IN-STORE SHOPPING ACTIVITY MODELING BASED ON DYNAMIC BAYESIAN NETWORKS

Ping Yan, Daniel D. Zeng
Department of Management Information Systems, University of Arizona
{pyan, zeng}@email.arizona.edu

Abstract

RFID technology has been recently adopted in retail environments to track consumer in-store movements, bringing about new exciting opportunities for spatial data mining-enabled marketing. In this paper, we propose a Dynamic Bayesian Networks (DBN)-based model of customer in-store shopping trips and activities. This model infers a customer's product purchase interest given the observations of customer in-store movement data collected through a RFID-based wireless network. We also report a preliminary evaluation of our approach using a real-world dataset. Our proposed approach can be potentially used to help create an intelligent shopping environment, in which store operators can target their marketing efforts at providing effective location-aware real-time product recommendation for individual customers.

Keywords: Dynamic Bayesian Networks, DBN, Radio Frequency Identification Devices, RFID, Retailing, Shopping Activity Modeling

1. Introduction

Radio Frequency Identification Devices (RFID) refers to technologies and systems that use radio waves to transmit and uniquely identify objects (Finkenzeller, 2003, Heijden, 2006). In the retail industry, RFID applications are traditionally centered on supply chain management and business integration (Koh et al., 2006). Recently, a new class of RFID application involving customer in-store movement tracking is emerging. In an RFID-enabled shopping space, each shopping cart is attached with an RFID tag. As a customer pushes a cart around the store, RFID antennas mounted on both sides of the aisles can pinpoint the cart’s location within inches. Several key companies deploying the in-store RFID systems for customer movement tracking include Intelligentz and Sorensen Associates (Sorensen 2003; Swedberg 2006). Microsoft has also recently collaborated with MediaCart and ShopRite to develop “smart carts” (Graham-Rowe, 2006).

Examining customer in-store shopping trips allows one to learn customer shopping behavior at an unprecedented level of granularity and detail without being obtrusive or incurring large data collection expenses. Recognizing and predicting customers’ shopping activities through modeling their travel choices and buying decisions during shopping trips can lead to many innovative applications in a pervasive retailing environment. One of these potential applications is in-store personalized product recommendations delivered right “on the spot.”

This paper reports our research on modeling customer shopping behavior based on Dynamic Bayesian Networks (DBN) with the aim to recognize and predict various types of customers shopping activities. DBN-based methods have been widely used as a tool for human behavior modeling (Murphy 2002), and are particularly suitable to study sequential actions and identify appropriate interpretations for such actions. The objective of our research is to develop and evaluate DBN-based inference mechanisms for in-store shopping trip modeling and prediction. We apply DBN to determine as early as possible which product a customer is attempting to purchase, and to predict which store location a shopper might visit in the next move. Formally, we model a shopper’s product interests and travel choices as a set of conditional decisions. A shopper is assumed to decide first his shopping goal - what products to purchase, then, conditioned on that choice, the travel choices, then the purchasing choices, and so on. Each choice is
modeled separately, resulting in a set of models of conditional and marginal choice probabilities which together can be used to express the choice probabilities for any type of shopping trips.

The rest of the paper is structured as follows. We first present our study’s theoretical background and discuss some related modeling work. We then present in detail our DBN-based model and its inference mechanism. A preliminary evaluation of our approach using a real-world dataset is then reported with some encouraging findings.

2. Theoretical Background and Related Work

The existing literature supports the view of considering a shopping trip as a process of goal setting and task accomplishment. One theory suggests that people like to be in situations in which they are constantly making progress towards their goals (Deci and Ryan 1985). Takahashi (Takahashi 1988) points out that there is empirical evidence that the choice of a destination depends substantially on the utility of possible later destinations on the same trip chain. Although empirically such behavior was only studied in out-of-store multi-destination shopping trip scenarios (Brooks et al. 2008), within a store, people are expected to exhibit similar trip chaining behavior. Shopping trips should be carefully examined from a spatial standpoint to shed lights on customer search and purchase behavior.

Limited work has been dedicated to studying in-store shopping behavior utilizing the traversal paths each shopper makes. This is mainly due to the difficulty of accessing consumer in-store shopping trip data. Recently, researchers start to leverage the datasets made available with RFID devices for shopping behavior study (Hui et al. 2008). The effectiveness of these analytical models for real-time shopper choice inference, however, is not yet clear.

We model customer purchase and in-store travel choice decisions using the DBN framework. A DBN extends a Bayesian network (BN) by including a temporal dimension. It consists of a sequence of BNs to represent the world. At each time slice, the same BN is used to model the dependencies among variables. In addition to intra-slice dependencies captured in the BNs, inter-slice connections are used to represent temporal dependencies in consecutive time slices (Murphy, 2002, Dean and Kanazawa, 1989).

An important property of DBN is that they readily support hybrid recommendation approaches combining, for instance, content-based recommendation (based on a single customer’s historical purchases) and collaborative recommendation (based on purchasing histories of multiple customers). User characteristics of the training population can be learned as conditional probability distributions and the initial beliefs of a BN in a collaborative manner. These beliefs can then be updated using a content-based approach as transactional data concerning a particular user are collected. This hybrid approach enables a predictive model to overcome the data collection problem associated with the content-based approach which requires a large amount of historic data from a single customer. At the same time, it enables the tailoring of a collaboratively-learned model for a single user. DBNs also exhibit adaptive behavior, as they can update the probability distributions as the time slices roll forward.

Successful applications of DBN inference methodologies include speech recognition, plan recognition (Lesh and Etzioni 1995), goal prediction from a given sequence of command actions in the UNIX domain (Blaylock and Allen 2005). The most widely used are the ones embedded in Microsoft products (e.g., Office Assistant and Vista Operating System) (Heckerman, 1999).

3. DBN-based Shopping Activity Modeling

In our study, the development of the DBN-based inference methodology consists of the following steps: (1) identify domain variables; (2) examine dependencies between the domain variables and the manner in which these domain variables change over time; (3) describe how the conditional probability distributions are constructed from the shopping trip data; and (4) develop the belief update procedural.
Formally, we denote the finite set of purchase goals as $G = \{g_0^0, \ldots, g_k^0\}$ (assuming there are $k$ products on shelf in the store). The finite set of visit locations during a shopping trip is denoted as $L = \{L_0, \ldots, L_l\}$ (assuming that the store space can be divided into $l$ store regions where a shopper can travel through, each region displaying one or more product categories).

