Predicting the Interest Level of Rental Listings

BIA 656 Final Report

Christina Eng

Ying Liu

Salman Sigari

Haoyue Yu

05/08/2017

Abstract

This Kaggle competition is co-hosted by Two Sigma and Renthop, a portfolio company of Two Sigma Ventures. The data was analyzed, feature engineering was conducted, and different kinds of machine learning algorithms were applied to predict how popular an apartment rental listing is. Price was discovered to be the most important feature to determine the popularity of the listing. After creating several predictive models for this dataset, the random forest algorithm was determined to perform best out of the methods evaluated.

1. Introduction

RentHop is a web and mobile-based search engine that allows users to search for apartments in New York City. Renthop classifies the listing data to find groups of posters that are similar. Because different areas have different acceptable demands when it comes to price and number of rooms, there are a decent amount of classes broken down by interest level and neighborhood. As a participant of the Kaggle competition, our team was required to build a model with high accuracy which predict the probability of each interest level for each rental listing based on the listing content like text description, photos, number of bedrooms, price, etc. This project aimed to help Renthop better handle fraud control, identify potential listing quality issues, and allowed owners and agents to better understand renters' needs and preferences.

2. Data

2.1 Data Source

The dataset came from Renthop.com, an apartment listing website. The data was provided through Kaggle. All apartment listings were located in New York City. There were two json files: a training set and testing set.

2.2 Exploratory Data Analysis

The training dataset consisted of 49,352 records and each record represented a rental listing. There were 14 variables and 1 target variable "interest_level" labeled as "High", "Medium", and "Low". 14 variables consisted of attributes for each apartment, which includes numeric, categorical, and text data. The full breakdown of data is described in Appendix A.

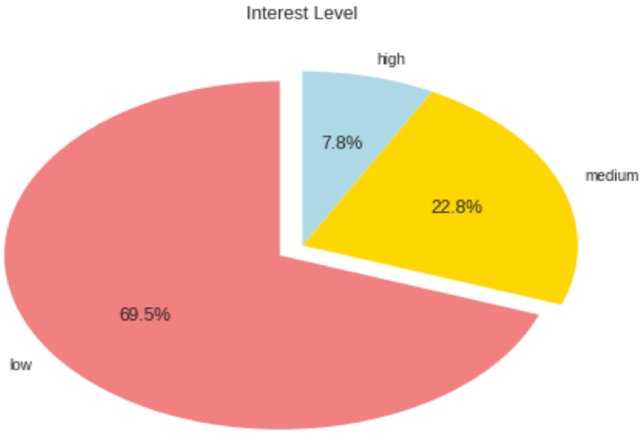

Figure 1: Proportion of rental listings with different interest level.

Figure 1 showed very few listings were with high interest level and most listings were labeled as low interest level in the training dataset.

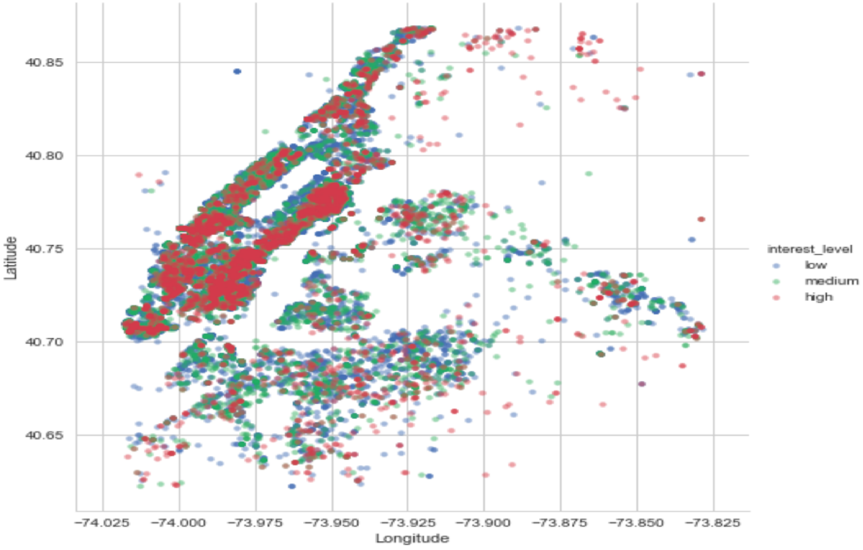

Figure 2: Distribution of rental listings with different interest level

Figure 2 showed that most rental listings with high interest level were located in Manhattan.

3. Data Preparation

Integrated data is critical to data mining. Therefore, feature engineering was conducted to make sure data is integrated to produce format required for data mining. There were two kinds of features: numerical and categorical features.

