Mining Customers’ Spatio-temporal Behavior Data using Topographic Unsupervised Learning

Guénaël Cabanes¹, Younès Bennani¹ and Frédéric Dufau-Joël²

¹ LIPN-CNRS, UMR 7030
99 Avenue J-B. Clément, 93430 Villetaneuse, France

² La Boite à Outils
2 Rue Raymond Pitet, 38000 Grenoble, France

Abstract

Radio Frequency IDentification (RFID) is an advanced tracking technology that can be used to study the spatio-temporal behavior of customers in a supermarket. The aim of this work is to build a new RFID-based autonomous system to follow individuals’ spatio-temporal activity, a tool not currently available, and to develop new methods for automatic data mining. Here, we study how to transform these data to investigate the customers’ behaviors. We propose a new unsupervised data mining method to deal with this complex and very noisy data. This method is fast, efficient and allows some useful analysis to understand how the customers behave during shopping.

1 Introduction

Radio Frequency IDentification (RFID) is an advanced tracking technology. The RFID tags, which consist of a microchip and an antenna, must be used with a reader that can detect simultaneously a lot of tags in a single scan. A computer has to be used to store the data about the position of each tag for each scan in a database. This allows different analyses. RFID, thanks to miniaturization, offers the advantage of automation and overcomes the constraints imposed by video analyzes.

In this paper we aim at studying the individuals’ spatio-temporal activity during their shopping in a supermarket. Until now, little research has been undertaken in this way. Usual questions are: how do customers really travel through the store, do they go through every area or do they skip from one area to another in a more direct manner, do they follow a single, dominant pattern, or are they rather heterogeneous? Because such dynamic data are difficult to collect, this kind of study implies that adequate observation tools are available; hence traceability technologies (RFID), which allow to automate the monitoring of individuals localization and movements simultaneously, are greatly needed. Those systems are now well developed and they are almost operational as well. For this work, we used a RFID device based on marketed products. It consists of a RFID readers network connected to a detector which sends the information to a computer. These readers detect the movements of RFID tags glued on the bottom of baskets at the disposal of the customers.

The analysis of the collected data uses modern methods of data mining, based on a new unsupervised algorithm. Unsupervised methods are highly suited when we do not have prior knowledge about the data. This is particularly true for data from experimental studies, for which we have generally little prior information.

The remainder of this paper is organized as follows. Section 2 presents the new algorithm. Section 3 describes the experimental protocol and some results. Conclusion and future works are given in section 4.

2 A fast density-based algorithm to model data structure

The basic assumption in this work is that data are described as vectors of numeric attributes and that the datasets to compare have the same type. First, each dataset is modeled using an enriched Self-organizing Map (SOM) model (adapted from [1]), constructing an
abstract representation which is supposed to capture the essential data structure. Then, the density function of each dataset is estimated from the abstract representation. Finally, different datasets are compared using a dissimilarity measure based upon the density functions.

The idea is to combine the dimension reduction and the fast learning SOM capabilities in the first level to construct a new reduced vector space, then apply other analysis in this new space. These are called two-levels methods. The two-levels methods are known to reduce greatly the computational time, the effects of noise and the “curse of dimensionality” [1]. Furthermore, it allows some visual interpretation of the result using the two-dimensional map generated by the SOM.

2.1 Algorithm schema

The algorithm proceeds in three steps:

1. The first step is the learning of the enriched SOM. During the learning, each prototype of the SOM is extended with novel information extracted from the data. These structural informations will be used in the second step to infer the density function. More specifically, the attributes added to each prototype are:

   - **Density modes.** It is a measure of the data density surrounding the prototype (local density). The local density is a measure of the amount of data present in an area of the input space. We use a Gaussian kernel estimator [12] for this task.

   - **Local variability.** It is a measure of the variability of data that are represented by the prototype. This variability can be defined as the average distance between the prototypes and the represented data.

   - **The neighborhood.** This is a prototype’s neighborhood measure. The neighborhood value of two prototypes is the number of data for which they are the two best match prototypes.

