Learning Intent to Book Metrics for Airbnb Search

Bradley C. Turnbull
Airbnb Inc.
San Francisco, California
bradley.turnbull@airbnb.com

ABSTRACT

Airbnb is a two-sided rental marketplace offering a variety of unique and more traditional accommodation options. Similar to other online marketplaces we invest in optimizing the content surfaced on the search UI and ranking relevance to improve the guest online search experience. The unique Airbnb inventory, however, surfaces some major data challenges. Given the high stakes of booking less traditional accommodations, users can spend many days to weeks searching and scanning the description page of many accommodation "listings" before making a decision to book. Moreover, much of the information about a listing is unstructured and can only be found by the user after they go through the details on the listing page. As a result, we have found traditional search metrics do not work well in the context of our platform. Basic metrics of single user actions, such as click-through-rates, number of listings viewed, or dwell time, are not consistently directionally correlated with our downstream business metrics. To address these issues we leverage machine learning to isolate signals of intent from rich behavioral data. These signals have key applications including analytical insights, ranking modeling inputs, and experimentation velocity. In this paper, we describe the development of a model-based user intent metric, "intentful listing view", which combines the signals of a variety of user micro-actions on the listing description page. We demonstrate this learned metric is directionally correlated with downstream conversion metrics and sensitive across a variety of historical search experiments.

CCS CONCEPTS

• **General and reference** → **Metrics**; **Experimentation**; • **Information systems** → **Electronic commerce**; • **Computing methodologies** → **Machine learning**.

KEYWORDS

User Intent Modeling, Experimentation, E-commerce Search

ACM Reference Format:
Bradley C. Turnbull. 2019. Learning Intent to Book Metrics for Airbnb Search. In *Proceedings of the 2019 World Wide Web Conference (WWW '19), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3308558.3313648

WWW '19, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6674-8/19/05.
https://doi.org/10.1145/3308558.3313648

1 INTRODUCTION

Airbnb[1] is a two-sided rental marketplace where users can book unique as well as more traditional accommodations. The guest booking "funnel" is generally defined in three stages:

Search - A user executes searches, likely specifying a location, dates, and guest count. The user may also apply filters to their search to express specific preferences, such as price maximum and/or minimum. Available accommodation listings from our inventory are retrieved, ranked via our relevance models, and presented to the user. A user will explore the results by clicking listings to open the details page, with information such as checkin and checkout times, available amenities, sleeping arrangements, reviews, and more.

Contact - Once a user finds a listing they are interested in, they send a message to the host requesting to book their place. Note that guests can also send messages to hosts asking for clarification of listing details or special requests without sending a formal request to book.

Book - Once the host approves the request, the guest receives a notification and can then finalize the booking. Some hosts opt-in to a program, called "instant book", where booking requests are automatically accepted. It is also possible that the host will reject the booking request, at which point the guest will have to contact other listings to find accommodations.

The search team at Airbnb focuses on optimizing the guest experience from search to booking. We analyze the efficiency at which guests move through the different stages of the conversion funnel to inform product and model improvements, and measure their success in online experiments. We generally observe contacts and bookings almost simultaneously. Therefore, relying primarily on contacts and bookings signals poses two major challenges:

1. **Sparsity** - very few visitors and searchers end up contacting or booking,
2. **Signal Time Lag** - searchers can spend days to weeks searching before contacting or booking, so it can take a similarly long time for lifts or drops in booking signals to manifest within experiments.

Consequently, we are strongly motivated to develop upper-funnel metrics which are less sparse, meaning observed for non-bookers as well, and can be realized faster than downstream business metrics. This is a classical challenge in product optimization shared by many online platforms[2].