3.1 Numerical Features

The dataset consisted of 6 numerical features: "bathrooms", "bedrooms", "latitude", "longitude", "listing_id", and "price". Data processing for numerical features included:

- Created new feature "room_sum" by adding up "bathrooms" and "bedrooms".
- Used "latitude" and "longitude" features to determine the zip code of each location with the geocoder package and generated a new feature "zip_id".
- Log of the "price" feature and generated the new feature "log price". This was done to smooth any extremely high prices.
- Divided the price by "room_sum" and added a new feature "price_per_room".
- Divided the price by "bedrooms" and added a new feature "price_t"

3.2 Non-numerical Features

The dataset consisted of 8 non-numerical features: "building_id", "created", "description", "features", "manager_id", "photos", "display_address", and "street_address". Data processing for categorical features included:

- Created features "created_year", "created_month", and "created_day" based on "created".
- Counted the number of photos for each rental listing and set a new feature " num_photos"
- Created a few dummy variables for "feature" column. There were a bunch of amenities for each rental listing. Based on the frequency of occurrence of each amenity, a few dummy variables were set for the most popular amenities. For example, for elevator, 1

represented if elevator was available for rental listings, otherwise 0. The features extracted from this column were “No Fee”, “Elevator”, “Hardwood Floors”, “Pre-War”, “Laundry in Unit”, “Dishwasher”, “Laundry in Building”, “Cats Allowed”, “Dogs Allowed”, “Doorman”, “Fitness Center”, “Dining Room”, “Roof Deck”, “Outdoor Space”, “Balcony”, “High Speed Internet”, and “Swimming Pool”.

- Encoded "manager_id" and "building_id" with value between 0 and the number of classes minus 1.

3.3 Target variable

Categorical target variable "interest_level" was transformed into numerical data. 0 represented "low" interest level, 1 represented "medium" interest level, and 2 represented "high" interest level.

After processing the data, more features were obtained. The new dataset is described in Appendix B.

4. Modeling

The goal was to determine probabilities of low, medium, and high levels of interests in renting each apartment, which would eventually decide whether this apartment was attractive. The training dataset was spilt into 80% training set and 20% testing set. To predict interest level, logistic regression, decision tree, naive bayes, k-nearest neighbors, bagging, adaboost, and random forest algorithms were applied.

4.1 Baseline Model

Logistic regression was the base model to test each classifier and its log loss and accuracy, to find out the most accurate model to help the company find the most interesting listings for its customers. Logistic regression is a very simple and reliable algorithm to use for solving linear problems with categorical outcomes. However, for a large dataset with 49,352 observations in training dataset and 74,659 observations in test dataset, logistic regression may not be the best choice for solving this problem. Hence, other methods were tested, such as decision tree, which was known to be an algorithm that better dealt with large datasets. However, in the RentHop problem, there were many complicated features. The trees may become too large even after some pruning, so decision tree may not be the best model. Naive bayes and k-nearest neighbor were also viable methods to look at. Naive bayes could give the smallest prediction error, but in this case, continuous features such as price, bedroom and bathroom numbers, log-price, and date could affect the result of naive bayes classifier. For this reason, KNN was used, which could be more suitable to deal with continuous variables. However, this algorithm also had disadvantages, such as it is expensive and slow when there is a large dataset.

4.2 Ensemble Methods

Ensemble methods were also attempted since they may outperform the single algorithms.

1) Bagging

Bagging was used on decision tree, which focused on resampling and could reduce variance in comparison to regular decision trees that were used before. It could also handle qualitative features that we used in feature function. The cons of this algorithm was that it will slightly increase bias. In addition, in bagging, if some features were correlated, the variance could not be reduced. In the RentHop dataset, it is highly possible that price and location were correlated, which would reduce the effect of bagging.

2) AdaBoost

Another method used to build the predictive model was AdaBoost. AdaBoost algorithm focuses on reweighing and reducing bias. It is more interpretable than bagging. Cons of the algorithm are that it can overfit if the number of trees is too large and overly complex weak hypotheses could affect the result.

3) Random Forest

The last algorithm considered was random forest. Using random forest, overfitting was avoided and it reduced bias in model. A very important point is that random forest can help decorrelate trees. In the RentHop problem, attributes such as price, log-price, the number of bathrooms, the number of rooms, location, building and management could have correlation. By using random forest algorithm, the problem of correlation can be resolved and it can determine the interest level of each apartment based on all of the features that were generated.

