User Modeling and User-Adapted Interaction (2019) 29:251–290
https://doi.org/10.1007/s11257-018-9206-9

Content-based artwork recommendation: integrating painting metadata with neural and manually-engineered visual features

Pablo Messina[1,2] · **Vicente Dominguez**[1,2] · **Denis Parra**[1,2] · **Christoph Trattner**[3] · **Alvaro Soto**[1,2]

Received: 15 December 2017 / Accepted in revised form: 18 July 2018 / Published online: 27 July 2018

Abstract
Recommender Systems help us deal with information overload by suggesting relevant items based on our personal preferences. Although there is a large body of research in areas such as movies or music, artwork recommendation has received comparatively little attention, despite the continuous growth of the artwork market. Most previous research has relied on ratings and metadata, and a few recent works have exploited visual features extracted with deep neural networks (DNN) to recommend digital art. In this work, we contribute to the area of content-based artwork recommendation of physical paintings by studying the impact of the aforementioned features (artwork metadata, neural visual features), as well as manually-engineered visual features, such as naturalness, brightness and contrast. We implement and evaluate our method using transactional data from *UGallery.com*, an online artwork store. Our results show that artwork recommendations based on a hybrid combination of artist preference, curated attributes, deep neural visual features and manually-engineered visual features produce the best performance. Moreover, we discuss the trade-off between automatically obtained DNN features and manually-engineered visual features for the purpose of explainability, as well as the impact of user profile size on predictions. Our research informs the development of next-generation content-based artwork recommenders which rely on different types of data, from text to multimedia.

Keywords Artwork · Recommender systems · Content-based recommender · Hybrid recommendations · Metadata · Visual features · Deep neural networks

✉ Pablo Messina
pamessina@uc.cl

1 IMFD, Santiago, Chile

2 Pontificia Universidad Católica (PUC), Santiago, Chile

3 University of Bergen, Bergen, Norway

1 Introduction

Despite the financial crisis of 2007–2008 which shook the markets worldwide, the global artwork market has kept growing over the years. For instance, in 2011, art received $11.57 billion in total global annual revenue, over $2 billion versus 2010 (Esman 2012). Particularly, online artwork sales are booming mostly due to the influence of social media and new consumption behavior of millennials (Weinswig 2016). Online art sales reached $3.27 billion in 2015, and at the current growth rate, it will reach $9.58 billion by 2020. Notably, although many online businesses utilize recommendation systems to boost their revenue, online artwork recommendation has received little attention compared to other areas such as movies (Amatriain 2013; Gomez-Uribe and Hunt 2016) or music (Maes 1994; Celma 2010).

There are several stores nowadays that sell artworks online, such as UGallery,[1] Singulart,[2] and Artspace.[3] However, finding the right artwork for people's personal taste is a tricky task, as several properties need to be considered. Recommender systems could indeed help in this task, since previous research have been tailored explicitly towards helping people find relevant artworks, specially in the context of museum collections (Aroyo et al. 2007; Albanese et al. 2011; Semeraro et al. 2012). Most of these works have dealt with recommendation in museum collections using traditional methods and data such as ratings, textual descriptions and social tags (Aroyo et al. 2007; Albanese et al. 2011; Semeraro et al. 2012). The earliest of these works was the CHIP project (Aroyo et al. 2007), which implemented well-known techniques such as content-based and collaborative filtering for artwork recommendation in the Rijksmuseum. More recently, He et al. (2016) used pre-trained deep neural networks (DNN), combined with collaborative information, for the recommendation of digital art online. This is a very promising technique, since the development of deep neural networks has increased by orders of magnitude the performance on visual tasks such as image classification (Krizhevsky et al. 2012) or scene identification (Sharif Razavian et al. 2014). However, He et al. (2016) only studied digital art rather than physical artifacts such as paintings or sculptures, which is what most of the aforementioned online art stores sell.

Unlike the aforementioned works (Aroyo et al. 2007; Albanese et al. 2011; Semeraro et al. 2012; He et al. 2016), in this article we address the problem of artwork recommendation for one-of-a-kind paintings in online art stores. We call a painting *one-of-a-kind* when only one instance is available. If the only user feedback in the datasets are purchases, then it is not possible to compute user co-occurrences, which is needed for methods such as collaborative filtering. For this reason, we address this problem using a content-based recommender, with a focus on different types of content—including metadata, automatically learned features from deep neural networks (DNN) as well as manually-engineered visual features (MEVF)—and also on how to combine them for personalized recommendation. With respect to content-based filtering techniques, these have been extensively studied in the area of recommenda-

[1] www.ugallery.com.

[2] www.singulart.com.

[3] www.artspace.com.

tion. Most of content-based recommendation algorithms in literature rely heavily on textual data (Aggarwal 2016), and more sophisticated semantics-aware techniques draw on external knowledge sources such as ontologies and data from the Linked Data cloud (de Gemmis et al. 2015). However, the artwork recommendation problem we study is not very favorable for the application of sophisticated text-based techniques, as the metadata available, while certainly useful, is rather limited. Instead, most of the useful content-based representation comes from features obtained directly from the images, which has naturally led us to rely mainly on techniques from the domain of computer vision.

Objective In this paper, we study the impact of different features for content-based recommender systems of physical artworks. In particular, we investigate the utility of artwork metadata (curated attributes and artist), neural (DNN) and manually engineered (MEVF) visual features extracted from images as well as user transactions from the online store *UGallery*.[4] In this work, we perform two evaluations: one with an offline dataset from the *UGallery* web site, and then a small online study with 8 expert curators from *UGallery*.

Research questions To drive our research four questions were defined. They are as follows:

- *RQ1* To what extent is it possible to predict people's purchases based on content-based features? Since we have several types of content features, we answer this question by splitting the analysis into two subgroups:
  - *RQ1.1* Which is the best metadata-based feature?
  - *RQ1.2* Which is the best visual feature?
- *RQ2* How do different sets of features (metadata vs. visual) compare in the artwork recommendation domain? Although both feature sets could potentially be useful, curated metadata is not always available. Visual features, which can be calculated for every image, have then the potential to alleviate the new item problem.
- *RQ3* Is there an optimal way of combining features with hybrid methods to maximize the recommendation performance?
- *RQ4* To what extent is an offline evaluation consistent with an expert user validation?

Contributions (1) In general, the work outlined in this article makes a contribution to the yet sparsely explored problem of recommending physical artworks to people online. To make this happen, we study and compare the utility of several sources of information (content metadata, visual features), typically available in online galleries. We do this by running an extensive set of simulated experiments with real-world data provided by a large online artwork store based in CA, USA called *UGallery*. (2) Furthermore, our work contributes to the one-of-a-kind recommender system problem—i.e., items that go out of stock with the first purchase—by using a content-based approach. Also (3) we introduce a hybrid artwork recommender method, which exploits the aforementioned features. Finally, (4) we conduct an online evaluation with UGallery curators to reveal whether the offline results are mirrored when tested

[4] http://www.UGallery.com/.

with real people. To the best of our knowledge, we are the first to study the utility of pre-trained DNN visual features and how these compare to manually-engineered visual features and metadata for artwork recommendation.

Outline Section 2 presents a formal definition of the content-based artwork recommendation problem. In Sect. 3 we survey relevant related work in the area. Section 4 presents the *UGallery* dataset. Then, in Sect. 5 we provide details of our recommendation methods, following Sect. 6 with our evaluation procedure. Section 7 presents the results, we discuss them in Sect. 8, and finally Sect. 9 concludes the article and presents ideas for future work.

2 Problem statement: content-based recommendation of artworks

Based on the formulation of the recommendation problem by Adomavicius and Tuzhilin (2005), we formalize our content-based recommendation problem with the following definitions.

Let U be the set of all users and I be the set of all items (physical artworks) available in the inventory. Let s be a function which measures the utility of an item i to a user u, $s : U \times I \rightarrow R$, where R is a totally ordered set (e.g., non-negative real numbers within a certain range). In other words, a utility function s, which, given a user $u \in U$ and an item $i \in I$, returns a predicted utility score r. Now, our end goal is to identify the set R_u of "top-k items" $\{i_1 \dots i_k\}$ which maximize the utility of the user u, i.e., the list of recommended items:

$$R_u = \underset{\{i_1 \dots i_k\}}{\operatorname{argmax}} \sum_{j=1}^{k} s(u, i_j) \tag{1}$$

Due to the one-of-a-kind nature of our artwork items, once an artwork item is purchased, it is immediately removed from the system. Hence, we cannot rely directly on co-occurrence methods such as collaborative filtering, and for this reason we formulate our utility function as a content-based recommendation problem. In a content-based recommender, the utility function $s(u, i)$ in Adomavicius and Tuzhilin (2005) is defined as:

$$s(u, i) = score(ContentBasedProfile(u), Content(i)), \tag{2}$$

where $score(x, y)$ usually represents a similarity function (such as cosine or BM25 in the case of documents), and $ContentBasedProfile$ of user u and $Content$ of item i can be respectively represented as vectors, such as TF–IDF vectors using the bag-of-words document model. In our case, $ContentBasedProfile(u)$ will be the set of artworks P_u already purchased by user u. $Content(i)$ is a vector representation of the artwork i, its dimensions can represent different features. In this particular research, these features can be: (i) manually curated labels, (ii) the artist (artwork's creator), (iii) visual features extracted with pre-trained DNNs, e.g. VGG and AlexNet, and (iv) manually-engineered visual features, e.g. attractiveness features and local binary patterns (LBP).

In Sect. 5 we will explain in detail which form the function $score(x, y)$ takes depending on the different features used.

3 Related work

In this section we provide an overview of relevant related work. The section is split into two parts: *Artwork Recommender Systems* (3.1) and *Visually-aware Recommender Systems* (3.2). Both sub-sections are important to better understand our contribution and the problem we are targeting with the paper. A final Sect. *Differences to Previous Research* (3.3) highlights what we add with our work to the already existing literature in the area.

3.1 Artwork recommender systems

Within the topic of artwork recommender systems, one of the first contributions in this area was made by the CHIP Project (Aroyo et al. 2007). The aim of the project was to build a recommendation infrastructure for the Rijksmuseum in the Netherlands. The project used several techniques such as content-based filtering based on metadata provided by experts, as well as collaborative filtering based on users' ratings given to artworks of the Rijksmuseum.

Another important contribution in the field is the work developed by Semeraro et al. (2012). In their paper, they introduce an artwork recommender system called FIRSt (Folksonomy-based Item Recommender syStem) which utilizes social tags given by experts and non-experts over 65 paintings of the Vatican picture gallery. They focused their research on making recommendations using textual features (textual painting descriptions and user tags), but did not employ visual features among their methods.

More complex methods were implemented recently by Benouaret and Lenne (2015), who improve the current state-of-the-art in artwork recommender systems using context obtained through a mobile application. The particular research question they address is to what extent it is possible to make museum tour recommendations more useful. Their content-based approach uses ratings applied by the users during the tour and metadata from the artworks people have rated, e.g. title or artist names. They address the artwork recommendation problem in museums, yet their solution cannot be fully applied to the *one-of-a-kind* problem in online stores as we approach it in this research.

Finally, the recent work of He et al. (2016) addresses digital artwork recommendations based on pre-trained deep neural visual features. In this case, the experiments were conducted on a virtual art gallery, with the advantage of items always available and explicit user feedback in the form of ratings.

3.2 Visually-aware recommender systems

Manually-engineered visual features extracted from images (texture, sharpness, brightness, etc.) have been used in several tasks for information filtering, such as

retrieval (Rui et al. 1998; La Cascia et al. 1998) and ranking (San Pedro and Siersdorfer 2009). More recently, very promising results have been shown for the use of low-level handcrafted stylistic visual features automatically extracted from video frames for content-based video recommendation (Deldjoo et al. 2016). By extracting and aggregating five stylistic visual features per video and using cosine similarity for pairwise comparison, Deldjoo et al. achieved higher recommendation accuracy than traditional recommendation methods based on high-level expert annotated metadata. Even better results are obtained when both stylistic visual features and annotated metadata are combined in a hybrid recommender, as shown in the work of Elahi et al. (2017).

In the latest years, many works in image processing and computer vision such as object recognition (Akay et al. 2016), image classification (Krizhevsky et al. 2012) and scene identification (Sharif Razavian et al. 2014) have shown significant performance improvements by using visual embeddings obtained from pre-trained deep convolutional neural networks (Deep CNN) such as AlexNet introduced by Krizhevsky et al. (2012) or VGG by Simonyan and Zisserman (2014). These are examples of transfer learning methods, i.e., visual embeddings trained for specific tasks (e.g. image classification) which perform well in other tasks (e.g. image segmentation) and have been adopted for the recommendation problem.

Motivated by these results, McAuley et al. (2015) introduced an image-based recommendation system based on styles and substitutes for clothing using visual embeddings pre-trained on a large-scale dataset obtained from Amazon.com. Recently, He and McAuley (2016) went further in this line of research and introduced a visually-aware matrix factorization approach that incorporates visual signals (from a pre-trained DNN) into predictors of people's opinions. Their training model is based on Bayesian Personalized Ranking (BPR), a model previously introduced by Rendle et al. (2009).

The latest work by He et al. (2016) deals with visually-aware artistic recommendation, building a model which combines ratings, social signals and visual features. Another relevant work was the research by Lei et al. (2016) who introduced comparative deep learning for hybrid image recommendation. In this work, they use a neural network architecture for making recommendations of images using user information (such as demographics and social tags) as well as images in pairs (one liked, one disliked) in order to build a ranking model. The approach is interesting, but they work with regular images, not artwork images.