We model a customer’s shopping activities using a two-level DBN model shown in Figure 1. The nodes represent different domain variables and the arcs indicate dependencies between the nodes (Murphy 2002). Temporal dependencies are represented by arcs connecting two time slices. The upper level of the model describes a purchase sequence corresponding to a customer’s individual purchase goals at different steps within a shopping trip. A purchase goal represents the product a shopper currently intends to purchase during a trip segment. A trip segment is a sequence of visitation-purchase pairs, within which a shopper completes a single shopping task, i.e., buys one product. The lower level of the model represents the transitions between the location a shopper just visited and the location this shopper is about to visit. The links from a goal node $G$ to the locations nodes mean that a purchase goal is achieved by visiting a certain sequence of store locations. The location $L_n$ that a customer visits at time slice $t$ depends on his or her location $L_{t-1}$ at the previous time slice, as well as the purchase goal $G$ they are currently aiming at. The Boolean trip switching node is set to be true whenever the customer reaches the purchase point where a purchase is made, indicating the end of the current trip segment. The next trip segment is chosen according to the segment transition conditioned on the current goal as well as the current visit location. Once the next trip segment is active, the switch node is suspended. In the model reported in this paper, the purchase goal of the customer can only change when the end of a trip segment is reached.

With the model specified, we now discuss the learning of the parameters, i.e., the conditional probability distributions of the DBN nodes from training shopping trips. Observations of a shopper’s visit locations or purchase decisions are added to the network as new evidence, and the probability distributions of the current purchase goal and visit locations are inferred given the evidence in real time. We then update the beliefs of a shopper’s next location and the product category he intends to purchase. The belief update algorithm used to process one trip segment is presented in Table 1.

**Table 1: Belief update procedural: inference of the next location and the current purchase goal within an individual trip segment**

1. **Update current location at time-slice #1:**
   
   $p(l_t | g_0^0, l_0) = \sum_{g_1^1} p(l_t | g_1^1, l_0) p(g_1^1 | g_0^0)$

   $p(g_1^1 | g_0^0, l_0, l_t) = \alpha p(l_t | g_1^1, g_0^0, l_0) p(g_1^1 | g_0^0, l_0) = \alpha p(l_t | g_1^1, l_0) p(g_1^1 | g_0^0)$, as $g_0^0$ d-separates $g_1^1$ and $l_0$, and $\alpha = 1/ p(l_t | g_1^1, g_0^0, l_0)$
2. From time-slice #2 till the completion of a trip segment:

\[
p(L_{n+1} = l_{n+1} \mid g^1, l_0, \ldots, l_n) = \sum_g p(l_{n+1} \mid g^1, l_0, \ldots, l_n) p(g^1 \mid g^0, l_0, \ldots, l_n)
\]

\[
L_{n+1}^* = \arg\max_p p(L_{n+1} = l_{n+1} \mid g^1, l_0, \ldots, l_n)
\]

\[
p(G^i = g^1 \mid g^0, l_0, l_1, \ldots, l_{n+1}) = \alpha p(l_{n+1} \mid g^1, g^0, l_0, \ldots, l_n) p(g^1 \mid g^0, l_0, l_1, \ldots, l_n),
\]
\[
\alpha = 1/ p(g^1, g^0, l_0, \ldots, l_n, l_{n+1})
\]

**Applying the Bayes’ Rule**

\[
p(G^i = g^1 \mid g^0, l_0, l_1, \ldots, l_{n+1}) = \alpha p(l_{n+1} \mid g^1, l_n) p(g^1 \mid g^0, l_0, l_1, \ldots, l_n)
\]

\[
G^* = \arg\max_p p(G^i = g^1 \mid g^0, l_0, l_1, \ldots, l_{n+1}),
\]

i.e., \( G^* = \arg\max_p p(l_{n+1} \mid g^1, l_n) p(g^1 \mid g^0, l_0, l_1, \ldots, l_n) \)

Note: If the trip segment begins as the shopper enters the store, \( G^0 \) and \( L_0 \) are set to NULL. Otherwise, if the segment begins upon a completion of one purchase, \( G^0 \) is set to the purchase made in the last segment. The location the shopper visited at the previous time step is assigned to \( L_0 \).

4. A Preliminary Evaluation

To evaluate our proposed DBN-based approach, we have conducted a preliminary computational study using real-world data. We acquired 843 RFID-collected shopping trips from the PathTracker system developed at Sorensen Associates, a retail marketing research firm. There are 1074 distinct products in the dataset. They are categorized into 38 product categories. In this research, we study customer purchase behavior at the product category level. It is an appropriate level of aggregation to deal with the data sparseness problem commonly seen in marketing studies. For each shopping trip, the shopping cart’s two-dimensional location coordinates are recorded at five-second intervals. The locations where a shopper picks up an item are also recorded. Other available information includes the time-of-visit of the trip; the length of the entire trip in seconds and in feet; the average travel speed over the trip, and the shopper’s stay time and travel velocities at each location. Associated with each shopping trip, each purchased product’s description, price and its category are also available.

Following the standard practice in the data mining literature, model performance is assessed using a 10-fold cross-validation procedure (Devijver and Kittler 2001). To first show how the predictions are updated during the process, we depict the predicted probabilities of the current goal (Figure 2(a)) and the next locations (Figure 2(b)) for a single trip segment (trained on 90% of the data). The real example used in Figure 2 represents a relatively successful prediction sequence performed for individual trip segments. Less successful runs take longer for the predictions to improve, exhibiting more fluctuations. Figure 2(a) shows that initially the model predicts a low probability for the purchase goal that is being attempted. The predicted probabilities keep rising after the first few steps and get above 80% after about one third of the trip segment are observed. The predicted probabilities of the actual goal and actual next locations in that trip segment are shown as solid lines, while the predicted probability of the most likely purchase and next locations are shown as dashed lines. As shown in Figure 2(a), the predicted most likely purchase matches the actual purchase except a small percentage of the time (note that the solid line and the dashed line overlap after 14 steps). Similarly, the predictions of the next locations turn out to be actual next locations most of the time (Figure 2(b)).
Figure 2 (a-left; b-right): A typical successful inference example showing the predicted probabilities of the current purchase goal (left), and the predicted probabilities of the next locations (right). The solid lines represent the probabilities of the actual current goal or next locations. The dashed lines represent the probabilities of the most likely purchase or next locations.

We measure the prediction accuracy by computing the average hit rates. A hit is a prediction that matches the product or the location that are actually purchased or visited as recorded in the testing dataset.

Formally, \[ \text{number-of-hits} = \sum_{i} I_i, \quad I_i = \begin{cases} 1, & \text{the prediction matches the actual value at step } i \\ 0, & \text{otherwise} \end{cases} \]

Thus, \( \frac{\text{number-of-hits}}{\text{number-of-steps}} \) suggests the prediction accuracy after a number of steps. We plot the average hit rates of the purchase and location predictions across all the testing trips in Figure 3. For evaluation purposes, we also implemented as a benchmark method, the widely-applied Bigram method used in collaborative filtering and language modeling. This method keeps frequency counts of the co-occurrence of any pair of items and predicts the next item in a sequence given the preceding item. The visit location sequence and the purchase sequence are modeled using Bigram separately, with the dependency between a purchase goal and the visit locations removed. As shown in Figure 3, the DBN method outperforms the Bigram.

Figure 3: Performance Comparison: (left) average prediction accuracy of purchases by trip segment percentage and (right) average prediction accuracy of locations by trip segment percentage.

