TO: Marc Lefar, CEO & President, and Will Lin, VP of Consumer Marketing

FROM: Michelle Chen, Hans Han, John Hu, Trishal Patel, and Max Santa Ana

DATE: November 10, 2017

SUBJECT: A Chicago-based recommendation to increase ApartmentGuide revisitors

To assist Rentpath in generating leads from first-time visitors to ApartmentGuide.com, our team examined website clickthrough data limited to Chicago, IL. We chose to focus on first-time visitors who revisit because they are more likely to leave a lead than churners (*Exhibit 1*). Under our assumptions, using predictive analytics to better target revisitors and reduce ad spending on churners will maximize profits for RentPath.

With this business objective in mind, we moved to data preprocessing to clean our data and removed irrelevant attributes with little influence on the outcome of the class label, U_Revisited. During the modeling phase, we used an iterative process to improve our model performance. We first leveraged singular classifiers and optimized their respective model parameters to achieve a higher recall product. Optimizing these models enabled us to run ensemble methods more efficiently, thus reducing the amount of parameters we needed to change. At the conclusion of our extensive modeling process, we discovered that bagged-random forest yielded the highest net benefit per year for Chicago of $98,752. Two key areas of opportunity were discovered as a result of our analysis. First, we recommend RentPath employ this analytical model in the Chicago region to refine the prediction of likely website revisitors, and allocate marketing resources appropriately to maximize lead conversion of first-time website visitors. Second, we discovered that U_Screen_D had a sizeable influence on predicting the class label. Further analysis revealed that smartphones are the most common viewing screen for customers, and the most common screen for submitted leads. Our recommendation is to drive app installs on smartphones through our targeted ad strategy.

Lead generation from first-time visitors maximizes the net benefit of our recommendation

The business objective of our analytics initiative is to increase lead generation for RentPath's clients. These clients (rental property managers) purchase subscription packages from RentPath to advertise properties on the company's affiliated sites (ie. ApartmentGuide (AG)). Specifically focusing on revisitors will increase lead generation potential for clients by driving traffic to the site (*Exhibit 1*). The success of our project is measured by the net benefit from implementing our analytics solution. Expanding RentPath's understanding of customer search behavior through secondary analysis is also an auxiliary success criterion.

In order to calculate the net benefit, we created a model using the"FirstVisits" website click-through data from AG to predict the likelihood of revisit. "FirstVisits" contains 100,000 rows and 230 attributes of the type nominal and integers, of which "U_Revisited" and "U_ID" are the

class label and ID, respectively. Evaluating the model performance produces a confusion matrix detailing the correctly and incorrectly predicted revisit and churn class labels. We can combine the confusion and cost matrix (*Exhibit 2*) to produce an accurate estimate of the net benefit of the analytics solution. Within the cost matrix, correctly predicting 'revisit' (True-Positive) yields 7.5 cents, or a 15% increase over the value of a potential revisitor ($0.50). This will result in a potential benefit of $3.3 million dollars a year (assuming 1.9 million unique visitors in Chicago, 25% potential revisitors). However, predicting that a churned customer is a potential revisitor (false positive), results in a loss of almost 2.5 cents, the value of the ad. Correctly predicting a churned customer (true negative) results in zero cost. Incorrectly predicting a potential revisitor to be a churned customer results in the potential revisitor to be worth the same, $0.50. (*Exhibit 2*).

Data cleaning allows for improved modeling performance and efficiency gains

We used RapidMiner to clean the data set prior to modeling. After restricting the dataset to only Chicago searches, we imputed missing values based on averages. We then proceeded to use the 'remove useless/correlated attributes' operators to reduce redundancy and multicollinearity among correlated independent attributes. We also removed potential duplicates from the dataset. Fitting models to this data set exhausted computational power and was too time-consuming. As a result, we decided to further refine our data set and remove additional attributes. We performed feature weighting by information gain to remove features which had 0 weighting with respect to U_Revisited (*Exhibit 3*) This process is further explained in the modeling section. Our final, cleaned Chicago data set contains 12,741 rows and 143 attributes (*Exhibit 4*).

