Probabilistic Latent Clustering of Device Usage

Jean-Marc Andreoli and Guillaume Bouchard

Xerox Research Centre Europe, Grenoble, France
FirstName.LastName@xrce.xerox.com

Abstract. We investigate an application of Probabilistic Latent Semantics to the problem of device usage analysis in an infrastructure in which multiple users have access to a shared pool of devices delivering different kinds of service and service levels. Each invocation of a service by a user, called a job, is assumed to be logged simply as a co-occurrence of the identifier of the user and that of the device used. The data is best modeled by assuming that multiple latent variables (instead of a single one as in traditional PLSA) satisfying different types of constraints explain the observed variables of a job. We discuss the application of our model to the printing infrastructure in an office environment.

1 Introduction

It is nowadays common that printing devices in an office or a workplace be accessed through the local network instead of being assigned and directly connected to individual desktops. As a result, a large amount of information can easily be collected about the actual use of the whole printing infrastructure, rather than individual devices. To be useful, this data needs to be analysed and presented in a synthetic way to the administrators of the infrastructure. We are interested here in analysing the correlation between users and devices in the data, ie. how the printing potential of users translates into actual use of the devices. We assume here that users are not strongly constrained in their use, the extreme case being when any user is allowed to print anything on any device in the infrastructure. The expected outcome of such an analysis may be diverse. For example, the administrator could discover communities of device usage, corresponding to different physical or virtual locations of the users at the time of the jobs, and, from these, form hypotheses on the actual behaviour of the users, both in the case of normal functioning of the infrastructure and in case of exceptions (device down or not working properly). This in turn could lead to more refined decisions as to the organisation of the infrastructure and to the instructions given to its users. It could also help work around failures of devices inside the infrastructure, by redirecting a job sent to a failing device toward a working one chosen in accordance with the community to which the job belongs.

A study on inhabitant-device interactions [3] shows that the recorded device usage can be mined to discover significant patterns, which in turn could be used to automate device interactions. To the authors knowledge, generic user-device interaction analysis in the presence of devices delivering possibly multiple services or levels of service has not been studied extensively. We propose here a cluster-based approach to find such patterns.

Problem statement Our overall goal is to analyse usage data in an infrastructure consisting of a set independent devices offering services of different or identical classes, and operated by a set of independent users. An interaction between a user and a device is called a job. The usage data consists of a log of these jobs over a given period of time. More precisely, we make the following assumptions.

- Let N_U, N_D, N_K denote the number of, respectively, users, devices and service classes, assumed invariable over the analysed period. Each user, resp. device, resp. service class, can therefore be identified by a number $u \in \{1, \ldots, N_U\}$, resp. $d \in \{1, \ldots, N_D\}$, resp. $k \in \{1, \ldots, N_K\}$. Each user, device, service class also has a print name, for display and reference purpose.
- Each device offers services of one or more classes. This is captured in a boolean matrix f of dimension $N_K \times N_D$ where f_{kd} is 1 if device d offers the service class k and 0 otherwise. This matrix is assumed static over the analysed period.
- All the jobs are recorded over the analysed period. Let n be the number of recorded jobs. Each job can therefore be identified by an index $i \in \{1, \ldots, n\}$. Each job i contributes exactly one entry in the log, consisting of the pair (u_i, d_i) identifying the user and device involved in that job.

A printing infrastructure in an office is a typical example where our method applies, and we use this example to illustrate the method. In that case, a service class could be a particular type of printing. For simplification purpose, in the examples, we consider only two service classes: black&white ($k = 1$) and colour ($k = 2$). Note that a colour printer can always also perform black&white jobs, meaning that if $f_{2d} = 1$, then $f_{1d} = 1$.