At the end of this process, each prototype is associated with a density and a variability value, and each pair of prototypes is associated with a neighborhood value. The substantial information about the structure of the data is captured by these values. Then, it is no longer necessary to keep data in memory.

2. The second step is the construction, from each enriched SOM, of a density function which will be used to estimate the input space density. This function is constructed by induction from the information associated to the prototypes of the SOM, and is represented as a mixture model of spherical normal kernel \( \{ K_i \}_{i=1}^{M} \), where \( K_i \) is a Gaussian function centered on a prototype \( w_i \) and \( M \) the number of prototype. The density function can therefore be written as:

\[
    f(x) = \sum_{i=1}^{M} \alpha_i K_i(x)
\]

where

\[
    K_i(x) = \frac{1}{\sqrt{2\pi} h_i} e^{-\frac{|x - w_i|^2}{2h_i^2}}
\]

The most popular method to fit mixture models (i.e. to find \( h_i \) and \( \alpha_i \)) is the expectation-maximization (EM) algorithm [4]. However, this algorithm needs to work in the data input space. As here we work on enriched SOM instead of dataset, we can’t use EM algorithm.

Thus, we propose the heuristic to choose \( h_i \) :

\[
    h_i = \frac{\sum_{j=1}^{N_i} \sqrt{N_i + N_j} (s_i N_i + d_{i,j} N_j)}{\sum_{j} v_{i,j}}
\]

Where \( N_i \) is the number of data point represented by prototype \( w_i \), \( s_i \) is the variability of \( w_i \) and \( d_{i,j} \) is the euclidean distance between \( w_i \) and \( w_j \). The idea is that \( h_i \) is the standard deviation of data represented by \( K_i \). These data are also represented by \( w_i \) and their neighbors. Then \( h_i \) depends on the variability \( s_i \) computed for \( w_i \) and the distance \( d_{i,j} \) between \( w_i \) and his neighbors, weighted by the number of data represented by each prototype and the connectivity value between \( w_i \) and his neighborhood.

Now, since the density \( D \) for each prototype \( w \) is known \((f(w_i) = D_i)\), we can use a gradient descent method to determine the weights \( \alpha_i \). The \( \alpha_i \) are initialized with the values of \( D_i \), then these values are reduced gradually to better fit \( D = \sum_{i=1}^{M} \alpha_i K_i(w) \). To do this, we optimize the following criterion:

\[
    \alpha = \arg\min_{\alpha} \frac{1}{M} \sum_{i=1}^{M} \left[ \sum_{j=1}^{M} (\alpha_j K_j(w_i)) - D_i \right]^2
\]

Thus, we now have a density function that is a model of the dataset represented by the enriched SOM. Some example of estimated density functions can be seen in Fig. 1 and 2.
2.2 Algorithm complexity

The complexity of the algorithm is scaled as $O(T \times M)$, with $T$ the number of step and $M$ the number of prototypes in the SOM. It is recommended to set up at least $T > 10 \times M$ for a good convergence of the SOM [7]. In this study we use $T = \max(N, 50 \times M)$ as in [13], with $N$ the number of data. This means that if $N > 50 \times M$ (large database), the complexity of the algorithm is $O(N \times M)$, i.e, it is linear in $N$ for a fixed size of the SOM. Then the whole process is very fast and is suited for the treatment of large databases. Also very large databases can be handled by fixing $T < N$ (this is similar as working on a random subsample of the database).

This is much faster than traditional density estimator algorithms as the Kernel estimator [12] (that also needs to keep all data in memory) or the Gaussian Mixture Model [5] estimated with the EM algorithm (as the convergence speed can become extraordinarily slow [11, 9]).

3 Analysis of customer’s displacements inside a supermarket

The purpose of this work is to explore data recorded via a RFID device to model and analyze the purchasing behavior of customers. In particular, we would know the time spent in each area of the store for detecting hot spots and cold spots. We also aim to analyze the customers trajectory patterns.

3.1 Measurement Device

The movement of customers during their shopping was monitored using RFID device\(^1\). To do this, some plastic baskets are at the disposal of the customers. Each basket have a RFID tag glued on it (Fig. 3).