Iterative process reveals the most important parameters for a robust model

We began our modeling process by performing preliminary runs for each of the classifiers to determine the models with the highest recall product to use in the ensemble methods. We first ran our models using a 80/20 split between training and testing data. During this initial modeling phase, we tracked the accuracy and recall product across the individual modeling techniques. We also utilized the optimization operator within RapidMiner to maximize the recall product and accuracy for individual models. For instance, use of the operator enabled us to discover that designating a dot kernel for the logistic regression model yielded a higher recall product compared to the radial and polynomial kernels. This is because the dot kernel picks up on the few, major patterns that are reflected within the data set. We optimized the following models using: decision tree, logistic regression, rule induction, naive bayes, and nearest-neighbor. Models such as logistic regression required an additional operator to convert the nominal attributes within our data set into numerical values. Using the same process for the support vector machine model, however, yielded 0% revisit recall which led to a very poor recall product and low accuracy, despite our attempts to sample through various means.

We decided to use stratified sampling on the training set prior to modeling. While trialing various sampling methods (absolute, relative, and probability), we kept the models and their respective parameters consistent. Through this method, we discovered that balanced sampling was the best practice to achieve the highest recall product. In the end, we used an absolute sampling method and designated 2602 instances for each of the class labels (revisit, churn). The reason we chose 2602 specifically is because the training set contains 2602 cases of revisits from the total Chicago first-timer visitors. The low frequency of revisits compared to churners means that models will be more likely to classify correctly or incorrectly classify instances as 'churn' rather than 'revisit.' Balanced sampling addresses this sizeable imbalance within the data set and enables our model to achieve a higher revisit recall, which is more in aligned with RentPath's business objective of increasing leading generation.

Next, we used the insights from individual models and sampling techniques leveraged to enhance the ensemble modeling process. The ensemble methods we experimented with include: boosting, bagging, stacking, and random forest. The random forest ensemble method achieved the best performance in terms of recall product. We found that minimal gain and the number of trees had the most influence on the model. Similarly, minimal gain impacted our decision tree by the greatest margin; however, in random forest models, increasing the number of trees dramatically increased run-times of our models, thus reducing their viability in practical use. After many iterations, we modified our training and testing datasets by removing excess attributes through feature weighting. As a result, the number of attributes decreased from 220 to 143. Our thought process was that decreasing the number of attributes in a conservative manner would reduce the computational power required to run models and improve our recall product. This helped greatly and we eventually settled on a bagged random forest model with 38.55% recall product.

Cross validating a bagged random forest model generates a net benefit of $98,752

During the evaluation phase, we first picked categories of models that achieved high recall product and accuracy measures; the categories we ended up with were random forest, bagged random forest, bagged decision tree, and boosted decision tree ensemble methods. Within these four categories, we chose to evaluate six of the top performing trials based on recall product. Of these six trials, the top three trials sorted from highest to lowest net benefit were a bagged decision tree, random forest, and bagged random forest trial (*Exhibit 5*).

Originally during the modeling phase, we split the data (split validation) into training and testing data sets so that the data sets were consistent across team members. The downside to split validation is that it often overestimates the test accuracy and recall product. To account for this overestimate, we performed cross validation during the evaluation phase on the aforementioned top three trials with the highest net benefit to give us a more accurate measure of the true performance and recall product for the models. During cross validation, we also adjusted the minimum and cross validation iteration parameters to maximize the product recall. These model adjustments further enabled us to achieve ideal revisit recall that exceeded the churn recall, which satisfies the business objective of our analytics initiative, which is to excel at determining

which first-time visitors are likely to revisit AG (as opposed to churners) (*Exhibit 1*). After cross-validation, cost-benefit analysis indicates that the bagged random forest model produces the greatest net benefit to RentPath of $98,752 (*Exhibit 5*). In this case, the cross-validated recall product for bagged random forest did not exceed that of the cross-validated bagged decision tree. This means that implementing a model with the highest recall product does imply the best course of action for profit maximization.

Three Best Models (Cross Validated)					
Random Forest		**Bagging (random forest)**		**Bagging (decision tree)**	
Recall Product	Yearly Net Benefit	Recall Product	Yearly Net Benefit	Recall Product	Yearly Net Benefit
35.22%	$81,311	37.35%	$98,752	37.37%	$98,197

To calculate the net benefit of the models, we with an arbitrary population of 10,000 users. We assumed that 25% of the sample would be revisitors and 75% are churners, similar to the ratio of the Chicago sample. We applied the recall percentages to the revisitors and churners, calculating true positives, false positives, true negatives, and false negatives to create a new confusion matrix. Then, we applied the confusion matrix to cost matrix to obtain the benefit. We then netted out the $1,250 "do nothing" or no model benefit to find the net benefit. Finally we extrapolated this benefit to the entire Chicago population to find the total monthly and then yearly benefit (*Exhibit 5*).

