# Serendipitous Exploration of Large-scale Product Catalogs


Marc von Wyl
University of Geneva
Department of Computer Science
Carouge, Switzerland
marc.vonwyl@unige.ch

Birgit Hofreiter
Vienna University of Technology
Institute for Software Technology and Interactive Systems, e-Commerce Group
Informatics Innovation Center (i2c)
Vienna, Austria
birgit.hofreiter@tuwien.ac.at

Stéphane Marchand-Maillet
University of Geneva
Department of Computer Science
Carouge, Switzerland
stephane.marchand-maillet@unige.ch



***Abstract*—Online shopping has developed to a stage where catalogs have become very large and diverse. Thus, it is a challenge to present relevant items to potential customers within a very few interactions. This is even more so when users have no defined shopping objectives but operate in an opportunistic mindset. This problem is often tackled by recommender systems. However, these systems rely on consistent user interaction patterns to predict items of interest. In contrast, we propose to adapt the classical information retrieval (IR) paradigm for the purpose of accessing catalog items in a context of un-predictable user interaction. Accordingly, we present a novel information access strategy based on the notion of *interest* rather than relevance. We detail the design of a scalable browsing system including learning capabilities joint with a limited-memory model. Our approach enables locating interesting items within a few steps while not requiring good quality descriptions. Our system allows customer to seamlessly change browsing objectives without having to start explicitly a new session. An evaluation of our approach based on both artificial and real-life datasets demonstrates its efficiency in learning and adaptation.**


## I. Motivation

The emergence of online shopping has offered new opportunities to propose services and products to customers. Currently, many online shops are not anymore restricted to a certain category of products. For example Amazon, initially focused on cultural and entertainment media (books, music, and video), is now offering products as diverse as home appliances or jewelry. Even more crucial, we usually find thousands of items within a product category, e.g. 38 million books and 3,5 million jewelry items on Amazon. Both the breadth of product lines and the depth within a product line not only boost the volume of the catalogs but also make it difficult for the customer to find products of interest without an accurate search protocol. Presenting relevant products to potential customers is the goal of recommender systems. Independent of their type (collaborative filtering systems, content-based recommender, etc), recommender systems usually operate on a user profile gained from previous shopping sessions. For this reason, recommender systems suffer from the cold-start problem, when new users and/or new products appear [1]. Furthermore, they extract user profiles and predictions from consistent interaction patterns to propagate judgments between users.

In contrast to the above, our approach does not require the definition of a user profile nor it imposes specific search sessions with pre-defined objectives. In other words, we present an efficient product access strategy enabling intuitive browsing by estimating the user's intention from his/her input to the system and displaying items that are considered as most interesting to him/her (and thus likely to be purchased). Our new information access strategy is based on the notion of current *interest* rather than on the notion of *relevance* classically used in Information Retrieval [2]. We detail the design of an information browsing system while keeping the following objectives in mind:

- (O1) We accommodate serendipity. We assume no pre-defined (fixed) objective of the user's chain of actions;
- (O2) The system matches classic (simple) interaction models;
- (O3) The system is scalable in terms of the volume of the product catalog.

Our approach results in an interactive navigation system, which let the user operate naturally over the product catalog while swiftly reacting to changes in the browsing objectives. The major difference with earlier approaches is a rapidly adapting system, that copes with radical changes, and is scalable to operate over realistic-scale product catalogs.

The remainder of the paper is structured as follows: in section II, we discuss relevant approaches for information characterisation and content access strategies in large repositories. In section III, we present our interaction model, which describes the type of interaction that is expected from the user and what information is carried over with this interaction. We formalise our navigation model, anticipating functional issues in section IV. In particular, we review its properties ensuring scalability and compatibility with other models. In section V, we propose a comprehensive assessment of the performance of our model in an adaptive browsing scenario. At every browsing step, the system aims at displaying the most useful items to the user with respect to past interaction. Although our study includes an inherent temporal dimension, which makes the evaluation context different from that of classical search systems, the evaluation may still be performed following IR assessment procedures. We therefore propose to evolve standard IR evaluation protocols in order to accommodate differences. Experimental results are then derived from this proposed evaluation strategy, implemented into two different setups. Finally, we present our conclusions in section VI.

## II. Related work

The multiplication of large-scale online information repositories has created the need for robust and adaptive access strategies in many applicative areas such as business [3], education or culture [4], [5], [6], or general public domain [7]. Search and retrieval operations have installed themselves as a base paradigm for accessing items from within a collection. They are mostly based on the notion of a query formulation [8]. The domain of knowledge management, via the definition of accurate data description schemes offers solutions for accurate query-based information access. However, the complexity and induced costs of design, creation, maintenance and compatibility of such descriptions generally impedes their usage and development. Hence, in the case of systems operating over poorly described or non-textual data, the idea of "query-by-example" has emerged [2], [9] as a help to formulate the query. Positive and negative examples are aggregated over intermediate search operations, in order to form a descriptive set for the sought items. Examples then become the base for online learning operations, so as to generalise "classes" of provided *relevant* and non-relevant items [10], [11], [12].

Browsing systems have been proposed and are also mostly based on the definition of a search objective [13]. Such systems are typically oriented towards the localisation of a known object, be it copy detection [14] or user's *mental image* localization [15], [16]. They iterate user *judgements* over appropriately-chosen sample sets of items to estimate the target item the user has in mind. This framework has been extended by [17] from image search to product browsing for mobile e-commerce. Our proposal differs from that work in the fact that, while our method may be used as a search system (see discussion in section VI), we assume no predefined customer target, in order to foster serendipity (objective (O1)).