Outline of the method The purpose of our analysis is essentially to discover clusters in the usage data. Since the observed data correspond to co-occurrences of discrete variables, we choosed an approach similar to Probabilistic Latent Semantics Analysis (PLSA) [2]. This model is particularly relevant here as its basic assumption has a straightforward interpretation in our context. Indeed, the PLSA assumption is that the data can be generated according to a process that first selects a (latent) cluster, then a user and a device, in such a way that, conditionally to the cluster, the choices of user and device are independent. There is a natural interpretation of such clusters as communities of usage which are associated to physical or virtual locations within the infrastructure. The PLSA assumption means that at a given location, users tend to choose devices in the same way, which is quite reasonable. For example, in an office infrastructure comprising multiple floors, each floor can correspond to a community, whose users share the same perception of the infrastructure and tend to choose printers in a similar fashion. PLSA clustering therefore offers a powerful tool to discover such communities of usage.

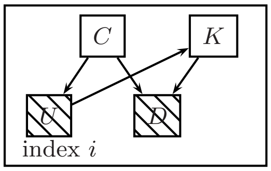

Fig. 1. Graphical representation of the variables dependencies.

However, another important determining factor for the choice of device is the nature of the job to be performed. This information may not be directly available from the logs, still it can be partially inferred from the knowledge of the service class supported by the chosen device. For example, a job sent to a non-colour printer is certainly a black&white job (assuming that users statistically do not make mistakes by launching a job on a device not supporting the service class of that job). As a consequence, the basic PLSA model in which a single latent variable (the cluster) explains the observed ones must be extended to account for the presence of additional latent variables with specific constraints attached to them. Here, the service class of the job is an additional latent variable, but, unlike the cluster, its range is known in advance (it is the set of possible service classes supported by the devices of the infrastructure) and its dependency to the chosen device is constrained by the knowledge of the service classes supported by each device. Studying this multi-factor constrained latent structure motivates our investigation.

2 Definition of the model and parameter estimation

The random variables of the model The recorded jobs are assumed to be independent and identically distributed. We consider 4 random variables that are instantiated for each job: two observed variables U and D defining the user id and the device id, and two latent (or unobserved) discrete variables C and K corresponding to the index of a job cluster and the job service class. We consider that the instantiation of these variables comes from the following generative process: 1. Generate the cluster index C, 2. Generate the user id U, 3. Generate the job service class K depending only on the user, 4. Generate the device choice D depending only on the cluster and the job service class. This process is equivalent to assuming that C is independent to K conditionally to U. A possible factorisation[1] of the joint distribution is $\boldsymbol{p}(U, D, C, K) = \boldsymbol{p}(C)\boldsymbol{p}(U|C)\boldsymbol{p}(K|U)\boldsymbol{p}(D|C, K)$ which is illustrated in the graphical model of Figure 1. Let $\pi^{(C)}$ be the parameters of the multinomial distributions $\boldsymbol{p}(C)$, ie. a vector of proportions of dimension N_C that sums to 1. The other parameters are conditional discrete distributions: $\boldsymbol{p}(U|C)$, $\boldsymbol{p}(K|U)$ and $\boldsymbol{p}(D|C, K)$ are parameterised by the conditional probability tables

[1] Another equivalent factorisation is $\boldsymbol{p}(U, D, C, K) = \boldsymbol{p}(U)\boldsymbol{p}(C|U)\boldsymbol{p}(K|U)\boldsymbol{p}(D|C, K)$, where the generative process starts with the choice of a user and then a cluster.