Our challenges and contributions are unique in that clicks, view duration, and traditional search metrics are not good leading indicators for bookings on our platform. Movements in click metrics are not consistently correlated with downstream business metrics,

[1] Airbnb.com
[2] https://www.slideshare.net/bonbonsuperbonbon/defining-true-north-metrics-to-quantify-engagement-at-linkedin

making it difficult to draw conclusions from their increases or decreases. We, therefore, develop a new metric based on a machine learning model, screening signals of friction from signals of intent. This metric captures a new stage in the booking funnel between Search and Contact, the searcher found a listing which is a good match for them. We can use this metric for a range of applications: drive more analytics insights of our booking funnel, increase experimentation velocity, and feed less sparse signals to our ranking models.

This paper outlines the challenges of using traditional search engagement metrics within the complex Airbnb booking funnel. However, by sacrificing some simplicity and turning towards a machine learning derived user intent metric, "intentful listing view", we were able to overcome these challenges. We demonstrate the machine learning based metric is directionally correlated with downstream business metrics and sensitive across a variety of historical search experiments. The contributions of our work are two-fold:

1. Developing a simple yet effective approach to capture intent from page engagement signals.
2. Creating a new conversion leading indicator for increased experimentation velocity and less sparse ranking optimization.

2 ENGAGEMENT SIGNALS IN AIRBNB SEARCH

The use of observable user behaviors, such as clicks, is a popular approach to evaluate the quality of online search retrieval and ranking systems [2]. These signals are inexpensive to obtain, especially compared to expert judgments, and can be gathered in large quantities. Given these attractive features, user behavior metrics are relatively well-adopted in the field of Web Search [4]. However, given the signal is naturally noisy and user behavior is multi-faceted, deriving metrics which truly reflect the relevance of results can be challenging [6].

The direct application of user behavior metrics to Airbnb search poses a few challenges. In the context of Web Search, the time it takes a user to complete a "search task" can be measured in minutes and at most a few query reformulations [8]. For Airbnb search we are not afforded such a luxury. Even the most eager to book user can spend 3 days searching, execute 12 unique searches, and view the description page of 11 listings before booking.

Motivated by click-through-rate we implemented a similar concept at Airbnb, number of listing views per user [5]. We evaluated the quality of this metric as a leading indicator of booking, via a meta-analysis of historical experiments. We ideally want to observe a strong correlation and directional agreement between movements in the leading indicator metric and booking conversion. Figure 1 shows there is very mild correlation between the listing view metric and booking conversion for 20 search experiments; confidence intervals are omitted as all results are statistically significant. In 40% of the experiments, represented by green dots, the number of listing views decreases yet booking conversion improves, or the inverse. There is also only a 0.06 Kendall's tau correlation between the metrics. The contradictions and lack of correlation make it difficult to utilize this click-based metric with confidence. At Airbnb clicks and views are not a strong signal of intent. Much of the information

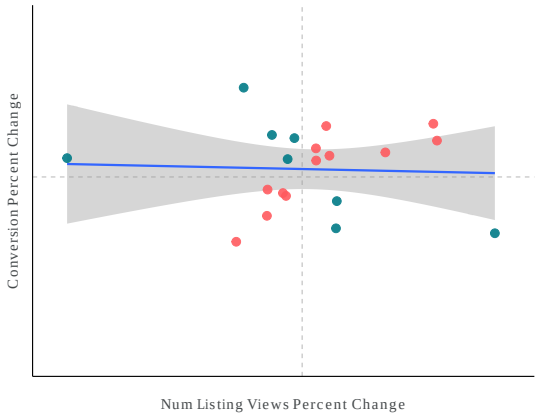

Figure 1: Correlation of Num Listing Views Metric with Booking Conversion Metric

about a listing is unstructured and can only be found after viewing the details on the listing page. Therefore, it is not clear if increases in number of listing views is a signal of consideration or increased friction.

In an effort to remove quickly abandoned listing views, we explored restricting to only long dwell-time views [9, 10]. Across a variety of duration thresholds and transformations as suggested by [11] and [12], we were unable to achieve reasonable correlation with business metrics. Listings have varying numbers of photos, description lengths, and number of reviews; this can cause unwanted bias and variation in comparing view duration across listings. We also explored Airbnb product specific engagement information such as clicking into the price details module on the listing page as well as searching for specific dates. Efforts produced metrics which either did not correlate with business metrics, or were too focussed on a specific path of engagement, and did not generalize well to all searchers. For example, some searchers without specific dates in mind will execute searches without entering dates, and determine their travel dates based on the availability of listings they like. Therefore, a metric related to searching for specific dates would not extend to that user segment.

