

Open in app ↗

Sign up

Sign in



Search



Learning To Rank Diversely



Malay Haldar · Follow

Published in The Airbnb Tech Blog

6 min read · Jan 30, 2023



Listen



Share

by [Malay Haldar](#), [Liwei He](#) & [Moose Abdool](#)

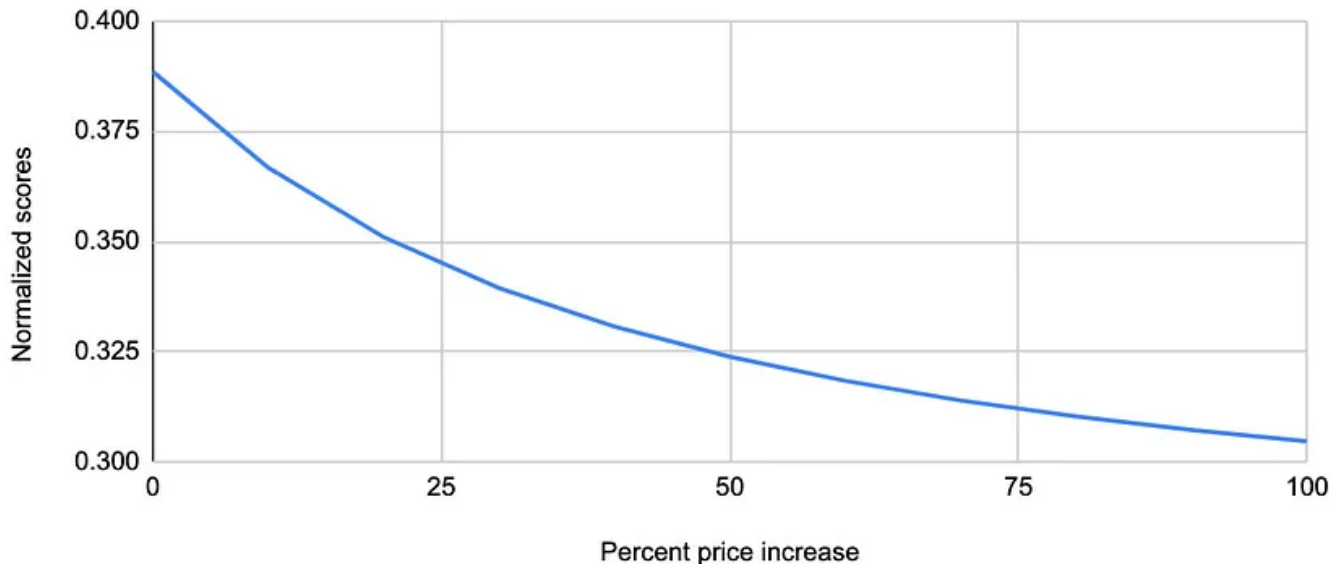
Airbnb connects millions of guests and Hosts everyday. Most of these connections are forged through search, the results of which are determined by a neural network-based ranking algorithm. While this neural network is adept at selecting *individual listings* for guests, we recently improved the neural network to better select the overall *collection of listings* that make up a search result. In this post, we

dive deeper into this recent breakthrough that enhances the diversity of listings in search results.

How Does Ranking Work?

The ranking neural network finds the best listings to surface for a given query by comparing two listings at a time and predicting which one has the higher probability of getting booked. To generate this probability estimate, the neural network places different weights on various listing attributes such as price, location and reviews. These weights are then refined by comparing booked listings against not-booked listings from search logs, with the objective of assigning higher probabilities to booked listings over the not-booked ones.

What does the ranking neural network learn in the process? As an example, a concept the neural network picks up is that lower prices are preferred. This is illustrated in the figure below, which plots increasing price on the x-axis and its corresponding effect on normalized model scores on the y-axis. Increasing price makes model scores go down, which makes intuitive sense since the majority of bookings at Airbnb skew towards the economical range.



