A Bayesian Predictor of Airline Class Seats Based on Multinomial Event Model

Bingchuan Liu
Ctrip.com
Shanghai, China
bcliu@Ctrip.com

Yudong Tan and Huimin Zhou
Ctrip.com
Shanghai, China
ydtan@Ctrip.com, hmzhou@Ctrip.com

***Abstract*—The allocation of class seats on a flight in airline industry is closely connected to multiple correlating factors, including yield management and airfare strategy from airlines, policy regulation and price modification from travel agencies, booking and reservation behavior from customers. Different from various machine learning methods targeting at direct fare or price prediction, we constructed a state predictor of class seats by applying a Naïve Bayes algorithm based on Multinomial Event Model on the core flight reservations inventory big data, to tell the probability of class availability within the next several hours or days. Four fundamental models and one integrated model are developed to propose an optimal decision to the airfare search engine layer, which makes the engine be capable of forecasting a smart buy-or-wait suggestion to customers. In our experimental route from SHA to TYO, the integrated model reaches an average of 95.42% accuracy.**

Keywords- Naïve Bayes; Multinomial Event Model; air fare; class seats prediction; generic learning algorithm

I. INTRODUCTION

In some business scenarios, we aim to build a predictor to forecast the airline ticket prices accurately in order to benefit customers by purchasing flight products with more optimal buy-wait strategies. A good deal of research work has been presented on understanding the behavior of prices, involving methods such as rule learning[1], reinforcement learning[2], regression model[3], pattern clustering[4], time series analysis[5] and combined multi-strategy [6]. Besides, several commercial products, for example, Kayak and Bing, have been released, though with few published references.

Nowadays airlines have employed various sophisticated and complicated systems, such as auto reservation, pricing regulation and yield management to maximize their global revenue[7], which lead to the air ticket price fluctuating dynamically and sensitively due to sorts of tiny disturbances. What's more, big player airlines introduce some noise into their pricing patterns sometimes to suffer from profit margins loss caused by prediction methods. Therefore, to predict the price directly in a right way, too much complicated, connected prior knowledge about the airlines' commercial decisions are required to be captured.

Based on our current logic of ticket price search engine, price of a flight depends on fare information and class state simultaneously. In most cases, effective air fare with an unavailable class seat indicates that the value and the price of the seat will go up. As a result, the frequent price change is caused by the number of open or closed classes instead of the unchanged air fare. Thus we consider the state of class seat, the more essential and dedicated regulator to the price, as our prediction target. There has been few study concerning the class seats, maybe because of lacking in core seats data to learn model from or the engine layer to combine model into.

In this paper we address the method and implementation of building a predictor on the inventory reservations dataset of flight class seats, which are usually treated as hidden variables in the price prediction models. We construct four basic models based on Naïve Bayes multinomial event model, considering different conditional selection and information gain from business view. We also establish a comprehensive model to improve the balance between the accuracy, precision and true positive findings. Moreover, we make an optimal decision according to the overall performance of five models and embed it into the airfare search engine layer in application level. Finally, an instance of a popular route is presented and evaluated to show its valid usage scenario. By giving the joint probability of classes availability in the situation where state of seats could change, the predictor is capable of informing customers of a potential saver in.

II. METHODS

A. Problem Statement

The rule about future state of a class seat, being available or not, can be learned according to the current string of observed seat states, which makes it possible to forecast the likelihood that an airfare price would rise or decline. We intend to construct a state predictor of class seats by applying a Bayesian classifier based on Multinomial Event Model in the context of core inventory reservations data, in order to tell the probability of seats availability within the next several hours or days.

B. Model Description

Although several machine learning algorithms have been deployed in various of classification problems, we adopt Naïve Bayes classifier which stands out as a generic learning

algorithm[8] for its easy implementation, linear computational complexity and being comparable to those ones with more elaborate learning algorithms[9, 10].

Abstractly, in a training set with size *m*, each data sample is constructed as a feature vector $\vec{F}$ which is composed of several feature variables $(F_1, \dots, F_n)$ with length *n*, and a class variable *C* which labels the category the data sample belongs to. In our cases, we define the set of possible class states allocated by airlines as feature vector where each class states are feature variables; define the class states within the next 72 hours, open or closed, as label class.