Taking a step-back, we invested analytical resources in mapping out users' specific product surface touch-points on the path to booking. We identified that viewing the listing description page is a touch point which every guest passes through on their way to booking, and is a step which is relatively high-up in the funnel. We also identified that users perform a range of actions on the listing description page: clicking photos, paginating reviews, etc. The list of actions is extensive and performed in varying sequences, such that it is difficult to comprehend globally positive and negative actions from basic summary statistics. Note that users will also sometimes abandon a listing description page, only to return to it a day later.

Zhou et al. [13] successfully improved e-commerce recommendations using micro-actions between users and items, not solely relying on purchase signals. In addition, Kim et al. [7] trained a machine learning classification model using search query attributes, page attributes, and dwell-time, to predict if a web search result

click is associated with user satisfaction. Motivated by these successes, we attempt to build a machine learning model which can combine the rich collection of user micro-actions on the listing description page to produce a more reliable engagement signal.

3 INTENTFUL VIEW MODEL

Our objective is to build a model which can identify guest listing views which signal that a user is considering booking a listing, versus those which the user has dismissed from consideration. We formalize this objective as a classification task.

3.1 Problem Formulation

Given our focus is on the searching to contacting stage of the funnel, we utilize the action of contacting a listing as our target outcome variable. Therefore, for a given user, u_i, and listing, l_k, we model the probability a user's view of the listing, v_{u_i,l_k}, will result in a listing contact, i.e.

$$Pr\left(u_i \text{ contacts } l_k \mid v_{u_i,l_k}\right).$$

The above classification formulation is straightforward, but there is the added complexity that some users contact a listing and some do not. We include all users in our training data, contacters and non-contacters, so as to avoid the selection-bias of only including users who contact. We emphasize learning the delineation between a contacter and non-contacter while still focussing on user-listing level predictions. This is achieved by applying some filtering in terms of positive and negative examples in our training data:

1. **Positives** - listing views of users which lead to contact of the viewed listing,
2. **Negatives** - listing views which did not lead to a contact, and the searcher did not contact any other listings for the given trip.

Note that it is possible for a user to contact multiple listings for a trip, and all such examples are included in the data as positives.

It can be seen that a contacter's only contribution are the listings he or she contacted, and listings that were viewed but not contacted for the given trip are omitted. While this can be viewed as over simplifying the problem, it helps reinforce the objective of separating the actions of guests which have reached the stage to contact, versus those who will not contact. Previous analyses at Airbnb have shown that most users who contact a listing will view and strongly consider a set of listings. But since he or she can only stay in one place for a given location and dates, most users choose to contact only one of the listings they were considering. Therefore, including negatives from users who contact crosses into the territory of learning preferences, and less so a threshold of intent. Their omission improves our signal to noise ratio for modeling general intent to contact.

Figure 2 illustrates which user listing views are included in the training data (opaque), and their associated label, versus those which are excluded (transparent). The "thumbs up" denotes that a user was implicitly interested in a listing. Note that the first and second user "liked" multiple listings, but only the contacted listings' views are included in the training data, while all the listing views from the non-contacting user are included.

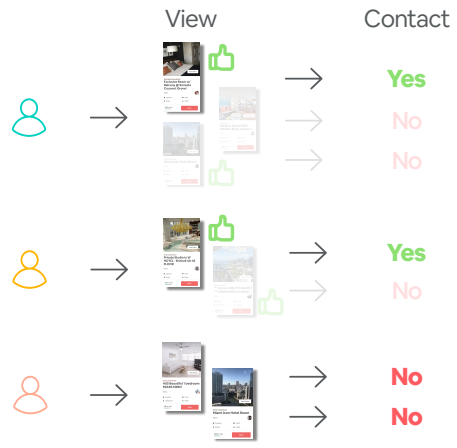

Figure 2: Illustration of Training Data Construction

Many guests view a listing multiple times before booking, this is a valuable signal of intent. We, therefore, update the probability of contact with each view, incorporating information about the current view as well as past views, up to a reasonable look-back time threshold. This updates our objective as follows:

$$Pr\left(u_i \text{ contacts } l_k \mid v_{u_i,l_k,t_1},\ldots,v_{u_i,l_k,t_m}\right),$$

where m denotes the rank number of the current view of listing l_k by user u_i at time t_j.