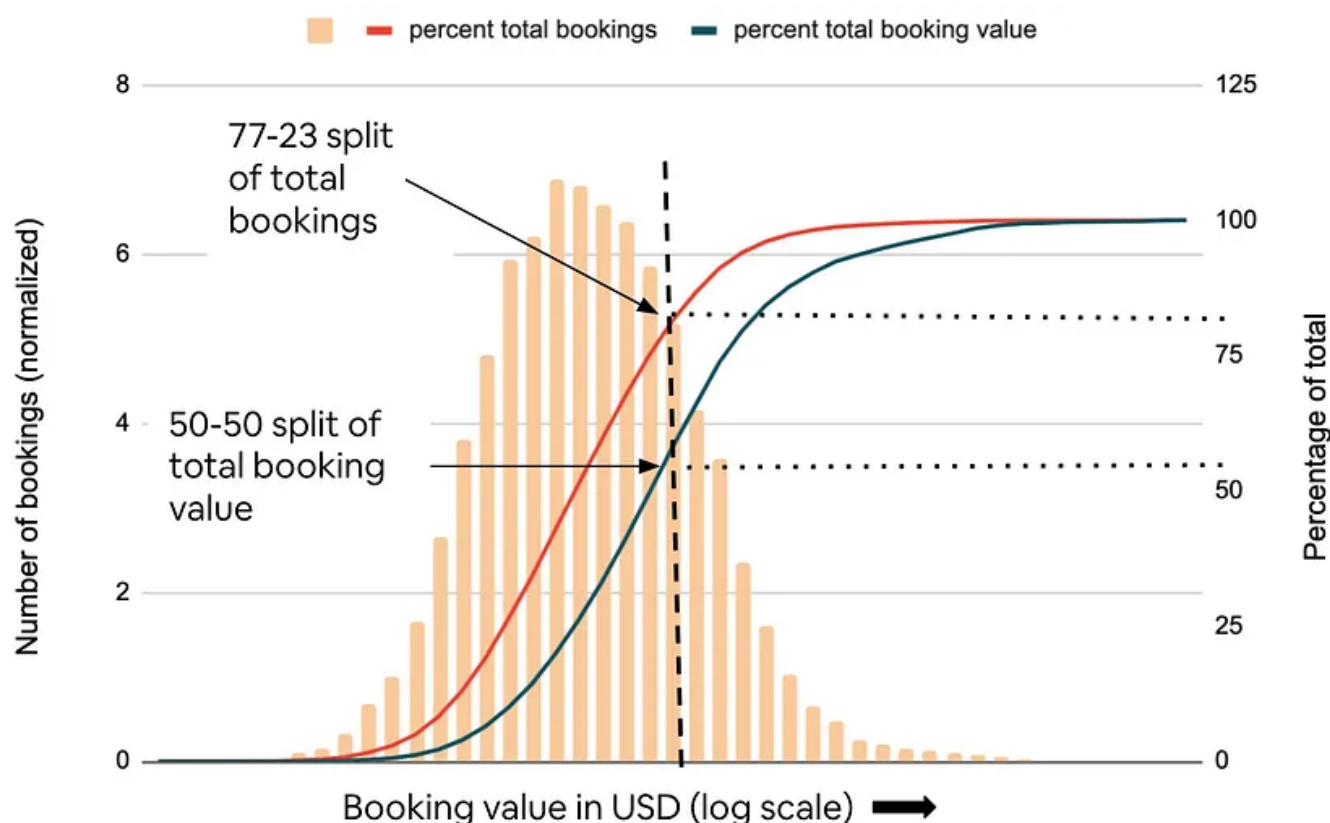
Relation between model scores and percent price increase

But price is not the only feature for which the model learns such concepts. Other features such as the listing's distance from the query location, number of reviews, number of bedrooms, and photo quality can all exhibit such trends. Much of the complexity of the neural network is in balancing all these various factors, tuning them to the best possible tradeoffs that fit all cities and all seasons.