5. Conclusion and Future Work

RFID applications have created many new exciting opportunities for spatial data mining-enabled marketing. The reported study represents the first attempt to apply a DBN decision-theoretic approach to modeling customer in-store shopping activities. The developed inference procedure is tested on a real RFID-collected shopping trip dataset with promising results in terms of prediction accuracies of customer...
purchase goals and in-store visit locations. Our work can be potentially applied to help create an intelligent shopping environment, in which store operators can target their marketing efforts at providing effective location-aware real-time product recommendation for individual customers.

We are currently working on extending the reported work in several directions. 1) We are developing more sophisticated DBN network structures to incorporate additional information encoded in the PathTracker dataset, for example, the shoppers’ moving velocity, their deliberation time over each purchase, and the price information. 2) We plan to add another network layer to differentiate different shopping activities as identified in the marketing literature, e.g., impulse shopping vs. planned shopping. 3) In this study, we assumed that during each trip segment, a shopper’s travel choice depends on a single purchase goal. We are now extending our model to handle multiple conjunctive shopping goals. 4) We are experimenting with additional benchmark methods and performing comparative evaluation studies.

Acknowledgements

The authors wish to acknowledge research support from US NSF (0839990, 0428241), DHS (2008-ST-061-BS0002), NNSFC (60621001, 60875049, 70890084), MOST (2006AA010106), MOH (2009ZX10004-315), and CAS (2F07C01, 2F08N03).

References

A PRIVACY PROTECTION TECHNIQUE FOR PUBLISHING DATA MINING MODELS AND SUPPORTING DATA
Yu Fu, Zhiyuan Chen¹, A. Gunes Koru, Aryya Gangopadhyay
Department of Information Systems, University of Maryland, Baltimore County
{yufu1, zhchen, gkoru, gangopad}@umbc.edu

Abstract
Data mining techniques have been widely used to extract useful knowledge (as mining models) from data. The model and the supporting data often need to be published together to allow others to verify the model or to use the data in their own research. However, the privacy of the published data needs to be protected. Existing privacy protection methods can protect privacy, but they do not guarantee that the same model can be built from sanitized data. Thus models cannot be verified. This paper proposes a technique that not only protects privacy, but also guarantees that the same model, in the format of decision trees, can be built from the sanitized data. Users can also apply other mining techniques to the sanitized data to achieve good results. This technique can be used to reduce research fraud and the cost of collecting research data.

Keywords: Privacy preserving data mining, data privacy, data mining

1. Introduction
Data mining techniques have been widely used to extract useful knowledge in the format of mining models from data. These models often need to be published to benefit the general public or to prove the effectiveness of a new product such as a new drug. The supporting data (i.e., the data used to create the model) often needs to be published as well for the following benefits:

- This can add credibility to the results and reduce research fraud. There has been high profile fraud in medical research recently (Black April 18, 2006). Such fraud could have huge negative impacts. Publishing supporting data allows others to verify the published results. This can reduce research fraud because researchers cannot fake the results (they can still use faked data, but to catch faked data the data needs to be made available as well).
- This will allow other researchers to use the data in their own research (e.g., to try a different mining technique). This will reduce the cost of collecting research data, which could be quite expensive. The cost of clinical trials conducted in US for new drugs was $25 billion at 2006, and was about $100 million to $800 million per drug (Fee March 01, 2007).

For example, suppose a clinical trial has been conducted by a group of researchers to evaluate the effectiveness of a new drug. They have built a decision tree to predict whether the new drug will be effective for a patient based on the patient’s medical conditions and genes (Bellazzi et al. 2008). They need to publish this decision tree in a medical journal. In addition, they will publish the raw data because this allows others to verify their model using the supporting data and brings more confidence in the results of research. Further, other researchers can use the clinical trial data in their own research, e.g., they can try a different mining technique (Vickers 2006).

However, since the supporting data often contains privacy sensitive information (e.g., patient’s medical conditions), it is necessary to protect the privacy before releasing the data. Actually, privacy issues have prevented many companies or researchers from sharing raw clinical data (Vickers 2006).

¹ This work was supported in part by the National Science Foundation under Grant Numbers IIS-0713345.
There has been a rich body of work on privacy protection techniques (Aggarwal et al. 2008). The existing techniques are sufficient to protect privacy, but they usually do not guarantee that the same mining model can be built from sanitized data. Thus other researchers cannot verify the published models.

For example, Figure 1 (a) shows a decision tree built from UCI Adult data set (Hettich et al. 1998), which predicts whether the annual income of a US household is over 50k. Figure 1 (b) and Figure 1 (c) show the tree built from data sanitized by two existing methods in (LeFevre et al. 2006a) and (LeFevre et al. 2006b), respectively. The trees in Figure 1 (b) and Figure 1 (c) are very different from the original one in Figure 1 (a). In addition, the prediction accuracies of these trees are also lower than that of the original tree.

This paper makes the following contributions:

- We propose a technique that both protects privacy and guarantees that decision tree, a popular data mining model, will be preserved (i.e., the tree built from the sanitized data is the same as that built from the original data).
- Experimental results show that users can apply other data mining techniques (e.g., Naïve Bayesian) to the data sanitized using our approach to achieve reasonably good results.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 describes our approach. Section 4 shows the experimental results and Section 5 concludes the paper.

2. Related Work

The two most popular privacy models are K-anonymity (Sweeney 2002) and L-diversity (Machanavajjhala et al. 2007). K-anonymity prevents linking attacks, which link attributes (called quasi-identifier) such as birth date, gender, and ZIP code with publicly available data sets. K-anonymity ensures that there are at least $K$ people with the same quasi-identifier. This can be done by generalization, i.e., replacing specific values with more general ones. For example, the exact age of a patient can be replaced with a range. Records with the same quasi-identifier values form an equivalence class. L-diversity further requires that the people with the same quasi-identifier contain at least $L$ well-represented sensitive values such that attackers cannot discover the values of sensitive attributes easily.

There has been a rich body of work to enforce K-anonymity and L-diversity models (Aggarwal et al. 2008). A workload aware anonymization approach was proposed in (LeFevre et al. 2006b), where the anonymization process is optimized for specific mining tasks. For example, the anonymization tries to

![Figure 1. Decision trees built from original Adult data and sanitized data](image-url)
maximize information gain (which is used in decision tree building) for classification. However, the information gain computed in the original data is typically different from the information gain computed in the sanitized data, due to the data distortion introduced by the anonymization process. Thus most existing techniques, including those in (LeFevre et al. 2006b), do not provide guarantee on model preservation. The only work we are aware of that preserves data mining models is to publish the contingency table for Naïve Bayesian classifiers (Mozafari et al. 2009). However, it is unclear how that method can be applied to other mining models such as decision trees.

3. Our Approach

We first define the problem. We then briefly describe the decision tree building algorithm and prove a theorem that describes the conditions under which the decision tree model can be preserved. Finally we present a method that preserves both privacy and the decision tree model.