Targeted Facebook ads for smartphone users capitalizes on attractive opportunities

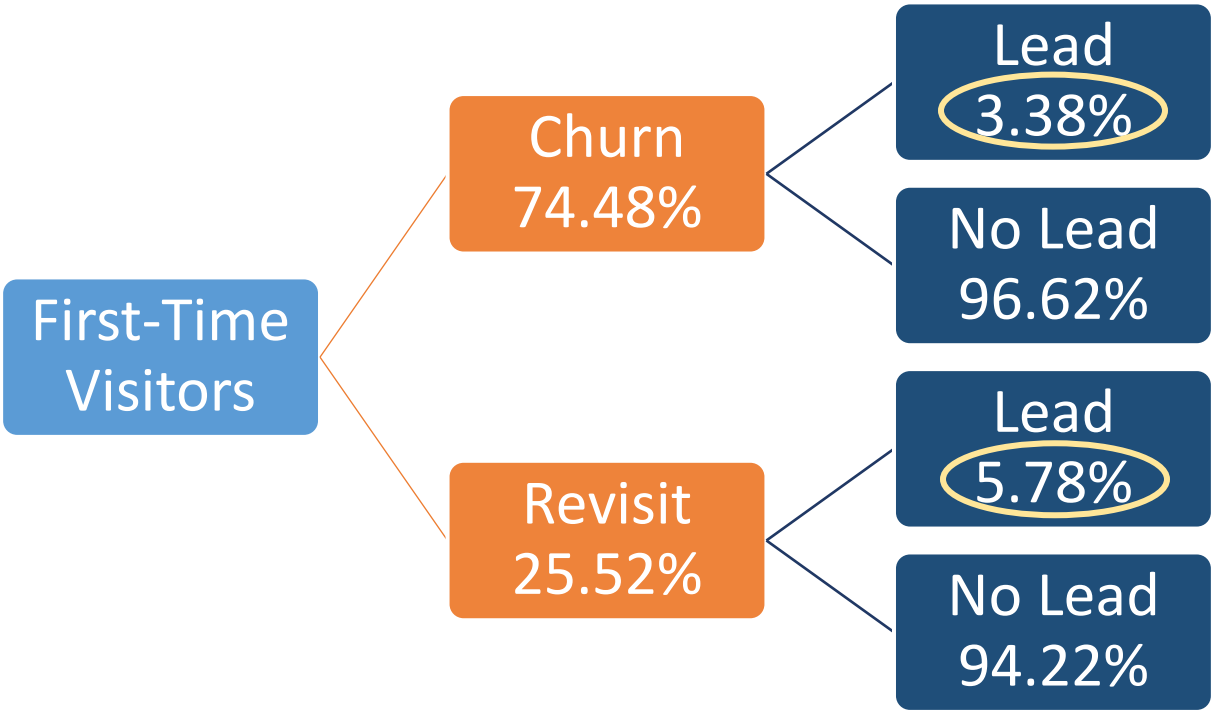
Our first recommendation is to focus on the potential revisitors to RentPath's websites. We recommend RentPath to apply the bagged, random forest model to Chicago users to predict potential revisitors. After obtaining these predictions, we recommend RentPath to track predict revisitors to company's website using cookies. Next, we suggest RentPath enter into a contract with Facebook to target these predicted potential revisitors using Facebook ads. We predict that 50% of RentPath potential revisitors will come back organically. Showing ads will improve the chance of these potential revisitors coming to our website by 20%. After considering the cost of the ad, the value of each correctly predicted potential revisitor increases by 15%, from $0.50 to $0.575. After considering the predictive ability of our model, it results in the monthly benefit of

of the model is $8,229. Applied to the population of 1.9 million Chicago users, the yearly benefit of the model is $98,752 (*Exhibit 5*).

