2017 16th IEEE International Conference on Machine Learning and Applications

Predicting Hotel Bookings Cancellation With a Machine Learning Classification Model

Nuno Antonio
ISCTE-IUL: Department of
Information Science and Technology
Lisbon, Portugal
nuno_miguel_antonio@iscte.pt

Ana de Almeida
ISCTE-IUL: Department of
Information Science and Technology
CISUC
Lisbon, Portugal
ana.almeida@iscte.pt

Luis Nunes
ISCTE-IUL: Department of
Information Science and Technology
Instituto de Telecomunicações
ISTAR
Lisbon, Portugal
luis.nunes@iscte.pt

***Abstract*—Booking cancellations have significant impact on demand-management decisions in the hospitality industry. To mitigate the effect of cancellations, hotels implement rigid cancellation policies and overbooking tactics, which in turn can have a negative impact on revenue and on the hotel reputation. To reduce this impact, a machine learning based system prototype was developed. It makes use of the hotel's Property Management Systems data and trains a classification model every day to predict which bookings are "likely to cancel" and with that calculate net demand. This prototype, deployed in a production environment in two hotels, by enforcing A/B testing, also enables the measurement of the impact of actions taken to act upon bookings predicted as "likely to cancel". Results indicate good prototype performance and provide important indications for research progress whilst evidencing that bookings contacted by hotels cancel less than bookings not contacted.**

Keywords—Bookings cancellation; hospitality; machine learning; predictive modeling; prototyping; revenue management;

I. INTRODUCTION

In the hospitality industry, booking cancellations have significant impact on demand-management decisions. They limit the production of accurate forecasts, a critical tool in terms of revenue management performance. To mitigate the difficulties caused by cancellations, hotels implement rigid cancellation policies and overbooking strategies [1]–[3], which later can generate a negative impact on revenue and social reputation, as well as damage the hotel business performance. Overbooking forces the hotel to deny service provision, which can be a terrible experience for the customer and have a negative effect on both the hotel's reputation and immediate revenue [4]. It can also mean future revenue loss from discontent customers who will not book again at the same hotel [1]. On the other hand, rigid cancellation policies, especially non-refundable policies, have the potential not only to reduce the number of bookings but also to diminish revenue due to the application of significant discounts on price [2].

Some of the previous published works on booking cancellations prediction approach it as a classification problem while most works consider it as a regression problem [5]. Yet, some of the later focused on global cancellation rate forecast and not on each booking's cancellation probability [6]. In fact, Morales and Wang stated that "it is hard to imagine that one can predict whether a booking will be canceled or not with high accuracy" [6, p. 556]. But, previous research [5], [6], demonstrated that using machine learning, statistics, data mining and data visualization this is possible with results surpassing that expectations.

By identifying which bookings are likely to be cancelled, revenue managers and other members of the hotel's staff can take measures to avoid potential cancellations such as offering services, room upgrades, discounts, entrances to shows/amusement parks, or other perks. However, these offers cannot always be applied due to the pricing insensitiveness of some customers (e.g., corporate guests). Nonetheless, booking classification is not the only possible benefit. By running the models daily against all reservations on-the-books, an important result emerges: the number of room nights predicted to be canceled for each of the following days. With this amount, hotels can deduce its value from their demand by calculating their net demand. Equipped with an accurate demand value, hotel managers can develop more effective overbooking and cancellation policies.

To the extent of the authors knowledge, there are no known scientific documented examples of the application of cancellation prediction models in a production environment. This study presents a cancellation prediction system, based on a machine learning model that uses data from the hotel's Property Management Systems (PMS) to predict hotel bookings with high likelihood of being cancelled. By developing a prototype and testing the system in two hotels, in a real production environment, this study demonstrates not only how a machine learning cancellation prediction model can be an excellent tool for rooms pricing and inventory allocation optimization tasks, but also the potential of machine learning as a tool in hotel revenue management.

II. BACKGROUND

Originally developed in 1966 by the aviation industry [7], revenue management was gradually introduced in other services industries, such as hotels, golf courses, restaurants and casinos [7]. In the hospitality industry (rooms division), revenue management general definition was adapted to "making the right room available for the right guest and the right price at the right time via the right distribution channel" [1, p. 2].