3.3 Differences to previous research

Almost all the surveyed articles on artwork recommendation have in common that they used standard techniques such as collaborative filtering and content-based filtering, but without exploiting visual features extracted from images. Unlike these works, we rely exclusively on content-based methods. We are unable to use traditional collaborative filtering, since there are no ratings or implicit feedback on the same item: once an item is purchased, it is out of stock due to its one-of-a-kind condition. In terms of content-based filtering, unlike the previous works we extract, compare and combine metadata, neural visual features and manually-engineered visual features.

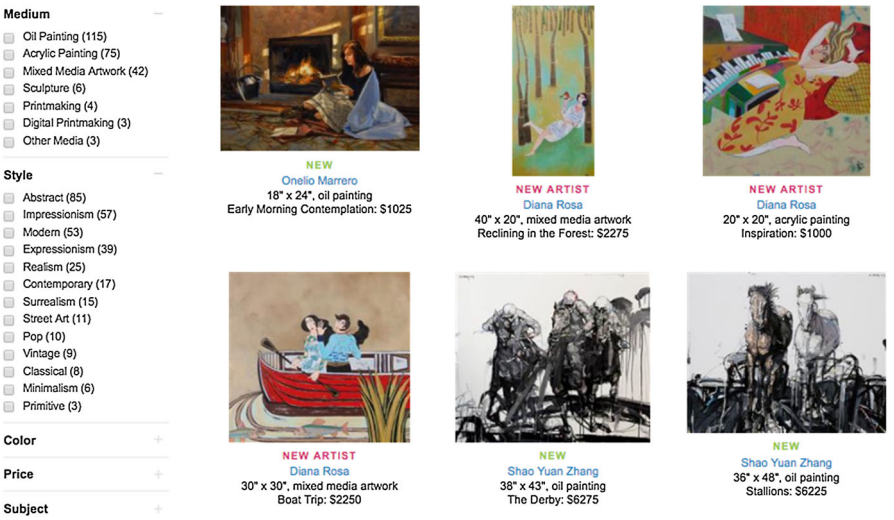

Fig. 1 Screenshot of the search interface of *UGallery*. Users can filter by different facets on the left side

With regards to the related work on visually-aware recommender systems, almost all of the surveyed articles have focused on tasks different from artwork recommendation, such as clothing recommendation and video recommendation.

Only one work, the research by He et al. (2016) resembles ours in terms of the topic (artwork recommendation) and the use of visual features. However, there are several important differences: (i) First, although they do use visual features from DNN embeddings, they do not use manually-engineered visual features, such as brightness or sharpness. (ii) Second, in addition to visual features, we also consider artwork metadata (artwork artists and curated attributes). (iii) Third, our research deals with physical (real-world) artworks, not digital art. Hence, when an artwork is sold, it goes out of stock, whereas in the work of He et al. the digital artworks can be "copied" to an unlimited amount. For us this is a big impediment to using collaborative filtering, which is why our research focuses on content-based recommendation instead. (iv) And fourth, in our work we also perform an online evaluation with expert curators to verify consistency with offline evaluation results.

4 Materials

The online web store *UGallery* has been selling artworks for more than 10 years (Weinswig 2016). They support emergent artists by helping them sell their artworks online. The *UGallery* website allows users (customers) to search for items and to browse the catalog based on different attributes with a predefined order: orientation, size, medium, style and others, as seen on the left side of Fig. 1. However, what their current system does not support is the exploration of items via personalized recommendations, which is exactly what we aim for in this paper.

UGallery provided us with an anonymized dataset of 1371 users, 3490 items and 2846 purchases (transactions) of artistic artifacts, where all users have made at least

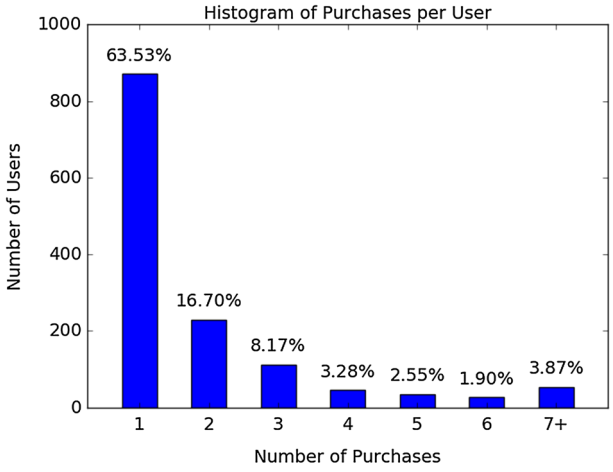

Fig. 2 Distribution of purchases per user. It resembles the typical skewed user consumption behavior in online websites

one transaction. In average, each user has bought 2–3 items in the latest years.[5] Figure 2 shows the distribution of purchases per user. The distribution is skewed since most users (871 in total) bought only one item, and only a few users (53 in total) have bought 7 or more items. Our data is not atypical, since it resembles the rating distribution of the Netflix prize or the Movielens dataset, where a few users account for most of the activity and most users have little or none (Harper and Konstan 2015; Bennett et al. 2007).

The artworks in the *UGallery* dataset were manually curated by experts. Hence, every artwork has been described with metadata *attributes* such as color, style and medium, to enable the user to filter and browse in the *UGallery* interface. In total, there are eleven attributes, which are described with their respective *attribute values* in Table 1. The attributes in rows 1–4 (*Color* to *Medium*) are self-explaining by reading the examples. Attributes in rows 5–11 (*Energy* to *Age Perception*) are grouped into a meta-category called *Mood*. It is important to note that only from the very latest years onwards the artworks started being filled with all their attributes more systematically. As such, there is a distribution of attributes present and absent in the artworks, which is shown in Table 2. While *Color* (97.16%) is present in almost all the artworks, *Subject* is only present in 16,56%. In addition to these curated attributes, the artwork metadata also includes another important source of information: the artwork's artist. In the *UGallery* dataset, each artwork is associated to a unique artist. In total, there are 423 artists, who have 8.25 artworks in average each for sale.

5 Artwork recommendation approaches

In this section we describe six different content-based artwork recommendation approaches, which we have implemented to tackle the one-of-a-kind recommendation problem. Table 3 contains an overview of symbols used in the following sub-sections.

[5] Our collaborators at UGallery requested us not to disclose the exact dates when the data was collected.

Table 1 Metadata attributes and attribute values for artworks in the *UGallery* dataset

Attribute	Type	Values
Color	Nominal	B&W, Beige, Black, Blue, Brown, Dark Blue, Dark Green, Dark Red, Green, Grey, Orange, Pink, Purple, Red, Turquoise, Violet, White, Yellow
Subject	Nominal	Animals, Architecture, Cuisine, Fantasy, Fashion, Flora, Landscape, Nature, Nudes, People, Religion, Seascape, Sports, Still Life, Travel, Western
Style	Nominal	Abstract, Classical, Expressionism, Impressionism, Minimalism, Modern , Non-representational, Pop, Primitive, Realism, Representational, Street Art, Street Photography, Surrealism, Vintage
Medium	Nominal	Acrylic Painting, Ceramic Artwork, Chalk Drawing, Charcoal Drawing, Colored Pencil, Digital Printmaking, Drawing Artwork, Encaustic Artwork, Gouache Painting, Ink Artwork, Marker Artwork, Mixed Media Artwork, Oil Painting, Other Media, Pastel Artwork, Pencil Drawing, Photography, Printmaking, Sculpture, Watercolor
Energy	Ordinal	Calm, Neutral, Energetic
Seriousness	Ordinal	Playful, Neutral, Serious
Warmness	Ordinal	Warm, Neutral, Cool
Purpose	Ordinal	Decorative, Neutral, Thought-Provoking
Complexity	Ordinal	Simple, Neutral, Complex
Formality	Ordinal	Formal, Neutral, Informal
Age perception	Ordinal	Young, Neutral, Old

Table 2 Statistics of attributes' presence among artworks in the *UGallery* dataset

	Color	Style	Subject	Mood	Medium
Present	3391 (97.16%)	646 (18.51%)	578 (16.56%)	1550 (44.41%)	3490 (100%)

5.1 Most popular curated attribute value (MPCAV)

The Most Popular Curated Attribute Value method is the first and most simple approach we tested. Together with Random, it is also used and referred to as a baseline throughout our paper. Since the concept of "popular item" is meaningless in a *one-of-a-kind* setting, instead we recommend based on the most popular *curated attribute values*. Given an artwork i and CAV_i^X the corresponding set of curated attribute values (where X can be either *Color*, *Subject*, *Style*, *Medium*, *Mood* or *All*), we compute the MPCAV score as the sum of the frequencies (popularities) of each of its curated attribute values. More formally, the MPCAV score is calculated as follows:

$$score(i)_{MPCAV} = \sum_{v \in CAV_i^X} \sum_{j \in P} \mathbb{1}(j, v), \tag{3}$$

Table 3 Symbols used in our artwork recommendation approaches

Symbol	Description
U, I	User set, item set
u, i	A specific user or item (resp.)
P	Set of all items purchased in the system up to an arbitrary point in time
P_u	Set of all items purchased by user u up to an arbitrary point in time, we refer to these items as the *user profile* or the *user model*, indistinctly
CAV_i^X	Set of all curated attribute values of type X present in item i, where X can be either *Color*, *Subject*, *Style*, *Medium*, *Mood* or *All* (all curated attributes at the same time)
a_i	The artist (creator) of item i
V_i	Vector of visual features of item i, either manually engineered or obtained with a pre-trained DNN
V_i^X	Vector of visual features (of item i) of the specific type X (where X can be e.g. *AlexNet*, *VGG*, *LBP* or *Attractiveness*)

where P is the set of products purchased so far, and $\mathbb{1}(j, v)$ is an indicator function, which returns 1 if item j has curated attribute value v or 0 otherwise. Intuitively, an item will have a higher score if its curated attribute values are more frequent (popular) among items already purchased in the system. Finally, we rank the items based on this score and recommend the top-n.

Because of the low granularity of the curated attribute values (which at least was the case with the UGallery dataset), one problem of this scoring function is that it may be prone to ties, i.e. many items with the same score. Therefore if there are too many items with the same score that do not fit into the top-n limit, as a workaround we uniformly sample a subset of these items just to fit the top-n recommendation.

5.2 Personalized most popular curated attribute value (PMPCAV)

This method is equivalent to MPCAV, with the only difference that we just look at the past purchases of user u instead of the past purchases of the whole system. More formally, the formula for the PMPCAV scoring function is:

$$score(u, i)_{PMPCAV} = \sum_{v \in CAV_i^X} \sum_{j \in P_u} \mathbb{1}(j, v), \tag{4}$$

which is almost exactly as Eq. 3, but here we consider only the set of items purchased by the user u, i.e., the set P_u. Then we can rank items and recommend the top-n based on this score. In case of ties, the same workaround as in MPCAV can be used (uniform sampling). On the other hand, if we are not able to build a user model because the user's purchased items lack proper tagging, a possible fallback option is to switch to MPCAV.

A weakness of this method compared to MPCAV is that it requires at least one previous purchase from the user to make recommendations. On the positive side, by considering the user's preferences, one should expect more accurate recommendations.

5.3 Personalized favorite artist (FA)

Besides curated attributes, the artwork metadata also includes another important source of information: the artist who created the painting. The FA method leverages this information by recommending artworks created by artists that the user has shown favoritism for. More formally, given a user u and an item i, the FA scoring function is defined as follows:

$$score(u, i)_{FA} = \sum_{j \in P_u} \mathbb{1}(j, a_i), \tag{5}$$

where $\mathbb{1}(j, a_i)$ is an indicator function that returns 1, if the artist a_i of artwork i is also the creator of artwork j (in our dataset, each artwork is associated to a single creator). Intuitively, an artwork has a higher score if the user has purchased more artworks from the same artist in the past. Then we rank and recommend the top-n artworks based on this score. If there are too many items with the same score, a subset of these items can be uniformly sampled to fit the top-n recommendation. On the other hand, if there are too few items with a positive score to recommend (e.g. because the user's favorite artists have sold almost all their artworks), we resort to the globally most favorite artists to rank the remaining artworks and fill the top-n recommendation.

5.4 Latent visual features: deep neural network embedding (DNN)

Since the dataset contains one image for every item, we tested visual features for artwork recommendation. One of the two visual embeddings used was a vector of features obtained from an AlexNet, a convolutional deep neural network developed to classify images (Krizhevsky et al. 2012). In particular, we use an AlexNet model pre-trained with the ImageNet dataset (Deng et al. 2009). Using the pre-trained weights for every image a vector of 4096 dimensions was generated with the Caffe[6] framework. As seen in Fig. 3, this vector corresponds to the output of the first fully connected layer of AlexNet, also known as fc6.

Although there are two fully connected layers (fc6 and fc7) we used fc6 rather than fc7 because previous works show better performance of this layer in a transfer learning setting, e.g., classifying regions using an embedding trained for a different task, object classification (Girshick et al. 2014). Our task is also transfer learning, since we are using an embedding originally trained for object classification, when our goal is recommendation. Figure 3 shows the architecture and the procedure to obtain the features from fc6.

We also tested the Visual Geometry Group (VGG) network (Simonyan and Zisserman 2014), a newer deep neural network architecture used to classify images. This

[6] http://caffe.berkeleyvision.org/.