The probability model for a classifier is considered to be a conditional model over a dependent class variable *C* conditional on feature vector $\vec{F}$, which can be expressed as the conditional distribution of *C* given feature variable F_1 through F_n. Applying Bayes' rule, $p(C|\vec{F})$ can be derived as:

$$p(C|\vec{F}) = \frac{p(C)p(\vec{F}|C)}{p(\vec{F})} \quad (1)$$

$p(\vec{F})$ in (1) is a constant due to its independence on class *C* and given values of each feature F_i, therefore the posterior $p(C|\vec{F})$ is equivalent to $p(\vec{F}|C)$. Use the chain rule for repeated application of the condition probability to describe the probability distribution $p(\vec{F}|C)$ in terms of conditional probabilities; then impose the naïve conditional independence assumptions that assuming each feature F_i is conditionally independent of every other feature given the category *C* to simplify the joint model. Consequently, the likelihood which models the feature distribution of a specified category can be written as a product of joint probability:

$$p(\vec{F}|C) = \prod_{i=1}^{n} p(F_i|C) \quad (2)$$

To estimate the parameters for the feature's probability distribution $p(F_i|C)$, a distribution that has generated the model should be assumed[11]. From the two first-order event models of the Naïve Bayes classifier used in discrete-valued feature vectors, we choose multinomial event model instead of multinomial event model since the former is found to be almost uniformly better than the latter in the problems of text classification[12-15]. Multinomial Event Model models the $p(F_i|C)$ with a multinomial distribution, where feature vectors represent the term frequencies. With the definition of corresponding $(v_1, \dots, v_n)$ being the feature values, the likelihood of observing a feature vector $\vec{F}$ is given by:

$$p(F_1 = v_1, \dots, F_n = v_n|C) = (\sum_{i=1}^{n} v_i)! \prod_{i=1}^{n} \frac{p(F_i|C)^{v_i}}{v_i!} \quad (3)$$

We adopt the maximum likelihood estimates of the parameters and introduce Laplace smoothing to avoid the situation that probability estimate hits zero when a feature variable never appears in the training data set. Accordingly, the estimates of each feature f_i of a specified category *c* can be calculated by:

$$p(F_i = f_i|C = c) = \frac{\sum_{j=1}^{m} 1\{F_i=f_i \wedge C=c\} \cdot v_i + 1}{\sum_{j=1}^{m} 1\{C=c\} \cdot (\sum_{i=1}^{n} v_i) + |\vec{F}|} \quad (4)$$

Finally, the class label on data set can be determined by calculating the class posterior probability and selecting the class with the highest probability.

- *Generic Model and Specific Model*

According to whether involving other business related information, we divide the models into two subtypes: generic model and specific model. For the generic model, the parameters are learned from the full data set in their entirety; For the specific model, the parameters were learned on every data subset, which were conditional filtered by multiple essential airfare attributes, including advance purchase weeks, peak or low seasons, holiday definition, etc.

- *Native Model and Weighted Model*

According to whether using the maximum likelihood estimates as the class probability, we divide the models into two subtypes: native model and weighted model. For the native models, classes' priors were calculated by assuming equal probability; For the weighted models, the frequencies of classes were calculated from the training set or subset, as the class probability.

- *Integrated Model*

Beyond the basic models, we build a comprehensively integrated model that takes consideration of the four basic models mentioned above by voting mechanism. Finally, the optimal model is determined between the integrated model and the basic model who own the highest precision and the largest amount of positive findings.

C. Algorithm Implementation

Our algorithm can be divided into learning process and reasoning process. Learning modules conduct parameter learning from the historical big volumes of data; Reasoning modules execute the prediction on the real-time data by learning models. For each flight number, class code, do:

Algorithm of Learning Process

1. **Prepare Model Data:**
   - 1.1 Merge data on composite dates;
   - 1.2 Refine data on query dates;
   - 1.3 Make statistical count on composite dates and query dates;
2. **Make Label Class:**
   - 2.1 Mark label class of each data sample on class state string within the next 72 hours
3. **Build Learning Model:**

   for *i* in (Generic Model, Specific Model):
   - 3.1 Initialize data subset on conditional filters on specified attributes dependent upon *i*;
   - 3.2 Calculate conditional probability $p(F_i|C)$;

   for *j* in (Native Model, Weighted Model):
   - 3.3 Calculate prior probability dependent upon *i*,*j*;