Problem Definition: Let $T$ be a table with attributes $A_1$ to $A_m$, $K$ and $L$ be two integers, and $P$ be a decision tree built from $T$. The goal is to create a sanitized table $T'$ such that $T'$ satisfies $K$-anonymity and $L$-diversity, and at the same time $P$ can be built from $T'$.

Decision tree building algorithms: Each internal node of a decision tree contains a test condition and the branch represents the outcome of the test. E.g., from the root node of the tree in Figure 1 (a), those with capital gains less than or equal to 5095 will go to the left child and those with capital gains greater than 5095 will go to the right child. The leaf will predict a class label.

The decision tree building algorithm recursively partitions rows in table $T$. At each step, an attribute $A_i$ other than the class label is chosen to optimize a splitting criterion. If $A_i$ is numerical, a value $v$ (typically as the average of two consecutive $A_i$ values) is selected such that those records with values less than or equal to $v$ go to left child and those with values greater than $v$ go to right child. If $A_i$ is categorical, either binary split is used where $A_i$’s values will be divided into two disjoint sets, or multi-way split is used where each value of $A_i$ will become a child node.

There are three commonly used splitting criteria: information gain, gain ratio, and Gini index. Here we just describe information gain while our approach also applies to the other two. Let $S$ be the set of records at an intermediate node in $P$, $t$ be the number of child nodes, $S_j (1 \leq j \leq t)$ be the set of records in child $j$, $A_i$ be the split attribute, $v$ be the split value, and $C_1, \ldots, C_q$ be the $q$ classes. Let $f(C_i, S)$ be the frequency of class $C_i$ in $S$. The information gain equals

$$InfoGain(S, v) = \sum_{i=1}^{q} \frac{f(C_i, S)}{|S|} \log_2 \frac{|S|}{f(C_i, S)} - \sum_{j=1}^{t} \frac{|S_j|}{|S|} \sum_{i=1}^{q} \left( \frac{f(C_i, S_j)}{|S_j|} \log_2 \frac{|S_j|}{f(C_i, S_j)} \right)$$

(1)

Conditions for preserving decision trees: The first sum in Equation (1) is constant for all splits, thus the splitting criterion only depends on $|S_i|$ and $f(C_i, S_i)$, i.e., the size of each child node and the distribution of class labels in each child node. This property also holds for gain ratio and Gini index.

Theorem 1: If the privacy protection algorithm satisfies the following 3 conditions, the decision tree generated from the sanitized data will be the same as that generated from the original data.

1. It leaves the class labels unchanged.
2. Let $A_i$ be a categorical attribute that appears in the tree.
   a. If multi-way split is used, $A_i$ cannot be generalized because each value of $A_i$ forms a new branch.
   b. If two-way split is used, let $VS_1$ and $VS_2$ be the sets of $A_i$ values in the child nodes $S_1$ and $S_2$. Let $VS'$ be the set of values (including $v$) that will be generalized to the same value $v'$. $v'$ must belong to the same branch as $v$. That is, if $v \in VS_1$ (or $VS_2$), then $VS' \subseteq VS_1$ (or $VS_2$).
3. Let $A_i$ be a numerical attribute that appears in the tree. Both of the following two conditions need to be satisfied.
a. The order of values of $A_i$ is preserved, i.e., if $v_j \leq v_2$ in original data, $v_1' \leq v_2'$ in the sanitized data where $v_j (j=1,2)$ is a value of $A_i$ and $v_i'$ is the sanitized value of $v_i$.

b. Let $v$ be a split value and $v_1 (v_2)$ be the maximal (minimal) value of $A_i$ in the left (right) child after the split, respectively. The sanitized values $v_1'$ and $v_2'$ must satisfy that $v_1 + v_2 = v_1' + v_2'$. This will preserve split value $v$ because $v = (v_1 + v_2)/2$.

**Proof:** These conditions ensure that both the child record sets ($S_i$) and distribution of class labels in each $S_i$ for all possible splits remain unchanged in the sanitized data. This ensures that the tree model will be preserved. Due to space limits, we will just explain condition 3 (a).

Condition 3 (a) ensures that the order for a numerical split attribute $A_i$ is preserved. The decision tree building algorithm will check all possible splits on $A_i$, thus preserving the order of $A_i$ will ensure that the same set of child record sets ($S_i$) will be generated. Since condition 1 also preserves the class labels, the distribution of class labels in $S_i$ remains unchanged. Thus the decision tree algorithm will select the same best split as in the original data.

For example, suppose there are six records $P_1, \ldots, P_6$. $P_1, P_2, P_3, P_4$ are in class $C_1$ and $P_5, P_6$ are in class $C_2$. $A_i$ be “capital gains”. Suppose that in the original data, the order of capital gains is $P_1, P_2, P_3, P_4, P_5, P_6$. The class labels in the order of capital gains are $C_1, C_2, C_1, C_2, C_1, C_2$. The best split in the original data generates two $S_i$: $\{P_1(C_2), P_2(C_2), P_3(C_1)\}$ and $\{P_4(C_1), P_5(C_1), P_6(C_2)\}$. If the order on capital gains is preserved in the sanitized data, the class label in the order of capital gains is still $C_1, C_2, C_1, C_2, C_1, C_2$. Thus the best split will remain unchanged (between $P_3$ and $P_4$). However, suppose the order of capital gains in sanitized data is changed to $P_1, P_3, P_5, P_6, P_2$, and $P_4$, the class label in the order of capital gains becomes $C_1, C_1, C_2, C_2, C_1, C_2$. The best split in the sanitized data becomes $\{P_1(C_1), P_3(C_1)\}$ and $\{P_2(C_2), P_4(C_2), P_5(C_1), P_6(C_2)\}$, which is different from the best split in the original data.

**Proposed Method:** Next we describe the Decision Tree Preserving Algorithm.

**DTP Algorithm:** Its input includes original data $T$, a decision tree $P$, privacy parameters $K$ and $L$.

1. For each attribute $A_i$ that is not the class label and is not used in $P$, replace its value with a single value (for categorical, use ALL; for numerical, use mean of $A_i$).

2. For numerical attribute $A_i$ that appears in the tree, do the following:
   a) For each node $x$ in $P$ that uses $A_i$ as split attribute, collect boundary values as the maximal $A_i$ value in the left child and the minimal $A_i$ value in the right child.
   b) Sort values of $A_i$ and divide them into intervals using boundary values collected in step 2a) and compute the mean of each interval, let them be $\mu_1, \mu_2, \ldots$
   c) For each node $x$ in $P$ that uses $A_i$ as split attribute, let $v_1 (v_2)$ be the maximal (minimal) $A_i$ value in the left (right) child of $x$. Let $I_1 (I_2)$ be the intervals with $v_1 (v_2)$ as the right (left) boundary. Compute $\delta = \min\{|v_1 - \mu_1, \mu_2 - v_2\|$. Replace values in $I_1$ with $v_1 - \delta$, and values in $I_2$ with $v_2 + \delta$.