Can One Size Fit All?

The way the ranking neural network is constructed, its booking probability estimate for a listing is determined by how many guests in the past have booked listings with similar combinations of price, location, reviews, etc. The notion of higher booking probability essentially translates to what the majority of guests have preferred in the past. For instance, there is a strong correlation between high booking probabilities and low listing prices. The booking probabilities are tailored to location, guest count and trip length, among other factors. However, within that context, the ranking algorithm up-ranks listings that the largest fraction of the guest population would have preferred. This logic is repeated for each position in the search result, so the entire search result is constructed to favor the majority preference of guests. We refer to this as the *Majority principle* in ranking — the overwhelming tendency of the ranking algorithm to follow the majority at every position.

But majority preference isn't the best way to represent the preferences of the entire guest population. Continuing with our discussion of listing prices, we look at the distribution of booked prices for a popular destination — Rome — and specifically focus on two night trips for two guests. This allows us to focus on price variations due to listing quality alone, and eliminate most of other variabilities. Figure below plots the distribution.



Pareto principle: 50/50 split of booking value corresponds to roughly 80/20 split of bookings

The x-axis corresponds to booking values in USD, log-scale. Left y-axis is the number of bookings corresponding to each price point on the x-axis. The orange shape confirms the log-normal distribution of booking value. The red line plots the percentage of total bookings in Rome that have booking value less than or equal to the corresponding point on x-axis, and the green line plots the percentage of total booking value for Rome covered by those bookings. Splitting total booking value 50/50 splits bookings into two unequal groups of ~80/20. In other words, 20% of bookings account for 50% of booking value. For this 20% minority, cheaper is not necessarily better, and their preference leans more towards quality. This demonstrates the *Pareto principle*, a coarse view of the heterogeneity of preference among guests.

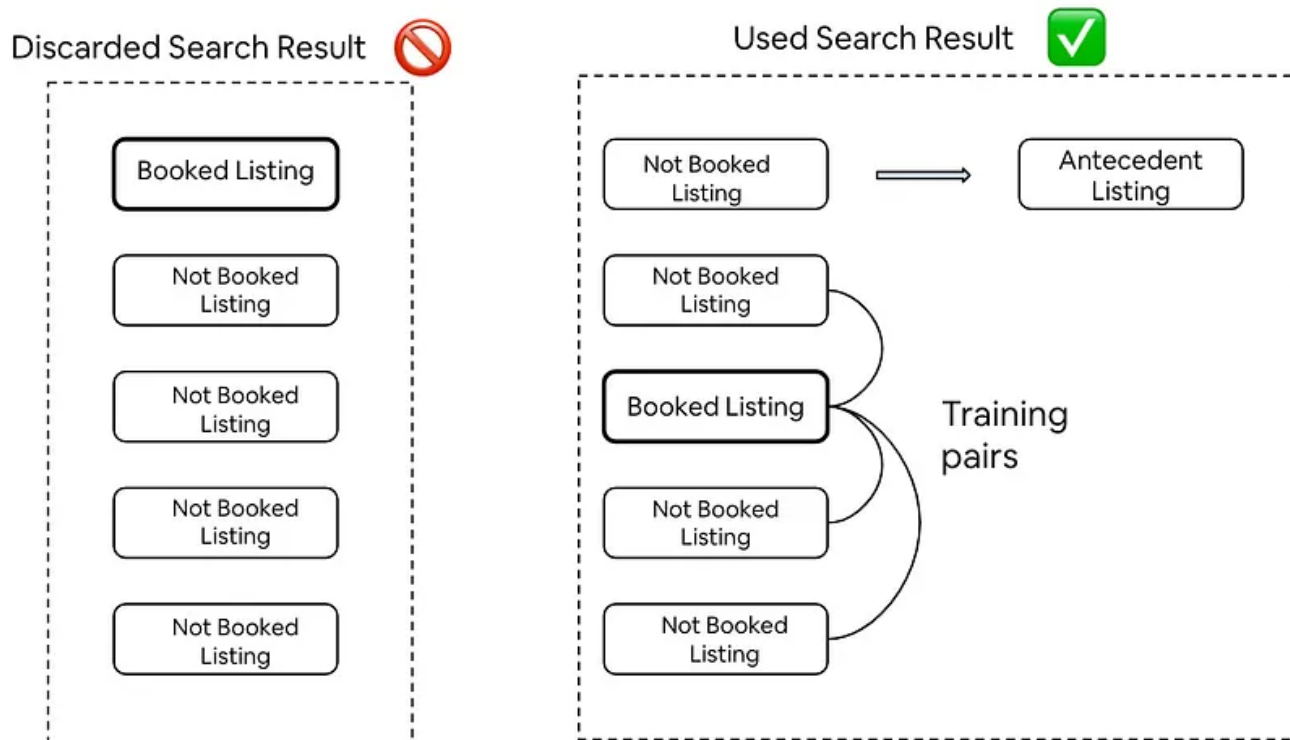
While the Pareto principle suggests the need to accommodate a wider range of preferences, the Majority principle summarizes what happens in practice. When it comes to search ranking, the Majority principle is at odds with the Pareto principle.