5. Evaluation

After the training models were created from each algorithm, the models were evaluated to determine which model had the best performance. For comparison, the base algorithm was chosen to be logistic regression because it is a common model for this type of problem. Two metrics were used to evaluate models.

5.1 Log-loss

$$Logloss = -\frac{1}{N}\sum_{i=1}^{n}\sum_{j=1}^{n} y_{ij} \, log(p_{ij})$$

where N is the number of listings in the test set, M is the number of class labels (3 classes), log is the natural logarithm, y_{ij} is 1 if observation i belongs to class j and 0 otherwise, and p_{ij} is the predicted probability that observation i belongs to class j.

Since the Kaggle competition used the log loss metric to rank the performance of the predictive models, the log loss was first calculated for each algorithm. The lower the log loss, the more accurate the classifier. To calculate log loss, a probability must be assigned to each class and the metric heavily penalizes classifiers that are confident about incorrect classifications.

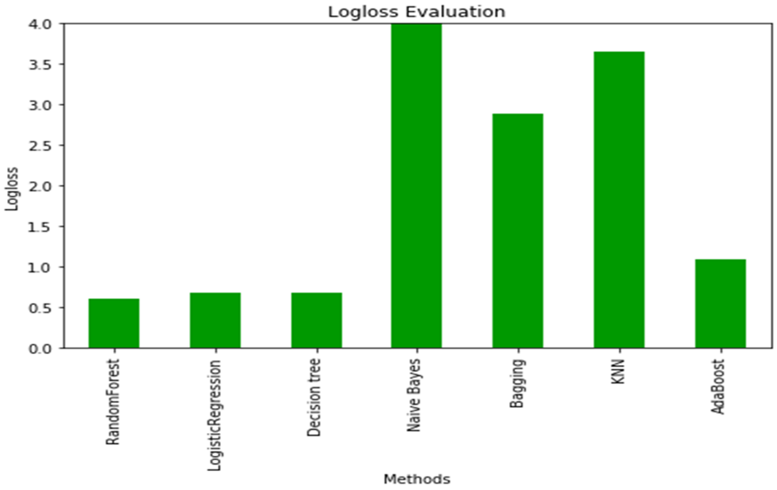

Figure 3: Log loss comparison for all the algorithms.

Figure 3 showed a plot of the log loss for each model. Decision tree had a similar log loss to logistic regression and all other models had a higher log loss. Random forest had the lowest log loss out of the models. Therefore, from a Kaggle competition point of view, the best model out of the ones evaluated was random forest.

5.2 Accuracy score

There were other ways to evaluate the classifiers to better understand the performance of the model. Another metric that was used was the accuracy score. The higher the accuracy score, the better the model.

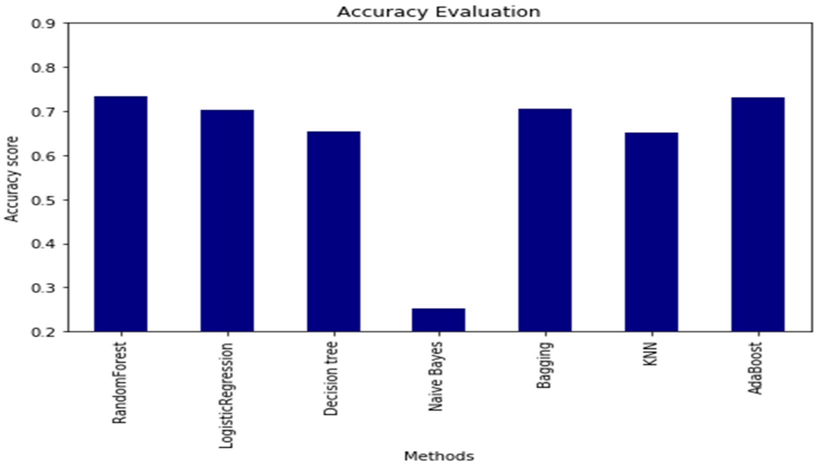

Figure 4: Accuracy score comparison for all the algorithms

Figure 4 showed a plot of the accuracy score for each model. Random forest was again the best model per the accuracy score metric.

5.3 Evaluation on Random Forest

To further understand the random forest model, the confusion matrix and classification report was generated. The confusion matrix is shown in Table 1. The classification report is shown in Table 2, which shows the precision, recall, and F1-score for each of the interest levels. Interpreting the two tables, the model seems to be relatively conservative. It is better at predicting low interest than medium or high interest. The training data has more training data on low interest apartments than higher interest ones, which is why it may be better at predicting low interest listings.