This study was in part supported by a Microsoft Azure Data Science Award grant conceded to the first author.

0-7695-6321-X/17/31.00 ©2017 IEEE
DOI 10.1109/ICMLA.2017.00-11

As in other service industries who have a fixed inventory and have a "perishable product", the hospitality industry accepts bookings in advance. These bookings represent a contract between the customer and the hotel [3]. This contract gives the customer the right to use the service in the future at a settled price, but most of the times, with an option to cancel the contract prior to the service provision. Although advanced bookings are considered the leading predictor of a hotel's forecast performance [2], this option to cancel the service puts the risk on the side of the hotel, leading to the hotel having to ensure that it has rooms to customers who honor their bookings. However, a cancelation or a no-show forces the hotel to endure the cost of having vacant rooms [3] (although there are differences between no-shows and cancellations, for the purpose of this study, both will be treated as cancellations).

A considerable number of studies have been published on the subject of bookings cancellation and demand forecast [8]–[17]. Nevertheless, a substantial part of these studies focuses in the airline industry [8]–[13], which although having some similarities to the hospitality industry, its different. A second consideration is that most of these studies employ "traditional statistics" methodologies. Few of them take advantage of machine learning methodologies and techniques [9], [10], [16]. The same is also valid for research regarding the component of demand forecast to predict cancellations [5], [6], [12], [18]–[22], but only the studies of [5], [6], [18], [20], [22] focus to the hospitality industry, and only [5], [6], [22] use hotel specific data (PMS data).

This study, by using hotel specific data to develop a machine learning based model and deploy this system in a prototype in a real production environment, aims to extend a previous presented approach [5], [6] and present a case study showing that booking cancellation prediction is possible, not only from theoretical standpoint, but also from an empirical one as well.

III. Methodology

The need to evaluate and test in a real production environment a machine learning prediction model of hotel bookings cancellation is undoubtedly a problem that can be addressed in the context of Design Science Research (DSR), as it requires the development of an artifact, in this case, a prototype of a component of a Revenue Management System (RMS), fulfilling the two requirements of DSR: Relevance – by addressing a real business need and Rigor – by the need to apply the proper body of knowledge in the artifact development [23], [24], in this case, tools and skills like data visualization, data mining and machine learning.

To build the model and test the system prototype in a real production environment the collaboration of two hotels from the one Portuguese hotel chain (whom required anonymity) was obtained. The prototype was implemented in one resort hotel and one city hotel, both with the official classification of 4 stars. These hotels provided access to their PMS data and agreed to allocate resources to work with the prototype in order to measure its performance in a production environment. Data for the resort hotel (H1) was available from July 2015 onwards and, for the city hotel (H2), from September 2015 onwards. Bookings cancellations distribution, for these hotels, per year, can be seen in Fig. 1.

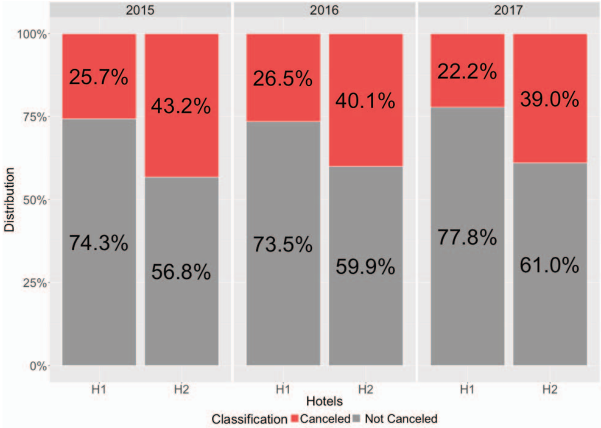

Fig. 1. Cancellations distribution, per hotel, per year.