Diversifying by Reducing Similarity

The lack of diversity of listings in search results can alternatively be viewed as listings being too similar to each other. Reducing inter-listing similarity, therefore, can remove some of the listings from search results that are redundant choices to begin with. For instance, instead of dedicating every position in the search result to economical listings, we can use some of the positions for quality listings. The challenge here is how to quantify this inter-listing similarity, and how to balance it against the base booking probabilities estimated by the ranking neural network.

To solve this problem, we build another neural network, a companion to the ranking neural network. The task of this companion neural network is to estimate the similarity of a given listing to previously placed listings in a search result.

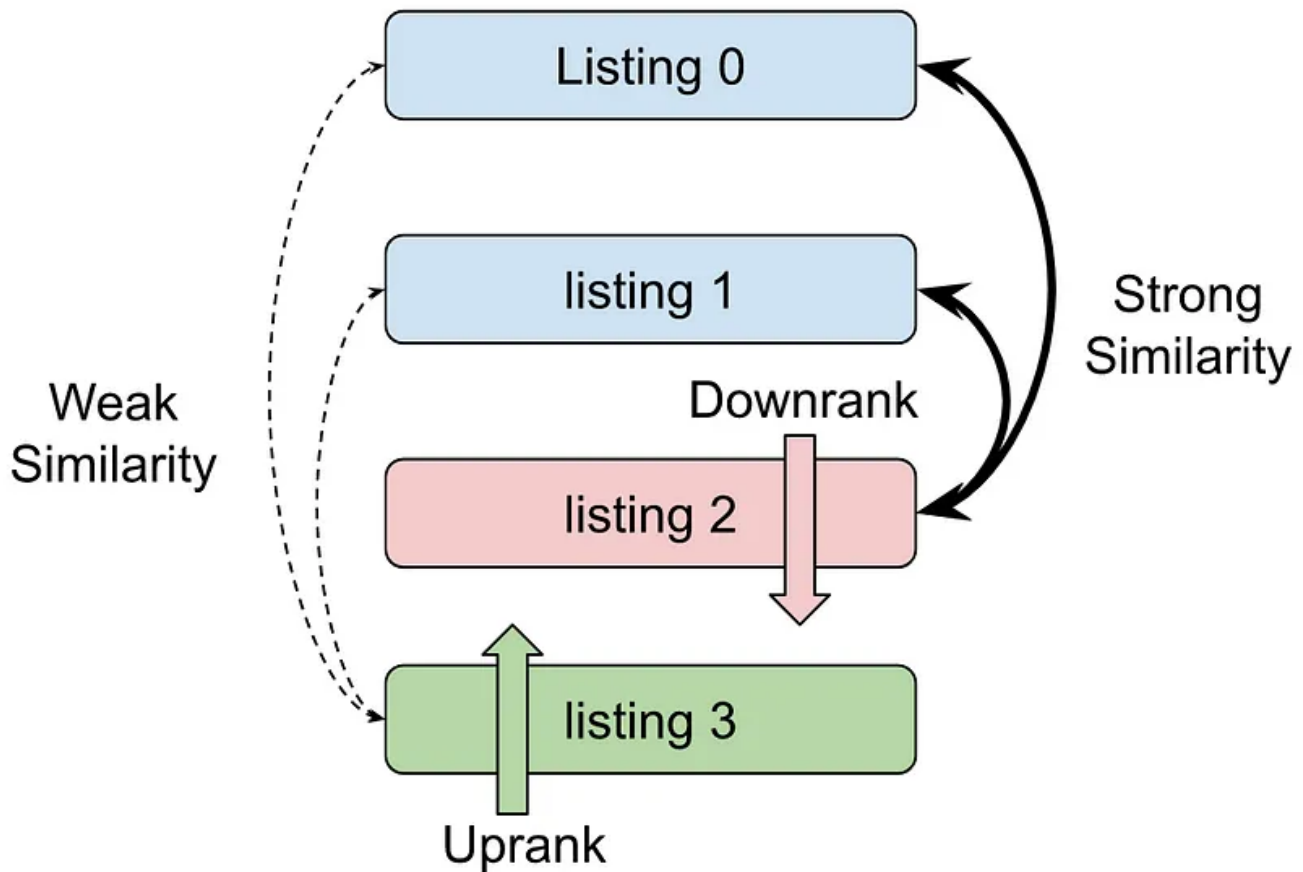
To train the similarity neural network, we construct the training data from logged search results. All search results where the booked listing appears as the top result are discarded. For the remaining search results, we set aside the top result as a special listing, called the antecedent listing. Using listings from the second position onwards, we create pairs of booked and not-booked listings. This is summarized in the figure below.