$\pi^{(U)}$, $\pi^{(K)}$ and $\pi^{(D)}$, respectively. The distribution of the devices is constrained by the knowledge of the service classes they support: $f_{kd} = 0$ implies $\pi^{(D)}_{dck} = 0$ for all $c \in \{1, \cdots, C\}$. Writing $\theta = (\pi^{(C)}, \pi^{(U)}, \pi^{(K)}, \pi^{(D)})$ the set of parameters involved in the model, the joint distribution is $\boldsymbol{p}(u, d, c, k|\theta) = \pi^{(C)}_{c} \pi^{(U)}_{uc} \pi^{(K)}_{ku} \pi^{(D)}_{dck}$. The maximum likelihood estimator is not always satisfactory when the number of jobs is small. We use instead a bayesian framework by defining a prior distribution on the parameters. Since they corresponds to conditional probability tables, we assume Dirichlet priors: $\pi^{(X)}_{.\mathrm{pa}(X)} \sim \mathcal{D}(m^{(X)}_{j}, j = 1, \ldots, N_j)$ where X represents one of the variables U, K, D and C. $\mathrm{Pa}(X)$ are the parents of the variable X. In the application below, the hyperparameter $m^{(K)}$ is set according to expected device usage and the others are set to 0.5 (Jeffrey's uniformative prior). In particular, it may happen that during the analysed period, a given user u never performs jobs of a given service class k (eg. never prints in colour), in which case the maximum likelihood estimator will yield $\pi^{(K)}_{ku} = 0$, meaning that user u *never* uses service class k. The prior knowledge on the users' needs in terms of service classes can be used to compensate for insufficient data. In the printer example below, the expected B&W/color job ratio will be used to define $m^{(K)}$. These can be seen as pseudo-counts of usage of each service class given *a priori* for a "prototypical" user.

Parameter estimation The MAP estimator $\hat{\theta} = \mathrm{argmax}_{\theta}\, \boldsymbol{p}(\theta|\mathbf{x})$ is obtained using the EM algorithm [1]. For brievity, the EM updates equations are not given in this paper. As usual with that algorithm, some care has to be taken in the initialisation. If the number of clusters N_C is known, the MAP estimator can be computed directly. If it is unknown, the MAP estimator must be computed for each possible value of N_C, and the model maximising the BIC score [5] is chosen. This criterion is given by:

$$\mathrm{BIC}(N_C) = \log \boldsymbol{p}(\mathbf{x}|\hat{\theta}; N_C) + \log \boldsymbol{p}(\hat{\theta}; N_C) - \frac{\nu(N_C)}{2} \log n$$

Here, $\log \boldsymbol{p}(\mathbf{x}|\hat{\theta}; N_C)$ is the likelihood of the estimated parameter, $\boldsymbol{p}(\hat{\theta}; N_C)$ is the probability *a priori* of the estimated parameters and $\nu(N_C)$ is the number of free parameters of the model. The selected number of clusters $\widehat{N_C}$ is the one that maximises $BIC(N_C)$. To compute a set of models with different complexities N_C, we first initialise a model with a relatively large complexity, and then decrease it step-by-step until having only one cluster. For each intermediate step, the criterion BIC is computed at the MAP solution obtained by the EM algorithm. Instead of re-initialising the model at each step, we use for level c the $c + 1$ different initialisations that are obtained by removing one cluster from the model learnt at level $c + 1$.

3 Exploitation of the model

There are various ways in which the probabilistic model, once estimated, can be used. We consider two in particular: outlier detection and smoothing.

Outlier detection An outlier[2] is a user whose usage profile observed in the log does not match its expected value by the model. Identifying outliers can help an administrator to understand individual needs that are not provided for by the current configuration of the infrastructure. Recall that the raw usage data is given by matrix n_{ud} which gives the number of jobs involving users u and device d. It is the realisation of the random variable $X_{ud} = \sum_{i=1}^{n} \mathbb{I}\{U_i = u, D_i = d\}$. Let n^*_{ud} be its expectation according to the model. We have $n^*_{ud} = \boldsymbol{E}\left[X_{ud}\right] = n\, \boldsymbol{p}(u, d|\hat{\theta})$. The matrix $n^*_{..}$ is the smoothed version of $n_{..}$ in which information orthogonal to the model space is considered as noise and eliminated. One possible way to compute outliers is to define a quality-of-fit measure of each user and then find the user above a given threshold. The standard chi-squared statistic is used to test if the actual usage of the devices fits that estimated by the model: $\chi^2_u = \sum_{d=1}^{N_D} (n^*_{ud} - n_{ud})^2 / n^*_{ud}$. A user is considered as outlier as soon as χ^2_u is superior to the inverse cumulative distribution of the chi-squared law with $N_D - 1$ degrees of freedom.