	Predicted			
True		High	Medium	Low
	High	284	457	523
	Medium	219	1061	2537
	Low	75	784	10840

Table 1: Confusion matrix from random forest model

	Precision	Recall	F1-score	Support
High	0.49	0.22	0.31	1264
Medium	0.46	0.28	0.35	3817
Low	0.78	0.93	0.85	11699
Avg/total	0.69	0.73	0.69	16780

Table 2: Classification report from random forest model

5.4 Feature importance

It is important to see what influences the interest level of renters so that Renthop knows what kind of apartments interest customers the most. They can use this information to inform people who post rental listings on their website how to attract more renters to their property. To see which attributes contribute the most to the interest level of the apartment, the importance of features was ranked using the extremely randomized trees algorithm. The plot generated in Figure 5 shows that price, created day and month, building ID, manager ID, number of photos, location, and number of rooms are the most important attributes. Out of the building features extracted from the features column of the data set, the most important feature was no-fee, elevator, hardwood floors, and pre-war style. Therefore, it is important to tell posters of rental listings to fill in all these details and to market the apartments using these details, if applicable. Renthop can anticipate popular rental listings with this information and then promote the anticipated popular listings on their website such that more renters will want to go on their website. With this tactic, they can potentially reach more users and then more posters would want to post on their website. This can help improve their profit since Renthop earns money from the rental listings. Renthop can also use this information to improve the ranking of listings to aid renters in finding apartments.

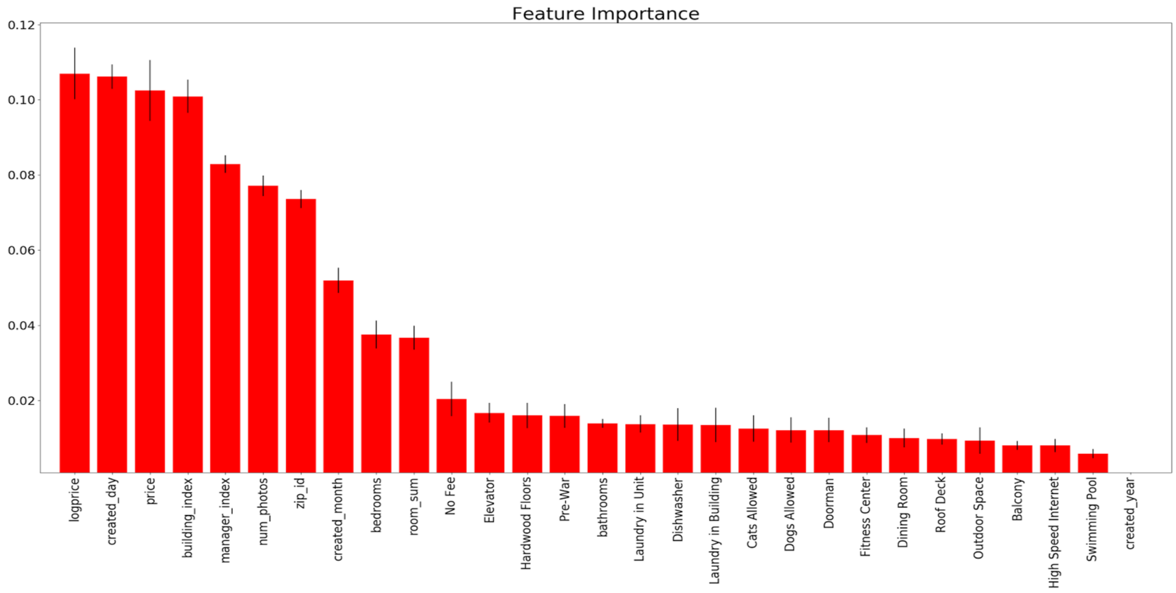

Figure 5: Feature importance ranking

Although this model can help Renthop improve their marketing of rental listings to gain more renters to look on their website, it will be difficult to measure the return on investment gained from this model. The cost will be the amount of time and resources spent on building and deploying the model. The profit from this initiative will be difficult to determine. There are other ongoing ways to gain more users, such as through advertisement. In addition, the amount of renters is dependent on the renter's market, which may differ month to month. A good way to see the effect of this initiative is to compare the amount of views and listings to the amounts from the previous year. It will be difficult to tell if it is directly from this initiative unless all previous marketing stays the same.

6. Deployment

The model created to predict the interest level of apartment listings can be deployed to predict the interest level of future apartment listings. Renthop ranks apartments for their viewers such that the most top quality apartments will be the first to show up. This model can help determine the quality of apartment listings because high quality is related to high interest. The model can automatically rank the apartment listings based on interest level and then put the highest ranked ones on the top of the page. This would keep the interest of renters to keep using their website to search for apartment listings.