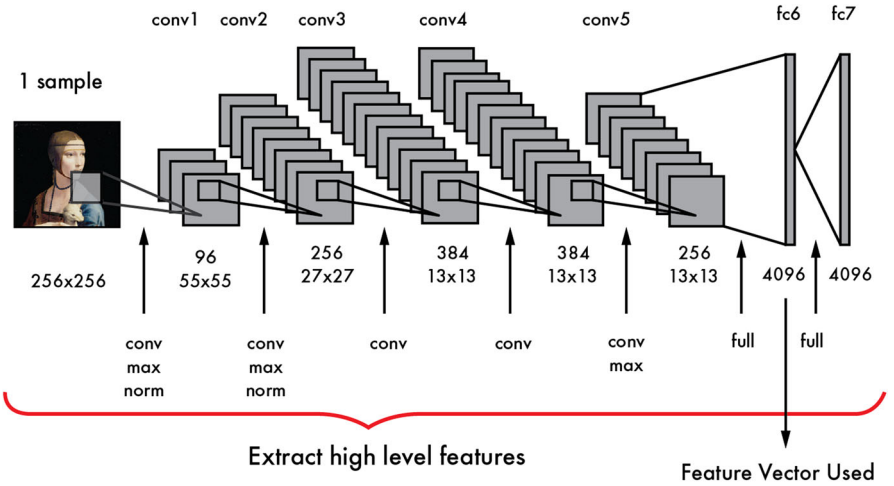

Fig. 3 AlexNet architecture. This shows the process to obtain the latent feature vector we use in our experiments, which corresponds to fc6. A convolutional window passes over the image, from each layer to the next layer, with different shapes and strides in every layer. This figure is inspired by Karnowski (2015)

network outperformed the results obtained by the AlexNet (Simonyan and Zisserman 2014), reaching even human level of performance in the task of image classification (Russakovsky et al. 2015), so it seemed reasonable to put this network to the test in the task of artwork recommendation as well. We used the first fully connected layer of this network, also known as fc14, to obtain a $\mathbb{R}^{4096}$ feature vector for each image.

DNN utility score We make recommendations by maximizing the utility score that an item provides to a user. Given a user u who has consumed a set of artworks P_u, and an arbitrary artwork i from the inventory, the score of this item i to be recommended to u is defined as:

$$score(u,i)_X = \begin{cases} \max\limits_{j \in P_u} \{sim(V_i^X, V_j^X)\} & (maximum) \\ \dfrac{\sum\limits_{j \in P_u} sim(V_i^X, V_j^X)}{\min\{K, |P_u|\}} & (average) \\ \dfrac{\sum\limits_{r=1}^{\min\{K, |P_u|\}} \max\limits_{j \in P_u}{}^{(r)} \{sim(V_i^X, V_j^X)\}}{\min\{K, |P_u|\}} & (average\ top\ K) \end{cases}, \quad (6)$$

where V_z^X is a feature vector of type X associated to item z. In this particular case V_z^X stands for the vector embedding of item z obtained with a pre-trained DNN of type X, where X can be either VGG or AlexNet. $\max^{(r)}$ denotes the r-th maximum value, e.g. if $r = 1$ it is the overall maximum, if $r = 2$ it is the second maximum, and so on. $sim(V_i, V_j)$ denotes a similarity function between vectors V_i and V_j. In this particular case, the similarity function used was cosine similarity, expressed as:

$$sim(V_i, V_j) = cos(V_i, V_j) = \frac{V_i \cdot V_j}{\|V_i\| \|V_j\|} \quad (7)$$

Essentially, the score in Eq. 6 looks at the similarity between item i and each item j in the user profile P_u, and then aggregates these similarities in three possible ways: taking either (a) the maximum, (b) the average or (c) the average of the top-K most similar items, where K can be tuned empirically.

In addition, we also studied the performance of using both DNNs at the same time. For this purpose, we implemented the following hybrid score:

$$score(u,i)_{DNN} = \alpha_1 \cdot score(u,i)_{VGG} + \alpha_2 \cdot score(u,i)_{AlexNet}, \tag{8}$$

where $score(u,i)_{VGG}$ and $score(u,i)_{AlexNet}$ are calculated following Eqs. 6 and 7, using VGG and AlexNet feature vectors, respectively, and α_1 and α_2 are weights to perform the linear combination between the two scores. After an optimization of the weights by grid search, this hybrid approach produced the best results, where the optimal values were $\alpha_1 = 0.8$ and $\alpha_2 = 0.2$.

5.5 Manually engineered visual features (MEVF)

The visual features obtained with DNN techniques are of latent nature, i.e., they are not easily interpretable in terms of more intuitive features such as image colorfulness or brightness. To mitigate this problem, one might want to take advantage of manually engineered visual features, which usually are much more intuitive and explainable than neural features. Moreover, they are suitable to be used in a search interface to support navigation. For example, imagine a use case where a content-based recommender uses the brightness of an image to find similar items. This information could be used to make an explanation—*you might like this image because of its brightness*—or to allow the user to filter search results based on the paintings' level of brightness.

In order to choose which visual features to extract, we surveyed related work and found features related to *attractiveness* as potentially useful.

Attractiveness San Pedro and Siersdorfer in San Pedro and Siersdorfer (2009) proposed several explainable visual features that can capture to a great extent the attractiveness of an image posted on Flickr. Following their procedure, for every image in our *UGallery* dataset we calculated: (a) average brightness, (b) saturation, (c) sharpness, (d) RMS-contrast, (e) colorfulness and (f) naturalness. In addition, we added (g) entropy, which is a good way to characterize and measure the texture of an image (Gonzalez et al. 2004). These metrics have also been used in another study (Trattner and Elsweiler 2017), where they are successfully used to nudge people with attractive images to take up more healthy recipe recommendations.

Since each feature varies within different value ranges (e.g. 0–1, 10–100), we applied a feature-wise min-max normalization to prevent biases in similarity calculations. Following, we provide a more detailed description of these attractiveness-based features:

- *Brightness* measures the level of luminance of an image. For images in the *YUV* color space, we obtain the average of the luminance component Y as follows:

$$B = \frac{1}{N} \sum_{x,y} Y_{x,y}, \tag{9}$$

where N is the amount of pixels and $Y_{x,y}$ is the value of the luminance in the pixel (x, y)

- *Saturation* measures the vividness of an image. For images in the *HSV* or *HSL* color space, we obtain the average of the saturation component S as follows:

$$S = \frac{1}{N} \sum_{x,y} S_{x,y}, \tag{10}$$

where N is the amount of pixels and $S_{x,y}$ is the value of the saturation in the pixel (x, y)

- *Sharpness* measures the detail level of an image. For an image in gray-scale, it can be obtained using a Laplacian filter and luminance around every pixel:

$$L(x, y) = \frac{\delta^2 I}{\delta x^2} + \frac{\delta^2 I}{\delta y^2} \tag{11}$$

$$Sh = \frac{\sum_{x,y} \frac{L(x,y)}{\mu_{x,y}}}{n}, \tag{12}$$

where n is the number of pixels and $\mu_{x,y}$ is the average luminance of the pixels around the pixel (x, y).

- *Colorfulness* measures how distant the colors are from the gray color. For images in the RGB space, it can be obtained with the following formulas:

$$C = \sigma_{rgyb} + 0.3 \cdot \mu_{rgyb} \tag{13}$$

$$\sigma_{rgyb} = \sqrt{\sigma_{rg}^2 + \sigma_{yb}^2} \tag{14}$$

$$\mu_{rgyb} = \sqrt{\mu_{rg}^2 + \mu_{yb}^2}, \tag{15}$$

where μ_{rg}^2, μ_{yb}^2 are the means of the components of the opponent color space. σ_{rg}^2, σ_{yb}^2 are the standard deviations of the component of opponent color space. This color space is defined as:

$$rg = R - G \tag{16}$$

$$yb = \frac{1}{2}(R + G) - B \tag{17}$$

- *Naturalness* measures the naturalness of an image by grouping the pixels into Sky, Grass and Skins pixels and applying the formula in San Pedro and Siersdorfer

(2009). First, using the HSL color space, the pixels are filtered considering only the ones with $20 \leq L \leq 80$ and $S > 0.1$. Then, they are grouped by their hue value in three classes "A - Skin", "B - Grass" and "C - Sky", which are defined as follows:

- pixels with $25 \leq hue \leq 70$ belong to the "A - Skin" set.
- pixels with $95 \leq hue \leq 135$ belong to the "B - Grass" set.
- pixels with $185 \leq hue \leq 260$ belong to the "C - Sky" set.

For each set, average saturation is calculated and denoted as μ_S. Then, local naturalness for each set is calculated using the following formulas:

$$N_{skin} = e^{-0.5\left(\frac{\mu_S^A - 0.76}{0.52}\right)^2} \tag{18}$$

$$N_{Grass} = e^{-0.5\left(\frac{\mu_S^B - 0.81}{0.53}\right)^2} \tag{19}$$

$$N_{Sky} = e^{-0.5\left(\frac{\mu_S^C - 0.43}{0.22}\right)^2} \tag{20}$$

After this, the Naturalness value is obtained by:

$$Na = \sum_i \omega_i N_i, \; i \in \{\text{“}Skin\text{”}, \text{“}Grass\text{”}, \text{“}Sky\text{”}\}, \tag{21}$$

where ω_i is the amount of pixels of set i divided by the total pixels in the image.

- *RMS-contrast* measures the variance of luminance in an image using the intensity of each pixel:

$$C^{rms} = \frac{1}{n} \sum_{x,y}^{n} (I_{x,y} - \bar{I}),$$

where $I_{x,y}$ is the intensity of the pixel (x, y) and $\bar{I}$ is the average intensity.

- *Entropy* The entropy of a gray-scale image is a way to measure and characterize the texture of the image (Gonzalez et al. 2004). Shannon's entropy is applied to the histogram of values of every pixel in a gray-scale image. The formula is defined as follows:

$$E = - \sum_{x \in [0..255]} p(x) \log p(x), \tag{22}$$

where $p(x)$ is the probability of finding the gray-scale value x among all the pixels in the image.

Attractiveness utility scores For the attractiveness features we studied the performance of (i) using each feature individually and (ii) using all features together. For the

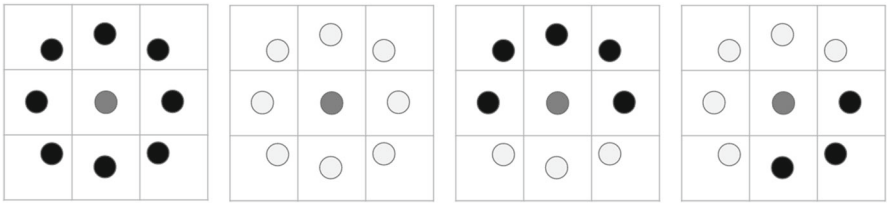

Fig. 4 Examples of pixelwise patterns extracted with local binary patterns (LBP). Each small square is a pixel, and these boxes with nine pixels each represent patterns. The black circles represent pixels with value over a threshold (= 1), while gray circles represent pixels with value below a threshold (= 0). The threshold is set by the value of the pixel in the center of the pattern

first case, we used $\mathbb{R}^1$ vectors of one single feature at a time. To calculate the similarity between two $\mathbb{R}^1$ vectors, we used Euclidean distance, formally expressed as:

$$sim\left(V_i^X, V_j^X\right) = \left\| V_i^X - V_j^X \right\|, \tag{23}$$

where V_i^X and V_j^X are $\mathbb{R}^1$ vectors of items i and j, respectively, containing a single feature of type X (where X can be either *average brightness*, *saturation*, *sharpness*, *RMS-contrast*, *colorfulness*, *naturalness* or *entropy*).

For the second case (all features together), we put the 7 attractiveness-based features into a single $\mathbb{R}^7$ vector, which we denote as $V_i^{Attract}$. Then, to calculate the similarity between two vectors $V_i^{Attract}$ and $V_j^{Attract}$ we used cosine similarity, as per Eq. 7:

$$sim\left(V_i^{Attract}, V_j^{Attract}\right) = cos(V_i^{Attract}, V_j^{Attract}) \tag{24}$$

As for the utility score $(score(u, i)_X)$ itself, we used the same similarity aggregation techniques outlined in Eq. 6 (*maximum*, *average* and *average-top-k*). This applies for both (i) $\mathbb{R}^1$ vectors of single features and (ii) $\mathbb{R}^7$ vectors with all attractiveness features, using the corresponding similarity function in each case.

LBP Another set of features we explored apart from those of attractiveness were the *Local Binary Patterns* (LBP) (Ojala et al. 1996). Although this is not an actual "explicit" visual feature, it is a traditional baseline in several computer vision tasks such as image classification, so we tested it for the task of recommendation, too. LBP is not represented as a scalar value, but rather as a feature vector of 59 dimensions. The values in the LBP feature vector represent counts in a histogram of the patterns found on an image. Figure 4 shows four of such patterns as example.

LBP utility score Since the output of LBP is a feature vector, we calculated the similarity between two vectors V_i^{LBP} and V_j^{LBP} as we did with most of the feature vectors, using cosine similarity (7). Namely:

$$sim\left(V_i^{LBP}, V_j^{LBP}\right) = cos\left(V_i^{LBP}, V_j^{LBP}\right) \tag{25}$$

Finally, the utility score $(score(u, i)_{LBP})$ is calculated using the same similarity aggregation techniques outlined in Eq. 6: *maximum*, *average* and *average-top-k*.