A. *Machine learning model*

To create the machine learning models this study employed CRISP-DM methodology [25] and used as a start point the models designed and presented in [5]. Albeit the authors have demonstrated that hotel booking cancellation prediction models, built with PMS data, can produce excellent results, with Accuracy and AUC (Area Under the Curve) values above 90%, this study revealed, by testing in a real production environment, that those models have a tendency to over-fit and not to generalize well for future bookings. Further analysis exposed two issues. One related to data leakage and one of what [26, p. xi] calls a "dataset shift" problem, i.e., "where the joint distribution of inputs and outputs differs between training and test stage". In this case, this shift between the distribution happened because [5] employed a stratified dataset splitting strategy for training and testing dataset splitting, but because hospitality business is changing rapidly and therefore also the demand patterns, this splitting does not guarantee a similar distribution among both datasets. In particular, due the boom in the tourism industry over the last years, demand has increased very fast, making prices (ADR – Average Daily Rate) and Lead time (number of days before arrival a booking is made) increase rapidly overtime. Also, the fast rate of appearance of new players (OTA's – Online Travel Agencies) and disappearance of others ("traditional" travel agencies and travel operators), or at least, the change of weight in terms of business they generate to hotels, influences the distribution of certain features like ADR, Lead time, Agency or Company, along the timeline. To address these issues two major changes were introduced to what [5] had previously been defined [5]: changes in the dataset construction and splitting and changes in feature selection and engineering.

1) *Dataset Construction and Splitting*

As acknowledge by [17], data for hotel forecasting has two dimensions: one related to when the booking was made and another related to the period of stay. This means that between the date of booking until the expected arrival date, a booking can be canceled (independently of any cancellation policies hotels might have regarding cancellations). For this reason, in any moment in time, a PMS database has three types of bookings:

- A – Effective: bookings created any moment in time, with an arrival date inferior or equal to current date, that already checked-out or are checked-in;
- B – Canceled: bookings created any moment in time, with an arrival date for any moment in time (past or future), that cancelled or were a no-show;
- C – Future arrivals: bookings created any moment in time, with any arrival date equal or superior to current date, which have not cancel until the current date, but who can cancel until the expected arrival date. This means these bookings are in "unstable" state where they are not cancelled, but some might.

Instead of using all three types of bookings in the dataset construction, as [5], "C" type bookings were removed so only bookings with known final outcome would be used to train the models. This way, observations from future arrival bookings are not used on model training, reducing the risk of leakage and of incorrect training, as some of these bookings will probably be canceled.

One other important change was how training and testing datasets were created. Instead of making a stratified split by the outcome (*IsCanceled*) based on the dimension of the booking creation date, an approach usually employed in time series was applied, convenience splitting [27]. This method involves the splitting of the dataset in discrete blocks. In this case, the dataset is ordered by arrival date of bookings and then, blocks of "month/year" are created. Then, a 75% stratified split of each block is merged into a training dataset and the remaining 25% into a test dataset. This allows the capture of what [28, p. 197] calls "non-stationary temporal data", data that "changes behavior with time and therefore should be reflected in the modeling data and sampling strategies".

2) *Feature Selection and Engineering*

Removal of bookings type "C" from the modeling dataset and the fact that data of each hotel only spanned for an arrivals period approximately of two years, made the training algorithm to perform poorly and to predict the vast majority of future arrivals as "likely to be canceled". To overcome this situation, several transformations were introduced in the Data Preparation phase.

First, *Country* was removed from the modeling dataset. It was found that "Portugal" was assigned as the country in the majority of bookings, at the time of their creation. Only at check-in, when guests presented their personal identification to the hotel staff, was the country correctly filled. This meant that the majority of canceled bookings, had the *Country* "Portugal", which clearly was a case of leakage.

Second, due to the fact that for model training, for future dates, only type "B" bookings existed, all features associated with time were removed from the modeling dataset, so that this should not be captured by the classification algorithm. This included the features *ArrivalDateDayOfMonth*, *ArrivalDateMonth*, *ArrivalDateWeekNumber* and *ArrivalDateYear*.

Third, features that did not bring any value or just introduced noise into model elaboration were also removed. This included *AssignedRoomType*, *RequiredCarParkingSpaces* and *ReservedRoomType*.