Smoothing Any statistic computed from the raw data $n_{..}$ can now be applied to the smoothed data $n^*_{..}$, yielding more precise information:

- *Correction of the primary devices* A good way to check the benefit of smoothing is to compare the primary device of a user u for a service class k, which is defined from the raw data by $r_{ku} = \operatorname{argmax}_d n_{ud} f_{kd}$ and from the smoothed data by $r^*_{ku} = \operatorname{argmax}_d n^*_{ud} f_{kd}$ The users for which $r_{ku} \neq r^*_{ku}$ have a non-standard behaviour which may be of interest to the administrator.
- *Visualisation of the infrastructure* A useful tool for an administrator is a 2D map of the infrastructure s/he administrates. Even if it does not corresponds exactly to the map of the physical setting, such a low dimensional representation provides the administrator with a synthetic view of the overall infrastructure usage. A map of users and devices based on matrix $n^*_{..}$ instead of $n_{..}$ is particularly interesting as $n_{..}$ contains outliers which usually have a strong impact on the dimension reduction algorithms. The use of a smoothed version of the data increases the precision and clarity of the map.
- *Estimating redirections in the infrastructure* Another important tool for administrators is the computation of the redirection matrix of the infrastructure for each of the N_K service classes. This matrix gives for each device d and service class k the device choice distribution conditionally to the fact that d is out of order. For a user u, a device d and a service class k, we derive from the raw data matrix and the *a priori* knowledge a raw estimate of the number of jobs of service class k involving user u and device d: $n_{udk} = n_{ud} m^K_k f_{kd} / \sum_{k'=1}^{N_K} m^K_{k'} f_{k'd}$. In the experiment below, the estimation of the redirection matrix based on $n^*_{..k}$ gives more sensible results than the same method based on $n_{..k}$.

[2] We consider here only *user* outliers. Other types of outliers, eg. devices, can also be treated in the same way.

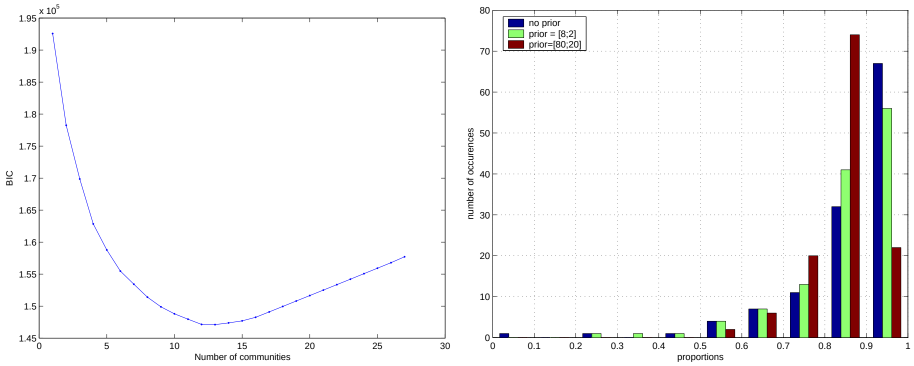

Fig. 2. (a)BIC score of the model for various numbers of clusters. (b) Effect of the prior on the estimations.

4 Experiment on a print infrastructure usage log

Printing logs from an office infrastructure were used to test our model. About 30 000 jobs were logged over a 5 month period, involving 124 users and 22 printers (5 of them colour). The initial number of clusters was the number of observed primary device configurations and was equal to $N_C^o = 21$. From an initial solution including all the previous configurations, the step-by-step procedure described above learnt 21 models with decreasing complexity. Figure 2(a) shows that there is clearly a minimum of the BIC score within the range of estimated models. The optimal value is $\widehat{N_C} = 13$ clusters. This number of clusters is relatively stable when considering only subsets of the data: from 5 000 to 30 000 jobs, the same number of clusters was selected.