Adaptive Hypermedia (AH) also relies on the navigation paradigm for information exploration to resolve the issue of complex query formulation. As given in [18]: "The core problem in finding the information you want, in all the above cases, is *describing* what you want. Results from search engines are often disappointing because most search requests are too short and unspecific to yield good results. [...]. The community of *user modeling* and *adaptive hypermedia* offers solutions for this problem: using information gathered about the user during the browsing process to change the *information content* and *link structure* on-the-fly. User modeling captures the mental state of the user, and thus allows that knowledge to be combined with the explicit queries (or links) in order to determine precisely what the user is looking for. To support the design of this user model-based adaptation, reference models like AHAM (De Bra et al. 1999 [19], Wu 2002 [20]) and Munich (Koch and Wirsing 2002 [21]), both based on the Dexter Model by Halasz and Schwartz (1994 [22]), have been introduced in an attempt to standardize and unify the design of adaptive hypermedia applications, used mostly in isolated information spaces such as an online course, an electronic shopping site, an online museum, *etc*".

In [23], a taxonomy of AH technologies is further presented. The taxonomy is analyzed in detail in [24], along with an extensive review of AH systems. Our current work embarks the objectives of AH, with importing adaptive strategies close to the *metric learning* setup [25], [26] as a replacement of the *link adaptation* technique proposed by AH navigation systems.

The above also involves the notion of *user modeling* and a comprehensive review on personalization research in e-commerce is presented in [27].

Information filtering also comes as a helper solution for the interactive formulation of search queries. Rules are defined over product characteristics, in order to define the class of the sought items as the intersection of solution sets for the rules. Rules are generally based on information *facets*. Facets are orthogonal, mutually exclusive dimensions of the data whose range is quantized in relevant intervals [28]. Facets may be determined from the data model itself by highlighting important characteristics of the data. In exploratory conditions however, i.e. when the data is not fully understood, it may be interesting to let facets emerge automatically or interactively for providing interpretation of its organisation and to facilitate its exploration [7]. Several routes may be taken to automatically determine data facets. They all consist in using the data or a representative sample in a mining process to identify a reduced set of orthogonal projection operators whereby every data item is identified by its set of projections.

Faceted search is extensively used over e-commerce sites when products bear inherent orthogonal characteristics. For example, this is the case for real estate commerce with facets such as product type, surface, region, price range,... In section VI, we also discuss how our model comes as a complement to faceted exploration.

Where item characterisation is difficult, deficient or not possible, recommender systems [29], [30] leverage the wisdom of the crowd and propagate user interests across a community. The main idea is to create a bipartite graph between products and customers where user ratings (judgments) are used as edge weights. Information is then propagated along this graph to group customers and/or products and thus, predict new edge weights (ie the judgement of a costumer over a product).

The framework of recommendation is used in [31] to cater for the lack of relevant or accurate information available to customers over "experience products". Authors demonstrate the effectiveness of their technique in the context of movie recommendations. An early study on how such systems may be formally evaluated is proposed in [32]. The compatibility of our model with this approach is also discussed in section VI.

In the following we elaborate the design of our browsing model, from its structure (section III) to its formalisation (section IV).

## III. Method

We aim to construct a system that sees user interaction as an *expression of interest* and uses this to provide further recommendations to the user. While recommender systems [30] propagate user interests across a community, here, recommendation is based on the navigation history, in the line of Adaptive Hypermedia Navigation [23].

The main features of our model are the use of positive only feedback and a user model with time-limited characteristics. The main idea is to discover, at every step of the navigation, what linear combination of product characteristics brings together past choices of the user and to adapt accordingly in a scalable fashion. We propose the following basic navigation scenario that may be implemented over essentially any type of visualization device. The main control at every iteration is a single click on a product (objective (O2)). Here, clicking an item means mining deeper in the direction of that item, in relation to a past navigation behavior. While allowing specific target search, this process allows for widening or reorienting the navigation by simply breaking the coherence in the history of actions. Thus, our system is very close to the "more like this" paradigm but adds memory in the browsing model.

Our system operates in the following three major steps, repeated at each iteration of the browsing process:

- (S1) The system samples the collection from the best of its current knowledge;
- (S2) The user clicks on one of the proposed products;
- (S3) Accordingly, the system updates its knowledge over the session and repeats from step (S1).

In the sequel, we detail how we model steps (S1) and (S3) (system adaptation), according to what is assumed from step (S2) (user input). Essentially, we are based on the setup of metric learning [25], [26] to adapt a measure over the collection and to be able to extract a collection sample suited for further navigation.

Note this paper focuses on the formal back-end model for catalog browsing and product recommendation. The issue of the display and how the interaction is captured [33], depending on the used device is subject of future work and is only briefly discussed in section VI.

## IV. Opportunistic browsing

The user (customer) is faced with a collection (catalog) of $N$ items (products) $\mathcal{C} = \{x_1, \cdots, x_N\}$, each bearing a set of intrinsic characteristics, qualitative (ie taken from a taxonomy: brand, model, ...) or quantitative (ie measured: size, color, weight, ...). Without loss of generality, we represent each item by a $M$-dimensional feature vector ($x_i = [x_{i1} \cdots x_{iM}]^\mathsf{T}$) by aggregating its characteristics.