MEVF hybrid utility score In addition to studying Attractiveness and LBP separately, we also studied the performance of using both feature sets at the same time. We tried two ways to combine the features: (i) concatenating Attractiveness ($\mathbb{R}^7$) and LBP ($\mathbb{R}^{59}$) into a single $\mathbb{R}^{66}$ vector and then recommending based on Eqs. 6 and 7, and (ii) computing a relevance score for Attractiveness and LBP separately and then merging the two scores with a convex linear combination based on Eq. 8. As we will show in Sect. 7, this hybrid approach achieved the best results.

5.6 Hybrid recommendations (hybrid)

Since different methods can measure different sources of similarity between items and the user profile, we developed a hybrid recommender model which integrates the previous approaches. The basic idea is to compute a hybrid score as a convex linear combination of the scores of individual methods. We took the best performing version of each individual method and tested multiple hybrid combinations of them.

Formally, given a user u who has purchased a set of artworks P_u, and an arbitrary artwork i from the inventory, we compute the hybrid score of item i for user u as a convex linear combination of multiple scores, which for the case of combining all features is given by:

$$\begin{aligned} score(u,i)_{Hybrid} = {} & \beta_1 \cdot score(u,i)_{\text{FA}} \\ & + \beta_2 \cdot score(u,i)_{\text{VGG}} \\ & + \beta_3 \cdot score(u,i)_{\text{AlexNet}} \\ & + \beta_4 \cdot score(u,i)_{\text{LBP}} \\ & + \beta_5 \cdot score(u,i)_{\text{Attract}} \\ & + \beta_6 \cdot score(u,i)_{\text{PMPCAV}}, \end{aligned} \tag{26}$$

where β are global (non-personalized) coefficients such that $0 \leqslant \beta_i \leqslant 1$ and $\sum_i \beta_i = 1$. The β coefficients were tuned by exhaustive grid search, and in the case of the hybrid with all features the best coefficients found were $\beta_1 = 0.207$, $\beta_2 = 0.269$, $\beta_3 = 0.165$, $\beta_4 = 0.145$, $\beta_5 = 0.062$ and $\beta_6 = 0.153$.[7] In the equation, $score(u,i)_{\text{VGG}}$, $score(u,i)_{\text{AlexNet}}$, $score(u,i)_{\text{LBP}}$ and $score(u,i)_{\text{Attract}}$ are calculated as in Eq. 6. Meanwhile, $score(u,i)_{\text{PMPCAV}}$ and $score(u,i)_{\text{FA}}$ had to be slightly modified to ensure normalized values in the range [0, 1]:

$$score(u,i)_{\text{PMPCAV}} = \frac{\sum_{v \in CAV_i^{All}} \sum_{j \in P_u} \mathbb{1}(j,v)}{\sum_{j \in P_u} |CAV_j^{All}|} \tag{27}$$

$$score(u,i)_{\text{FA}} = \frac{\sum_{j \in P_u} \mathbb{1}(j,a_i)}{|P_u|}, \tag{28}$$

[7] To obtain the weights for the different methods, we initialize the coefficients based on the individual performance (concretely, Recall@10) of each method and then we iterate with a grid search, each time narrowing the weight search space in a greedy fashion. The performance tend to converge after two to three iterations.

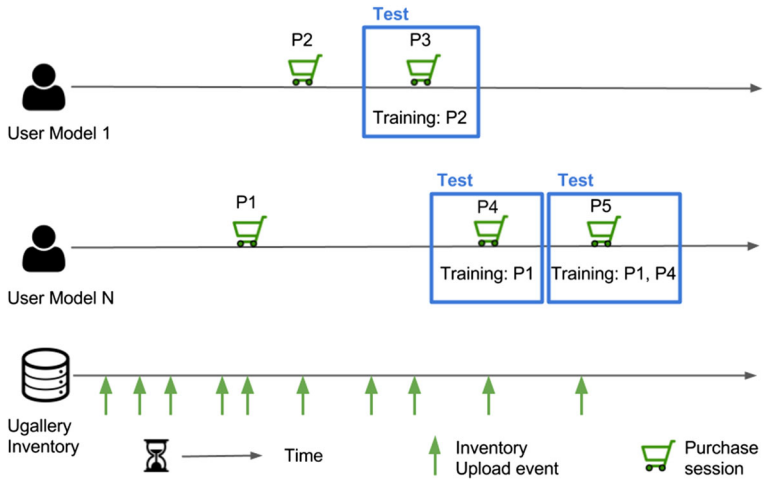

Fig. 5 offline evaluation procedure. Each surrounding box represents a test, where we predict the items of the purchase session. In the figure, we predict which artworks User 1 bought in purchase P3. 'Training:P2' means we used items from purchase session P2 to train the model

which are almost the same as Eqs. 4 and 5 but with the addition of a normalizing denominator that represents the theoretical maximum of the score in each case.

6 Evaluation methodology

The evaluation had two stages. The first was an offline evaluation, conducted using a dataset of transactions (purchases) as described in Sect. 4. With this offline evaluation we can answer research questions RQ1, RQ2 and RQ3. The second stage was performed with expert curators from the UGallery store. We developed a web interface where the experts could rate recommendations based on algorithms selected from the offline evaluation, and we analyzed consistency between results of both stages (RQ4).

6.1 Offline evaluation

The evaluation protocol we follow in this paper is the one usually used in order to evaluate predictive models and recommender systems offline in a time-based manner (Macedo et al. 2015). Hence, the *UGallery* dataset was split into training and test samples according to the time line of every user, as seen in Fig. 5. With this setting, we attempt to predict the items purchased by the user in every transaction, where the training set contains all the artworks bought by a user previous to the transaction to be predicted.

Figure 5 shows that for every user we test the predictions made for every purchase session excepting the first one of each user. For instance, for User 1 we tested the predicted items of purchase *P3* using items in *P2* as training. In the same Figure, for User N we performed two predictions tasks: the first one predicting items bought in

Fig. 6 Screenshot of the upper part of the interface used in the expert evaluation. On the left the items liked by the user. The large table to the right shows one column per each method used to make recommendations

purchase *P4* using *P1* as training, and then testing a prediction on purchase *P5* using *P1* and *P4* as training. In our evaluation, most of the experiments considered only users who had at least 2 purchase sessions. Users who only had a single purchase session in their whole history were considered *cold start* users (only MPCAV and Random were able to make predictions in those cases, since they are non-personalized methods).

6.2 Online evaluation

The online evaluation involved 8 expert curators from UGallery. We asked each expert to send us a list of 10 of their preferred paintings from the current UGallery dataset, which they sent us via email. For each expert we created five lists of recommendations based on different methods: FA, MEVF, DNN, and the hybrids DNN + MEVF, and FA + DNN + MEVF. Each recommendation list had 10 items, and the experts had to rate each painting recommended with stars in a scale from 1 to 5. We used ratings rather than likes/dislikes to evaluate the recommendations in order to give experts the chance to express their perception of relevance with higher granularity. Unlike regular art consumers for which a preference rating of two or three stars might be hard to discriminate, experts are more likely to understand detailed levels of relevance of the paintings recommended. In total, each expert rated 50 items. A screenshot of the rating interface for a ficticious user called "Madeline" is shown in Fig. 6. We stored the user id, item id and the ratings over every painting for each method, to calculate the evaluation metrics and compare the results.

6.3 Evaluation metrics

Table 4 shows a summary of symbols used in this section. As suggested by Cremonesi et al. (2010) for Top-N recommendation, for our offline evaluations we used Recall@k ($R@k$), Precision@k ($P@k$) and F1-score@k ($F1@k$), as shown in the equations below:

$$p@k(t) = \frac{|r_t^k \cap R_t|}{k} \tag{29}$$

Table 4 Evaluation metrics symbol table

Symbol	Description
t	A test case during the execution of an offline evaluation of a certain recommendation algorithm
u_t	User whose shopping basket is predicted during offline test case t
r_t^k	List of top-k items recommended to user u_t at offline test case t
R_t	The set of relevant items (i.e. items in the shopping basket) of user u_t during offline test case t
T_u	The set of all test cases performed with purchase sessions of user u
U_r	Set of all users who received at least 1 recommendation during a certain offline evaluation (i.e., all $u \in U$ such that $\|T_u\| \geq 1$)
$i_{t,z}$	Item appearing at position z in the recommended list at offline test t
vc_i	The visual cluster that item i belongs to
PS	Total number of purchase sessions in the system

$$r@k(t) = \frac{|r_t^k \cap R_t|}{|R_t|} \tag{30}$$

$$f1@k(t) = 2 \cdot \frac{p@k(t) \cdot r@k(t)}{p@k(t) + r@k(t)} \tag{31}$$

$$P@k = \frac{1}{|U_r|} \sum_{u \in U_r} \left(\frac{1}{|T_u|} \sum_{t \in T_u} p@k(t) \right) \tag{32}$$

$$R@k = \frac{1}{|U_r|} \sum_{u \in U_r} \left(\frac{1}{|T_u|} \sum_{t \in T_u} r@k(t) \right) \tag{33}$$

$$F1@k = \frac{1}{|U_r|} \sum_{u \in U_r} \left(\frac{1}{|T_u|} \sum_{t \in T_u} f1@k(t) \right), \tag{34}$$

where $p@k(t)$, $r@k(t)$ and $f1@k(t)$ are precision, recall and f1-score at k, respectively, measured during the test case t, whereas $P@k$, $R@k$ and $F1@k$ are the overall aggregations of precision, recall and f1-score at k, respectively, by first calculating user averages and then the average of these averages. These are the evaluation metrics that we report in Sect. 7.

In addition, we also report *Normalized Discounted Cumulative Gain (nDCG)* (Manning et al. 2008) which is a ranking-dependent metric that not only measures how relevant the items are but also takes the position of the items in the recommended list into account. The *nDCG* metric with a cut-off of k items in the recommended list is based on the *Discounted Cumulative Gain (DCG@k)* which is defined as follows:

$$DCG@k(t) = \sum_{z=1}^{k} \frac{2^{B_t(i_{t,z})} - 1}{log_2(1+z)}, \tag{35}$$

where $B_t(i_{t,z})$ is a function that returns the graded relevance of item $i_{t,z}$ appearing at position z in the recommended list during the test case t. In our case, $B_t(i_{t,z})$ basically returns 1 if item $i_{t,z}$ was present in the shopping basket of test case t, and 0 otherwise. $nD@k$ is calculated as $DCG@k$ divided by the ideal $DCG@k$ value $iDCG@k$ which is the highest possible $DCG@k$ value that can be achieved if all the relevant items were recommended in the correct order (i.e., all shopping basket items appearing first in the recommended list). Taken together, the overall $nDCG@k$ is defined as follows:

$$nD@k = \frac{1}{|U_r|} \sum_{u \in U_r} \left(\frac{1}{|T_u|} \sum_{t \in T_u} \frac{DCG@k(t)}{iDCG@k(t)} \right) \tag{36}$$

In addition, we calculated *user coverage* (UC), expressed as:

$$UC = \frac{|U_r|}{|U|} \tag{37}$$

User Coverage is defined as the number of users for whom at least one recommendation could be generated ($|Ur|$) divided by total number of users $|U|$ (Lacic et al. 2015).

We also report *session coverage* (SC), expressed as:

$$SC = \frac{\sum_{u \in U_r} |T_u|}{PS} \tag{38}$$

Session Coverage is defined as the number of purchase sessions in which the recommender was able to generate a recommendation (i.e., total number of valid test cases) divided by the total number of purchase sessions of the system (PS).

Content-based recommendation techniques are usually much more susceptible to overspecialization than other recommendation techniques, such as e.g. collaborative filtering (Parra and Sahebi 2013). Therefore, in order to measure the degree of this effect we also calculated several diversity metrics.

The first of these metrics is the *Artist Diversity* ($\text{D@k}_{\text{artist}}$), defined as:

$$\text{D@k}_{\text{artist}} = \frac{\sum_{u \in U_r} \sum_{t \in T_u} \left| \bigcup_{i \in r_t^k} \{ a_i \} \right|}{\sum_{u \in U_r} |T_u|}, \tag{39}$$

where a_i is item i's artist. The Artist Diversity measures the average number of distinct artists per recommendation. This metric is useful for getting a notion of how diverse a recommendation is in terms of the different artists recommended. The larger the metric, the more the chances of recommending items from novel artists to users.

Similarly, we also calculate *Color Diversity* and *Medium Diversity*, which are formally defined as:

$$\underset{\text{color}}{\text{D@k}} = \frac{\sum_{u \in U_r} \sum_{t \in T_u} \left| \bigcup_{i \in r_t^k} \text{CAV}_i^{\text{color}} \right|}{\sum_{u \in U_r} |T_u|} \tag{40}$$

$$\underset{\text{medium}}{\text{D@k}} = \frac{\sum_{u \in U_r} \sum_{t \in T_u} \left| \bigcup_{i \in r_t^k} \text{CAV}_i^{\text{medium}} \right|}{\sum_{u \in U_r} |T_u|}, \tag{41}$$

where CAV_i^{color} and CAV_i^{medium} are defined in Table 3. *Color Diversity* and *Medium Diversity* measure the average number of distinct color values and medium values per recommendation, respectively. We do not use other curated attributes apart from color and medium because these are the only ones that are present in (almost) all artworks, as already shown in Table 2.