3.2 Feature Engineering

We implement granular logging of user micro-actions on the listing description page, which enables us to build a rich feature set. Motivated by Zhou et al. [13] we log dwell time on the entire listing page as well as page scroll depth. We also log number of clicks, dwell time, expand/collapse, and pagination for each individual section of the listing page, e.g. photo carousel, description text, reviews, etc. From this logging we generate four classes of features:

1. **Actions on Most Recent View** - This includes features of user actions for the most recent view of the listing.
2. **Actions from Previous Views** - We predict contact with respect to the current view as well as all previous views of the listing by the user. We therefore include a class of features which are aggregated summary statistics of the user actions from previous views of the listing. We use the same numeric features from the "Recent View" class, calculating the min, max, and mean value across all previous views. We also found it improved model performance to include as their own individual class the feature values for the very first view of the listing, in addition to the aggregates.
3. **Time Gap Info of Previous Views** - Information about the timing pattern of a user's multiple views of a listing also impacts model performance. We include features with basic time information: number of previous views of the listing, time since last view of the listing, and min, max, and average time between views of the listing. For cases when a user has

no previous views of the listing, these values are all set to null.

4. **Trip Information** - User behavior which signals intent can be different depending on how far out the trip is (lead time), the length of the trip, and/or how many guests are attending. We include features with this information, as well as how many unique checkin and checkout dates a user enters on the listing's availability module. Users with a strong affinity for a listing will sometimes modify their travel dates to fit the availability constraints of the listing.

3.3 Model Training

We train on user logs from a fixed time period and evaluate on an independent test set from a future time period; care is taken to ensure no user is represented in both sets. We perform some basic filtering by removing examples from users with a single view of a listing with duration less than 5 seconds; this threshold is chosen because it removes very low-intent negative examples without affecting the coverage of positive examples. In order to avoid overfitting to users with many views, we weight training examples by the reciprocal of the total number of examples per user. All continuous and count features are log-transformed. We train our model using the XGboost[3] library with a log-loss objective function. Tuning parameters are selected by evaluating performance on a separate validation set.

3.4 Evaluation of Model Performance

We evaluate the AUC performance of our XGboost classification model compared to a variety of alternative models:

1. Total View Duration - We score the intent level of listing views by the sum total view duration of the current view and previous views of the listing; this is equivalent to the traditional dwell-time weighting of clicks.
2. Number of Repeat Views - We score views by total number of previous views of the listing by the user
3. Decision Tree Depth 2, 3, and 4 - Using the same features as in the full XGboost model, we train a few simple decision trees of varying depth with Gini impurity as the split criterion [1].

Figure 3 shows the ROC curve and AUC performance of each model on the test set. The XGboost model out performs all other models. This is not surprising given the increased complexity of the model, but the dramatic improvement in performance lends a strong argument to continue with the more complex model in lieu of something simpler.

3.5 Evaluation of Model Robustness

The listing description page is a product surface which is still under development. We test product changes attempting to improve booking conversion which can re-order sections, collapse/expand sections, alter the content within a section, or delete a section altogether. Our model includes features related to user actions on the listing description page UI, so we want to be sure the model is for the most part robust to UI changes.