Some issues that may arise during deployment is if the model erroneously ranks the listing as a high interest when indeed it is a low interest or if it ranks the listing as low interest when indeed it is a high interest. The latter is not as common but as shown through the model evaluation, it was common to rank the some of the low interest apartments as high interest apartments. This may be solved by having a training model with more high interest level apartment listings, which would involve collecting more data or resampling the data to make it more balanced. If the model ranks a lot of low interest apartments highly, then it may make customers lose interest in searching on their website since they may not be looking at listings that they are interested in. In addition, people who post low interest apartment listings may be deterred from posting on Renthop. Since Renthop earns profit from posters of apartment listings, Renthop should encourage them to post on their website while helping them understand how to make their apartments listings more interesting to renters.

Trying to help posters make a possible low interest apartment listing in to a higher interest listing may bring up ethical problems. Renthop needs to be honest to their users and posters. Even though Renthop can give advice to posters on how to a make their listings more attractive, they need to make sure that the posters are still being honest and not making up any facts to make their listings more attractive. Therefore, if Renthop wants to inform their posters how to make their postings more attractive, Renthop needs to make sure they know the consequences of giving incorrect information. Otherwise, if incorrect information is posted on the website, it may hurt Renthop and its image. These risks should be handled by having someone verify the information on the postings before they are posted.

7. Conclusion

Different kinds of models were tried to predict the probability in this project. The best performance is Random Forest with 74% accuracy score. Based on feature importance measured by extremely randomized trees algorithm, price was determined to be the most important feature to influence the popularity of the listing. Therefore, Renthop may apply this evaluated model to more accurately anticipate how popular a rental listing is in the future.

8. Future work

There are some limitations in this project, which limits the result to be better.

- The accuracy score of best evaluated model Random Forest is not high enough (74%)
- The issue of unbalanced data was not solved. Few listings were with high interest level and most listings were labeled as low interest level in the training dataset.
- Multiple algorithms were compared but not combined to see if combination of different algorithms would improve the performance.

Therefore, several future works are necessary to be done.

- XGBoost is a good model that could be tried to avoid the problem of overfitting. To achieve better results, parameters tuning for XGBoost could also be considered.
- Second, it is worthwhile to combine models rather than only compare their performance.
- Then, keeping exploring and optimizing more features is another good method to improve the accuracy of models.
- Last but not least, spending more time on clustering the dataset especially the samples with "Low" labels could be considered so that the dataset could be more balanced.

References

https://www.kaggle.com/c/two-sigma-connect-rental-listing-inquiries

http://scikit-learn.org/stable/modules/ensemble.html

http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html

http://scikit-learn.org/stable/modules/feature_selection.html

Appendix

Appendix A: Full breakdown of original data fields

- bathrooms: number of bathrooms
- bedrooms: number of bathrooms
- building_id

- created
- description
- display_address
- features: a list of features about this apartment
- latitude
- listing_id
- longitude
- manager_id
- photos: a list of photo links. You are welcome to download the pictures yourselves from Renthop's site, but they are the same as imgs.zip.
- price: in USD
- street_address
- interest_level: this is the target variable. It has 3 categories: 'high', 'medium', 'low'

bathrooms	bedrooms	building_id	created	description	display_address	features	street_address
1.5	3	53a5b119ba8f7b 61d4e010512e0dfc85	2016-06-24 07:54:24	A Brand New 3 bathrooms...	Metropolitan Avenue	Doorman, Elevator, Fitness Center,	792 Metropolitan Avenue
interest_level	latitude	listing_id	longitude	manager_id	photos	price	
medium	40.7145	7211212	-73.9425	5ba989232d0489da1b5f2c45f6688adc	[https://photos.renthop.com/2/721	3000	

Appendix B: One sample of new data fields with features_to_use

Features_to_use = ["bathrooms", "bedrooms", "price", 'logprice',"room_sum", "num_photos","Elevator","Dogs Allowed","Hardwood Floors","Cats Allowed","Dishwasher","Doorman","No Fee","Laundry in Building","Fitness Center","Pre-War","Laundry in Unit", "Roof Deck”, "Outdoor Space", "Dining Room", "High Speed Internet","Balcony","Swimming Pool","Created_year", "Created_month","Created_day","building_index","manager_index","zip_id"

bathrooms	bedrooms	building_index	created_year	Elevator	manager_index	zip_id	log-price
1.5	3	14	2016	1	19	8	2.477
interest_level	room_sum	price_per_room	price_t	num_photos	created_month	price	
medium	4.5	666.666	1000	5	6	3000	