For most of the parameters, we used uninformative priors, since the amount of data was sufficient. The only parameter with informative prior was $\pi^{(K)}$. To check the effect of the priors on the estimation, we tried three different values of the hyper-parameter m^K. We compared $m^K = (1,1)$, ie. no prior, equivalent to maximum likelihood estimation, $m^K = (8,2)$, ie. small prior, and $m^K = (80,20)$, ie. strong prior. The ratio 80%/20% means that B&W jobs are *a priori* considered 4 times more frequent than colour jobs for any user. The histogram of the values $\pi_{u1}^{(K)}$ is represented on Figure 2(b) and shows that the priors prevent the parameters from being 0 (only colour jobs) or 1 (no colour job). With the small prior, a user having 25% only of B&W jobs still appears. This corresponds to a user who generally prints B&W jobs to a colour printer. In the sequel, we use the results obtained with the "strong" prior.

Discussion of the results Most of the cluster parameters are summarised in Table 1. Among the 13 clusters, we can see that the first 4 B&W/Colour pairs represent nearly 50% of the jobs. Some remarks:

cluster	B&W printer	colour printer	%	user IDs (% of usage)				
C1	Pre(99%)	Lib(98%)	12.7	ej(13%)	cu(9%)	bw(8%)	cm(8%)	el(8%)
C2	Stu(100%)	Lib(100%)	10	be(16%)	ds(9%)	cp(7%)	au(7%)	dc(6%)
C3	Tim(85%)	Ver(99%)	15.6	db(9%)	ar(9%)	bm(8%)	az(8%)	er(7%)
C4	Vog(99%)	Rep(52%)	13.8	cg(25%)	aw(20%)	ei(18%)	dy(15%)	ep(4%)
C5	Hol(100%)	Lib(100%)	7.7	ch(51%)	ay(31%)	bs(13%)	ec(2%)	bw(0%)
C6	Her(98%)	Tel(98%)	7	ef(26%)	dq(18%)	ce(11%)	dt(10%)	dm(8%)
C7	Geo(97%)	Ver(96%)	5.6	ac(65%)	bv(31%)	dx(2%)	eq(0%)	ec(0%)
C8	Bib(99%)	Rep(100%)	6.8	ag(42%)	bu(38%)	dh(10%)	ec(9%)	et(0%)
C9	Mes(73%)	Ver(84%)	4.5	dx(72%)	em(26%)	ba(0%)	do(0%)	bt(0%)
C10	Lem(97%)	Rep(100%)	3.5	an(92%)	ei(5%)	et(1%)	ch(0%)	bt(0%)
C11	Hod(89%)	Ver(69%)	5.5	eq(20%)	et(14%)	cy(13%)	cc(12%)	ek(9%)
C12	Mid(76%)	Fig(91%)	1.7	da(99%)	ba(0%)	do(0%)	dx(0%)	em(0%)
C13	Sta(99%)	Tel(95%)	5.6	av(12%)	de(10%)	ea(10%)	bh(8%)	cz(8%)

Table 1. Summary of the estimated parameters for each job cluster. The "B&W printer" and "colour printer" are the printers that are most used in each cluster, where the percentage indicates how often this "preferred" printer is chosen, as given by parameter $\pi^{(D)}_{dck}$. The % column gives the probability of each cluster, ie. parameter $\pi^{(C)}_{c}$, and the "main users" are the first 5 users of each cluster with the percentage of launched jobs, ie. parameter $\pi^{(U)}_{cu}$.

- Each cluster is dominated by the use of a "preferred" printer. One example is cluster $C2$, where 100% of the jobs are sent to printer Stu for B&W printing and Lib for colour printing.
- As an exception, cluster $C4$ associated to the B&W printer Vog has two main colour printer (Lib and Rep) with equal importance. The reason of this behaviour cannot be found in the model, but indicates to the administrator that there is a non-standard use of colour printers among the users of Vog.
- Clusters $C3$ and $C12$ contain colour printers (Lib at 4.7% and Ver at 2%) among the B&W printers. This may indicate the use of colour printer when the nearest B&W device is unavailable.
- There are two clusters composed of only one user: "an" in C10 and "da" in C12. In fact, these users have a specific position in the company, and each of them has her own printer, resp. Lem and Mid. These users are not considered outliers since they print a sufficient number of jobs to create individual clusters.

