**Overview:**

Makemytrip users searching for the flights make activity in the app to look for the flights and book them. This activity can be across multiple search contexts, search context is defined as combination of from\_city:to\_city:travel\_date. User might be searching for multiple dates and destinations. We have this activity data and would like to identify the search context user would finally book for.

**Problem statement:**

For a sampled point in time of a user, you are given the data of the user activity in the past 7 days and the activity in next 7 days. You need to build a model that scores each of the search context user made in the past 7 days. This score would be used to rank the search contexts and high scored search context should have higher booking chances in the next 7 days compared to the low scored one.

**Data:**

File type: parquet

Each line item is a time in point sample of a user:

{

“context”: <list of user activity in past 7 days>,

“affinity”: <list of user activity in next 7 days>

**}**

Each activity by the user looks like this:

{

“features”: array of size 24,

“from\_id”: <id of the source airport>,

“to\_id”: <id of the destination airport>

}

Array feature definition:

Index 0-5: (lat, lon, area) from and to city features

Index 6: travel date in integer format (days relative to 2023-01-01)

Index 7-9: travel date features day of the week, day of the month, day of the year

Index 10: search timestamp in integer format (days relative to 2023-01-01)

Index 11-14: other search related fields like travel class, pax etc

Index 15: Activity type

0: user viewed listing page for search context

1: user went a step ahead and looked for a flight

2: user went till the add traveler details page

3: user made a booking

Index 16-23: price related features

\*\*To get the search context of an activity, you may concatenate from\_id:to\_id:features[6]

Analyzing the Problem Statement:

Source: “You need to build a model that scores each of the search context user made in the past 7 days.”

Learning: The training data for building the model would be the activities in last 7 days.  
If needed a validation split, then it should only be created from within this last 7 days activities.

The activities of the next 7 days, would be used as a Test set to get an approximation of “Best Model’s Performance on Unseen Data”. This test set will be used only once.

Source: “high scored search context should have higher booking chances in the next 7 days compared to the low scored one.”

Learning: The Activity type column has 4 values provided: [0, 1, 2, 3].

3 refers to a booking.

The problem statement could be formulated in 2 ways:

Should we try to learn a Ranking Model, that scores 0 class obs < 1 class obs < 2 class obs < 3 class obs ??

In this approach, we could have more info about the class in which the search context is falling and interventions could be made to move the search context to the next higher class

Another option is to learn a Binary Class Model that scores whether an observation belongs to class 3 or not ?

Here the main goal would be to separate the search context that will complete the booking from all other search contexts, which would not go till the final stage.

Since the problem statement specifically mentions --> “high score --> higher booking chances”, we’ll formuated this as a Binary class problem with ACTIVITY TYPE == 3 as class 1 and all others as class 0.

Other