4. Make Posterior Probability:

4.1 Construct joint string of class state;
for *i* in (Generic Model, Specific Model):
for *j* in (Native Model, Weighted Model):

4.2 Calculate joint probability $p(C|\vec{F})$ for each data sample dependent upon *i*,*j*;

4.3 Pick the class with higher posterior as the reason class;

4.4 Build an integrated model based on the above 4 basic models by comprehensive balance between the amount of positive finding and precision;

5. Evaluate Learning Model:

for *i* in (Generic Model, Specific Model):
for *j* in (Native Model, Weighted Model):

5.1 Initialize data subset on conditional filters on specified attributes dependent upon *i*;

5.2 Make conditional count and calculate confusion matrix;

5.3 Determine the optimal model among one integrated model and four basic models as final winner.

Algorithm of Reasoning Process

1. Prepare Model Data:
the same as Learning Process

2. Make Label Class:
the same as Learning Process

3. Make Posterior Probability:

3.1 Calculate joint probability $p(C|\vec{F})$ for each data record, on the conditional probability of the optimal model.

3.2 Pick the class with higher posterior as the reason class.

III. EXPERIMENTS

We took a popular route from Shanghai (SHA), China to Tokyo (TYO), Japan for instance. Four representative flights from different airlines (NH, JL, DL and MU) were selected to display since they have the close daily departure time around 9-10 a.m.

A. Data

Airline class inventory reservations data were collected from our databases by data provider interface, covering one-year timespan (from July, 2015 to June, 2016) and multiple connection kernels (TravelSky, Amadeus, etc.). Scope of coach classes on non-stop flights by non-code sharing flights were restricted. Also the advance days between query date and flight day were limited within 4 to 119 days. Then a key reduction method that merging data records with same flight date and query hour into one training sample based on weighted time-distance was applied to correct the data unbalance influence caused by our scheduled downloading strategy. Table 1 displays the data size and reduction ratio after these prerequisites.

TABLE I. SIZE OF DATA SET

	Flight Number			
	NH922	**JL872**	**DL296**	**MU523**
Source Size	398704	397787	271463	678946
Refined Size	260443	178770	160828	288007
Reduction Ratio	65.32	44.94	59.24	42.42

Airfare impacting attributes used in specific models, including advance purchase weeks, peak or low seasons, holiday definition were extracted from ATPCO data.

B. Results

We built five models for every combination of flight origin, flight destination, flight number and class code. Only up to ten states can be seen according the protocol, no extra dimensionality reduction is needed.

In the SHA to TYO example, the five models achieved nearly equivalent accuracies. Table II presents the accuracy of integrated model, whose average accuracy over all class codes reached 0.9696, 0.9608, 0.9373, 0.9510 respectively. Note since each airline has its own defined class codes, a sequence of letters was used here to refer to each mapping level of coach classes of different airlines.

We explored that a large majority of unchanged class states contributed to the observed high accuracy in Table II, however it is the changed state, especially the state from being zero to non-zero which infers a price drop, that would bring a saver to the customer. Considering this valuable situation, we conducted a filter on the data samples with unavailable current seat.

Took a further look at flight NH922 since similar patterns were found among the other flights. Accuracy decreased along the class fare ascending sequence, as table III shows. Moreover, for each class we analyzed the probability of its being in the closed state, in the open state after being closed and the product of the above two probabilities. The state pattern seen in Fig. 1 looks consistent with the gradual behavior of accuracy.