3. For each categorical attribute $A_i$ that appears in the tree, if two-way split is used, divide values of $A_i$ into groups such that the values in the same group appear in the same branches in $P$ (this can be done by sorting values on the branches they appear). Replace values of $A_i$ in the same group with the same generalized value.

4. Group all records on quasi-identifier attributes. For each group, check whether it satisfies $K$-anonymity and $L$-diversity. If so, the sanitized data is returned. Otherwise, the users can call DTP with a smaller tree until the privacy requirement is satisfied.

We have proved that DTP satisfies all conditions in Theorem 1 and thus preserves the decision trees. Figure 2 shows how it works for the example of records $P_1$ to $P_6$. The best split in original data is between $P_3$ and $P_4$. Thus step 2a) will pick the boundary values $P_3$ and $P_4$ as boundaries (let them be $v_1$ and $v_2$, respectively). In step 2b), two intervals get created: $I_1$ containing $P_1$ to $P_3$ and $I_2$ containing $P_4$ to $P_6$. The mean of each interval is also computed. Step 2c) computes the gap between $v_1 (v_2)$ and the mean of $I_1 (I_2)$. Let $\delta$ be the smaller of these two gaps. It then generalizes values in $I_1$ to $v_1 - \delta$, and values in $I_2$ to $v_2 + \delta$. 

64

19th Workshop on Information Technologies and Systems
Clearly, the new split value in the sanitized data \( (v_1-\delta + v_2+\delta)/2 \) is the same as the old split value \( (v_1+v_2)/2 \). The order is also preserved because \( v_1 \leq v_2 \) and \( v_1-\delta \leq v_2+\delta \).

**Privacy Protection:** Step 4 will ensure that the sanitized data satisfies K-anonymity and L-diversity. One possible attack is when the attacker knows the order of a numerical attribute (e.g., knowing \( P_1 \) has the smallest capital gains). DTP preserves the order for such attributes. However, all values in the same interval are generalized to the same value (e.g., \( P_1 \) to \( P_3 \) all have the same capital gains after sanitization). Thus the attacker can only locate the group of rows with smallest capital gains, but cannot decide which one in the group is \( P_1 \). Since there are at least \( K \) people in each group, the probability of identifying \( P_1 \) is at most \( 1/K \).

**Complexity Analysis:** Let \( n \) be the number of rows and \( m \) be the number of attributes. The cost of DTP is \( O(mn\log n) \). Detailed analysis is omitted.

### 4. Experimental Evaluation

**Data:** We used the Adult data (Hettich et al. 1998) because it has become the de facto benchmark in the literature. It contained 30,717 records (rows with missing values are excluded as in the literature), and 5 numerical attributes and 7 categorical attributes. We used “occupation” as the sensitive attribute and the rest as quasi-identifiers. Our method was implemented in R. The experiment was run on a desktop PC with 3.2G Hz CPU and 2 GB RAM, running Windows XP.

**Methods:** The decision tree is built to predict whether the income is over 50K. We compare our method (DTP) to the InfoGain method in (LeFevre et al. 2006b) because it has the best prediction accuracy among existing methods. InfoGain partitions data into groups such that information gain is maximized. It then generalizes quasi-identifier attributes in each group. It does not satisfy Condition 3 in Theorem 1 (preserving order for numerical attributes), thus it does not preserve decision trees.

**Metrics:** We generated 10 pairs of testing and training data sets as 10-fold cross-validation. We assumed that the training data would be published, and other researchers could apply the published decision tree to the test data. For each pair of data sets, the training data was sanitized and a decision was built from it. This tree was then applied to the test data and accuracy was computed. Finally the accuracy of all testing sets was averaged. We used K-anonymity and L-diversity (averaged over 10 sanitized training sets) to measure the degree of privacy protection. Larger \( K \) and \( L \) mean more protection. We also built a Naïve Bayesian classifier on the sanitized data and computed its accuracy.

**Results:** The execution time of our method was less than 12 seconds in all experiments. Figure 3 reports the accuracy of decision trees built from sanitized data. We varied the sizes of trees (as the number of leaf nodes). The accuracy for the tree built from the original data is also reported as the baseline. The tree using DTP has higher accuracy than the tree using InfoGain. More importantly, DTP *always* preserves the decision tree model while InfoGain *never* preserves the model in all experiments. The accuracy using data sanitized by DTP is the same as that using original data because the decision trees are the same. Figure 4 reports K-anonymity results. \( K \) decreases as the tree becomes larger because as the tree grows, more intervals will be generated by DTP and the degree of generalization becomes less. The \( K \) values for DTP are slightly worse than those of InforGain for trees with 4 or 5 leaves, because DTP preserves the tree model and thus does less generalization. This is the price we pay for preserving mining models. In terms of L-diversity, the “occupation” attribute is not used in the decision tree and is thus suppressed by both methods. This is the best a privacy protection method can do. The best strategy for the attackers is to...
assume that the “occupation” always has the most frequent value. We use the strong form of \( L \)-diversity where the fraction of the most frequent values in each equivalence class must be less than \( 1/L \) (Machanavajjhala et al. 2007). In all experiments, the values of \( L \) for both methods are quite close and are in the range of 2 to 3.

When one applies our technique, he can decide the size of tree based on the tradeoff between prediction accuracy and the degree of privacy protection. E.g., for this data set the tree with 4 leaves seems to give the best tradeoff. Smaller tree is usually preferred because it is less likely to overfit the data. It also leads to better privacy protection. Thus a rule of thumb is to use the smallest tree that provides sufficient accuracy.

Figure 5 reports the accuracy of Naïve Bayesian classification using sanitized data as training data. The accuracy of using data sanitized by DTP is better than that of InfoGain, and is close to the accuracy using the original data. Naïve Bayesian (as well as other mining techniques) relies on the correlation between other attributes and class labels to predict class labels. DTP preserves the decision tree model, which already captures such correlations.

8. Conclusion

This paper proposes a privacy protection technique that both preserves decision tree models and protects privacy. This allows users to verify the published models. Further, experimental results suggest that users can also try other mining techniques on sanitized data and obtain good results. This technique can reduce both research fraud and the cost of collecting research data.

References

Towards Designing Ranking Systems for Hotels on Travel Search Engines: Combining Text Mining and Image Classification with Econometrics

Anindya Ghose, Panagiotis G. Ipeirotis, Beibei Li
Department of Information, Operations, and Management Sciences
Stern School of Business, New York University
{aghose, panos, bli}@stern.nyu.edu

Abstract

In this paper, we empirically estimate the economic value of different hotel characteristics, especially the location-based and service-based characteristics given the associated local infrastructure. We build a random coefficients-based structural model taking into consideration the multiple-levels of consumer heterogeneity introduced by different travel contexts and different hotel characteristics. We estimate this econometric model with a unique dataset of hotel reservations located in the US over 3 months and user-generated content data that was processed based on techniques from text mining, image classification, and on-demand annotations. This enables us to infer the economic significance of various hotel characteristics. We then propose to design a new hotel ranking system based on the empirical estimates that take into account the multi-dimensional preferences of customers and imputes consumer surplus from transactions for a given hotel. By doing so, we are able to provide customers with the “best value for money” hotels. Based on blind tests of users from Amazon Mechanical Turk, we test our ranking system with some benchmark hotel ranking systems. We find that our system performs significantly better than existing ones. This suggests that our inter-disciplinary approach has the potential to improve the quality of hotel search.