Fourth, features *LeadTime* and *ADR* were replaced by engineered features. *LeadTime* was replaced by *LiveTime* and *ADR* was replaced with *ADRThirdQuartileDeviation*. Because *LeadTime* is a static value representing the number of days between the date of the booking creation and the expected booking arrival date, it did not capture the time before arrivals bookings were canceled, thus not allowing the model to understand when were bookings canceled. Since, the introduction of such feature on canceled bookings would introduce leakage, this feature was transformed to have value that should variate according to the type of booking: for "A" bookings had the *LeadTime* value; for "B" bookings had the elapsed number of days between the date of booking creation and the cancellation date; for "C" bookings had the elapsed number of days between the data of booking creation and the processing date. Because prices have been increasing, prices vary by different factors (time of the year, room type, distribution channel, among others), and because time based features were removed from the modeling dataset, ADR distribution and amplitude, did not allow the classification algorithm to capture any relation to cancellations. Yet, business-wise, is known that bookings with an expensive price (compared to similar bookings for the same period of time) tend to cancel more. For this reason, several tests were made to create a feature that would incorporate price, in a manner that reflected this notion and that could be in a scale common to any time of the year, independently of the amplitude of prices. This resulted in the feature *ADRThirdQuartileDeviation*, which is a ratio of each booking ADR, by the third quartile value, of all bookings of the same distribution channel, same reserved room type, for the same expected week/year of arrival.

Fifth, because some categorical features, like *Agency* and *Company* had a high degree of cardinality, which contributed for the model to take longer to train and to have a tendency to over-fit, these features were re-encoded. As recognized by [29], how predictors are encoded can have a significant impact in the model performance. Sometimes, combinations of predictors are more effective than the individual values of features. Using R "vtreat" package [30], categorical features with a level frequency of at least 0.02 were encoded into an indicator column (one-hot encoding) and replaced by two numerical features, one with their logit-odds in outcome from mean distribution on the observed value of the original amount, and the prevalence fact of the categorical value in the dataset (whether the categorical level is common or rare in the dataset). This made possible to mitigate the effects of high cardinality in features like *Agency* and *Company*, and generalize the effects of features with some levels not commonly used, like *Meal* and *MarketSegment*.

These transformations resulted on a modeling dataset with more features than other studies [5], containing besides the *IsCanceled* label, the features: *ADRThirdQuartileDeviation*, *Adults*, *Babies*, *BookingChanges*, *Children*, *DaysInWaitingList*, *IsRepeatedGuest*, *IsVIP*, *LiveTime*, *PreviousCancellationRatio*, *StaysInWeekendNights*, *StaysInWeekNights*, *TotalOfSpecialRequests*, *WasInWaitingList*, and the encoded features of the categorical features *Agent*, *Company*, *CustomerType*, *DepositType*, *DistributionChannel*, *MarketSegment* and *Meal*.

B. Protoype Requirements and Specifications

The prototype designed in this study was not meant to be the prototype of a full RMS, but rather the prototype of a component of a RMS, for a specific task: identify bookings that might cancel and enable users to visualize net demand. For this reason, the prototype included a web visualization component where hotel users and researchers could login to visualize demand and net demand reports, model performance metrics, and check which bookings were identified as "likely to cancel". However, although users had access to the total number of bookings identified as "likely to cancel", details were only shown for 50% of those bookings (acted as the verification group). The remaining 50% acted as the control group, thus enabling A/B testing.

C. System Architechture

Based on the prototype requirements and specifications, technical requirements, and the need to evaluate how the system could be integrated into a RMS an architecture built on top of the Microsoft Azure cloud platform was designed. This architecture makes uses of several Microsoft Azure services:

- One HDInsight Hadoop and Spark cluster with R Server. This enables Hadoop/Spark-based big data processing. R is used in the Spark context to implement the model with XGBoost [31], a powerful tree boosting machine learning method, taking advantage of the cluster capabilities to distribute processing among the different machines, to daily build a new classification model and execute predictions in a fast manner.
- One SQL database. Where processing and performance logs are stored. It also stores all prediction results, together with actions made by the users.
- One web server. This server publishes the visualization layer in the form of a dynamic website, built using C# asp.net, where users can view demand and predictions, as well as give feedback on actions made upon bookings identified as "likely to cancel".