[3]https://xgboost.readthedocs.io/en/latest/

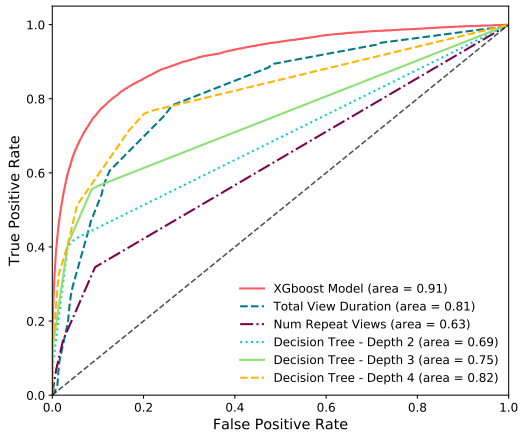

Figure 3: Model Performance Evaluation

Table 1: Model Evaluation on Different Listing Page Design

	Current Listing Page	Previous Listing Page
AUC	0.91	0.89

We compare the AUC performance of our model on the current listing page UI, on which it was trained, versus a previous version. The previous UI version differs fairly drastically from the current version in that there are additional sections and all sections are expanded by default, so the listing description page is longer. Table 1 shows that while there is an expected slight decline in model performance on the different design, the change is not too dramatic with an AUC drop of only 2%.

Since the model performance can be slightly influenced by UI changes, we invest in monitoring metrics related to model performance. Via Airflow[4], we calculate and report daily the model's predictive performance (AUC), as well as descriptive statistics and distributional characteristics of the prediction output of the model and feature inputs. In the future we intend to explore semi-regular automatic retraining of the model via a similar process.

4 APPLICATION TO EXPERIMENTATION

Similar to other online marketplaces, A/B experimentation is a vital component of optimizing the Airbnb search experience. As previously outlined, the long booking conversion funnel can pose challenges in terms of experimentation velocity. Leading indicator metrics help us more quickly determine which product changes are tracking towards improved booking conversion and those which are likely hurting conversion. It also enables us to understand how experiments are affecting different subsets of users, especially those with less traffic. In this section we demonstrate how we use the output from the intentful view model to build a strong leading indicator metric for search experiments, "intentful viewer".

An effective leading indicator metric is consistently correlated with downstream business metrics and ideally more sensitive, i.e.

[4]http://airbnb.io/projects/airflow

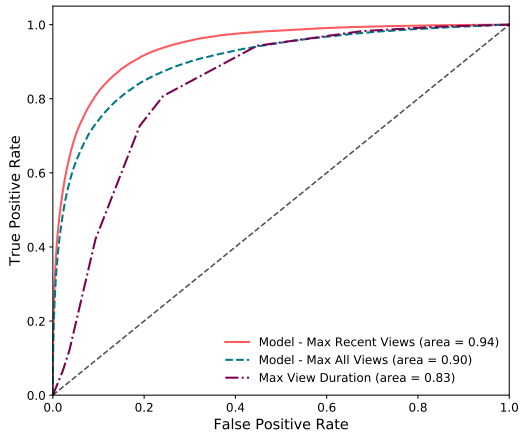

Figure 4: User-Level Prediction of Contacter Evaluation

reaches statistical significance before downstream metrics [3]. Depending on the product change there are cases in which the correlation will not hold, but the goal is for those cases to be rare and easily explainable. We therefore evaluate the quality of our proposed leading indicator metric based on those two factors.

4.1 User Level Prediction

Doing a view which signals intent to contact can be considered a milestone for a user in the search to book funnel. We, therefore, convert the output of our model from user-listing level to simply user level. We do not modify the model, but instead aggregate the model predictions across all listings viewed by the user. We make the reasonable assumption that a users overall intent to contact any listing is bounded from below by their maximum intent to contact an individual listing. More formally, for a user, u_i, with views across listings $\mathcal{L}_i = l_1, \ldots, l_r$, then

$$Pr\left(u_i \text{ contacts any listing}\right) >= \max_{l_k \in \mathcal{L}_i} Pr(u_i \text{ contacts } l_k). \quad (1)$$

This formulation is likely a conservative probability of contacting, but enables us to generate a user level metric which is easier to interpret within the setting of an online experiment.