Construction of training examples from logged search results

We then train a ranking neural network to assign a higher booking probability to the booked listing compared to the not-booked listing, but with a modification — we subtract the output of the similarity neural network that supplies a similarity estimate between the given listing vs the antecedent listing. The reasoning here is that guests who skipped the antecedent listing and then went on to book a listing from results down below must have picked something that is dissimilar to the antecedent listing. Otherwise, they would have booked the antecedent listing itself.

Once trained, we are ready to use the similarity network for ranking listings online. During ranking, we start by filling the top-most result with the listing that has the highest booking probability. For subsequent positions, we select the listing that has the highest booking probability amongst the remaining listings, after discounting its similarity to the listings already placed above. The search result is constructed iteratively, with each position trying to be diverse from all the positions above it. Listings too similar to the ones already placed effectively get down-ranked as illustrated below.



Reranking of listings based on similarity to top results

Following this strategy led to one of the most impactful changes to ranking in recent times. We observed an increase of 0.29% in uncanceled bookings, along with a 0.8% increase in booking value. The increase in booking value is far greater than the increase in bookings because the increase is dominated by high-quality listings which correlate with higher value. Increase in booking value provides us with a reliable proxy to measure increase in quality, although increase in booking value is not the target. We also observed some direct evidence of increase in quality of bookings — a 0.4% increase in 5-star ratings, indicating higher guest satisfaction for the entire trip.

Further Reading

We discussed reducing similarity between listings to improve the overall utility of search results and cater to diverse guest preferences. While intuitive, to put the idea in practice we need a rigorous foundation in machine learning, which is described in [our technical paper](#). Up next, we are looking deeper into the location diversity of results. We welcome all comments and suggestions for the technical paper and the blog post.

Interested in working at Airbnb? Check out [these open roles](#).

Machine Learning

Airbnb

Deep Learning

Engineering

AI



Follow



Written by Malay Haldar

497 Followers · Writer for The Airbnb Tech Blog

math + fiction = reality
