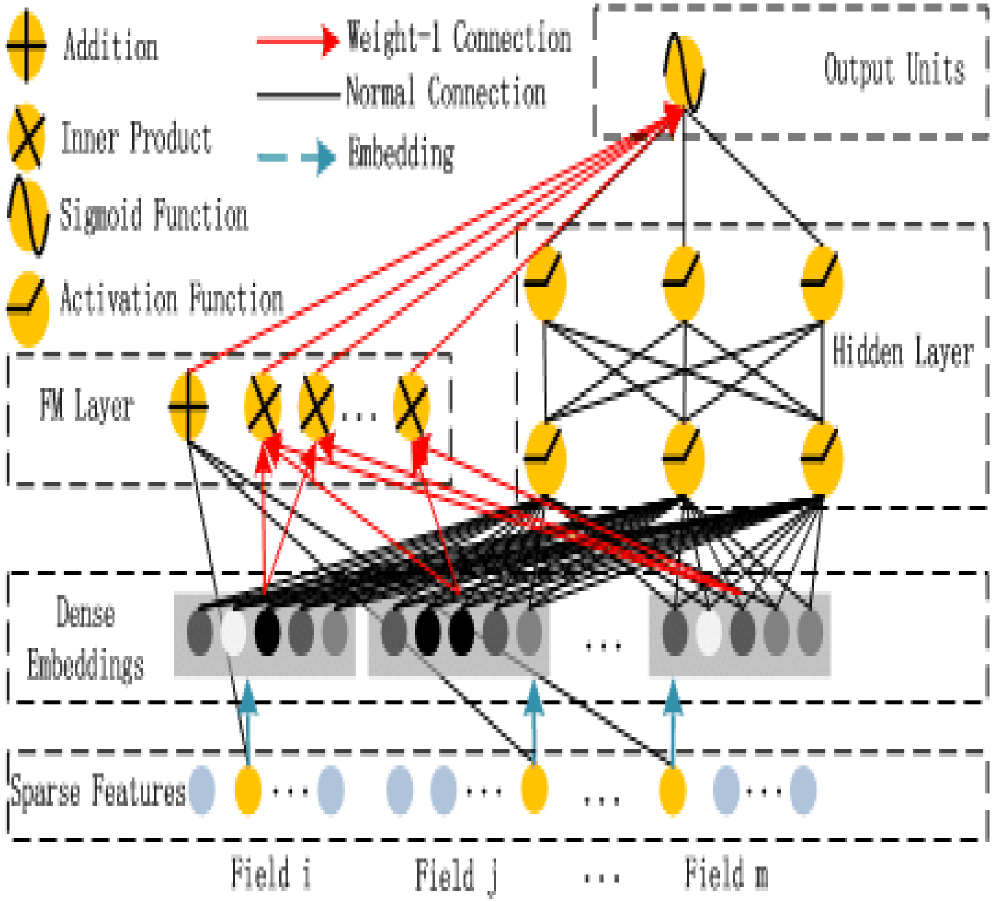
- Features were divided into sparse (categorical) and dense (continuous) features, Label Encoding for sparse features, and normalization for dense numerical features were applied.



- The wide and deep learning has two individual components.
- The wide network is a linear estimator or a single layer feed-forward network which assigns weights to each features and adds bias to them to model the matrix factorization method, as illustrated in above figure (left).
- These feature set includes raw input features and transformed. The deep component is a feed-forward neural network, as shown in above figure (right).
- By jointly training the wide and deep network, it takes the weighted sum of the outputs from both wide model and deep model as the prediction value.

Deep Factorization Machine (DeepFM)

- Compared to the latest Wide & Deep model from Google, DeepFM does not require tedious feature engineering.
- DeepFM has a shared raw feature input to both its “wide” and “deep” components.
- The wide component of DeepFM is an FM layer and the Deep Component of DeepFM can be any neural network.



Evaluation

- Mean Absolute Error (MAE)
  - ✓ The mean of the absolute value of the errors
- Mean Squared Error (MSE)
  - ✓ The mean of the squared errors.
- Root Mean Squared Error (RMSE)
  - ✓ The square root of the mean of the squared errors
- Area under the ROC Curve (AUC)
  - ✓ AUC measures the entire two-dimensional area underneath the entire ROC curve

Results

Model	RMSE	MAE	MSE	AUC
Wide and Deep	0.260	0.140	0.070	0.78
DeepFM	0.241	0.131	0.061	0.82

Technology Stack

❑ Programming

- Python Analysis (Numpy, Pandas)
- Python Visualization (Matplotlib, Seaborn, Plotly)
- Python Machine Learning (SciKit-Learn, Keras)

❑ Software & Tools

  - Python
  - Jupyter Notebook

Generating Recommendation

Finally, we recommended top 5 hotel cluster each of the users. The following table, shows the recommendation for user_id= 1048.

User ID	Hotel Cluster (item_id)
User_id 1048 recommended	hotel cluster: 45
User_id 1048 recommended	hotel cluster: 17
User_id 1048 recommended	hotel cluster: 40
User_id 1048 recommended	hotel cluster: 11
User_id 1048 recommended	hotel cluster: 98

Resources

- https://arxiv.org/abs/1606.07792
- https://arxiv.org/pdf/1703.04247.pdf
- https://medium.com/analytics-vidhya/wide-deep-learning-for-recommender-systems-dc99094fc291
- https://towardsdatascience.com/brief-on-recommender-systems-b86a1068a4dd
- https://towardsdatascience.com/how-to-build-from-scratch-a-content-based-movie-recommender-with-natural-language-processing-25ad400eb243
- https://blog.cambridgespark.com/nowadays-recommender-systems-are-used-to-personalize-your-experience-on-the-web-telling-you-what-120f39b89c3c
- https://towardsdatascience.com/various-implementations-of-collaborative-filtering-100385c6dfe0
- https://towardsdatascience.com/recsys-series-part-5-neural-matrix-factorization-for-collaborative-filtering-a0aebfe15883
- https://towardsdatascience.com/evaluation-metrics-for-recommender-systems-df56c6611093
- https://medium.com/@m_n_malaeb/recall-and-precision-at-k-for-recommender-systems-618483226c54
- https://towardsdatascience.com/ranking-evaluation-metrics-for-recommender-systems-263d0a66ef54
- https://towardsdatascience.com/ranking-evaluation-metrics-for-recommender-systems-263d0a66ef54

Thanks for listening!