Keywords: Structural modeling, text mining, hotel search engine, user-generated content, ecommerce.

1. Introduction

It is now widely acknowledged that local search for hotel accommodations is a component of general Web searches that is increasing in popularity as more and more users search and reserve their trips online. Online travel search engines provide only rudimentary ranking facilities, typically using a single ranking criterion such as distance from the city center, star ratings, price per night, etc. This approach has quite a few shortcomings. First, it ignores the multidimensional preferences of the consumer in that a customer’s ideal choice may consist of several hotel-specific attributes. Second, it largely ignores characteristics related to the location of the hotel, for instance, in terms of proximity to a “beach” or proximity to a “downtown shopping area”. These location-based features represent important characteristics that can influence the desirability of a particular hotel. In this paper, using demand estimation techniques, we propose to estimate the weight that consumers place on different internal (service) and external (locational) characteristics of hotels. Thereafter, based on the estimation of consumer surplus we compute the “best value for money” of a particular hotel. The eventual outcome of our analysis is to design a new ranking system for hotels based on this concept. Such a ranking can be displayed in response to a user query on hotel search engines.

We undertake this study in the context of demand for hotel rooms using a unique dataset consisting of actual transactions and different kinds of user-generated content such as product reviews describing hotel service characteristics as well social geo-tags and on-demand annotations describing location characteristics. The theory that product reviews affect product sales has received support in prior empirical studies (for example, Chevalier and Mayzlin 2006). However, these studies have only used the numeric review ratings (e.g., the valence and the volume of reviews) in their empirical analysis. An emerging stream of work has begun to examine whether the textual information embedded in online user-generated content can have an economic impact (Ghose et al 2006, Das & Chen 2007, Archak et al. 2008, Ghose and Ipeirotis 2008, Ghose 2009). But these studies do not focus on estimating the impact of reviews in influencing real transactions nor do they aim to design new IT systems based on economic variables. Hence, another research objective in this
paper is to analyze the extent to which user-generated content can help us learn consumer preferences for different hotel attributes.

Our work involves three stages. First, we need to identify the important hotel characteristics, both internal (service) and external (locational) that influence demand. Second, we need to empirically quantify the extent to which these characteristics influence demand. Finally, we aim to improve local search for hotels by incorporating the economic impact of these characteristics on consumer surplus from hotel transactions and designing a ranking system that incorporates “value for money” as a criterion for ranking. Any successful attempt to address these issues needs to answer the following questions: How can we automatically extract information about hotel attributes captured in textual content of product reviews and social tags, and visual content of satellite images? How can we incorporate extracted variables in a structural demand estimation model so as to be able to precisely identify parameter estimates?

A key challenge is in bridging the gap between the essentially textual and qualitative nature of review and image content and the quantitative nature of structural demand estimation models. With the rapid growth and popularity of user-generated content on the Web, a new area of research applying text mining techniques product reviews has emerged (for example, Hu & Liu 2004, Pang & Lee 2004, Das & Chen 2007). Similarly, advances in image classification have been made using non-parametric classifiers such as decision trees and support vector machines (Lu and Weng 2007). We use techniques from both these streams of work in finalizing our dataset (see below).

2. Data Description

We have complete information on all transactions conducted over a 3 month period from 2008/11–2009/1 for 2117 hotels in the US via Travelocity. These hotels were randomly selected by Travelocity. Further, we have data on hotel attributes from three sources: (i) service descriptions based on mining users’ hotel reviews from Travelocity, (ii) location descriptions based on social geo-tags identifying different “external amenities” (such as shopping malls, restaurants, tourist attractions, etc) from Geonames.org, and (iii) user-contributed opinions on important hotel characteristics from Amazon Mechanical Turk such as whether a hotel is located “near the interstate highway”, “near public transportation”, etc. Since some location-based characteristics, such as “proximity to the beach” and “distance from downtown”, are not directly measurable based on reviews, tags or opinions, we use image classification techniques to infer such features from the satellite images of the area. We extracted hybrid satellite images (sized 256 × 256 pixels) using the Visual Earth Tile System, for each of the hotel venues, with 4 different zoom levels for each. These images were then used to extract information about the surroundings of the hotels. We performed a study to examine the performance of the classifiers. To perform the classification, we classified the out-of-sample images using Amazon Mechanical Turk; our results show that our SVM classifier has an accuracy of 91.2% for “Beach” image classification and 80.7% for “Downtown” image classification.

With regard to the service-based hotel characteristics, we extracted them from the website of TripAdvisor using fully automated JavaScript parsing engines. Since hotel amenities are not directly listed on TripAdvisor website, we retrieved them by following the link provided on the hotel web page, which randomly directs to one of its cooperative partner websites (i.e., Travelocity, Orbitz, Expedia, etc.). We looked into two text-style features: “subjectivity” and “readability” of reviews. In order to capture more objectively the review text-style, we used a multiple-item method for subjectivity and readability. In order to decide the probability of subjectivity for review text, we trained a classifier using as “objective” documents the hotel descriptions of each of the hotels in our data set. We randomly retrieved 1000 reviews to construct the “subjective” examples in the training set. We conducted the training process by using a 4-gram Dynamic Language Model classifier provided by the LingPipe toolkit. Thus, we were able to acquire a subjectivity confidence score for each sentence in a review, thereby deriving the mean and standard deviation of this score, which represent the probability of the review being subjective. Finally, previous research suggested that the prevalence of reviewer disclosure of identity information is associated with changes in subsequent online product sales (Forman et al. 2008). Therefore, we decide to include one particular characteristic capturing the level of reviewers’ disclosure of their identity information on these websites – “real name or location.” These data sources were merged to create the main dataset (Table 1).
### Table 1. Summary of Different Sources for Extracting Hotel Characteristics