Many other informations about the print usage can be extracted from a deeper analysis of the paramters, depending on the infrastructure administrator's goal.

Outlier identification We applied the method described in Section 3. Only 3 users were rejected from the 80% confidence test: "bx", "aw" and "bd". User "bx" is in fact a generic login for a group of people. Users "aw" and "bd" are using specific printers Pho and Leq that are rarely used by other users. They were not put into a specific cluster and are therefore considered as outliers from a usage point of view.

B&W jobs

az	Ver →Tim
ba	Lib →Pre
bd	Pho→Hod
ci	Ver →Tim
dr	Tel →Sta
eh	Hod→Stu
es	Lib →Pre

colour jobs

al	Lib→Tel	co	Lib→Ver
aw	Lib→Rep	cv	Lib→Ver
ba	Lib→Ver	cw	Lib→Ver
bd	Lib→Ver	db	Lib→Ver
bu	Lib→Rep	dj	Lib→Tel
by	Lib→Rep	dk	Lib→Ver
cc	Lib→Ver	ek	Lib→Ver
cf	Lib→Tel	eo	Tel→Ver

Table 2. Users for which the estimated primary printer is different from the observed one.

Correction of the primary devices Following the method of Section 3, Table 2 lists, for each of the two service classes k (B&W and colour), the users whose estimated primary printer differs from their observed one (ie. $r^*_{ku} \neq r_{ku}$). The "raw" primary device r_{ku} is on the left-hand side of the arrow while the "smoothed" one r^*_{ku} is on the right-hand side. In the B&W case, some colour printers[3] such as Lib or Ver are replaced by a more suitable B&W printer. In the colour case, printer Lib is often replaced by another colour printer which is generally closer to the users. The specific role of printer Lib may be due to the fact that it has a high-quality output, contrary to other colour printers. Our model is in fact biased in that case, as it does not distinguish within the service classes the speed or quality of individual printers. This could be improved by introducing more service classes.

Visualisation of the infrastructure We tried several dimension reduction techniques. PLSA is sometime refered a multinomial PCA (mPCA). With our model, the user repartition $\pi^{(U)}$ can also be interpreted as latent coefficients and plotted if we set $N_C = 2$. However, this technique (as well as standard PCA) gave unsatisfactory results, due to the fact that the first two eigenvalues of the covariance matrix contain less that 50% of the data variance. We also tried *Kernel PCA* with a Gaussian Kernel, but the amount of explained information remained below 60%, which cannot yield a reliable map. Instead, we used a simple nonlinear dimensionality reduction technique called *Sammon's mapping* [4] to $n_{..}$ and $n^*_{..}$ and compared the results (Figure 3). The global distortion value equals 9.9% with the raw data matrix and 4.0% with the smoothed matrix. The printer positions were computed on the reduced space by a weighted means of the community positions. The map based on smoothed data is much more readable than the original one using the raw data. In the latter, users are spread out around their "preferred" printer, but the relation between clusters in confused and hidden by undesired links between users and wrongly estimated "preferred" color printer (e.g. the links to printer Libe). Because of this noise effect, the map does

[3] Recall that colour printers also support the B&W service class and hence can appear as primary printers for that class