A fully automated ETL (Extract, Transform and Load) process was created in both hotels PMS databases, to daily, extract all bookings from the hotels PMS', transform data into a CSV dataset file, and then loaded into the Hadoop cluster.

D. Prototype Development and Deployment

The prototype was fully written in R. It runs on the R edge node of the cluster and every day, after receiving the datasets from the hotels, it automatically trains a model for each of the hotels. This training includes the incorporation of new bookings and changes to existing bookings and features encoding. It also incorporates a weighing mechanism to give more importance to recent bookings (in terms of creation date) and a cost-sensitive learning by example weighting [32]. In this case, this method involves incorporating previous predictions hits and errors, by assigning a penalization weight to every false negatives and false positives observations, and assigning a bonuses weight to true positive predictions.

As [28, p. 498] recognizes "Even the most accurate and effective models don't stay active indefinitely". To overcome this challenge, the system incorporates what [28, p. 508] calls the "Champion-challenger" approach. Rather than waiting for a decrease in model performance to build a new model, a challenger model is built every day, as new data is available. Using this new data, the system executes an automatic hyper parameter tuning using 10-fold cross validation. The system then trains new a model with the new hyper parameters and compares its Accuracy and AUC results with a model trained with the hyper parameters used on the day before (old model). If Accuracy and AUC from the new test dataset is better than the last 7 days' average and is better in at least 4 of the previous 7 days. The challenger model (new), will be used. Otherwise, the model developed with the hyper parameters used on the previous day will be used.

Prototype was deployed in April 2017. Following a set of tests and adaptations, hotels started to use the model in the 1st of May 2017. The prototype is schedule to run until the end of September 2017. By that time, it is expected that results can give a good insight on the system performance and its impact on business.

IV. Results and discussion

Common machine learning performance metrics like Accuracy and AUC, as exemplified in Table I, for the model daily run of the 1st of July 2017, although being slightly inferior to those of [5], present good results. However, models are now less prone to overfitting and do not show problems of over-classification of future arrivals as "likely to cancel". For example, for this date, the percentage of future arrivals on-the-books, identified as "likely to cancel", was 11.5% for H1, and 28.5% for H2, which is in-line with those hotels cancellations rates (see Fig. 1).

Although hotels had access to the prototype since the beginning of May 2017, it was only in the beginning of June 2017 that hotels started paying more attention to prototype predictions and increased the number of actions on bookings identified as likely to be canceled. For this reason, in terms of predictions analysis, only arrivals between the 1st of June 2017 until the 16th of July 2017 will be presented. Because the system is designed to learn continuously, either by the daily incorporation of new bookings, changes of existing bookings, or by the incorporation of previous predictions results, the classification of a booking is not definitive. In any given day, the classification can change from "likely to cancel" to "likely not to cancel", or vice-versa. For this reason, to analyze results, and consider the booking classification prediction, a specific measure was created. This measure, Minimum Frequency (MF), is a ratio calculated by dividing the number of days the model

TABLE I. Performance Metrics For the 1st July 2017

Hotel.	*Dataset*	*Acc.*	*AUC*	*Prec.*	*Sensit.*	*Specif.*
H1	Train	0.846	0.910	0.839	0.626	0.950
	Test	0.842	0.877	0.811	0.603	0.941
H2	Train	0.857	0.934	0.876	0.793	0.909
	Test	0.849	0.922	0.869	0.779	0.905

classified the booking as “likely to cancel”, by the total number the booking was processed by the model.

Performance metrics for the period aforementioned, as demonstrated in Table II, show that models classification predictions for future arrivals, in this case, with a MF of 50%, present inferior results when compared to the modeling results (Table I). This can indicate that models, although better, still do not generalize well, but at the same time, these results may be being negatively influenced by the actions taken by users to avoid cancellations. Nonetheless, results by MF, per group, per hotel, as demonstrated in Tables III and IV, show that the effective cancellations ratio (%Canc. Total) increases as MF increases, which means as a prediction is more frequent, models are more precise. Yet, A/B testing does not show any statistically significant difference between groups for any of the MF thresholds. This performance could be explained by the low number of bookings were actions were taken to avoid cancellations (34 for H1 and 40 for H2, as shown in Table V).