### A. Adaptive Model

Given $A \in \mathcal{R}^{M\times M}$, a symmetric positive semi-definite matrix, the Mahalanobis distance between items $x_i$ and $x_j$ is defined by:

$$d_A^2(x_i, x_j) = (x_i - x_j)^\mathsf{T} A (x_i - x_j) \tag{1}$$

and corresponds to a weighted form of the Euclidean distance. As posed in the metric learning setup [25], [26], we use the Mahalanobis distance $d_A(x_i, x_j)$ as a base for measuring the similarity between items $x_i$ and $x_j$ and we further wish to use matrix $A$ as a way of adapting to user feedback.

We start with a relevance feedback-like scenario. Typically, by iterating the item selection (step (S2)), the user generates a navigation history in the form of an ordered item subset:

$$\mathcal{H}_t = \{x^{(1)}, x^{(2)}, \cdots, x^{(t)}\}$$

where $x^{(t)} \in \mathcal{C}$ is the product selected at iteration $t$. By learning a scaling matrix $A^{(t)}$ as a function of its previous state $A^{(t-1)}$ and the current available history of actions $\mathcal{H}_t$, the system will then adapt its base distance $d_{A^{(t)}}(.,.)$ (step (S3)) to group items in $\mathcal{H}_t$ close together according to a defined policy (detailed below). This learning step will generalise user preferences over non-selected items. From there on, the system will sample the collection $\mathcal{C}$ (step (S1)) by ranking items by their distance $d_{A^{(t)}}$ from the latest chosen sample $x^{(t)}$ and display top-ranked items, seen as the best current prediction of an accurate selection of products from within the catalog.

Now, simple algebra shows that, if $A$ is similar to another $\mathcal{R}^{M\times M}$ matrix $B$ via an orthogonal matrix $V \in \mathcal{R}^{M\times M}$ (ie $A = V^\mathsf{T} B V$), then $d_A(x_i, x_j) = d_B(y_i, y_j)$, where $y_i = V x_i$. In other words, by changing the distance function in Equation (1), we apply a linear transform on the feature space. In particular, if $A$ is taken as $\Sigma^{-1}$, the inverse of the $M \times M$ feature covariance matrix computed over $\mathcal{C}$, then the eigendecomposition $A = U^\mathsf{T}\Lambda U$ allows for defining decorrelated features

$$y_i = \Lambda^{-1/2} U^\mathsf{T} x_i, \tag{2}$$

so that $d_A(x_i, x_j) = d_B(y_i, y_j)$ with $B = \mathrm{Id}$, the identity matrix.

We use this property to simplify the computation of our new distance function and make it scalable. We use $B^{(t)} = \mathrm{diag}(1/\sigma_1^{(t)2}, \cdots, 1/\sigma_M^{(t)2})$, an inverse diagonal covariance matrix, as scaling matrix. Since we start with decorrelated features, initially, $B^{(0)} = \frac{1}{M}\mathrm{Id}$ (ie $\sigma_k^{(0)} = \sqrt{M}\ \forall k$). We thus obtain

$$d^2_{B^{(t)}}(y_i, y_j) = \sum_{k=1}^{M} \left( \frac{1}{\sigma_k^{(t)}} (y_{ik} - y_{jk}) \right)^2 \tag{3}$$

In order to evolve the scaling matrix $B^{(t)}$, two constraints need to be preserved:

- (C1) $B^{(t)}$ is a positive semi-definite matrix for all $t$, to keep $d_{B^{(t)}}$ a distance function;

(C2) $\mathrm{trace}(B^{(t)}) = \sum_k 1/\sigma_k^{(t)2}$ has to remain constant for all $t$, to avoid space shrinking and to preserve consistent scaling through iterations. Without any loss of generality, we choose $\mathrm{trace}(B^{(t)}) = 1$.

At iteration $t$, to evaluate whether two documents $y^{(t-1)}$ and $y^{(t)}$ are close to each other in terms of feature $k$, we compare their base distance $|y_{ik}^{(t)} - y_{jk}^{(t-1)}|$ with the corresponding scaling factor $\sigma_k^{(t)}$. If $|y_{ik}^{(t)} - y_{jk}^{(t-1)}| < \sigma_k^{(t)}$, we assume a consistent browsing behavior and feature $k$ should be reinforced to keep the documents close to each other. Otherwise, the influence of feature $k$ in Eq.(3) should be lowered.

We propose two schemes to evolve our scaling matrix $B^{(t)}$:

*a) Additive scheme:* We directly use the scaling criterion $|y_{ik}^{(t)} - y_{jk}^{(t-1)}| < \sigma_k^{(t)}$ as a gradient for scaling factors $\sigma_k^{(t)}$. We thus update $\sigma_k^{(t)}$ following:

$$\sigma_k^{(t)} = \sigma_k^{(t-1)} + \lambda \left( |y_{ik}^{(t)} - y_{jk}^{(t-1)}| - \sigma_k^{(t-1)} \right) \tag{4}$$

where $\lambda$ is a learning rate. The diagonal form of $B^{(t)}$ and constraint (C1) impose $\lambda \in [0, 1]$ to keep $\sigma_k^{(t)} > 0$ and we satisfy constraint (C2) by a global normalisation step.

Alternatively, Eq.(4) may be read as:

$$\sigma_k^{(t)} = (1 - \lambda)\sigma_k^{(t-1)} + \lambda |y_{ik}^{(t)} - y_{jk}^{(t-1)}| \tag{5}$$

so that the new scaling factor $\sigma_k^{(t)}$ is taken as a convex combination of the previous factor and the perceived similarity of selected items $y^{(t)}$ and $y^{(t-1)}$ against feature $k$.