The supermarket used for this experiment is a store specializing in the sale of decorative objects (6000m\(^2\)), with an average of 2500 visits each month. A RFID device with 4 readers was installed in the different area of the store to analyze the movement of customers. This will allow us to have a global view of the customers in the store and time spent in different area. The first tests carried out on site confirm that the positioning of readers is very delicate and only allow testing to find the optimum location.

---

\(^1\) Made by SpaceCode: http://www.spacecode-rfid.com/
Fig. 4 shows the final readers position inside the shop. Note that the readers numbers (1, 4, 9 and 10) represent the last number in their IP address. The information recorded by readers are handled by a RFID electronics and then sent to a computer which creates and stores the data files.

### 3.2 Data

The data indicate, for each scan (about one scan per second), the ID number of the tag detected, the IP address of the reader that have detected the tag and the date and time of the detection (Fig. 5). If, during a scan, none is detected, nothing appears in the data file.

3.2 Data postprocessing

As a customer moves inside the store, he is detected successively by different readers. However, depending on the crossed area, one tag can be detected by more than one reader, approximatively at the same time. This adds much more information about the actual location of the customer, but this also makes the moving sequence much harder to understand. Furthermore, data recorded is very noisy, because of RFID signal perturbations due to metallic structures and human bodies (see Fig. 6 for an example). Then, these trajectories can’t be handle by traditional sequence mining algorithms, which are unable to deal with strong noise. As our algorithm use prototype-based models, it is very efficient to overcome the problem of noisy data (see [2]).

In this work, we want to analyze the variation of the customers’ spatial behavior over time. To do that, a current location must be defined. The current location represents the area where the customer is at a given time. This location must be inferred from the complex and noisy RFID signal. To do this, we compute a vector representing how many times and how long each RFID reader has detected the customer’s tag during a 3 minutes time window centered on the current time, this window is calculated each 10 seconds of the customer’s trajectory.

Obviously, this definition implies some correlations between the description of two current locations if they are separated by less than 3 minutes, as the two related windows overlap. This will allow us to detect when a customer moves from one area to another, by detecting sudden changes in the description of the current location.

3.4 Detection of individual homogeneous behavioral sub-sequences

In order to regroup similar current location and to detect changes over time, we applied the algorithm DS2L-SOM [1] on time windows from each individual sequences. DS2L-SOM first compute an enriched SOM (as in §2.1) from the set of location windows descriptions, then use the density information to detect sudden change in the location windows, so as to regroup similar windows. Figure 7 gives an example of the results obtained with DS2L-SOM for one customer’s trajectory.

In this example, the customers start to stay in a location detected by reader 1 and sometimes by reader 4. This homogeneous structure was automatically found by DS2L-SOM, all time windows corresponding to this location are regrouped into one cluster (called A by the algorithm), we will call it “location A”. After about 5 minutes, the customer moves to another location (called B) and so on...
Thus, the proposed method is a powerful tool for detecting homogeneous behavioral sub-sequence inside a spatio-temporal RFID monitoring.

### 3.5 Detection of similar sub-sequences at collective level

The method used in §3.4 allows us to analyze efficiently the behavior of each individual. Anyway, we now need a method to compare all these individual sequences so as to perform an analysis at the collective level.

The idea is to define a similarity measure between two set of prototypes (from the enriched SOM) that represent two individual behavioral sub-sequences (i.e. clusters). For this job, we compute for each cluster the related density function as in §2.1.

We can now define a measure of dissimilarity between two sub-sequences $K$ and $L$, represented by two set of SOM’s prototypes: $P_K = \{w^K_i\}_{i=1}^{M_K}, f^K$ and $P_L = \{w^L_i\}_{i=1}^{M_L}, f^L$. With $M_K$ and $M_L$ the number of prototypes $w$ representing $K$ and $L$, and $f^K$ and $f^L$ the density function of $K$ and $L$.