In addition to Artist, Color and Medium, it is also possible to learn visual categories directly from images by means of unsupervised techniques, e.g. clustering. To this end, we crawled 10,316 images from the *UGallery* website (a superset of the 3490 images used in offline evaluations), and for each of these images we obtained a feature vector of 8258 dimensions ($\mathbb{R}^{8258}$) by concatenating AlexNet ($\mathbb{R}^{4096}$) + VGG ($\mathbb{R}^{4096}$) + LBP ($\mathbb{R}^{59}$) + Attractiveness ($\mathbb{R}^{7}$). Then we calculated a z-score normalization and used PCA to reduce the vector dimensionality to $\mathbb{R}^{100}$, so as to retain the most relevant visual features according to the natural distribution of images. Finally, we used *Gaussian Mixture* clustering to fit 400 clusters to this augmented image dataset (using more clusters did not yield significant improvements in silhouette scores). Thus, we calculate *Visual Cluster Diversity*, which is formally defined as:

$$\underset{\substack{\text{visual}\\\text{cluster}}}{\text{D@k}} = \frac{\sum_{u \in U_r} \sum_{t \in T_u} \left| \bigcup_{i \in r_t^k} \{ vc_i \} \right|}{\sum_{u \in U_r} |T_u|}, \tag{42}$$

where vc_i is defined in Table 4. This metric measures the average number of distinct visual clusters per recommendation.

In addition to clustering, the aforementioned $\mathbb{R}^{100}$ visual feature vectors can also be used for pairwise comparisons. Thus, we also calculate *Visual Pairwise Diversity* which we formally define as follows:

$$\underset{\substack{\text{visual}\\\text{pairwise}}}{\text{D@k}} = \frac{\sum_{u \in U_r} \sum_{t \in T_u} \underset{\substack{\text{visual}\\\text{pairwise}}}{\text{D@k}}(t)}{\sum_{u \in U_r} |T_u|} \tag{43}$$

$$\underset{\substack{\text{visual}\\\text{pairwise}}}{\text{D@k}}(t) = \frac{\sum_{y=1}^{k-1} \sum_{z=y+1}^{k} 0.5 \cdot \left[1 - cos\left(V_{i_{t,y}}^{\text{PCA(100)}}, V_{i_{t,z}}^{\text{PCA(100)}} \right) \right]}{\frac{k \cdot (k-1)}{2}}, \tag{44}$$

where $\underset{\substack{\text{visual}\\\text{pairwise}}}{\text{D@k}}(t)$ is the average of the pairwise cosine distances between the top-k items of test case t's recommended list, $i_{t,y}$ and $i_{t,z}$ are the items at positions y and

z, respectively, of test case t's recommended list, $V_i^{\mathrm{PCA}(100)}$ is item i's $\mathbb{R}^{100}$ visual feature vector obtained with PCA, and $cos(x, y)$ stands for cosine similarity.

Finally, we can also compute a pairwise diversity metric based on the whole metadata. By combining Artist, Colors and Medium in a single set of metadata attribute values per item, we can use Jaccard Index to calculate *Jaccard Pairwise Diversity*, which we formally define as:

$$\underset{\text{jaccard}}{\overset{\text{D@k}}{\text{pairwise}}} = \frac{\sum_{u \in U_r} \sum_{t \in T_u} \underset{\text{jaccard}}{\overset{\text{D@k}}{\text{pairwise}}}(t)}{\sum_{u \in U_r} |T_u|} \tag{45}$$

$$\underset{\text{jaccard}}{\overset{\text{D@k}}{\text{pairwise}}}(t) = \frac{\sum_{y=1}^{k-1} \sum_{z=y+1}^{k} \left[1 - \text{jaccard\_index}(S_{i_{t,y}}, S_{i_{t,z}})\right]}{\frac{k \cdot (k-1)}{2}} \tag{46}$$

$$\text{jaccard\_index}(S_i, S_j) = \frac{\left|S_i \cap S_j\right|}{\left|S_i \cup S_j\right|} \tag{47}$$

$$S_i = \text{CAV}_i^{\text{color}} \cup \text{CAV}_i^{\text{medium}} \cup \{ a_i \} \tag{48}$$

In addition to these offline evaluation metrics, we also report Precision@k and nDCG@k for the online evaluation with 8 UGallery expert curators. In this setting, the metrics were calculated as follows:

$$nD@k = \frac{1}{8} \sum_{x=1}^{8} \frac{DCG@k(x)}{iDCG@k(x)} \tag{49}$$

$$DCG@k(x) = \sum_{z=1}^{k} \frac{2^{B_x(i_{x,z})} - 1}{log_2(1+z)} \tag{50}$$

$$P@k = \frac{1}{8} \sum_{x=1}^{8} p@k(x) \tag{51}$$

$$p@k(x) = \frac{1}{k} \sum_{z=1}^{k} \mathbb{1}_x(i_{x,z}), \tag{52}$$

where x stands for the x-th expert curator, $i_{x,z}$ is the item appearing at position z in the list recommended to expert x, $B_x(i_{x,z})$ returns the original rating $S_x(i_{x,z})$ given by expert x to item $i_{x,z}$ if $S_x(i_{x,z}) >= 4$, or 0 otherwise, and $\mathbb{1}_x(i_{x,z})$ is an indicator function that returns 1 if rating $S_x(i_{x,z}) \geq 4$, or 0 otherwise (i.e., we used 4 as the relevance threshold for the calculation of these metrics).

7 Results

In this section, we report the results focusing on different aspects. With respect to research question RQ1—analyzing the impact of each single feature–, we analyze: a)

Table 5 nDCG (nD), Recall (R), Precision (P), F1 Score (F1) and Coverage (UC and SC) for metadata based methods: MPCAV (by attribute), PMPCAV (by attribute), and FA. The best result for each metric and method group are highlighted. The superindex indicates the ID of the method with the closest but still significantly smaller result. For instance, FA R@10 = .2067[12] tells that FA is significantly larger than at least (12) PMPCAV(All) R@10 = .0785, as well as significantly larger than all the other methods with R@10 < .0785

ID	Method	nD@10	R@10	P@10	F1@10	UC	SC
1	MPCAV(Subject)	.0115	.0172	.0023	.0041	.9985	.9991
2	MPCAV(Medium)	.0106	.0211	.0025	.0043	**.9993**	**.9995**
3	MPCAV(Style)	.0096	.0176	.0025	.0042	.9978	.9972
4	MPCAV(Color)	.0095	.0190	.0023	.0040	**.9993**	**.9995**
5	MPCAV(Mood)	**.0148**	**.0279**	**.0034**	**.0059**	.8483	.8229
6	MPCAV(All)	.0087	.0157	.0020	.0034	**.9993**	**.9995**
7	PMPCAV(Subject)	.0099	.0136	.0021	.0036	.0890	.1407
8	PMPCAV(Medium)	.0190	.0363	.0044	.0094	**.2640**	**.3593**
9	PMPCAV(Style)	.0237	.0485	.0060	.0118	.0766	.1168
10	PMPCAV(Color)	.0264	.0486[3]	.0063	.0108	.2619	.3570
11	PMPCAV(Mood)	**.0507**	.0774	**.0098**	**.0169**	.1327	.1822
12	PMPCAV(All)	.0448[14]	**.0785**[5]	.0095[5]	.0165[5]	**.2640**	**.3593**
13	FA	**.1380**[11]	**.2067**[12]	**.0259**[11]	**.0446**[11]	**.2640**	**.3593**
14	Random	.0122	.0214	.0027	.0046	1.0000	1.0000

Stat. significance by multiple t-tests, Bonferroni corr
$\alpha_{bonf} = \alpha/n = 0.05/91 = .00055$

metatadata features (personalized and non-personalized), and b) visual features (DNN and MEVF). For RQ2, we compare between visual features and metadata. Regarding research question RQ3, we test several combinations of features to identify the best hybrid recommender in terms of ranking and accuracy. In Sect. 7.5 we assess RQ2 and RQ3 with respect to metrics of diversity. Finally, regarding research question RQ4, the online validation, we report and discuss the results of recommendations evaluated by expert curators from UGallery.

7.1 Metadata features (RQ1.1)

Table 5 summarizes all the results for this analysis of metadata features. Here we report MPCAV, its personalized version PMPCAV, and Favorite Artist (FA).

Most Popular Curated Attribute Value (MPCAV) We tested the performance of MPCAV features separately as well as combined (*MPCAV(All)*). Table 5 shows that these results are not significantly different from random prediction in the performance metrics reported (nDCG, Recall, Precision, F1score).

Personalized MPCAV (PMPCAV) As for PMPCAV, in Table 5 we observe that the use of personalization causes a general, although rather small, improvement in the ranking metrics over MPCAV. However, personalization has the negative side effect of dropping user and session coverages. This is partly caused by the *user cold start*

problem, which is inherent to personalization, but also because of the absence of tags for many artworks (Table 2) which hinders the PMPCAV method from tracking users' attribute preferences and making recommendations in many cases. This prevents these small performance improvements from having statistical significance, with the remarkable exception of *PMPCAV(All)*, which by combining all attributes achieves top user and session coverages among personalized methods, and most importantly, significantly better ranking metrics than both MPCAV and Random.

Favorite Artist (FA) One result that stands out overall is the performance of the artist feature. In this method, we tested whether making personalized recommendations from the user's most frequently purchased artists could yield good results. Our results indicate that FA is actually the single most accurate method ($nD@10 = 0.1380$, $R@10 = 0.2067$), between 3 and 4 times better than the second best metadata based method—*PMPCAV(All)*.

MPCAV versus PMPCAV The most outstanding lesson about these methods is the relatively poor performance obtained with expertly annotated attributes with respect to a random baseline, although personalization (PMPCAV) produces a significant improvement. Our results support the importance of personalization to improve the performance, as seen in Table 5. As additional evidence, all the other more sophisticated personalized methods (MEVF, DNN, FA and Hybrids) are significantly better than MPCAV, as shown in Table 7.

7.2 DNN and MEVF visual features (RQ1.2)

To the best of our knowledge, our work presents the first analysis comparing manually engineered visual features (brightness, contrast, etc.) versus automatically extracted features (DNN) for the task of recommending artworks. Table 6 presents the results, where it is clear that DNN embeddings yield a significant improvement over MEVF features, either combined or in isolation, almost doubling their performance in almost all the accuracy and ranking metrics. These results are in line with the current state-of-the-art of deep neural networks in computer vision, which report better results than other methods in several tasks (Sharif Razavian et al. 2014; He and McAuley 2016).

Combining AlexNet and VGG shows a small improvement over using either DNN separately, but the statistical tests show no significant differences between them.

Combining MEVF features improves their performance compared to using them in isolation. This effect is remarkably clear in the case of Attractiveness, where using each feature in isolation shows poor results (not significantly different from Random) but combining them all leads to significant improvements. Moreover, combining Attractiveness and LBP yields the best MEVF results, with an improvement of about 400% above Random.

When comparing MEVF features one-by-one, we observe that *LBP* performs the best (about 300% better than Random) because it encodes texture patterns and local contrast very well, although as a feature it's harder to explain than e.g. image *brightness* or *contrast*.

In summary, these results provide evidence in favor of the use of pre-trained deep neural networks for transfer learning. Their only drawback is the great difficulty in

Table 6 nDCG (nD), Recall (R), Precision (P), F1 Score (F1) for image based methods. User and Session Coverage are all the same for every experiment, UC = .2640 and SC = .3593. The best absolute result of each metric is highlighted. The superindex indicates the ID of the method with the closest but still statistically significant difference. For instance, DNN-2 R@10 = $.1671^{4}$ indicates that DNN-2 is significantly larger than at least (4)MEVF (LBP + Att:all) R@10 = .0998, as well as significantly larger than all the other methods with R@10 < .0998

ID	Method	nD@10	R@10	P@10	F1@10
1	DNN-2(VGG + AlexNet)	$\mathbf{.1187}^{4}$	$\mathbf{.1671}^{4}$	$\mathbf{.0210}^{4}$	$\mathbf{.0365}^{4}$
2	DNN(VGG)	$.1123^{4}$	$.1614^{4}$	$.0203^{4}$	$.0352^{4}$
3	DNN(AlexNet)	$.1094^{4}$	$.1571^{4}$	$.0201^{4}$	$.0348^{4}$
4	MEVF(LBP + Att:all)	$.0674^{7}$	$.0998^{7}$	$.0118^{10}$	$.0213^{10}$
5	MEVF(LBP)	$.0500^{7}$	$.0897^{7}$	$.0104^{7}$	$.0183^{7}$
6	MEVF(Att: all)	$.0424^{7}$	$.0637^{9}$	$.0085^{7}$	$.0146^{7}$
7	MEVF(Att: contrast)	.0120	.0230	.0027	.0048
8	MEVF(Att: naturalness)	.0106	.0204	.0025	.0044
9	MEVF(Att: saturation)	.0091	.0197	.0021	.0037
10	MEVF(Att: brightness)	.0085	.0186	.0031	.0052
11	MEVF(Att: sharpness)	.0095	.0178	.0021	.0039
12	MEVF(Att: entropy)	.0106	.0137	.0019	.0034
13	MEVF(Att: colorfulness)	.0073	.0132	.0020	.0035
14	Random	.0122	.0214	.0027	.0046

Stat. significance by multiple t-tests, Bonferroni corr
$\alpha_{bonf} = \alpha/n = 0.05/91 = .00055$

interpreting the neural image embedding in order to explain recommendations to users. Recent works unveil which features are learned by certain neurons (Olah et al. 2017), but knowing whether those features are actually influencing the user towards a purchase decision is still difficult to know. In general terms the features automatically learned by neural networks are discriminating but difficult to explain, and this lack of transparency and explainability might potentially hinder the user acceptance of these recommendations (Konstan and Riedl 2012; Verbert et al. 2013; Nunes and Jannach 2017).