<table>
<thead>
<tr>
<th>Category</th>
<th>Variables - Hotel Characteristics</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transaction Data</td>
<td>● Transaction Price (per room per night)</td>
<td>Travelocity</td>
</tr>
<tr>
<td>Service-based</td>
<td>● Hotel Class</td>
<td>TripAdvisor</td>
</tr>
<tr>
<td></td>
<td>● Internal Amenities (“ice machine,” “pets allowed,” “fitness center,” “free breakfast”, “wheelchair accessible”, etc)</td>
<td></td>
</tr>
<tr>
<td>Review-based</td>
<td>● Number of Customer Reviews</td>
<td>Travelocity and TripAdvisor</td>
</tr>
<tr>
<td></td>
<td>● Overall Reviewer Rating</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Disclosure of Reviewer Identity Information</td>
<td></td>
</tr>
<tr>
<td>Subjectivity</td>
<td>● Mean Probability</td>
<td>Text Mining Analysis</td>
</tr>
<tr>
<td></td>
<td>● Std. Dev. of Probability</td>
<td></td>
</tr>
<tr>
<td>Readability</td>
<td>● Number of Characters</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Number of Syllables</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Number of Spelling Errors</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Average Length of Sentence</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● SMOG Readability Index</td>
<td></td>
</tr>
<tr>
<td>Location-based</td>
<td>● Near the Beach</td>
<td>Image Classification</td>
</tr>
<tr>
<td></td>
<td>● Near Downtown</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● External Amenities (Number of restaurants, shopping malls, historical sites, etc)</td>
<td>Geonames and Virtual Earth Interactive SDK</td>
</tr>
<tr>
<td></td>
<td>● Number of Local Competitors Within 2 miles</td>
<td>Amazon Mechanical Turk (AMT)</td>
</tr>
<tr>
<td></td>
<td>● Near the Interstate Highway</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● Near Public Transportation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>● City Annual Crime Rate</td>
<td>FBI online statistics</td>
</tr>
</tbody>
</table>

3. Model

In this section, we discuss how we develop our structural model and how we apply it to empirically estimate the distribution of consumer preferences for different hotel characteristics.

#### 3.1 Random Coefficients-Based Structural Model

We define a consumer’s decision-making behavior in the hotel market to be in accordance with the following two-step procedure. In the first step, the consumer aims to find a subset of hotels that best matches her travel context. For instance, if a consumer wants to go on a business trip, he would be more interested in a subset of hotels that specialize in business services; while if he plans to take his four-year kid for a family fun trip, he would be more likely to look for those hotels which are regarded as being kid-friendly. We have eight such unique category types in our data (Family Trip, Business Trip, Romantic Trip, Tourist Trip, Trip with Kids, Trip with Seniors, Pets Friendly and Disabilities Friendly). Then, in the second step, once the consumer has picked a corresponding subset of hotels which satisfy his travel requirement, he makes a further decision based on his evaluation of the value provided by the hotels.

Let the utility $u_{ijt}$ for consumer $i$ from choosing hotel $j$ with category type $k$ in market $t$ be as in (1):

$$u_{ijt} = X_{ijt} \beta_i - \alpha_i P_{ijt} + \xi_{ijt} + \varepsilon_{ijt}^k,$$

where, $i$ represents a consumer, $j$ represents hotel $j$ with category type $k$ ($1 \leq k \leq 7$), and $t$ represents a hotel market which in our case is defined as a “city-night” combination. In this model, $\beta_i$ and $\alpha_i$ are random coefficients that capture consumers’ heterogeneous tastes towards different observed hotel characteristics, $X$, and towards the average price per night, $P$, respectively. $\xi$ represents the set of hotel characteristics that are unobservable to the econometrician. $\varepsilon_{ijt}^k$ with a superscript $k$ represents a travel context level “taste shock”. Consistent with prior research (Berry and Pakes 2007), we assume that $\beta_i$ and $\alpha_i$ are distributed among
consumers per some known statistical distribution, i.e., \( \beta \sim (\beta, |\beta|, \sigma_\beta) \) and \( \alpha \sim (\alpha, \bar{\alpha}, \sigma_\alpha) \). Our goal is then to estimate the means \((\beta, \alpha)\) and the standard deviations \((\sigma_\beta, \sigma_\alpha)\) of these two distributions. The means correspond to the set of coefficients on hotel characteristics and on hotel price, which measure the average weight placed by consumers; while the standard deviations provide a measure of the consumer heterogeneity in those weights. Furthermore, these heterogeneities result from some particular demographic attributes of consumers. Hence, we assume that \( \sigma_\alpha \sim \alpha, I \), where \( I \) represents the income whose distribution can be inferred from consumer demographics; \( \sigma_\beta \sim \beta, V \), where \( V \sim N(0,1) \) represents some random factor that will influence people’s preferences towards individual hotel characteristics. Therefore, we rewrite our model in the following form:

\[
u_{it} = \delta_{kt} + X_{ij} \beta_i v_i - \alpha_i I_{ij} P_{kj} + \epsilon_{it}, \tag{2}\]

where \( \delta_{kt} = X_{ij} \bar{\alpha} - \alpha_i I_{ij} P_{kj} + \epsilon_{it} \), represents the mean utility of hotel \( j \) with category type \( k \) in market \( t \). \( \beta_i \) and \( \alpha_i \) are the set of parameters to be estimated.

3.2 Estimation

Due to lack of space, we describe the estimation procedure very briefly. As mentioned before, our goal here is to estimate the mean and variance of \( \beta \) and \( \alpha \). We apply estimation methods similar to those used in Berry and Pakes (2007). This problem can be essentially reduced to a procedure of solving a system of nonlinear equations. In general, with a given starting value of \( \theta_0 = (\alpha^0, \beta^0) \), we look for the mean utility \( \delta \) such that the model predicted market share equates the observed market share. From there, we form a GMM objective function using the moment conditions such that the mean of unobserved characteristics is uncorrelated with instrumental variables. Based on this, we identify a new value of \( \theta_1 = (\alpha^1, \beta^1) \), which is used as the starting point for the next round iteration. This procedure is repeated until the algorithm finds the optimal value of \( \theta \) that minimizes the GMM objective function. To find a solution, we applied the contraction mapping method suggested by Berry et al. (1995).

4. Empirical Analysis

4.1 Results

The estimation results are shown in Table 2. An important factor in influencing demand is the textual content and style of customer reviews. We see that “Complexity”, “Syllables” and “Spelling Errors” have a negative sign. This implies that most consumers would prefer to read reviews with shorter sentences, less syllables and fewer spelling errors in total. On the other hand, variables “Characters” and “SMOG Readability index” present a positive influence. This implies that consumers appreciate longer reviews with more characters, and with a more professional writing style. For the subjectivity features, both “Mean Subjectivity” and “Subjectivity standard deviation” turn out to be negative. Therefore, consumers prefer to obtain as much objective information as possible from others’ experiences.