Furthermore, Fig. 2 demonstrates the Precision (Positive Predictive Value) (in bar) and the true positive counts (in curve). PPVs present obvious difference among different class codes, maybe airlines define various business usages on each class, with a basic trend along the ascending air fare. It can be inferred that some class codes (i.e. J) may have their special

TABLE II. ACCURACY OF INTEGRATED MODEL

	Flight Number			
Class Code	**NH922**	**JL872**	**DL296**	**MU523**
M	NA	NA	0.9186	0.8973
L	0.9522	0.9174	0.9251	0.9598
K	0.9577	0.9296	0.9500	0.9153
J	0.9498	0.9308	0.9388	0.8838
I	0.9513	0.9415	0.9341	0.9230
H	0.9564	0.9539	0.9164	0.9421
G	0.9638	0.9665	0.9409	0.9524
F	0.9749	0.9752	0.9198	0.9668
E	0.9788	0.9817	0.9254	0.9779
D	0.9816	0.9770	0.9547	0.9853
C	0.9890	0.9847	0.9345	0.9892
B	0.9893	0.9862	0.9594	0.9757
A	0.9906	0.9846	0.9666	0.9945

TABLE III. ACCURACY OF INTEGRATED MODEL ON CHANGED CLASSES

Class Code	Model: Integrated Overall	Generic Native	Generic Weighted	Specific Native	Specific Weighted
L	0.9319	0.9318	0.9318	0.9317	0.9319
K	0.9398	0.9399	0.9399	0.9399	0.9399
J	0.9145	0.9148	0.9148	0.9144	0.9145
I	0.8891	0.8888	0.8889	0.8890	0.8891
H	0.8510	0.8504	0.8506	0.8506	0.8508
G	0.8007	0.7962	0.7973	0.8005	0.8011
F	0.7713	0.7623	0.7646	0.7712	0.7718
E	0.7273	0.7131	0.7172	0.7264	0.7331
D	0.7018	0.6852	0.6908	0.7006	0.7097
C	0.6990	0.6775	0.6852	0.6955	0.7071
B	0.6225	0.6021	0.6087	0.6203	0.6361
A	0.5734	0.5554	0.5600	0.5713	0.5973

usage by airlines. Specific models which incorporated fare knowledge as information gains achieved around 1% improvement in Precision and obtained more true positive findings comparing to Generic models. Slight differences can be told between Native models and Weighted models. Every class seat got an optimal model finally, and in most cases it was the integrated overall model which balanced the performance between the higher accuracy and more true positive findings. In the cases where the specific weighted model performed competitive well, we believe a sophisticated and dedicated fare regulation on those classes has been executed by airlines and captured by our models.

IV. CONCLUSION

In this paper we investigated the airline ticket prices on the view of class seats' inventory reservations instead of building a direct price predictor like most research do. The multinomial event model Na ïve Bayes classifier we adopted achieved an average of 95.42% accuracy in the instance with its easier implementation and linear computational complexity. The model prediction results can be embedded into search engine layer to give effective buy-or wait suggestion to customers, thus the predictor devotes itself into an instance of big data application. For those classes who has been set complicated fare strategies by airlines, the specific weighted models give a remarkably close description by introducing the fare and true positive findings.

In future work we plan to extend the model into a multiple-category task to fit the more complicated business scenario. Furthermore, we will investigate the role of class string length in classification through the event model that normalize the state occurrence counts in a class seats string. Moreover, we will seek to consider using Bayesian classifiers that are less restrictive than Na ïve Bayes. With the promotion of the predictor into more routes, we may face the challenge that independence assumption of Na ïve Bayes could be violated since some airlines correlate their seat inventory of each class[16]. In summary, we believe that prediction of class seats and fare price in airline industry is a fertile Big Data application area in future research.