7.3 Comparing visual features versus metadata (RQ2)

Visual Features versus Curated Attributes From Table 7, which shows results of the overall analysis, we observe that both DNN and MEVF methods significantly outperformed curation-based methods (PMPCAV and MPCAV). MEVF reports significantly better metrics than some combinations of manually curated metadata—versus PMPCAV + MPCAV (Mood) and MPCAV (Mood)—but not significantly better than PMPCAV(All). On the other hand, DNN methods always improve over both MEVF and PMPCAV(All), showing the potential of neural networks for automatic extraction of high quality features.

In general, these results indicate that it is possible to leverage automatic visual feature extraction techniques from artwork images to achieve higher accuracy and

Table 7 nDCG (nD), Recall (R), Precision (P), F1 Score (F1), User Coverage (UC) and Session Coverage (SC) for all methods. The best three absolute results of each metric are highlighted. The superindex indicates the ID of the method with the closest but still significantly smaller result. For instance, Hybrid_1 R@10 = $.2414^6$ tells that Hybrid_1 is significantly larger than (6)Hybrid_5 R@10 = .1767, as well as significantly larger than all the other methods with R@10 < .1767

ID	Method	nD@10	R@10	P@10	F1@10	UC	SC
1	Hybrid_1(FA + DNN-2 + MEVF + PMPCAV)	**.1660**^{6}	**.2414**^{6}	**.0296**^{5}	**.0515**^{6}	**.2640**	**.3593**
2	Hybrid_2(FA + DNN-2 + MEVF)	**.1695**^{5}	**.2379**^{6}	**.0296**^{5}	**.0513**^{6}	**.2640**	**.3593**
3	Hybrid_3(FA + DNN-2)	**.1680**^{5}	**.2373**^{6}	**.0292**^{5}	**.0507**^{6}	**.2640**	**.3593**
4	Hybrid_4(FA + MEVF)	.1569^{10}	.2252^9	**.0272**^{11}	.0474^{11}	**.2640**	**.3593**
5	FA	.1380^{11}	.2067^{11}	.0259^{11}	.0446^{11}	**.2640**	**.3593**
6	Hybrid_5(DNN-2 + MEVF + PMPCAV)	.1207^{11}	.1767^{11}	.0215^{11}	.0376^{11}	**.2640**	**.3593**
7	Hybrid_6(DNN-2 + MEVF)	.1204^{11}	.1713^{11}	.0215^{11}	.0374^{11}	**.2640**	**.3593**
8	DNN-2(VGG + AlexNet)	.1187^{11}	.1671^{11}	.0210^{11}	.0365^{11}	**.2640**	**.3593**
9	DNN(VGG)	.1123^{11}	.1614^{11}	.0203^{11}	.0352^{11}	**.2640**	**.3593**
10	DNN(AlexNet)	.1094^{11}	.1571^{11}	.0201^{11}	.0348^{11}	**.2640**	**.3593**
11	MEVF	.0674^{13}	.0999^{13}	.0122^{12}	.0213^{13}	**.2640**	**.3593**
12	PMPCAV(All)	.0448^{14}	.0785^{13}	.0095^{13}	.0165^{13}	**.2640**	**.3593**
13	PMPCAV + MPCAV(Mood)	.0168	.0301	.0037	.0063	**.8483**	**.8229**
14	MPCAV(Mood)	.0148	.0279	.0034	.0059	**.8483**	**.8229**
15	Random	.0122	.0214	.0027	.0046	**1.0000**	**1.0000**

Statistical significance was obtained using multiple pairwise t-tests with Bonferroni correction
$\alpha_{bonf} = \alpha/n = 0.05/105 = .00048$

ranking metrics in the task of future shopping basket prediction, and noteworthily, without using expert annotated metadata, the production of which can be very time-consuming.

Visual Features versus Favorite Artist (FA) In total contrast to curated attributes, recommending based on the user's favorite artists surprisingly outperforms both MEVF and DNN in terms of ranking metrics in the offline evaluation, as can be seen in Table 7. In fact, FA ($nD@10 = 0.1267$ and $R@10 = 0.2067$) results in significantly better metrics than the best DNN ($nD@10 = 0.1074$ and $R@10 = 0.1671$) by more than a 20%. These offline results may be explained by the fact that users are probably biased to keep exploring and finding items they like from artists they are already familiar with. However, when we look at the online results with expert curators (Table 9), the differences between FA and visual methods become much narrower, where in fact DNN and the hybrid DNN + MEVF show better results than FA in practically all metrics. This shows that FA is a very good heuristic for filtering the item search space when predicting next purchases (as reflected offline), but its lack of any visual content awareness renders it incapable of performing fine-grained visual discrimination, which is reflected in the less favorable results in the online evaluation compared to DNN and MEVF.

7.4 Hybrid recommendations (RQ3)

The Hybrid recommenders, summarized in Table 7, show a clear tendency: when features are combined into hybrids, they tend to perform better than the features used individually. Some of these improvements are statistically significant, such as hybrids 2–3 in nD@10 and hybrids 1–3 in P@10 with respect to FA. In other cases there are improvements but not strong enough to be deemed statistically significant, as in the cases of hybrids 1–4 in R@10 with respect to FA and hybrids 5–6 in all metrics with respect to DNN based methods. Although there are no significant differences among the top-4 Hybrid methods, there is a trend towards showing Hybrid_2(FA + DNN-2 + MEVF) as the best combination. The methods that do not include FA but include a combination of different visual features (i.e., Hybrid_5, Hybrid_6 and DNN-2) significantly outperform MEVF and curated metadata based methods (PMPCAV), and show no statistically significant differences with respect to FA, although they are clearly more expensive to implement. Under the light of these offline results, it is then interesting answering whether the online validation with expert users is consistent or not, i.e., if FA has such a good performance compared to hybrid methods.

7.5 Effect on diversity (RQ2 and RQ3)

Table 8 presents the results of several features and combinations of them upon six metrics of diversity. Recall@10 is also reported in the table as a reminder of the ranking performance of each method. The results can be summarized as follows: In terms of visual diversity, we can clearly see the effect of DNN: all methods that use DNN features show lower visual diversities than those that do not. This is an expectable result, as pre-trained CNNs are powerful off-the-shelf tools for extracting high qual-

Table 8 Diversity results of experiments for all methods. Recall@10 (R@10) is also included as a reminder of the accuracy of each method. The three smallest (underline) and largest (bold) diversity results for each metric are highlighted

ID	Method	R@10	D@10 visual cluster	D@10 visual pairwise	D@10 artist	D@10 jaccard pairwise	D@10 color	D@10 medium
1	$Hybrid_1$(FA + DNN-2 + MEVF + PMPCAV)	**.2414**^{6}	8.0169^{5}	.3659^{6}	5.3503^{2}	.7256^{5}	8.8620^{12}	1.9388
2	$Hybrid_2$(FA + DNN-2 + MEVF)	**.2379**^{6}	7.6901	.3442^{9}	5.0430^{3}	.7654^{3}	9.3594^{1}	2.4583^{6}
3	$Hybrid_3$(FA + DNN-2)	**.2373**^{6}	7.6471	.3429^{7}	4.9414^{5}	.7617^{1}	9.3867^{5}	2.4258^{6}
4	$Hybrid_4$(FA + MEVF)	.2252^{9}	8.1497^{5}	.3953^{1}	4.9206^{5}	.7603^{1}	9.2005^{1}	2.4622^{6}
5	FA	.2067^{11}	7.8244	.3931^{1}	2.4889	.7024	9.1482^{1}	2.1274^{1}
6	$Hybrid_5$(DNN-2 + MEVF + PMPCAV)	.1767^{11}	8.2865^{9}	.3474^{9}	7.3646^{1}	.7810^{2}	8.7370^{12}	2.1992^{1}
7	$Hybrid_6$(DNN-2 + MEVF)	.1713^{11}	7.9948^{2}	.3274^{10}	7.4714^{1}	.8423^{6}	9.5638^{4}	2.8307^{4}
8	DNN-2	.1671^{11}	7.9609^{2}	.3247^{10}	7.4440^{1}	.8393^{6}	9.5247^{4}	2.7682^{4}
9	DNN(VGG)	.1614^{11}	8.0911^{8}	.3352^{7}	7.6198^{7}	.8461^{8}	9.6914^{8}	2.8177^{4}
10	DNN(AlexNet)	.1571^{11}	8.0689^{5}	.3150	7.6073^{6}	.8487^{7}	9.4473^{5}	2.8934^{8}
11	MEVF	.0999^{13}	9.1730^{6}	.4243^{4}	**8.5527**^{14}	.8805^{10}	**10.0338**^{14}	3.0403^{10}
12	PMPCAV(All)	.0785^{13}	9.2653^{6}	.4490^{11}	**8.3537**^{9}	.7479^{1}	8.2380	1.8687
13	PMPCAV + MPCAV(Mood)	.0301	**9.5037**^{12}	**.4519**^{11}	8.2669^{9}	**.8888**^{10}	**10.0261**^{14}	**3.9387**^{15}
14	MPCAV(Mood)	.0279	**9.5327**^{12}	**.4511**^{11}	8.2499^{9}	**.8924**^{13}	9.8410^{5}	**4.1363**^{13}
15	Random	.0214	**9.8266**^{14}	**.4917**^{13}	**9.6439**^{11}	**.9111**^{14}	**10.6294**^{11}	**3.6224**^{11}

Statistical significance was obtained using multiple pairwise t-tests with Bonferroni correction
$\alpha_{bonf} = \alpha/n = 0.05/105 = .00048$

ity visual features. There is a notable exception, though. FA, which recommends by sampling artworks from the user's favorite artists, shows lower visual cluster diversity and comparable visual pairwise diversity with respect to DNN based methods. This result indicates that artists in our dataset paint visually similar artworks, making recommendation lists based on the same artist less diverse compared to using other methods.

Moreover, when FA and DNN are combined, as in Hybrid_3, the resulting recommender achieves the lowest visual cluster diversity ($\text{D@10}^{\text{visual}}_{\text{cluster}} = 7.6471$) and better predictive accuracy than each individual method in isolation, providing evidence that users are more likely to purchase similar-looking artworks from artists they are familiar with, although recommending based on this heuristic can lead to lower visual diversity. In the case of MEVF, we observe an improvement in accuracy and decrease in visual diversity with respect to Random and curated metadata based methods, but the effect is not as strong as that of DNN based methods.

Regarding diversity metrics based on metadata, the most informative metric is Artist Diversity ($\text{D@10}^{}_{\text{artist}}$). From this metric we can notice a very interesting trend: the fewer artists used in a recommendation, the more accurate the recommendation becomes. This trend holds until we get to FA, with the lowest artist diversity ($\text{D@10}^{}_{\text{artist}} = 2.5$ approximately). However, the trend gets reversed when we get to the top 4 hybrid recommenders, all of them recommending from about 5 artists on average. This seems to indicate the existence of an optimal combination in artist diversity in order to achieve optimal recommendation accuracy. This result is also good news from a business standpoint: the top hybrid recommenders can achieve higher accuracy while still being able to promote paintings from a reasonably diverse group of artists. With respect to Color ($\text{D@10}^{}_{\text{color}}$) and Medium ($\text{D@10}^{}_{\text{medium}}$) diversities, these metrics do not reveal very insightful patterns, besides the fact that the lowest values are reached when PMPCAV(All) is used. On the contrary, Jaccard Pairwise Diversity ($\text{D@10}^{\text{jaccard}}_{\text{pairwise}}$) do seem to show a pattern similar to Artist Diversity, although the apparent correlation is probably due to the influence of the artist in the bag of attributes used for Jaccard Index calculations.

7.6 Validation with expert users (RQ4)

Table 9 presents the results of the online evaluation with 8 expert curators from *UGallery*, showing the mean over four metrics: nDCG@5, nDCG@10, Precision@5 and Precision@10. As explained in Sect. 6.2, each user had to rate 10 recommended items from each of the five methods shown in Table 9 (i.e., they rated 50 items in total). The most important aspect to highlight is that combining FA with visual features in a single hybrid (FA + DNN + MEVF) outperforms all the other features, either hybrid or single, in all four metrics, which is consistent with the offline results. Another interesting result is that DNN shows better performance than FA, which is the opposite to the offline evaluation. We think that this might be due to the lack of diversity that FA promotes, but also to the potential noise present when sampling artworks from artists to fit a top-n recommendation without awareness of the visual content. It is also remarkable that the isolated features show smaller differences between them in this

Table 9 nD@5, nD@10, P@5 and P@10 for algorithms tested with 8 UGallery experts

Name	nD@5	nD@10	P@5	P@10
Hybrid(FA + DNN + MEVF)	**0.9042**	**0.8913**	**0.7500**	**0.6750**
Hybrid(DNN + MEVF)	0.6747	0.6638	0.5000	0.4250
DNN	0.7176	0.6947	0.5000	0.4000
FA	0.4276	0.5662	0.3000	0.4000
MEVF	0.5498	0.5314	0.3500	0.2625

For nD@k, all ratings ≤ 3 were set to 0. For P@k, only ratings ≥ 4 were regarded as relevant
Bold to highlight the highest value of each metric

user experiment than in the offline evaluation. In terms of nDCG@5, nDCG@10 and Precision@5, DNN seems to outperform both FA and MEVF, while it has the same performance as FA in terms of Precision@10. Given the small sample size, we cannot report tests of statistical significance, but the trend of results points toward implementing a hybrid recommender with FA and visual features for the best performance without hindering diversity.