*b) Multiplicative scheme:* We may also use the ratio $(|y_{ik}^{(t)} - y_{jk}^{(t-1)}|/\sigma_k^{(t)})^\lambda$ as learning transform, so that

$$\sigma_k^{(t)} = \left( \frac{|y_{ik}^{(t)} - y_{jk}^{(t-1)}|}{\sigma_k^{(t-1)}} \right)^\lambda \sigma_k^{(t-1)} \tag{6}$$

The same conditions apply on $\lambda$ to satisfy constraints (C1) and (C2). Similarly, we obtain:

$$\sigma_k^{(t)} = \left( \sigma_k^{(t-1)} \right)^{(1-\lambda)} \left( |y_{ik}^{(t)} - y_{jk}^{(t-1)}| \right)^\lambda \tag{7}$$

Our scheme pertains an implicit abstraction of features using the decorrelation step (Eq.(2)). This operation is similar to what is proposed in the LSI approach [34]. This facilitates the design of relevant characteristics by subsequent linear learning operations (eg, Eq.(4)). In our scheme, hand-crafted specific product features may be preserved by simply excluding them from this pre-processing and then merged "as is" for subsequent operations.

The above transforms apply between iterations $(t - 1)$ and $(t)$. We now generalise them over the complete history $\mathcal{H}_t$ to include a temporal dimension in the learning operation.

### *B. Limited-memory model*

Whenever the browsing history $\mathcal{H}_t$ is consistently directed towards a specific goal (eg, all selected items $x^{(t)}$ –resp. $y^{(t)}$– belong to the same class), the above scenario corresponds to a search scenario with positive relevance feedback only [2], [9]. It is also compatible with the target search browsing [16], [17] since it adapts a global distance function over the search space. However, in the case where the customer's objective changes and thus, the consistency of $\mathcal{H}_t$ is destroyed, relevance feedback or target search systems hardly adapt due to their inherent learning properties.

Our model makes it easy to introduce a temporal weighting scheme that weights the influence of past choices lower than the influence of recent user actions. We introduce a forgetting framework that gives more weight to recent choices. We weight the past choices using a sigmoid function $S(\tau) = \frac{1}{1+e^{-\tau}}$ over interval $\tau \in [-a, +a]$ and weight $\mathcal{H}_t$ on this function over a fixed number $N_h$ of steps $t, \ldots, t - N_h$ (see Fig. 1).

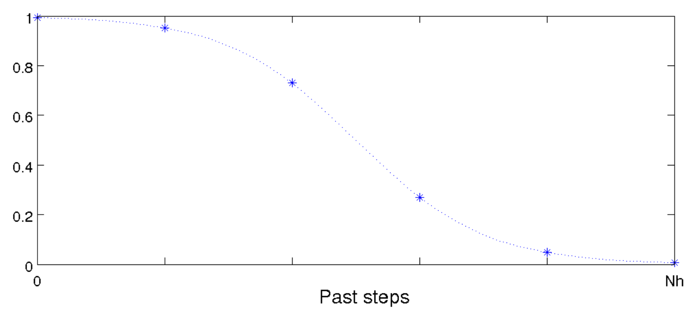


Fig. 1. Step weighting scheme ($a = 20$)

Hence at iteration $t$, we compute weights $\omega_t, \cdots, \omega_{t-N_h}$ using the weighting function $\omega_{t-l} = W.S(\frac{2al}{N_h} - a)$ for $l = 0, \ldots, N_h$ and $W$ such that $\sum_l \omega_{t-l} = 1$. Adapting from Eq.(5), we take a new value for $\sigma_k^{(t)}$ as a convex combination of the previous value $\sigma_k^{(t-1)}$ and a weighted average of feature importance criteria $|y_{ik}^{(t)} - y_{jk}^{(t-1)}|$. We thus obtain the following additive update rule for $\sigma_k^{(t)}$:

$$\sigma_k^{(t)} = (1 - \lambda)\sigma_k^{(t-1)} + \lambda \sum_{l=0}^{N_h} \omega_{t-l} |y_{ik}^{(t)} - y_{jk}^{(t-l-1)}| \tag{8}$$

Similarly, adapting from Eq.(7), we obtain the following update rule for $\sigma_k^{(t)}$:

$$\sigma_k^{(t)} = \left( \sigma_k^{(t-1)} \right)^{(1-\lambda)} \left( \prod_{l=0}^{N_h} \omega_{t-l} |y_{ik}^{(t)} - y_{jk}^{(t-l-1)}| \right)^\lambda \tag{9}$$

In practice, Eq.(9) suggests to iteratively multiply decreasing values together so that values of $\sigma_k^{(t)}$ quickly reach 0, but for a dominating feature $k$, where $\sigma_k^{(t)} \to 1$, due to normalisation. The effect is that feature $k$ is exponentially reinforced, unless a dampening function, such as the log function, is used. It should be noted however that $\log(\sigma_k^{(t)})$ in Eq.(9) has the same structure of the r.h.s of equation Eq.(8). Hence, we will only evaluate the performance of the additive scheme using Eq.(8) and a normalisation step such that $\mathrm{trace}(B^{(t)}) = 1$ for all $t$.

A further part of the evaluation will focus on testing the influence of the choice of values for $N_h$ and $\lambda$.

### C. Scalability

The above model is a simplification of typical distance metric learning algorithms [25]. A first difference with the usual context within metric learning applies is that, in our context, we do not look for classification or clustering, whether or not based on constraints [26], [35].