The dissimilarity between $K$ and $L$ is given by:

$$D_{\text{sim}}(K, L) = \frac{\sum_{i=1}^{M_K} f^K(w^K_i) \log \left( \frac{f^K(w^K_i)}{f^K(w^K_i)} \right)}{M_K} + \frac{\sum_{j=1}^{M_L} f^L(w^L_j) \log \left( \frac{f^L(w^L_j)}{f^L(w^L_j)} \right)}{M_L}$$

The idea is to compare the density functions $f^K$ and $f^L$ for each prototype $w$ of $K$ and $L$. If the distributions are identical, these two values must be very close. This measure is an adaptation of the weighted Monte Carlo approximation of the symmetrical Kullback–Leibler measure (see [6]), using the prototypes as a sample of the database.

We used this measure to compute a dissimilarity matrix with all the subsequences of all the recorded customers in the first day of recording. Then we used a modified version of DS2L-SOM that learn prototypes from a dissimilarity matrix [3] and find clusters of homogeneous prototypes [1]. This time prototypes represent a set of individual sub-sequences. This allows us to compare the subsequences of different customers.

These clusters are then used to rename all the subsequences, so as to give the same name to subsequences that belong to the same cluster. Here we found 6 clusters that represent 6 well-defined homogeneous locations. The similarity measure can now be used to label each new customer’s sequence recording after the first day. This is fast enough to be made in real time during the customer’s shopping.

### 3.6 Some results

The analysis method allowed us to find 6 well-defined homogeneous locations (named sectors). This means that we are able to define more well-localized area than the number of reader (50% more), this is a good information extraction. The sectors can be described as follow (see also Fig. 8):

- **S1**: Detected by reader 9 only, it corresponds to the entrance of the store. Baskets waiting for new customers are detected in this sector.
- **S2**: Detected by reader 1 only. In this sector customers can find flowers and vases.
- **S3**: Mainly detected by readers 4 and 10. In this sector customers can find wrought iron objects.
- **S4**: Mainly detected by reader 1, sometimes by reader 9 (wood furnitures).
- **S5**: Mainly detected by readers 9 and 10, sometimes by reader 4 or 1 (dishes and small objects).
- **S6**: Mainly detected by readers 1 and 4, sometimes by reader 9 (Mirrors and linens).

Fig. 8 shows an estimation of the location of the different sectors. We also compute the frequency of transition between one sector to another. This gives us an idea of how customers move inside the store. We can see for example that customer always take the same route in the first part of the store (S1 and S2), but act
more freely at the bottom of the store (S3 to S6). S5 seems to be a key area as there are many transitions from and to it, and it is connected to 4 over the 5 other sectors.

Figure 8. Estimation of sectors location. The thickness of arrows is proportional to transitions frequency. Each sector’s size is proportional to the mean time spent inside it.

Finally, we compute the mean time spent in each sector so as to find hot and cold spots. This shows us that S5 is a very hot spot (48% of the time) unlike S2(6%) and S4(2%) which are very cold spots. Note that we don’t use S1 for this analysis, as this sector includes waiting baskets.

The importance of the area S5 is what we could expected according the goods arrangement and floor plan. Indeed, S5 is the location surrounded with the most varieties of goods and is the only location that allows customers to move to any other directions. In other word, this is the centre point where most customers will definitely pass through.

4 Conclusion and future work

The new method used in this article seems to be a very efficient data mining tool for following RFID-based technology studies, as it is fast, requires little memory space and is suitable for continuous learning. Here, we highlighted some characteristics of spatial organization of customers during their shopping in a big store. We were able to deal with complex and noisy temporal data. We also succeeded to define more well-defined areas than the number of RFID readers we had.

The next step in our study will be to use a clustering method to precisely define different kinds of trajectories, in order to compare each one with the buying behavior at the end of the shopping. This will be done on a big database (4 months recording) to increase validity of the results. Finally, we wish to model the sequences segmentation found in §3.5 with a Markov Model method [10], in order to be able to describe very precisely the dynamic behavior of each customer.

Acknowledgment

This work was supported in part by the CADi project (No ANR - 07 TLOG 003) financed by the ANR (Agence Nationale de la Recherche).

References