8 Summary and discussion

The main findings with respect to our RQs can be summarized as follows:

- **RQ1. Metadata** In general, using the most popular curated attribute values (MPCAV) performed not significantly different than random prediction. The personalized version PMPCAV, specially the one using all attributes, performed significantly better than the non-personalized version MPCAV, but still the results were rather poor. Notably, just recommending based on a user's favorite artists produced very high ranking metrics.
- **RQ1. Visual features** The features automatically obtained from pre-trained neural networks (DNN) significantly outperformed manually-engineered visual features (MEVF). This is an interesting result, considering that the AlexNet and VGG neural networks were trained for object classification, not for recommendation. This supports the use of transfer learning.
- **RQ2. Visual features versus metadata** Visual features performed better than curated attributes. This is an important result, since it points towards using features extracted directly from the images rather than spending resources for manually tagging the images. However, the single best predictive feature overall was Favorite Artist, so combining the strengths of both visual features and FA seems like a promising approach.
- **RQ3. Hybrids** Hybrid methods combining multiple features outperformed individual features. The hybrid method which combined FA, DNN and MEVF produced the best results (a variant including PMPCAV performed equally well), in both offline and online evaluations.
- **RQ4. Expert online evaluation** The expert evaluation allows us to show the consistency of the offline results when assessed by real people. There was consistency

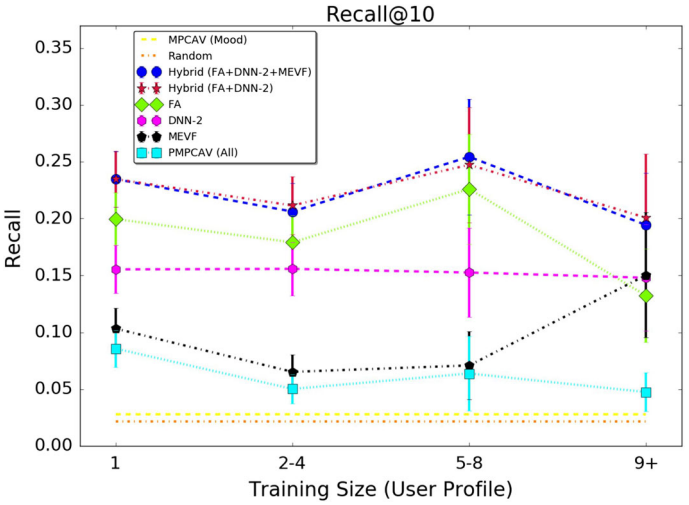

Fig. 7 Recall@10 of different methods at different user profile sizes

in terms of the best hybrid (FA + DNN + MEVF), which outperformed the other 4 alternatives. Also notable was the small difference among isolated features (DNN, MEVF, FA) compared to their offline results.

Taken together, our results show that a recommender system which utilizes several types of content could indeed support people who buy artworks online based on their personal taste. Moreover, we have some additional thoughts with respect to the intriguing high predictive power of favorite artist and the risk of relying solely on features such as those from neural networks.

Our offline evaluation results indicate that the method FA (based simply on sampling artworks from the favorite artists of a user) performs really well, with a 20%-30% improvement over the next competitor DNN, whereas the best Hybrid improves FA by a smaller margin of 10%–23%. We investigated further whether the size of the user profile (items in training) could give us more evidence of this effect. Our intuition behind this analysis is that artists have in average 8 artworks for sale, and if a customer buys them all, then it will be more difficult to predict the next potential favorite artist.

Figure 7 shows the Recall@10 of different methods considering different user profile sizes. The plot shows that DNN, MEVF, and PMPCAV return always very consistent results independent of profile size and that, among them, DNN performs the best. FA and hybrid methods perform better than DNN and MEVF up to user profiles with 5–8 items. However, with larger user profiles (9 +) DNN and MEVF seem to improve or maintain results, whereas other methods such as FA and the Hybrids suffer an important decrease. This decrease in methods using FA could be explained by the fact that artists tend to sell their artworks over time, leading to a natural shortage of available artworks from users' favorite artists, and therefore forcing users to explore new artists instead. It can also reflect a natural evolution of users' taste or the curiosity for exploring new artists over time. In contrast, the apparent stability displayed by visual features, especially DNN, seems to indicate that users are relatively more stable

in terms of their visual tastes over time. An interesting line of work could be exploring more sophisticated recommendation methods that can take the temporal dimension into account, such as the use of temporal decay to account for the effect of users' preference drift over time (Koren 2010; Larrain et al. 2015). Another possible factor contributing to the better performance of visual methods with larger user profiles can be found in the scoring function of DNN and MEVF, shown in Eq. 6. As a reminder, the score assigned to an item is calculated as either (a) the maximum, (b) the average or (c) the average of the top-k most similar items in the user profile. Based on our experiments, the best results were usually obtained with the average of the top 2 and top 3 most similar items in the user profile. This way of calculating the score can help to make recommendations that are supported by subsets of similar-looking items from the user profile (which can be thought of as emergent "mini-clusters" that capture "sub-tastes" of the user). This strategy turns useful when dealing with large user profiles, where a naive average can introduce too much noise and a greedy maximum can overlook emergent patterns across different user's purchases.

Furthermore, we show evidence that deep neural networks can be of great value in the field of personalized artwork recommendations, since they decrease the cost of domain expert knowledge to identify the visual features which can be most successful, with a small compromise on diversity. However, in order to make recommendations really useful and not only persuasive (Tintarev and Masthoff 2015), researchers and developers need to make sure that people can inspect and have a sense of control (Knijnenburg et al. 2012; Parra and Brusilovsky 2015), which is achieved by combining latent easy-to-engineer information (such as features from deep learning models) with actual explicit features, such as artist, color, style, or brightness. One way we have thought of to provide users with such control is by using techniques such as t-SNE with an interactive interface. t-SNE (Maaten and Hinton 2008) is a dimensionality reduction technique commonly used to visualize what DNN embeddings might encode (He et al. 2016; He and McAuley 2016; Nguyen et al. 2016). This technique could be used to help users visualize high-dimensional data in a lower-dimensional space in order to understand recommendations, explore them and inspect them, features associated with improved user satisfaction (Knijnenburg et al. 2012; Verbert et al. 2013). For instance, Fig. 8 uses t-SNE to reduce DNN embeddings and then display an anonymized user profile and the images predicted by three different methods: DNN, MPCAV and MEVF. We could perform a similar process over the MEVF embedding and show users differences between both representations, as well as allowing them rich exploration. We foresee building rich visual applications providing user control, transparency and explainability, important characteristics to build user trust and acceptance on recommendations (Tintarev and Masthoff 2015; Ekstrand et al. 2015).

An important aspect to bear in mind when interpreting our results is that they relate to only one single artwork retailer website, although one of the most popular on the Web. This might hinder the generalizability of our results. In addition, other forms of user evaluation are needed in order to test whether user evaluation correlates with our offline results, such as a large controlled laboratory study as well as a field online study using A/B testing. Another aspect to bear in mind is that the presented work is not intended to provide precise guidelines for an industrial implementation of an artwork recommender system, which would require pondering other aspects, such as algorithm

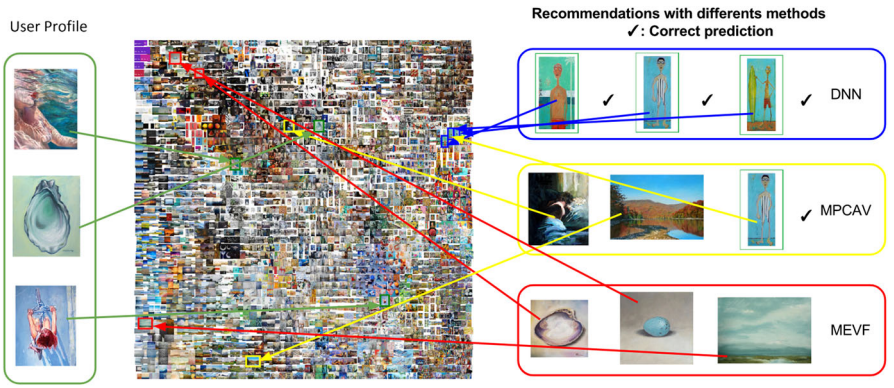

Fig. 8 t-SNE map of the DNN image embedding displaying paintings of an anonymized user profile (green), and recommendations contextualized with three methods: DNN (blue), MPCAV (yellow) and MEVF (red). Check marks indicate correct predictions. (Color figure online)

complexity and scalability. Rather, our focus has been to provide insights into which features and combinations thereof are the most promising if such a recommender system were to be implemented. That being said, one possible strategy for generating recommendations more efficiently is to quickly pre-filter the search space first, for example based on artists or using approximate nearest neighbor techniques in an image embedding space, and then perform a second re-ranking step over the filtered dataset with a more expensive ranker such as any of the hybrids proposed in this work.

9 Conclusions and future work

In this article, we have presented several notable results in the area of content-based artwork recommendation under the one-of-a-kind item problem. We have investigated the potential of several different features for this task. As our results reveal (in the context of a physical artwork online store named UGallery), individual expert-annotated metadata attributes perform not better than random predictions, unless they are combined in a personalized manner, which can improve the results by a small margin. However, recommending solely based on the favorite artists (FA) of the user can yield, surprisingly, very good results, at the expense of a small diversity in recommendation lists. Moreover, we found that visual features are more useful in predicting future purchases than expert-annotated metadata. Among the visual features investigated, image embeddings from Deep Convolutional Neural Networks work better than manually-engineered visual features, but overall, the hybrid combination of FA + DNN + MEVF produces the best results. Finally, a user study with expert curators from UGallery supports the use of a hybrid combining FA + DNN + MEVF for the optimal results.

In a deeper analysis, our study of the user profile sizes revealed that time may play an important role in recommending artwork to people. Though further investigation is needed, our results that consider different user profile sizes for training the user models can produce important differences in terms of Recall@k. As such we are interested

in investigating the time dimension in more detail, which has not been the focus of this work so far. The previous work by Hidasi et al. (2016) which introduces a neural network model for feature-rich session-based recommendations could be a starting point in this direction.

In this work we focused on comparing useful content features rather than on developing state-of-the-art recommendation models. As several new neural network architectures have been introduced to the recommender and visualization communities, we could apply some of these approaches to our problem. One example of such architectures is Convolutional Autoencoders, which are able to learn compact representations of images in an unsupervised manner, as in the work of David and Netanyahu (2016), who use the unsupervised compact image embedding learned by a convolutional autoencoder as a basis for a supervised painter classification task. Another option is to use generative models. Although generative models are usually designed, as the name implies, to generate samples of a certain distribution, there are works that show they can also be used to learn representation embeddings of images (Radford et al. 2015; Mathieu et al. 2016). All of these procedures could allow us to learn different image embeddings to eventually use them for learning a recommendation model.

We can also test a Siamese network architecture to learn an image embedding that locates similar images close to each other. There are many works confirming the success of this approach, such as the work by Schroff et al. (2015) in face recognition, or the work by Koch et al. (2015) in one-shot image recognition. Yang et al. also used successfully a Siamese network architecture for food recommendation based on images (Yang et al. 2015). The key point in the Siamese network approach lies in determining whether two images belong to the same class or not. Given the good results achieved by the artist attribute in our experiments, a natural choice for the class would be the artist, i.e., pulling images from the same artist together and pushing images from different artists apart.

However, there is a potential disadvantage in the Siamese network approach with the artist as the class label: the network might fail to learn fine-grained visual features to be able to rank images when they belong to the same artist. Moreover, there can be cases in which two similar-looking paintings belong to different artists, in which case we would still want the images to be close to each other in the embedding space. In this case, Triplet loss can be an alternative. By using triplets of the form $(query, +similar, -similar)$ the network can learn fine-grained visual features to rank images even when they belong to the same class, as shown in the work of Wang et al. (2014). Moreover, Triplet loss has been successfully applied in industry, as in the visual recommender system and search engine at Flipkart, India's largest e-commerce company (Shankar et al. 2017). Another interesting approach we could take is multitask learning. Rush (1979) observed that people's ability to accurately recognize artistic style could be enhanced if the exposition to image instances was accompanied with contextual side information (metadata). Inspired by this observation, very recently (2017) Strezoski et al. achieved state-of-the-art results in the Rijksmuseum challenge (Mensink and Van Gemert 2014) by fine-tuning the last layer of a pre-trained CNN as a common representation for solving multiple and complementary recognition tasks (e.g. period, materials and artist recognition) concurrently. We could try to reproduce

their work and study how the learned embedding performs in a recommendation setting.

There is also room for improvement with regards to the use of pre-trained CNNs. First of all, there are new state-of-the-art architectures we could use, such as Inception-v4 and Inception-ResNet-v2 (Szegedy et al. 2017). Furthermore, there is no need for limiting ourselves to just the last fully-connected layers in a pre-trained CNN, we can also leverage the rich stylistic, lower-level features captured by the convolutional layers that are closer to the input image, as shown by the work of Gatys et al. (2015). We foresee combining these ideas with siamese loss, triplet loss or even multitask learning in order to learn high quality artistic image embeddings. Another aspect for improvement is with respect to manually-engineered visual features. In addition to Attractiveness and LBP, there are more state-of-the-art handcrafted feature extraction techniques we could test, such as Histogram of Oriented Gradients (HOG) (Dalal and Triggs 2005), Scale-Invariant Feature Transform (SIFT) (Lowe 2004), Binarized Statistical Image Features (BSIF) (Kannala and Rahtu 2012), Extended Local Ternary Patterns (ELTP) (Liao and Young 2010), among others. This would allow us to make a more robust comparison between MEVF and DNN.

