Report Phase 2

Assigned	C Софья Полозова
Projects	Phase II - Data engineering/preparation
Status	Done

Chapter 3: Data preparation

The data preparation stage encompasses all tasks involved in creating the ultimate dataset (the data that will be used in machine learning pipelines) from the original raw data. These tasks are often carried out iteratively and without a set sequence. They involve selecting tables, records, and attributes, as well as transforming and cleaning the data in preparation for the modeling phase.

Section 3.1. Select data

The full list of the selected data is:

`Price`

`Beds`

`Baths`

`sq.ft`

`Floor`

`Move_in_date`

`Amenity`

`Address`

`City`

`Units`

`Northern_Exposure`

`Southern_Exposure`

`Eastern_Exposure`

`Western_Exposure`

`Balcony`

`Walk_In_Closet`

`Fireplace`

`City_Skyline`

`Kitchen_Island`

`Stainless_Appliances`

`Renovated`

`Office_Space`

`Days_Till_Available`

`Day_of_the_week_recorded`

`Unique_ID`

`Estiamted_Vacancy`

All these cols provide useful information for future modeling phase. But there are cols that are dropped because of the following reasons:

`Unnamed: 0` - need to be removed, because it is just index of the row.

`building_id`, `unit_id`, `Apartment Name` - dublicate the information from different text columns. These cols duplicate the col `Unique_ID`.

`Day_Recorded` - is dropped because this field can be calculated using columns `Move_in_date(new)` and `Days_Till_Available`.

`URL` - This feature does not provide useful information for a model as it is. Therefore, the feature is dropped.

Section 3.2. Clean data

The subsequent columns were filled with the mode value from each column:

1. `Days_Till_Available`
2. `Northern_Exposure`
3. `Southern_Exposure`
4. `Eastern_Exposure`
5. `Western_Exposure`
6. `Balcony`

7. `Walk_In_Closet`

8. `Fireplace`

9. `City_Skyline`

10. `Kitchen_Island`

11. `Stainless_Appliances`

12. `Renovated`

13. `Office_Space`

14. `Unique_ID`

These columns contain categories that represent unique choices. Filling missing values with the most common option is a logical approach for these types of features. Also, we have less than 4% missing values in these cols. Therefore, the approach is the most appropriate for our task.

Column `Move_in_date` has some missing values, but there is another Date Column which is `Day_Recorded`, which shows that when the sale is recorded and there is another columns `Days_Till_Available`(which has missing values too, but we can impute them with most frequent strategy), which shows that when they move in. So if we add the `Days_Recorded` in `Days_Till_Available`, we will eventually get the columns `Move_in_date`, without any Null values.

Section 3.3. Construct data

The `Move_in_date(new)` column was split into `Move_in_date_month`, `Move_in_date_day`, and `Move_in_date_year` that correspond to the month, day, and year. But we dropped `Move_in_date_year` because the year always remains the same.

The `City` column underwent one-hot encoding as it consists of categorical variables without any inherent order. By using one-hot encoding, we prevent the introduction of misleading ordinal relationships and generate distinct binary features for each category. This enables the model to understand and analyze the unique impact of each category on the price independently.

We used label encoding for `Day_of_the_week_recorded`, arranging the values in order of frequency.

There are text features:

1. `Address`

2. `Unique_ID`

3. `Amenity`

Encoding these text features using embeddings(Word2Vec is a popular technique for natural language processing (NLP) that involve using neural networks to create word embeddings. These embeddings are densevector representations of words that capture their meanings and relationships based on the context in which they appear in a large corpus of text.) because they contain useful information for the future models.

Next, we need to process the vector columns into separate single-value columns to further build the model.

Section 3.4. Standardize data

Standardization of a dataset is a common requirement for many machine learning estimators: they might behave badly if the individual features do not more or less look like standard normally distributed data (e.g. Gaussian with 0 mean and unit variance). We applied Standard Scaling for numeric columns except the target - `Price`, and binary columns: `Northern_Exposure`, `Southern_Exposure`, `Eastern_Exposure`, `Western_Exposure`, `Balcony`, `Walk_In_Closet`, `Fireplace`, `City_Skyline`, `Kitchen_Island`, `Stainless_Appliances`, `Renovated`, `Office_Space`, `Boston`, `Denver`, `Los Angeles`, `New York City`, `Orange County`, `San Diego`, `San Francisco`, `Seattle`, `Washington DC`.

Columns for scaling:

`Beds`

`Baths`

`sq.ft`

`Floor`

`Days_Till_Available`

`Units`

`Estiamted_Vacancy`

`Move_in_date_day`

`Move_in_date_month`

Gantt chart and the entire project page at link.