Figure 4 shows the ROC performance of the user-level prediction with target variable of contacter vs non-contacter. For users with multiple views of a listing, we find it improves predictions to only use the probability of contact from the most recent view of each listing in equation 1. We also include the comparison to an alternative model where user intent is ranked by maximum view duration on an individual listing. The intentful view model user-level prediction, based on most recent view, has a 13% better AUC than dwell time.

4.2 Metric Definition

We attempt to establish a probability threshold such that any user with a predicted probability above the threshold is declared "intentful". Therefore, within a given experiment we can compare the share of users in control and treatment who have reached this stage, with the conclusion that a product change which drives more users into an intentful state will lead to an increase in booking conversion.

Based on our previously outlined goals, we ideally want the selected threshold to result in a metric which is both:

1. Directionally correlated with our downstream booking conversion metrics, and
2. more sensitive than downstream metrics.

Quantifying if two metrics are directionally correlated is straightforward. The directional correlation of the relative lift of two metrics, m_1 and m_2, between branches A and B within an experiment, E_k, can be expressed as

$$D\left(m_1, m_2, E_k\right) = I\Big(\text{sgn}\big(\, L(m_1, E_k)\,\big) = \text{sgn}\big(\, L(m_2, E_k)\,\big)\Big), \quad (2)$$

where $L(m_j, E_k)$ denotes the lift of metric m_j, and $I(\cdot)$ and $\text{sgn}(\cdot)$ are the indicator function and sign operator respectively. We drop the references to branches A and B for simplicity.

Similar to Kharitonov et al. [6], we measure the sensitivity of a metric, m_j, on a given experiment using the z-score statistic, defined as

$$Z\left(m_j, E_k\right) = \frac{\bar{m}_{j_B} - \bar{m}_{j_A}}{\sqrt{\text{Var}\left(\bar{m}_{j_B} - \bar{m}_{j_A}\right)}},$$

where $\bar{m}_{j_A}$ and $\bar{m}_{j_B}$ are the samples means of metric m_j for experiment branches A and B respectively, in experiment E_k. A higher absolute value of the z-score implies higher confidence in the difference of the metric means between control A and treatment B.

When comparing sensitivity, we focus on the inequality comparison of sensitivity (e.g. greater-than) as opposed to the raw magnitude. For a given experiment, we therefore compare if the z-score of our derived metric is larger than that of our downstream metric. Based on the direction of the downstream metric z-score we transpose the scores such that an increase is always desired, i.e. branch B > A.

We present our downstream booking conversion metric as m_C, and derived intentful viewer metric as $m_I(\tau)$, which depends on a probability threshold τ. Using our previously defined expressions, for a set of experiments $\mathcal{E} = \{E_1, \ldots, E_N\}$, we wish to optimize two expressions:

$$\frac{1}{N} \sum_{E_k \in \mathcal{E}} D(\, m_I(\tau),\, m_C,\, E_k\,), \quad (3)$$

$$\frac{1}{N} \sum_{E_k \in \mathcal{E}} I(\, Z\left(m_I(\tau), E_k\right) > Z\left(m_C, E_k\right)\,). \quad (4)$$

We evaluate the above expressions over a range of τ values for a set of historical search experiments. Ideally, we would only use ground-truth experiments for which the booking conversion metrics are statistically significant. We, however, only have 20 such experiments with the required logging to score our intentful view model. In order to increase our sample size we include experiments which did not meet the traditional 0.05 significance level, expanding to experiments with a booking conversion metric p-value up to 0.3. This doubles our sample size of experiments, increasing it to 42.