Originally, the two hotels who agreed to participate, committed to employ resources to contact bookings identified as “likely to cancel”, like offering discounts on services, such as SPA, meals, room upgrades or even, complementary services. However, due the human resources costs and the services associated costs, hotels decided to restrict contacts to email and instead of making complementary offers, or reasonable discounts, decided to stick the contact to try to give a better service or just do some upselling (but very discretely, with little discounts). Hotels analyzed bookings classified as “likely to cancel” and according to the information on the booking, asked for missing pieces of information that could provide a better service, like the expected time of arrival, children age or type of bed, or just validated the credit card if one was available. The vast majority of guests answered back on the same day or the following day providing the information required and thanking the hotels for the interest, but others took the opportunity to cancel. This is not a bad result, as it allows hotels to put the room to sale again.

TABLE II. PERFORMANCE METRICS FOR ARRIVALS (MF 50%)

Hotel	*N*	*Accuracy*	*Precision*	*Sensitivity*	*Specificity*
H1	1848	0.736	0.248	0.132	0.894
H2	4185	0.712	0.319	0.174	0.882

TABLE III. H1 A/B GROUPS’ EFFECTIVE CANCELLATIONS SUMMARY

MF	*Group*	*Canc.*	*Not Canc.*	*Total*	*% Canc.*	*%Canc. Total*
0%	A	207	747	954	21.7%	20.7%
	B	187	766	953	19.6%	
50%	A	28	79	107	26.2%	25.7%
	B	26	77	103	25.2%	
75%	A	22	53	75	29.3%	29.9%
	B	22	50	72	30.6%	
90%	A	19	38	57	33.3%	35.0%
	B	16	27	43	37.2%	
100%	A	12	18	30	40.0%	37.5%
	B	9	17	26	34.6%	

TABLE IV. H2 A/B GROUPS’ EFFECTIVE CANCELLATIONS SUMMARY

MF	*Group*	*Canc.*	*Not Canc.*	*Total*	*% Canc.*	*%Canc. Total*
0%	A	513	1584	2097	25.4%	24.1%
	B	495	1593	2088	23.7%	
50%	A	80	199	279	28.7%	31.9%
	B	95	174	269	35.3%	
75%	A	52	109	161	32.3%	36.4%
	B	68	101	169	40.2%	
90%	A	43	64	107	40.2%	44.5%
	B	55	58	113	48.7%	
100%	A	35	38	73	47.9%	52.9%
	B	46	34	80	57.5%	

TABLE V. B GROUP ACTIONS RESULTS SUMMARY (MF 50%)

H	*Action*	*Canc.*	*Not Canc.*	*% Canc.*	*Chi-square (p)*	*Odds Ratio*
H1	No	23	46	33.3%	0.00224	8.0
	Yes	2	32	5.9%		
H2	No	89	140	38.9%	0.00357	3.6
	Yes	6	34	15.0%		

As presented in Table V, even though the number of bookings which were acted upon until now can still be considered low, the Chi-Squared test, shows that there is a statistically significant difference ($p<0.05$) between bookings which were contacted (actioned upon) and bookings which were not contacted. In fact, odds ratio shows that for H1 not contacting a guest has a booking cancellation enhancer factor in a magnitude of 8.0, and of 3.6 for H2, which clearly underlines the value of contacting bookings identified as “likely to cancel”.

V. CONCLUSION

Although this study is still in progress and these are preliminary results, it clearly shows the benefits of having such a system. Looking at the system as a machine learning prototype, designed in accordance to DSR to address an unsolved problem in a unique an innovative way [24], this system shows both in a theoretical as in a practical way, how a machine learning system to predict hotel booking cancellations, can be designed, implemented, and how it impacts business.