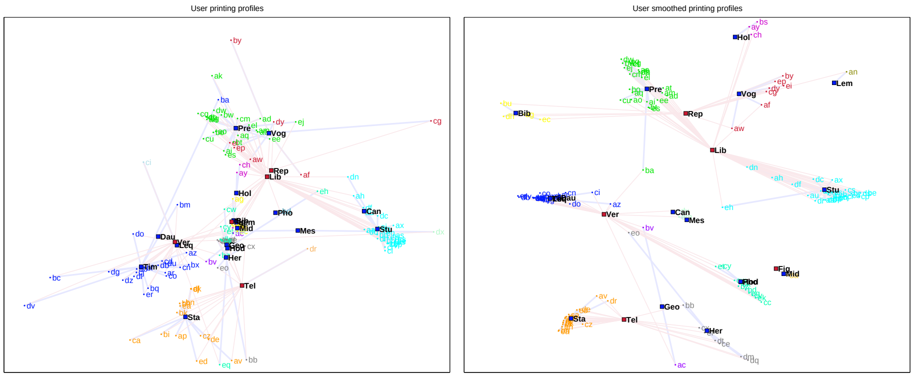

Fig. 3. Low-dimensional representations of the printers and their users. **Above**: based on the raw data matrix. **Below**: smoothed by the probabilistic model. Service classes (B&W and colour) are represented with different colours. Each user has two links: one to its primary B&W printer and one to its primary colour printer. The colour of the users is given by the primary B&W printer.

Raw data

	Lib	Ver	Rep	Tel	Fig
Libe	0	42	55	2	1
Vertical	74	0	21	0	4
Reportage	82	18	0	0	0
Telerama	88	12	0	0	0
Figaro	25	72	3	0	0

Smoothed data

	Lib	Ver	Rep	Tel	Fig
Libe	0	23	75	1	0
Vertical	69	0	26	1	5
Reportage	90	10	0	0	0
Telerama	71	29	0	0	0
Figaro	19	76	5	0	0

Table 3. The redirection matrices for colour service class based on, respectively, raw and smoothed data.

not concentrate the information into clearly distinct clusters. In the smoothed data map, on the other hand, clusters of usages are more visible. Moreover, the two builings of the real office environment are more separated, mainly due to the corrective effect of the "preferred" user printers.

Estimating printer redirections The expression $R_{dd'k} \propto \sum_u n_{udk} n_{ud'k} \mathbb{I}\{d \neq d'\}$ gives the redirection matrix for the service class k. This formula can be justified by assuming that the choice of the redirection printer d' conditionally to the initial printer d follows a multinomial distribution with parameters proportional to $n_{ud}\mathbb{I}\{d \neq d'\}$. Looking at the raw redirection matrix in the colour case on Table 3, printer Libe is redirected at 42% onto printer Vertical which is in another building. This quantity is decreased to 23% using the smoothed matrix, while Reportage is increased from 55% to 75%, which is more sensible since Reportage is much closer to Libe (in the same building). We see that the model uses information about the B&W printers proximity to guess proximity of colour printers.

This is of great interest because B&W data is more abundant, leading to an increased precision of the knowledge of the B&W behaviour, which indirectly increases the precision of the estimation of the redirection in the colour case.

5 Conclusion

In this paper, we proposed to analyse usage data in an infrastructure consisting of users operating devices offering services of different classes. We defined precisely the assumptions on the available data, then built a probabilistic latent class model to cluster the jobs (a job is an interaction user-device). From this model, multiple analysis tools were derived that can help administrators monitor the usage. Instead of studying each user profile individually, the model gives a small number of relevant usage patterns which "compress" the probability distribution into a small number of parameters. One important feature of the proposed model is that it takes into account the device functionalities, without assuming that the specific functionality required by each job is observed.

The case study on a printing infrastructure showed relevant informations about the actual usage of the printers. The model efficiently summarised the whole printing behaviour of the employees, identified non-standard printer usage and proposed changes to the "preferred" user printers that are coherent with the other profiles. The model was used as input to build a map of the printer and user positions, much more readable than those obtained by model-free dimensionality reduction techniques. Finally, the data smoothed by the model gave more sensible results in the estimation of the redirection matrix.

References

1. A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, B 39:1–38, 1977.
2. T. Hofmann. Probabilistic latent semantic analysis. In *Proc. of Uncertainty in Artificial Intelligence, UAI'99*, Stockholm, 1999.
3. E.O. Heierman III and D.J. Cook. Improving home automation by discovering regularly occurring device usage patterns. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003), Melbourne, Florida, USA*, pages 537–540, 2003.
4. J. W. Sammon. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, 18(5):401–409, 1969.
5. G. Schwartz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.