Our model is also designed to cope with scalability (objective (O3)). Basically, we secure the possibility for most computations to be done offline and then efficiently organise for online adaptation. The modification of matrix $B^{(t)}$ is straightforward using Eq.(8) and requires $O(M)$ operations.

The computation of distance $d_{B^{(t)}}(.,)$, as given in Eq.(3), requires the computation of the sum over values of $(y_{ik}-y_{jk})$, which are fixed and may be properly indexed prior to starting the process. To further help the actual distance computation and because we target nearest neighbour ranking, we could make full use of approximate metric indexing in every isolated feature space (ie $(y_{ik}-y_{jk})$) to prepare the data. This is not addressed here and is the subject of current and future work [36].

## V. Evaluation and Experiments

### A. Protocol

We demonstrate the information access performance of our system and also its capabilities in adapting to inconsistent browsing behaviour. Serendipitous search adds the complexity that the process may not genuinely be seen as a retrieval process with a defined final objective. There is a definite need for the development of specific evaluation procedures related to our process, essentially involving active manual judgments, where, for example, the user would be able to actively declare his/her objective, which would be later used to evaluate the result of each browsing step. Instead, in this paper, we adapt classical information retrieval measures to the context of browsing product catalogs, in order to obtain quantitative evaluation. As an initial simplification to propose an automated quantitative evaluation of our proposal, we map our problem onto a retrieval process where, at every step, relevance is known, although it may change from step to step. We use item categories and base relevance on categories. We thus operate our system over datasets formed out of items $(y_i)$, each associated with a class label. We then simulate the interaction of a user first seeking an item into one class and then, after some iterations, switches interest towards another class of items.

Accordingly, at every iteration, the user is presented the top items from the ranked list, where, ideally, all items belong to the class of the last selected item. In order not to limit the possible choice for a new selection, the user is also presented a sampling of the catalog (showing all possible classes) from where the new selection could also be made. The exact methodology for sampling the catalog, either independent of $\mathcal{H}_t$ or related to $\mathcal{H}_t$, is not discussed here and is an interesting topic for further research. Here, at the time where the user switches topic, an item from another class is simply selected randomly.

This basic scenario is a drastic worst case since, at the time of the switch, the system needs to quickly adapt at best to the radically new class, without the user to explicitly cancel previous history. Hence, the system should find a trade-off between relying on previous history for a more robust learning and being able to make drastic turns to adapt to the customer's behavior. The simulation is done by hiding class labels to the system and only using them to make a selection $y^{(t)}$ of the desired class at each iteration. Care is taken not to iterate over the same items $y^{(t)}$ in the history.

At each time step, a ranked list of estimated most interesting items is inferred. We evaluate the precision of this list with respect to the current selection $y^{(t)}$. More precisely, the precision at rank $r$ is defined by

$$P(r) = \frac{\text{number of relevant items in the top } r \text{ ranked list}}{r}$$

Ideally, $P(r) = 1$ at any rank of relevant items (items of the same class than $y^{(t)}$). The average precision (AP) is the average of precision at ranks of relevant items:

$$AP = \frac{\sum_{r=1}^{N}(P(r) \times \text{rel}(r))}{\text{number of relevant items}}$$

where $\text{rel}(r)$ indicates whether item at rank $r$ is relevant ($\text{rel}(r) = 1$) or not ($\text{rel}(r) = 0$). To emancipate from the influence of the choice made in interaction, we repeat the experiment $N_E$ times and average over experiments. The mean average precision (mAP) is thus defined as:

$$mAP = \frac{\sum_{i=1}^{N_E} AP_i}{N_E}.$$

We therefore display the results under the form of mAP values at every iteration $(t)$ for various datasets and parameters of our system. The goal is to demonstrate that, at every iteration, the user is presented a list of relevant items with respect to the current choice $y^{(t)}$.

### B. Datasets and results

We simulate the catalog by generating a set of $N$ items with $M$-dimensional decorrelated features, from two different origins. To make a comprehensive proof-of-concept, we first generate an *artificial dataset* by drawing features from probability distributions. Then, in order to obtain a more realistic scenario, we wish to apply our model over real data. However, commercial data such as e-commerce catalogs, added with item descriptions is not easily available and steps in the direction of creating such corpora should be taken. We still create a real-world set of features arising from the *MNIST dataset*, widely used in information clustering and classification for the elegant properties of its feature set. For both datasets, the resulting features describe the items, organised into classes and interacted with following the above protocol.

*Artificial dataset:* We consider a raw space of $N_a$ attributes. We characterise $N_a$-dimensional items of $N_c$ classes by varying the distribution of attributes for each of the classes. Attribute distributions are modeled as mixtures of Gaussian parametrised by random numbers of mixtures, mean, variance

TABLE I
PARAMETERS FOR THE ARTIFICIALLY GENERATED DATASET

| Dataset | | | | Experiment | | | |
|---|---|---|---|---|---|---|---|
| $N_a$ | $N_c$ | $N$ | $M$ | $N_1$ | $N_2$ | $N_E$ | $a$ |
| 50 | 10 | 3000 | 15 | 20 | 20 | 500 | 6 |

and mixing coefficients. Some classes may share attribute distributions to emulate transverse characteristics such as color, size or weight.