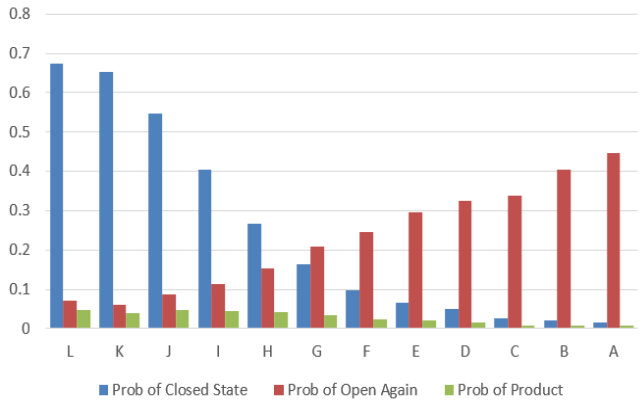
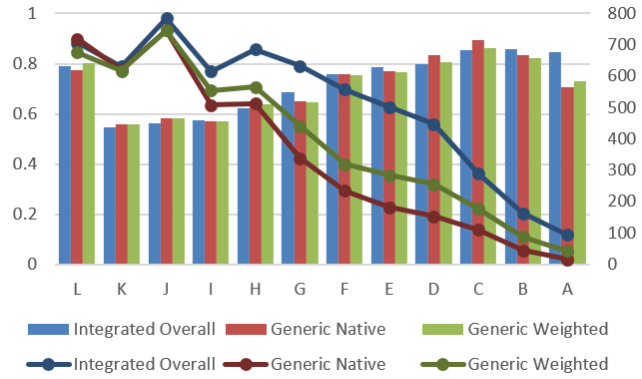
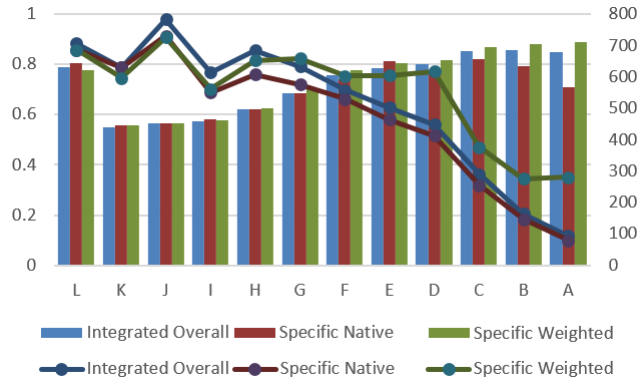


Figure 1. State Pattern of Class Seat



(a) Integrated Model and Generic Models



(b) Integrated Model and Specific Models

Figure 2. Precision and True Positives of all models

ACKNOWLEDGMENT

Many thanks to Dr. Yudong Tan and Mr. Huimin Zhou for their great support and helpful comments on the work.

REFERENCES

[1] W. W. Cohen, "Fast effective rule induction," in *Proceedings of the twelfth international conference on machine learning*, 1995, pp. 115-123.

[2] Sutton, R.S. and A.G. Barto, Reinforcement learning: An introduction. Vol. 1. 1998: MIT press Cambridge R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction* vol. 1: MIT press Cambridge, 1998.

[3] W. Groves and M. Gini, "A regression model for predicting optimal purchase timing for airline tickets," Technical Report 11-025, University of Minnesota, Minneapolis, MN2011.

[4] T. Wohlfarth, S. Clémençon, F. Roueff, and X. Casellato, "A data-mining approach to travel price forecasting," in *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on*, 2011, pp. 84-89.

[5] Y. Chen, J. Cao, S. Feng, and Y. Tan, "An ensemble learning based approach for building airfare forecast service," in *Big Data (Big Data), 2015 IEEE International Conference on*, 2015, pp. 964-969.

[6] O. Etzioni, R. Tuchinda, C. A. Knoblock, and A. Yates, "To buy or not to buy: mining airfare data to minimize ticket purchase price," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, pp. 119-128.

[7] A. W. Donovan, "Yield management in the airline industry," *Journal of Aviation/Aerospace Education & Research*, vol. 14, 2005.

[8] A. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," *Advances in neural information processing systems*, vol. 14, p. 841, 2002.

[9] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A Bayesian approach to filtering junk e-mail," in *Learning for Text Categorization: Papers from the 1998 workshop*, 1998, pp. 98-105.

[10] J. Koutsias, K. Chandrinos, and C. Spyropoulos, "An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages," 2005.

[11] G. H. John and P. Langley, "Estimating continuous distributions in Bayesian classifiers," in *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, 1995, pp. 338-345.

[12] A. McCallum and K. Nigam, "A comparison of event models for naive bayes text classification," in *AAAI-98 workshop on learning for text categorization*, 1998, pp. 41-48.

[13] V. Metsis, I. Androutsopoulos, and G. Paliouras, "Spam filtering with naive bayes-which naive bayes?," in *CEAS*, 2006, pp. 27-28..

[14] P. Pantel and D. Lin, "Spamcop: A spam classification & organization program," in *Proceedings of AAAI-98 Workshop on Learning for Text Categorization*, 1998, pp. 95-98.

[15] P. Graham, "A plan for spam," ed, 2002.

[16] P. Domingos and M. Pazzani, "Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier," ed, 1996.