Since we have stronger confidence in the results of some experiments versus others, we weight the observations by $\omega = (1 - \text{p-value})$, using the booking conversion metric p-value for the experiment. The intuition behind this weighting comes from the notion that

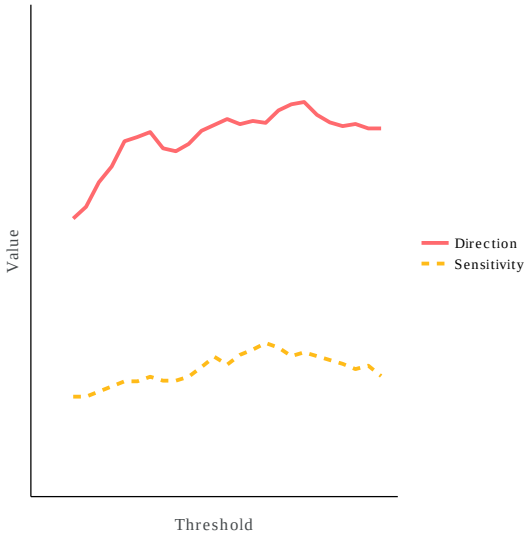

Figure 5: Intentful Viewer Directionality Agreement and Sensitivity for Range of τ Values

the p-value reflects the evidence against the null hypothesis. So the smaller the p-value, the more confidence we have in the experiment results, and therefore the experiment receives more weight. We modify expressions 3 and 4 as follows,

$$\frac{1}{\sum \omega_{m_C,E_k}} \sum_{E_k \in \mathcal{E}} \omega_{m_C,E_k} \, D(\, m_I(\tau),\, m_C,\, E_k \,), \tag{5}$$

$$\frac{1}{\sum \omega_{m_C,E_k}} \sum_{E_k \in \mathcal{E}} \omega_{m_C,E_k} \, I(\, Z(m_I(\tau), E_k) > Z(m_C, E_k) \,). \tag{6}$$

Figure 5 shows the evaluation of expressions 5 and 6 across the 42 experiments for a range of values of τ. There is a general concave shape for both directionality and sensitivity. Given the somewhat flat trend of sensitivity after a certain threshold, we select τ such that it maximizes directionality, we refer to this selected threshold as τ^*.

4.3 Results

For our selected τ^*, we plot in Figure 6 the correlation of the percent change in the intentful viewer metric versus the downstream booking metric across 20 statistically significant experiments. It can be seen our new metric achieves a strong correlation and directional agreement with the conversion metric, especially when compared to the number of listing views metric from Figure 1. In Table 2 we show a comparison of summary statistics of the two metrics. Our intentful viewer metric is able to achieve a directional agreement for 95% of the experiments, whereas the listing view count metric is only 60%. We also see that our new metric is more sensitive than the conversion metric 35% of the time. This shows there are still improvements to be made in terms of increasing the sensitivity, perhaps incorporating the magnitude of the difference of the z-score in the optimizing expression can help. We leave that for future work at this moment.

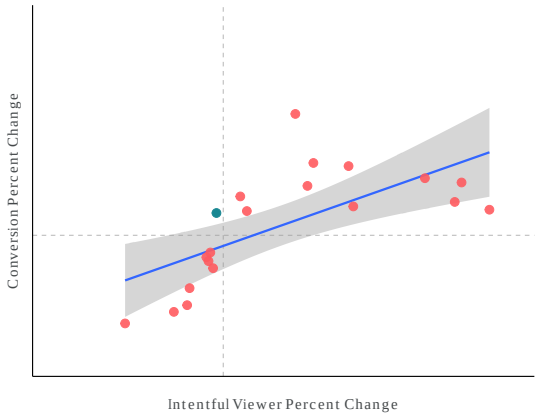

Figure 6: Correlation of Intentful Viewer Metric with Booking Conversion Metric

Table 2: Intentful Viewer vs Num Listing Views Metric

	Intentful Viewer	Num Listing Views
Direction Agreement	0.95	0.60
Kendall's Tau Cor	0.62	0.06
Greater Sensitivity	0.35	0.20

5 CONCLUSIONS AND FUTURE WORK

We present an effective approach, using a machine learning classification model, to capture user intent via user engagement signals. The predictive performance and robustness of the model are evaluated extensively. We use the output of the model to build a user intent metric, "intentful viewer". We demonstrate via a meta-analysis of historical experiments that this metric is strongly correlated with business output metrics.