From a machine learning standpoint, this study shows how dataset splitting and feature engineering are important in machine learning models’ creation. It also stresses the need for machine learning researchers/practitioners to have some domain knowledge. One other important aspect, is the technological aspect behind the development of this prototype. The open source, Hadoop/Spark cluster, running R Server, allows the processing to be distributed through the different cluster machines, taking advantage of the computational power available and of the powerful XGboost tree boosting machine learning method, demonstrating how a machine learning system can run automatically, incorporating new data every day, together with previous predictions errors and hits to improve itself continuously.

From a business standpoint, its common understanding that a hotel does not have resources to contact every guest prior to his/her arrival. However, guests contacted by hotels, as this study demonstrated, even without being offered nothing substantial, cancel much less than guests not contacted. Also, a known fact, is that hotels to not have contacts of all future guests and that some guests are less sensitive to price discounts and offers (e.g. corporate guests). Therefore, hotels cannot expect to contact all guests identified as "likely to cancel", but still, the integration of such a machine learning prediction system in a RMS, can help hoteliers reduce the number of bookings to be contacted and with that, contribute to lower cancellation rates, at controlled costs.

Results on the testing dataset as exhibited in Table I, presented good results for both hotels, with an Accuracy above 0.84. Yet, Accuracy for expected arrivals, as exhibited in Table II, was 0.736 for H1, and 0.712 for H2. This shows that there is some space for improvement. In terms of future research, these models could benefit from the introduction of features from other data sources related to factors that affect customers booking/cancellation decisions, like competitors' prices, competitors' social reputation, weather, among others. Another feature that has the potential to improve model accuracy, taking in consideration the impact actions can have on customers' decision not to cancel, is a feature that identifies if an action to avoid cancellation was already taken on the booking. Given sufficient time, the system can have the potential to generate a labeled database with the actions made in each booking to avoid cancellation. This database could be used to develop another model to, in combination with this model, suggest which actions should be appropriate to take in each identified booking.

References

[1] R. Mehrotra and J. Ruttley, *Revenue management (second ed.)*. Washington, DC, USA: American Hotel & Lodging Association (AHLA), 2006.

[2] S. J. Smith, H. G. Parsa, M. Bujisic, and J.-P. van der Rest, "Hotel cancelation policies, distributive and procedural fairness, and consumer patronage: A study of the lodging industry," *J. Travel Tour. Mark.*, vol. 32, no. 7, pp. 886–906, Oct. 2015.

[3] K. T. Talluri and G. Van Ryzin, *The theory and practice of revenue management*. New York, NY: Springer, 2005.

[4] B. M. Noone and C. H. Lee, "Hotel overbooking: The effect of overcompensation on customers' reactions to denied service," *J. Hosp. Tour. Res.*, vol. 35, no. 3, pp. 334–357, Nov. 2010.

[5] N. Antonio, A. Almeida, and L. Nunes, "Predicting hotel booking cancellation to decrease uncertainty and increase revenue," *Tour. Manag. Stud.*, vol. 13, no. 2, pp. 25–39, 2017.

[6] N. Antonio, A. de Almeida, and L. Nunes, "Using data science to predict hotel booking cancellations," in *Handbook of Research on Holistic Optimization Techniques in the Hospitality, Tourism, and Travel Industry*, P. Vasant and K. M, Eds. Hershey, PA, USA: Business Science Reference, 2016, pp. 141–167.

[7] W.-C. Chiang, J. C. Chen, and X. Xu, "An overview of research on revenue management: current issues and future research," *Int. J. Revenue Manag.*, vol. 1, no. 1, pp. 97–128, 2007.

[8] L. Garrow and M. Ferguson, "Revenue management and the analytics explosion: Perspectives from industry experts," *J. Revenue Pricing Manag.*, vol. 7, no. 2, pp. 219–229, Jun. 2008.

[9] C. Hueglin and F. Vannotti, "Data mining techniques to improve forecast accuracy in airline business," in *Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining*, 2001, pp. 438–442.

[10] B. Freisleben and G. Gleichmann, "Controlling airline seat allocations with neural networks," in *Proceeding of the Twenty-Sixth Hawaii International Conference on System Sciences, 1993*, 1993, vol. iv, pp. 635–642 vol.4.