Once classes are characterised, we generate populations for each of the classes to form a global population of $N$ $N_a$-dimensional items $x_i$. Then, we use PCA to decorrelate the features and obtain a population $\mathcal{C}$ of $N$ $M$-dimensional items $y_i$. Using the resulting set, we obtain an approximate 15% error rate with a basic $k$NN classifier, showing that there is room for more learning operations.

Based on this artificial dataset, we run experiments by simulating a user who selects products of class $A$ and then switches to class $B$ after $N_1$ iterations and stays with class B items during $N_2$ extra iterations. We run this experiment several times ($N_E$ times) by varying labels $A$ and $B$ to emancipate from the particular structures of the classes and present results as mAP, averaged over all runs.

Figure 3 first shows the results for the parameters given in Table I when varying our memory size parameter $N_h$ and a fixed learning rate $\lambda = 0.5$.

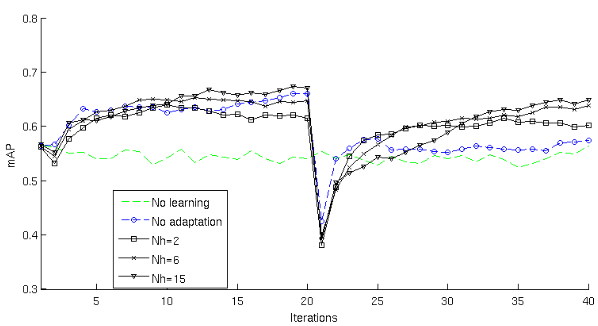


Fig. 2. mAP obtained for the artificial dataset when varying our memory size parameter $N_h$ ($\lambda = 0.5$)

Our baselines are respectively the “No-learning” system where $B^{(t)}$ does not evolve during iterations (equivalent to $\lambda = 0$) and a “No-adaptation” system, taking all history steps into account (equivalent to $\omega_{t-l} = 1$ for all $t$ and $l$ and $N_h = \infty$). We first see that learning happens since the mAP increases rapidly after the first iterations. When the customer switches the focus of interest (iteration 20), the mAP drops as the evaluation is done with $B^{(t)}$ learnt on class A. However, our system recovers quickly after only a few iterations, whereas the “No-adaptation” system cannot recover without starting a new session. In order to set $N_h$, one sees that large values give more inertia to the system. When $N_h = 15$, the system takes longer to recover. However, larger $N_h$ also guarantees more robust learning. We see that $N_h = 6$ is a good trade-off between system inertia and learning performance.

We now turn to varying parameter $\lambda$, the learning rate. Figure 3 shows that lower values of the learning rate are to be

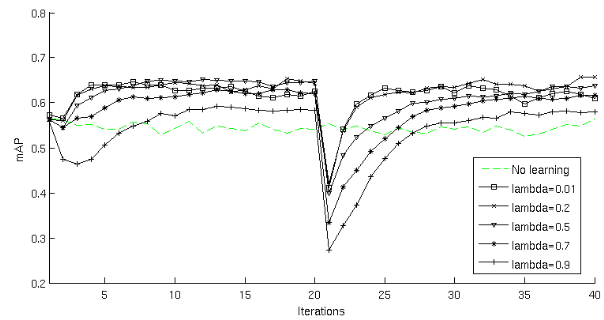


Fig. 3. mAP obtained for the artificial dataset when varying the learning rate $\lambda$ ($N_h = 6$)

preferred. A high $\lambda$ (eg $\lambda = 0.9$) induces too much instability in the learning and decreases the average performance. As can be read from the plot, a value of $\lambda \simeq 0.2$ leads to a good learning performance while preserving the stability of the system.

As further evidence on the learning operations and the adaptivity of the system to user inconsistency, Figure 4 illustrates the evolution of features, averaged over 50 runs when switching from class A=3 to class B=5, with $\lambda = 0.3$.

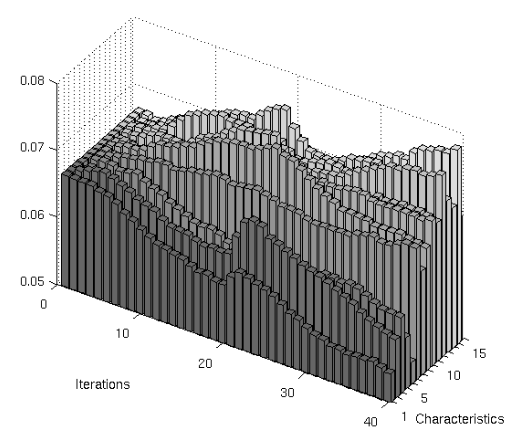


Fig. 4. Evolution of features for the artificial dataset when switching from class A=3 to B=5 (50 runs, $\lambda = 0.3$, $N_h = 6$)

*MNIST dataset:* In order to obtain a realistic setup, we use real data as a pool of features. Unlike in some other domains (eg movie rating), no e-commerce data collections is publicly available for standardized quantitative system evaluations. Hence, we turn to popular available datasets used for the evaluation of similar systems. The MNIST dataset is a set of hand-written digit (class “0” to “9”) images of size 28×28 classically used for information processing tasks.

MNIST images are generally characterised directly by their pixel values with aligning all columns as a $784 \times 1$ vector of grey-scale values. Using the 30 first decorrelated features, one classically obtains more than 95% accuracy with a $k$NN classifier, showing that these features are indeed very expressive. Although the semantic interpretation of the MNIST dataset does not fully match our context, it is a relevant pool of features that may be used for a blind evaluation of our system.