In the future, as more historical experiments are available, we will explore alternative methods for determining the optimal probability threshold for our "intentful viewer" metric. We also currently predict at the listing-level and then aggregate model predictions up to the user-level. It is possible to jointly model, via a conditional probability representation, a user's overall intent and how that intent disperses to individual listings. We can also iterate on improving model predictive performance by transitioning to neural network sequence models. This would provide a more advanced method for incorporating a user's historical actions on a listing, as the user history can be fed as a raw sequence to the model as opposed to a set of summary statistics as we currently employ. Finally, we will explore the use of the intentful view predictions as a more reliable implicit feedback signal in our search ranking models.

ACKNOWLEDGMENTS

We would like to thank Claire Lebarz, Thomas Legrand, Navin Sivanandam, Richard Dear, and the entire Airbnb Search Team for their feedback and suggestions throughout the development of this work.

REFERENCES

[1] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. 1984. Classification and regression trees. (1984).

[2] Olivier Chapelle, Thorsten Joachims, Filip Radlinski, and Yisong Yue. 2012. Large-Scale Validation and Analysis of Interleaved Search Evaluation. *ACM Transactions on Information Systems* 30, 1 (Feb. 2012).

[3] Alex Deng and Xiaolin Shi. 2016. Data-Driven Metric Development for Online Controlled Experiments: Seven Lessons Learned. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. ACM, New York, NY, USA, 77–86. https://doi.org/10.1145/2939672.2939700

[4] Alexey Drutsa, Gleb Gusev, and Pavel Serdyukov. 2015. Future User Engagement Prediction and Its Application to Improve the Sensitivity of Online Experiments. In *Proceedings of the 24th International Conference on World Wide Web (WWW '15)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 256–266. https://doi.org/10.1145/2736277.2741116

[5] Diane Kelly and Jaime Teevan. 2003. Implicit Feedback for Inferring User Preference: A Bibliography. *SIGIR Forum* 37, 2 (Sept. 2003), 18–28. https://doi.org/10.1145/959258.959260

[6] Eugene Kharitonov, Alexey Drutsa, and Pavel Serdyukov. 2017. Learning Sensitive Combinations of A/B Test Metrics. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (WSDM '17)*. 651–659.

[7] Youngho Kim, Ahmed Hassan, Ryen W White, and Imed Zitouni. 2014. Modeling dwell time to predict click-level satisfaction. In *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, 193–202.

[8] Ron Kohavi, Alex Deng, Brian Frasca, Roger Longbotham, Toby Walker, and Ya Xu. 2012. Trustworthy Online Controlled Experiments: Five Puzzling Outcomes Explained. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '12)*. ACM, New York, NY, USA, 786–794. https://doi.org/10.1145/2339530.2339653

[9] Chang Liu, Jingjing Liu, Nicholas Belkin, Michael Cole, and Jacek Gwizdka. 2011. Using dwell time as an implicit measure of usefulness in different task types. *Proceedings of the American Society for Information Science and Technology* 48, 1 (2011), 1–4.

[10] Ryen W White and Diane Kelly. 2006. A study on the effects of personalization and task information on implicit feedback performance. In *Proceedings of the 15th ACM international conference on Information and knowledge management*. ACM, 297–306.

[11] Songhua Xu, Hao Jiang, and Francis Chi-Moon Lau. 2011. Mining user dwell time for personalized web search re-ranking. In *International Joint Conference on Artificial Intelligence (IJCAI 2011)*. AAAI Press/International Joint Conferences on Artificial Intelligence.

[12] Xing Yi, Liangjie Hong, Erheng Zhong, Nanthan Nan Liu, and Suju Rajan. 2014. Beyond clicks: dwell time for personalization. In *Proceedings of the 8th ACM Conference on Recommender systems*. ACM, 113–120.

[13] Meizi Zhou, Zhuoye Ding, Jiliang Tang, and Dawei Yin. 2018. Micro Behaviors: A New Perspective in E-commerce Recommender Systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18)*. ACM, New York, NY, USA, 727–735. https://doi.org/10.1145/3159652.3159671