The second recommendation is derived from analyzing the feature weighting for attributes. The five most impactful attributes on the class label are as follows: U_Screen_D, U_OS_D, U_ProfileKey_D, U_BrowserV_D, and U_OSV. We found U_Screen_D to be the most influential attribute using both information gain and chi-square weightings. We chose to further examine the U_Screen attribute due to its 1.0 feature weighting and determined that smartphones were used the most when customers visited the website for the first time, and also had the greatest proportion of lead submissions (Exhibit 3). This observation aligns with current trends in increasing smartphone usage throughout the day. Next, we qualitatively compared the two methods of accessing Apartmentguide.com through a smartphone: mobile browser and mobile app. Out of these two options, the mobile app offers a better user experience through a more intuitive interface. For example, the ApartmentGuide App uses an intuitive mapping system to help users search for locations. In contrast, the website search is not as user friendly with its traditional zip-code filter system. Given the app's intuitive user interface and the higher chance of revisit from using the app, we recommend that RentPath leverage Facebook ads to increase awareness and drive app downloads and usage. The result is an increase in customer loyalty and frequency of revisits to platforms beyond the app, which will ultimately generate more leads and revenue for RentPath and its clients.

Exhibits

Exhibit #1: First-time revisitors are more likely to be associated with a lead than churners



In our analysis, we found that a first-time visitor to the Apartmentguide.com website had two options: revisit at a future date (25.52% of the instances), or churn (74.48%). We defined churn as first-time website visitors that no longer use the service after their initial visit. For people who churned, we found that 3.38% submitted a lead (contacted the Rentpath client via email or phone) and 96.62% left without submitting a lead. The people who revisited the site at a future date submitted leads 5.78% of the time, while 94.22% of the revisits led to no lead submissions.

Exhibit #2: The use of targeted ads increases a potential revisitor's value from $0.50 to $0.575

Cost Matrix	**Actual**		
Predicted		Actual Done	Actual Looking
	Predicted Done	$0.00	$0.50 [1] per revisitor
	Predicted Looking	$.025 [3] per ad	$0.575 [2] per revisitor

[1] Value of a visit is $1 (RentPath Data), 50% chance of a visitor going back to RentPath organically

[2]A targeted ad can move the chance of a revisitor going back to RentPath's website from 50% to 60%, subtracted cost of ad of $0.025, a conservative number ($1*60% - $.025 = $0.575)

[3]A retargeted customer will be shown 10 ads, cost per 1000 impressions is $2.50 (average, https://monetizepros.com/display-advertising/average-cpm-rates/), **2.50 / 1000 * 10 = $.025**

Exhibit #3: Feature-weighting by information gain reveals top 10 attributes with greatest influence

Attributes	Feature Weighting (Information Gain)
U_Screen_D	1
U_OS_D	0.997051889
U_ProfileKey_D	0.903970475
U_BrowserV_D	0.645697218
U_OSV_D	0.613490198
U_Browser_D	0.550061186
U_Referrer_D	0.473135338
U_Exit_D	0.286883946
U_Landing_D	0.251218995
U_PageCt_pdp	0.180774694

Exhibit #4: Descriptive statistics show importance of data preprocessing before modeling phase

	Original Data Set	**Chicago Subsetted Data**
Description	First time visitors click-through data	Original data set filtered by Chicago, where U_MSA_D= "Chicago"
[#Row, # Columns]	[100,000 , 230]	[12,748 , 143]
Types of Attributes	• ID: U_ID • Class label: U_Revisited • Int: U_Session T • Nominal: U_ProfileKey_D • Date_time: U_Datetime_start	• Class label: U_Revisited • Int: U_Session T • Nominal: U_ProfileKey_D • Date_time: U_Datetime_start

Summary Statistics	• Average session time = 337 seconds
	• Breakdown of devices used to access the site
	  ○ Smartphone = 50.72%
	  ○ Desktop = 42.56%
	  ○ Tablet = 6.72%
	• Percentage of revisits = 25.43%
	• Average number of leads submitted = 0.1315 leads
	• Portion of users who submitted at least one lead = 2.52%
	• Average click count for clicking photos = 16.69
		• Average session time = 318.4 seconds
		• Breakdown of devices used to access the site
		  ○ Smartphone = 52.09%
		  ○ Desktop = 40.94%
		  ○ Tablet = 6.97%
		• Percentage of revisits = 25.52%
		• Average number of leads submitted = 0.1433 leads
		• Portion of users who submitted at least one lead = 381
		• Average click count for clicking photos = 15.86

Missing Data	• Total: 16,974 Missing Data
	  ○ U_Referrer_D: 12960
	  ○ U_BrowserV_D: 3103
	  ○ U_Landing_D: 487
	  ○ U_Exit_D: 350
	  ○ U_OSV_D: 70
	  ○ U_Browser_D: 4
		• Total: 66 Missing Data
		  ○ U_Referrer_D: 0
		  ○ U_BrowserV_D: 1
		  ○ U_Landing_D: 0
		  ○ U_Exit_D: 0
		  ○ U_OSV_D: 65
		  ○ U_Browser_D: 0