There are at least five location-based characteristics which have a positive impact on hotel demand: “External Amenities,” “Near Public transportation,” “Near Highway”, “Near Downtown” and “Near Beach” showing that consumers prefer to stay in hotels with these features. A few location-based characteristics have a negative impact on hotel demand. Not surprisingly, one of them is the “Annual Crime Rate.” The higher the average crime rate reported in a local area, the lower the desirability of consumers for staying in a hotel located in that area. Another factor that has a negative impact is the number of “Local Competitors” within 2 miles. These estimates imply that controlling for price and content of user reviews, the geographical and other location attributes of a hotel can make a big difference in attracting consumers.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient (Std. Err)I</th>
<th>Coefficient (Std. Err)II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Room Price Per Night</td>
<td>-0.1768*** (.0289)</td>
<td>-0.0080 (.0144)</td>
</tr>
<tr>
<td>Number of Characters in Review</td>
<td>0.0155*** (.0020)</td>
<td>0.0108*** (.0015)</td>
</tr>
</tbody>
</table>
### Table 1: Summary Statistics for Hotel and City Characteristics

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Travelocity</th>
<th>TripAdvisor</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Review Complexity</strong></td>
<td>-0.0121***</td>
<td>-0.0070***</td>
</tr>
<tr>
<td><strong>Number of Syllables in Review</strong></td>
<td>-0.0482***</td>
<td>-0.0331***</td>
</tr>
<tr>
<td><strong>SMOG Readability Index</strong></td>
<td>0.1137***</td>
<td>0.0650***</td>
</tr>
<tr>
<td><strong>Number of Spelling Errors in Review</strong></td>
<td>-0.1575***</td>
<td>-0.1250***</td>
</tr>
<tr>
<td><strong>Mean Subjectivity of Review</strong></td>
<td>-0.8268***</td>
<td>-0.22265†</td>
</tr>
<tr>
<td><strong>Subjectivity Deviation of Review</strong></td>
<td>-0.2298**</td>
<td>-0.2221***</td>
</tr>
<tr>
<td><strong>Hotel Class</strong></td>
<td>0.0421***</td>
<td>0.0049***</td>
</tr>
<tr>
<td><strong>Number of Competitors of Hotel</strong></td>
<td>-0.0853***</td>
<td>-0.1435***</td>
</tr>
<tr>
<td><strong>City Annual Crime Rate</strong></td>
<td>-0.1523***</td>
<td>-0.0598***</td>
</tr>
<tr>
<td><strong>Number of Internal Amenities</strong></td>
<td>0.0022 (.0020)</td>
<td>0.0023 (.0010)</td>
</tr>
<tr>
<td><strong>Number of External Amenities</strong></td>
<td>0.0066***</td>
<td>0.0052***</td>
</tr>
<tr>
<td><strong>Near Beach or Not</strong></td>
<td>0.0693**</td>
<td>0.1035***</td>
</tr>
<tr>
<td><strong>Near Public Transportation or Not</strong></td>
<td>0.01495†</td>
<td>0.00003**</td>
</tr>
<tr>
<td><strong>Near Interstate Highway or Not</strong></td>
<td>0.1332***</td>
<td>0.0848***</td>
</tr>
<tr>
<td><strong>Near Downtown or Not</strong></td>
<td>0.0275 (.0287)</td>
<td>0.0713***</td>
</tr>
</tbody>
</table>

***, **, *, and † denote significance at 0.1%, 1%, 5% and 10% levels, respectively. Control variables include volume and valence of reviews and whether reviewer disclosed his identity or not.

The above results are based on the dataset of hotels from Travelocity, which may or may not have online customer reviews on its website. As a robustness check, we also collected reviews from a third party site - TripAdvisor, which is regarded as the world’s largest online travel search engine. We therefore narrowed down the sample to consist of those hotels that have at least one review from either Travelocity or TripAdvisor. The estimation results from this filtered dataset (II) are shown in column 2 of Table 2. Further, we conducted the similar estimations after incorporating the textual content of reviews from TripAdvisor. All the results were qualitatively very consistent with our findings above.

### 4.2 Consumer Surplus-Based Hotel Ranking

After we have estimated the parameters in the model, we can derive the consumer surplus from our model. The mean utility provides us a good estimation of how much consumers in general can benefit from choosing this particular hotel, and the standard deviation of utility describes the variance of this benefits from different consumers. In our case, we are interested to know what the excess utility, or consumer surplus, is for consumers on an aggregate level to choose a certain hotel. We thereby propose a new ranking approach for hotels based on the consumer surplus of each hotel for consumers on an aggregate level. This ranking idea is based on how much “extra value” consumers can obtain after paying for that hotel, which is what consumers really care about. If a hotel provides a comparably higher surplus for consumers on an aggregate level, then it should appear on the top of our ranking list for that city.

### 4.3 Evaluation With User Study

To evaluate the quality of our ranking technique, we conducted a user study using Amazon Mechanical Turk (AMT). First, we generated different rankings for the top-20 hotels, in various areas, according to a set of baseline criteria: price low to high, price high to low, maximum online review count, hotel class, hotel size (number of rooms), number of internal amenities, and popularity rank (generated by TripAdvisor). We then computed the consumer surplus for each hotel, and ranked the hotels in each city according to their surplus. Then, we performed blind tests, presenting various lists to 100 anonymous AMT users and asking them which ranking list they prefer. Further, we asked users to compare pairs of lists and tell us which of the hotel ranking lists they prefer the most. We tested the results for a few large cities like New York, Chicago, Dallas, Atlanta, Los Angeles, San Francisco and Washington DC. The results were highly encouraging. For example, in New York city, more than 80% of the customers preferred our ranking when listed side-by-side with the other, competing baseline techniques (p = 0.001, sign test).
We also asked consumers why they chose a particular ranking, to understand better how users interpret the surplus-based ranking. In our NYC experiment, the majority of the users indicated that they preferred the diversity of the returned results given that the list consisted of a mix of hotels cutting across several price ranges. In contrast, the other ranking approaches tend to list hotels of only one type (e.g., very expensive hotels). We found that a ranking system generated with “value for the money” returns a better variety of hotels, covering 30% 5-star, 40% 4-star, and 30% 3-star hotels in a given city. It generally starts out with lower class hotels and increases to 5-star hotels, providing a logical way to present the information on the screen which will help customers in their decision-making procedure. Based on the qualitative opinions of the users, it appears that diversity in hotel choices is indeed an important factor that improves the satisfaction of consumers, and an economic approach for ranking introduces diversity naturally. This result seems intuitive: if a specific segment of the market systematically appeared to be underpriced, then market forces would move the prices for the whole segment accordingly. However, this effect may be less pronounced with individual hotels, especially under a personalized consumer surplus calculation.

5. Conclusion

To summarize we show information about hotel characteristics captured from different sources of data can be incorporated in a demand estimation model to empirically estimate the economic value of different hotel characteristics, including both service based and location-based characteristics. Our research allows us to not only quantify the economic impact of hotel characteristics, but also by reversing the logic of this analysis, allows us to identify the characteristics that most influence the demand for a particular hotel. After inferring the economic significance of each characteristic, we incorporate the economic value of hotels characteristics into a local ranking function based on estimation of consumer surplus from transactions of that hotel. The key idea is that hotels that provide consumers with a higher surplus would be placed higher on the screen in response to consumer queries. We then conduct blind tests using users from AMT to examine how well our ranking system performs and find that our system performs significantly better than existing benchmark ones. We are currently working on extending our system into a more personalized ranking system that incorporates surplus for each individual consumer of a hotel and then displays hotel recommendations in a personalized manner for each consumer. We hope that our inter-disciplinary methods and approach can improve the quality of results displayed for hotel search engines on the Internet.

References