[11] C. Lemke, "Combinations of time series forecasts: When and hhy are they beneficial?," Bournemouth University, 2010.

[12] J. Subramanian, S. Stidham Jr, and C. J. Lautenbacher, "Airline yield management with overbooking, cancellations, and no-shows," *Transp. Sci.*, vol. 33, no. 2, pp. 147–167, 1999.

[13] M. G. Yoon, H. Y. Lee, and Y. S. Song, "Linear approximation approach for a stochastic seat allocation problem with cancellation & refund policy in airlines," *J. Air Transp. Manag.*, vol. 23, pp. 41–46, Aug. 2012.

[14] Zvi Schwartz, Muzaffer Uysal, Timothy Webb, and Mehmet Altin, "Hotel daily occupancy forecasting with competitive sets: a recursive algorithm," *Int. J. Contemp. Hosp. Manag.*, vol. 28, no. 2, pp. 267–285, Jan. 2016.

[15] L. N. Pereira, "An introduction to helpful forecasting methods for hotel revenue management," *Int. J. Hosp. Manag.*, vol. 58, pp. 13–23, Sep. 2016.

[16] W. Caicedo-Torres and F. Payares, "A machine learning model for occupancy rates and demand forecasting in the hospitality industry," in *Advances in Artificial Intelligence - IBERAMIA 2016*, 2016, pp. 201–211.

[17] L. R. Weatherford and S. E. Kimes, "A comparison of forecasting methods for hotel revenue management," *Int. J. Forecast.*, vol. 19, no. 3, pp. 401–415, Jul. 2003.

[18] D. R. Morales and J. Wang, "Forecasting cancellation rates for services booking revenue management using data mining," *Eur. J. Oper. Res.*, vol. 202, no. 2, pp. 554–562, Apr. 2010.

[19] D. Zenkert, "No-show forecast using passenger booking data," Lund University, 2017.

[20] H.-C. Huang, A. Y. Chang, C.-C. Ho, and others, "Using artificial neural networks to establish a customer-cancellation prediction model," *Przeglad Elektrotechniczny*, vol. 89, no. 1b, pp. 178–180, 2013.

[21] H. Chatterjee, "Forecasting for cancellations," in *AGIFORS 2001 Reservations and Yield Management Conference*, Bangkok, Thailand, 2001.

[22] P. H. Liu, "Hotel demand/cancellation analysis and estimation of unconstrained demand using statistical methods," *Revenue management and pricing: Case studies and applications*. Cengage Learning EMEA, pp. 91–101, 01-Jan-2004.

[23] A. Cleven, P. Gubler, and K. M. Hüner, "Design alternatives for the evaluation of Design Science research artifacts," in *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology*, New York, NY, USA, 2009, p. 19:1–19:8.

[24] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Q.*, vol. 28, no. 1, pp. 75–105, Mar. 2004.

[25] P. Chapman *et al.*, "CRISP-DM 1.0: Step-by-step data mining guide," *The Modeling Agency*, 2000. [Online]. Available: https://the-modeling-agency.com/crisp-dm.pdf. [Accessed: 10-Sep-2015].

[26] J. Quiñonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, Eds., *Dataset shift in machine learning*. Cambridge, Massachusetts: MIT Press, 2009.

[27] Z. Reitermanová, "Data splitting," in *WDS'10 Proceeding of Contributing Papers*, Praha, 2010, vol. Part I, pp. 31–36.

[28] D. Abbott, *Applied predictive analytics: Principles and techniques for the professional data analyst*. Indianapolis, IN, USA: Wiley, 2014.

[29] M. Kuhn and K. Johnson, *Applied predictive modeling*. New York, NY: Springer New York, 2013.

[30] J. Mount and N. Zumel, *vtreat: A statistically sound "data.frame" processor/conditioner*. 2017.

[31] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.

[32] N. Abe, B. Zadrozny, and J. Langford, "An iterative method for multi-class cost-sensitive learning," in *Proceedings of the tenth ACN SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, WA, 2004, pp. 3–11.