Suspicious Data	• For U_ID=1485579478129090, U_Session_T = -2 • Often times, users with high U_ActionCt_lead_submissions had low or 0 clicks and also for views, given the proportion of lead submissions.	• Often times, U_ActionCt_lead_submissions had low or 0 clicks and also for views, given the proportion of lead submissions. • Upon subsetting the data to only include Chicago searches, there were no abnormal session times as with the original data.

Exhibit #5: Net benefits produced by our best performing models show strong profitability

Cost Matrix

	Actual Norevisit	Actual Revisit
Predicted Norevisit	$0.00	$0.50
Predicted Revisit	($0.025)	$0.575

Split Validation Confusion Matrix

Random Forest

	Actual Done	Actual Looking	Precision:
Predicted Done	4643	933	83.26%
Predicted Looking	2857	1567	64.58%
Recall:	61.91%	62.67%	
		Accuracy:	62.10%
Net Revenue:	$ 1,296.09		
Net Revenue w/"do nothing":	$ 1,250.00		
Net Benefit:	$ 46.09		
Net Benefit for population (mo.):	$ 8,756.62		
Net Benefit for population (yr.):	$ 105,079.50		

Bagging- Random Forest

	Actual Done	Actual Looking	Precision:
Predicted Done	4541	910	83.30%
Predicted Looking	2960	1590	65.05%
Recall:	60.54%	63.59%	
		Accuracy:	61.30%
Net Revenue:	$ 1,295.24		
Net Revenue w/"do nothing":	$ 1,250.00		
Net Benefit:	$ 45.24		
Net Benefit for population (mo.):	$ 8,596.31		
Net Benefit for population (yr.):	$ 103,155.75		

Bagging - Decision Tree

	Actual Done	Actual Looking	Precision:
Predicted Done	5207	1103	82.53%
Predicted Looking	2293	1398	62.13%
Recall:	69.43%	55.90%	
		Accuracy:	66.05%
Net Revenue:	$ 1,297.49		
Net Revenue w/"do nothing":	$ 1,250.00		
Net Benefit:	$ 47.49		
Net Benefit for population (mo.):	$ 9,023.81		
Net Benefit for population (yr.):	$ 108,285.75		

Cross Validation Confusion Matrix

Random Forest

	Actual Done	Actual Looking	Precision:
Predicted Done	4129	901	82.09%
Predicted Looking	3371	1599	67.83%
Recall:	55.05%	63.97%	
		Accuracy:	57.28%
Net Revenue:	$ 1,285.66		
Net Revenue w/"do nothing":	$ 1,250.00		
Net Benefit:	$ 35.66		
Net Benefit for population (mo.):	$ 6,775.87		
Net Benefit for population (yr.):	$81,310.50		

Bagging- Random Forest

	Actual Done	Actual Looking	Precision:
Predicted Done	4070	779	83.93%
Predicted Looking	3430	1721	66.59%
Recall:	54.27%	68.83%	
		Accuracy:	57.91%
Net Revenue:	$ 1,293.31		
Net Revenue w/"do nothing":	$ 1,250.00		
Net Benefit:	$ 43.31		
Net Benefit for population (mo.):	$ 8,229.38		
Net Benefit for population (yr.):	$98,752.50		

Bagging - Decision Tree

	Actual Done	Actual Looking	Precision:
Predicted Done	4118	799	83.76%
Predicted Looking	3382	1702	66.53%
Recall:	54.91%	68.06%	
		Accuracy:	58.20%
Net Revenue:	$ 1,293.07		
Net Revenue w/"do nothing":	$ 1,250.00		
Net Benefit:	$ 43.07		
Net Benefit for population (mo.):	$ 8,183.06		
Net Benefit for population (yr.):	$98,196.75		

*Chicago population (users) calculated using (Chicago samples/ Total Samples) * 14.7 million unique monthly users

*Net Benefit for population (month) calculated using (net benefit / 10,000 in arbitrary sample) * Chicago users

On our honor, we pledge that we have neither given nor received aid on this project.

Michelle Chen, Hans Han, John Hu, Trishal Patel, and Max Santa Ana