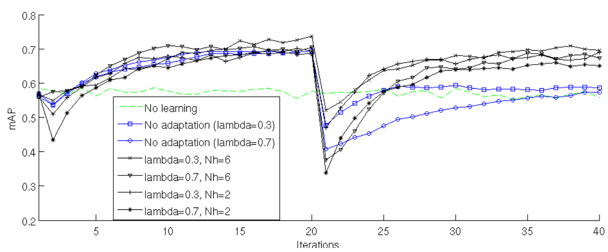


Fig. 5. mAP obtained for the MNIST dataset with several values of learning rate $\lambda$ and memory size $N_h$

Figure 5 shows the various mAP levels obtained for several values of the learning rate $\lambda$ and memory size $N_h$. We see that these results are similar to those previously obtained with a moderate sensitivity against the specific values of our parameters $\lambda$ and $N_h$ (the best empirical performance is again obtained for $\lambda = 0.3$ and $N_h = 6$). As before also, Figure 5 demonstrates the usefulness of our approach against the "No-adaptation" baseline at the time where user switches focus towards a different objective.

## VI. Discussion and conclusion

In product catalogs with thousands of items, it is of evident importance to present the potential customer with products that are of interest to him/her. Accordingly, we tackled the challenge of efficiently browsing through online shopping catalogs. We presented the design of a navigation system where the interaction is simplified to a click-through operation. This elementary feedback is interpreted as an *expression of interest*, which departs from either the relevance feedback for search where no temporal information is exploited and from judgements for recommendation where the information flows from a community to an active user. Here, the user takes a series of actions and the system reacts accordingly, accounting for most recent actions first. The system learns an approximate metric that group elements selected in the navigation history. The context of metric learning is used to justify approximations.

We proposed an adapted evaluation scheme oriented towards mAP values, and implemented it using artificial and real data. Our evaluation demonstrates that the two main features of our strategy are indeed operational. Namely, learning is achieved so that the customer is offered relevant products and adaptability is achieved so that the system copes with customer's changes of objectives, which alleviates the concept of sessions (classically present in search operations).

The evaluation protocol used here is biased towards the idea of retrieval and we plan to complete it with actual user experiments. Such an evaluation would also address the question on the form of display to present to the user, which was not directly addressed here. The serendipitous aspect of our process further makes the classical IR evaluation protocols not fully adapted to demonstrate the quality of our proposal. The issue of designing a more accurate quantitative evaluation protocol still needs to be addressed, possibly related to several alternative evaluation frameworks [8], [32].

There are many evidences on the utility of navigation-based systems, in complement to search and retrieval operations [13], [17]. Our approach readily includes all features of a search system. If the feedback provided is consistent towards a search objective, it is integrated as positive relevance feedback and converges towards that objective, like target search systems [16], [17]. Our approach is also compatible with any other type of information access mechanism. Although the initial display may simply show a random sample of a catalog (potentially biased towards trendy or sale products), any initial information access step may be made before browsing. Our system may be offered as an interaction module starting from the result of a search operation or a facet-based filtering. Conversely, it may also provide a base information for continuing with filtering or search. In complement, the type of collected feedback is based on the notion of interest. This notion is key in recommender systems and, should the customer make a final purchase, the search history may then be exploited for knowledge propagation in a recommendation framework [29].

This work demonstrates the validity of our approach. There are many ways forward from this study. First, formal modeling and evaluation of information browsing systems is not mature enough yet. User studies over appropriate data are necessary to clearly validate the utility of such approaches. For implementing such real-case tests, actual e-commerce data is needed and may be captured from actual sites.

Our methodology will be further developed by formalising the inclusion of diversity in the browsing results, in order to foster serendipitous discoveries. The complementary issues of how to optimally display results and capture interaction in such systems are then to be addressed to complete the design of a fully validated browsing system.

## VII. Acknowledgments


This work has been partly funded by SNF fund No. 200021-119906, in parallel with the Swiss NCCR(IM)2 and the ICT COST Action IC1002 "Multilingual and Multifaceted Interactive Information Access" (MUMIA) .

## References


[1] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, "Methods and metrics for cold-start recommentations," Tampere, Finland, 2002, pp. 253–260.

[2] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: A power tool in interactive content-based image retrieval," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 644–655, 1998.

[3] S. Seidel, F. M. Muller-Wienbergen, M. Rosemann, and J. Becker, "A conceptual framework for information retrieval to support creativity in business processes," in *Proceedings 16th EuropeanConference on Information Systems*, Galway, Ireland, 2008.

[4] P. Brusilovsky, L. N. Cassel, L. M. Delcambre, E. A. Fox, R. Furuta, D. D. Garcia, F. M. S. III, and M. Yudelson, "Social navigation for educational digital libraries," *Procedia Computer Science*, vol. 1, no. 2, pp. 2889 – 2897, 2010.

[5] A. Carbonaro, "Collaborative and semantic information retrieval for technology-enhanced learning," in *Proceedings of the 3rd International Workshop on Social Information Retrieval for Technology-Enhanced Learning (SIRTEL'09)*, Aachen, Germany, 2009.

[6] N. Manouselis, H. Drachsler, R. Vuorikari, H. G. K. Hummel, and R. Koper, *Recommender Systems Handbook*. Berlin: Springer, 2011, ch. Recommender Systems in Technology Enhanced Learning, pp. 387–415, (P. B. Kantor, F. Ricci, L. Rokach, and B. Shapira (Eds.)).

[7] R. van Zwol and B. Sigurbjörnsson, "Faceted exploration of image search results," in *Proceedings of the 19th international conference on World Wide Web (WWW'2010)*, 2010.

[8] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval: the concepts and technology behind search*, 2nd ed. Addison Wesley, 2011.

[9] R. Datta, D. Joshi, and J. Wang, "Image retrieval: Ideas, influences and trends of the new age," *ACM Computing Surveys (CSUR)*, vol. 40, no. 2, pp. 1–60, 2008.

[10] E. Bruno, J. Kludas, and S. Marchand-Maillet, "Combining multimodal preferences for multimedia information retrieval," in *Proceedings of the international workshop on Workshop on multimedia information retrieval*, 2007.

[11] M. von Wyl, H. Mohamed, E. Bruno, and S. Marchand-Maillet, "A parallel cross-modal search engine over large-scale multimedia collections with interactive relevance feedback," in *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ser. ICMR '11. ACM, 2011, pp. 73:1–73:2.

[12] F. Yan, K. Mikolajczyk, and J. Kittler, "Multiple kernel learning via distance metric learning for interactive image retrieval," in *Proceedings of the 10th international conference on Multiple classifier systems (MCS'11)*. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 147–156.

[13] D. Heesch, "A survey of browsing models for content based image retrieval," *Multimedia Tools and Applications*, vol. 40, pp. 261–284, 2008.

[14] J. Sivic and A. Zisserman, "Video google: a text retrieval approach to object matching in videos," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, oct. 2003, pp. 1470 –1477 vol.2.

[15] M. Ferecatu and D. Geman, "A statistical framework for image category search from a mental picture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 1087–1101, jun 2009.

[16] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos, "The bayesian image retrieval system, pichunter: Theory, implementation and psychophysical experiments," *IEEE transactions on image processing*, vol. 9, pp. 20–37, 2000.

[17] C. Lofi, C. Nieke, and W.-T. Balke, "Mobile product browsing using bayesian retrieval," in *12th Conference on Commerce and Enterprise Computing (CEC 2010)*, Shanghai, China, 2010, pp. 96–103.

[18] P. De Bra, L. Aroyo, and V. Chepegin, "The next big thing: Adaptive web-based systems," *Journal of Digital Information*, vol. 5, no. 1, 2004.

[19] P. De Bra, G.-J. Houben, and H. Wu, "AHAM: A Dexter-based reference model for adaptive hypermedia," in *Proceedings of the 10th ACM conference on Hypertext and Hypermedia*, Darmstadt, 1999.

[20] H. Wu, "A reference architecture for adaptive hypermedia applications"," Ph.D. dissertation, Eindhoven University of Technology, 2002.

[21] N. Koch and M. Wirsing, "The munich reference model for adaptive hypermedia applications," in *2nd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, 2002, pp. 213–222.

[22] F. Halasz and M. Schwartz, "The Dexter hypertext reference model," *Communications of the ACM*, vol. 37, no. 2, pp. 30–39, 1994.

[23] P. Brusilovsky, "Adaptive hypermedia," *User Modelling and User-Adapted Interaction*, vol. 11, pp. 87–110, 2001.

[24] N. Stash, "Incorporating cognitive/learning styles in a general-purpose adaptive hypermedia system," Ph.D. dissertation, Eindhoven University of Technology, The Netherlands, 2007.

[25] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall, "Learning a Mahalanobis metric from equivalence constraints," *Journal of Machine Learning Research*, vol. 6, pp. 937–965, 2005.

[26] B. McFee and G. Lanckriet, "Metric learning to rank," in *Proceedings of the 27th International Conference on Machine Learning (ICML'10)*, June 2010.

[27] C. Adolphs and A. Winkelmann, "A rigorous literature review on personalization research in e-commerce (2000-2008)," *Journal of Electronic Commerce Research*, vol. 11, no. 4, pp. 326–341, 2010.

[28] M. A. Hearst, "UIs for faceted navigation: Recent advances and remaining open problems," in *Workshop on Computer Interaction and Information Retrieval, HCIR*, Redmond, WA, 2008.

[29] D. Morrison, T. Tsikrika, V. Hollink, A. P. Vries, E. Bruno, and S. Marchand-Maillet, "Topic modelling of clickthrough data in image search," *Multimedia Tools and Applications*, pp. 1–23, 2012.

[30] D. H. Park, H. K. Kim, I. Y. Choi, and J. K. Kim, "A literature review and classification of recommender systems research," *Expert Systems with Applications*, 2012, (in press).

[31] J. Selke and W.-T. Balke, "Turning experience products into search products: Making user feedback count," in *13th IEEE Conference on Commerce and Enterprise Computing (CEC 2011)*, Luxembourg, Luxembourg, 2011.

[32] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 5–53, Jan. 2004.

[33] J. Zhang, *Visualization for information retrieval*, ser. The information retrieval series. Springer, 2008.

[34] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *JASIS*, vol. 41, no. 6, pp. 391–407, 1990.

[35] M. Bilenko, S. Basu, and R. J. Mooney, "Integrating constraints and metric learning in semi-supervised clustering," in *Proceedings of 21st International Conference on Machine Learning (ICML-2004)*, Banff, Canada, July 2004, pp. 81–88. [Online]. Available: http://www.cs.utexas.edu/users/ai-lab/pub-view.php?PubID=51547

[36] H. Mohamed and S. Marchand-Maillet, "Distributed media indexing based on MPI and MapReduce," in *10th Workshop on Content-Based Multimedia Indexing (CBMI 2012)*, 2012.