Finally, we are also interested in improving the statistical significance of our results, both offline and online. Therefore we are planning on conducting a larger analysis with more transactional data from our partners at *UGallery*, and we also want to conduct large scale user studies on online platforms such as Amazon Mechanical Turk. In fact, a massive online study could give us the chance to study other aspects, such as different ways of using MEVF and metadata to generate explanations for recommendations and study their effects on user experience.

Acknowledgements This research has been supported by the Chilean research agency Conicyt, under Fondecyt Grant 11150783, and partially funded by the Millennium Institute for Foundational Research on Data (IMFD). We also acknowledge the help from Felipe del Río and Domingo Mery, who helped us frame some evaluations and provided us with some interesting ideas for future work.

References

Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)

Aggarwal, C.C.: Content-based recommender systems. In: Recommender Systems, pp. 139–166. Springer, Berlin (2016). https://doi.org/10.1007/978-3-319-29659-3_4

Akay, S., Kundegorski, M.E., Devereux, M., Breckon, T.P.: Transfer learning using convolutional neural networks for object classification within X-ray baggage security imagery. In: Proceedings of the IEEE International Conference on Image Processing (ICIP), pp. 1057–1061 (2016)

Albanese, M., d'Acierno, A., Moscato, V., Persia, F., Picariello, A.: A multimedia semantic recommender system for cultural heritage applications. In: Proceedings of the Fifth IEEE International Conference on Semantic Computing (ICSC), pp. 403–410 (2011)

Amatriain, X.: Mining large streams of user data for personalized recommendations. ACM SIGKDD Explor. Newsl. **14**(2), 37–48 (2013)

Aroyo, L., Wang, Y., Brussee, R., Gorgels, P., Rutledge, L., Stash, N.: Personalized museum experience: the rijksmuseum use case. In: Proceedings of Museums and the Web (2007)

Bennett, J., Lanning, S., et al.: The netflix prize. In: Proceedings of KDD Cup and Workshop, vol. 2007, p. 35 (2007)

Benouaret, I., Lenne, D.: Personalizing the museum experience through context-aware recommendations. In: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 743–748 (2015)

Celma, O.: Music recommendation. In: Music Recommendation and Discovery, pp. 43–85. Springer, Berlin (2010). https://doi.org/10.1007/978-3-642-13287-2_3

Cremonesi, P., Koren, Y., Turrin, R.: Performance of recommender algorithms on top-n recommendation tasks. In: Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys '10, pp. 39–46 (2010)

Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR) **1**, 886–893 (2005)

David, O.E., Netanyahu, N.S.: DeepPainter: Painter Classification Using Deep Convolutional Autoencoders, pp. 20–28. Springer, Berlin (2016)

de Gemmis, M., Lops, P., Musto, C., Narducci, F., Semeraro, G.: Semantics-aware content-based recommender systems. In: Recommender Systems Handbook, pp. 119–159. Springer, Berlin (2015)

Deldjoo, Y., Elahi, M., Cremonesi, P., Garzotto, F., Piazzolla, P., Quadrana, M.: Content-based video recommendation system based on stylistic visual features. J. Data Semant. **5**(2), 99–113 (2016)

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 248–255 (2009)

Ekstrand, M.D., Kluver, D., Harper, F.M., Konstan, J.A.: Letting users choose recommender algorithms: an experimental study. In: Proceedings of the 9th ACM Conference on Recommender Systems, RecSys '15, pp. 11–18 (2015). https://doi.org/10.1145/2792838.2800195

Elahi, M., Deldjoo, Y., Bakhshandegan Moghaddam, F., Cella, L., Cereda, S., Cremonesi, P.: Exploring the semantic gap for movie recommendations. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys'17, pp. 326–330 (2017)

Esman, A.R.: The World's Strongest Economy? The Global Art Market. https://www.forbes.com/sites/abigailesman/2012/02/29/the-worlds-strongest-economy-the-global-art-market/ (2012). Accessed 21 March 2017

Gatys, L.A., Ecker, A.S., Bethge, M.: A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576 (2015)

Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587 (2014)

Gomez-Uribe, C.A., Hunt, N.: The netflix recommender system: algorithms, business value, and innovation. ACM Trans. Manag. Inf. Syst. (TMIS) **6**(4), 13 (2016)

Gonzalez, R.C., Eddins, S.L., Woods, R.E.: Digital Image Publishing Using MATLAB. Prentice Hall, Upper Saddle River (2004)

Harper, F.M., Konstan, J.A.: The movielens datasets: history and context. ACM Trans. Interact. Intell. Syst. **5**(4), 19:1–19:19 (2015)

He, R., Fang, C., Wang, Z., McAuley, J.: Vista: A visually, socially, and temporally-aware model for artistic recommendation. In: Proceedings of the 10th ACM Conference on Recommender Systems, RecSys '16, pp. 309–316 (2016)

He, R., McAuley, J.: VBPR: Visual Bayesian personalized ranking from implicit feedback. In: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, pp. 144–150 (2016)

Hidasi, B., Quadrana, M., Karatzoglou, A., Tikk, D.: Parallel recurrent neural network architectures for feature-rich session-based recommendations. In: Proceedings of the 10th ACM Conference on Recommender Systems, RecSys'16, pp. 241–248 (2016)

Kannala, J., Rahtu, E.: Bsif: Binarized statistical image features. In: Proceedings of 21st International Conference on Pattern Recognition (ICPR), pp. 1363–1366 (2012)

Karnowski, J.: AlexNet + SVM. https://jeremykarnowski.files.wordpress.com/2015/07/alexnet2.png (2015). Accessed 1 Dec 2017

Knijnenburg, B.P., Bostandjiev, S., O'Donovan, J., Kobsa, A.: Inspectability and control in social recommenders. In: Proceedings of the Sixth ACM Conference on Recommender Systems, pp. 43–50 (2012)

Koch, G., Zemel, R., Salakhutdinov, R.: Siamese neural networks for one-shot image recognition. In: Proceedings of ICML Deep Learning Workshop, vol. 2 (2015)

Konstan, J.A., Riedl, J.: Recommender systems: from algorithms to user experience. User Model. User Adapt. Interact. **22**(1–2), 101–123 (2012)

Koren, Y.: Collaborative filtering with temporal dynamics. Commun. ACM **53**(4), 89–97 (2010)
Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of Advances in Neural Information Processing Systems 25(NIPS), pp. 1097–1105 (2012)
La Cascia, M., Sethi, S., Sclaroff, S.: Combining textual and visual cues for content-based image retrieval on the world wide web. In: Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries, pp. 24–28 (1998)
Lacic, E., Kowald, D., Eberhard, L., Trattner, C., Parra, D., Marinho, L.B.: Utilizing online social network and location-based data to recommend products and categories in online marketplaces. In: Atzmueller M., Chin A., Scholz C., Trattner C. (eds) Mining, Modeling, and Recommending 'Things' in Social Media. MUSE 2013, MSM 2013. Lecture Notes in Computer Science, vol 8940. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-14723-9_6
Larrain, S., Trattner, C., Parra, D., Graells-Garrido, E., Nørvåg, K.: Good times bad times: a study on recency effects in collaborative filtering for social tagging. In: Proceedings of the 9th ACM Conference on Recommender Systems, RecSys'15, pp. 269–272 (2015)
Lei, C., Liu, D., Li, W., Zha, Z.J., Li, H.: Comparative deep learning of hybrid representations for image recommendations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2545–2553 (2016)
Liao, W.H., Young, T.J.: Texture classification using uniform extended local ternary patterns. In: Proceedings of IEEE International Symposium on Multimedia (ISM), pp. 191–195 (2010)
Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
Maaten, L.V.D., Hinton, G.: Visualizing data using t-sne. J. Mach. Learn. Res. **9**(Nov), 2579–2605 (2008)
Macedo, A.Q., Marinho, L.B., Santos, R.L.: Context-aware event recommendation in event-based social networks. In: Proceedings of the 9th ACM Conference on Recommender Systems, pp. 123–130 (2015)
Maes, P., et al.: Agents that reduce work and information overload. Commun. ACM **37**(7), 30–40 (1994)
Manning, C.D., Raghavan, P., Schütze, H., et al.: Introduction to Information Retrieval, vol. 1. Cambridge University Press Cambridge, Cambridge (2008)
Mathieu, M.F., Zhao, J.J., Zhao, J., Ramesh, A., Sprechmann, P., LeCun, Y.: Disentangling factors of variation in deep representation using adversarial training. In: Proceedings of Advances in Neural Information Processing Systems, pp. 5040–5048 (2016)
McAuley, J., Targett, C., Shi, Q., Van Den Hengel, A.: Image-based recommendations on styles and substitutes. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 43–52 (2015)
Mensink, T., Van Gemert, J.: The rijksmuseum challenge: Museum-centered visual recognition. In: Proceedings of International Conference on Multimedia Retrieval, p. 451 (2014)
Nguyen, A., Yosinski, J., Clune, J.: Multifaceted feature visualization: uncovering the different types of features learned by each neuron in deep neural networks. arXiv preprint arXiv:1602.03616 (2016)
Nunes, I., Jannach, D.: A systematic review and taxonomy of explanations in decision support and recommender systems. User Model. User Adapt. Interact. **27**(3), 393–444 (2017)
Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recognit. **29**(1), 51–59 (1996)
Olah, C., Mordvintsev, A., Schubert, L.: Feature visualization. Distill (2017) . https://doi.org/10.23915/distill.00007
Parra, D., Brusilovsky, P.: User-controllable personalization: a case study with setfusion. Int. J. Hum. Comput. Stud. **78**, 43–67 (2015)
Parra, D., Sahebi, S.: Recommender systems: sources of knowledge and evaluation metrics. In: Advanced Techniques in Web Intelligence-2, pp. 149–175. Springer, Berlin (2013). https://doi.org/10.1007/978-3-642-33326-2_7
Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, pp. 452–461 (2009)
Rui, Y., Huang, T.S., Ortega, M., Mehrotra, S.: Relevance feedback: a power tool for interactive content-based image retrieval. IEEE Trans. Circuits Syst. Video Technol. **8**(5), 644–655 (1998)
Rush, J.C.: Acquiring a concept of painting style. Stud. Art Educ. **20**(3), 43–51 (1979)

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015)

San Pedro, J., Siersdorfer, S.: Ranking and classifying attractiveness of photos in folksonomies. In: Proceedings of the 18th International Conference on World Wide Web, WWW'09, pp. 771–780 (2009)

Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 815–823 (2015)

Semeraro, G., Lops, P., De Gemmis, M., Musto, C., Narducci, F.: A folksonomy-based recommender system for personalized access to digital artworks. J. Comput. Cult. Herit. (JOCCH) **5**(3), 11 (2012)

Shankar, D., Narumanchi, S., Ananya, H., Kompalli, P., Chaudhury, K.: Deep learning based large scale visual recommendation and search for e-commerce. arXiv preprint arXiv:1703.02344 (2017)

Sharif Razavian, A., Azizpour, H., Sullivan, J., Carlsson, S.: CNN features off-the-shelf: an astounding baseline for recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 806–813 (2014)

Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Proceedings of AAAI, vol. 4, p. 12 (2017)

Tintarev, N., Masthoff, J.: Explaining recommendations: design and evaluation. In: Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook. Springer, Boston (2015). https://doi.org/10.1007/978-1-4899-7637-6_10

Trattner, C., Elsweiler, D.: Investigating the healthiness of internet-sourced recipes: implications for meal planning and recommender systems. In: Proceedings of the 26th International Conference on World Wide Web, pp. 489–498 (2017)

Verbert, K., Parra, D., Brusilovsky, P., Duval, E.: Visualizing recommendations to support exploration, transparency and controllability. In: Proceedings of the 2013 International Conference on Intelligent User Interfaces, pp. 351–362 (2013)

Wang, J., Song, Y., Leung, T., Rosenberg, C., Wang, J., Philbin, J., Chen, B., Wu, Y.: Learning fine-grained image similarity with deep ranking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1386–1393 (2014)

Weinswig, D.: Art Market Cooling, But Online Sales Booming. https://www.forbes.com/sites/deborahweinswig/2016/05/13/art-market-cooling-but-online-sales-booming/ (2016). Accessed 21 March 2017

Yang, L., Cui, Y., Zhang, F., Pollak, J.P., Belongie, S., Estrin, D.: PlateClick: bootstrapping food preferences through an adaptive visual interface. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM'15, pp. 183–192 (2015)

Pablo Messina is pursuing a Master Degree in Computer Science at Pontificia Universidad Católica de Chile (PUC) with a focus on Recommender Systems and Deep Learning. He holds a B.S. in Engineering also at PUC. He has participated in the ACM International Collegiate Programming Contest (ICPC) obtaining the second place in Chile twice. He is a member of the SocVis Research Group at PUC Chile. His main research interests include applications of machine learning and deep learning to solve real world problems.

Vicente Dominguez is currently pursuing a Master Degree in Computer Science at Pontificia Universidad Católica de Chile (PUC). He holds a B.S. in Engineering also at PUC. He is part of SocVis Research Group, a group of researchers focused on recommender systems, social networks, HCI and visual analytics. His main research interests include machine learning and recommender systems.

Denis Parra is Assistant Professor at the Department of Computer Science, in the School of Engineering at PUC Chile. He obtained a Ph.D. in Information Science from University of Pittsburgh, USA, in 2013. His main research interests are Recommender Systems, Information Visualization and Applications of Machine Learning and Data Mining. He has published in important conferences in the area such as

RecSys, IUI, Hypertext and UMAP, as well as in journals such as IJHCS, ESWA, and ACM TiiS. He is currently leading the SocVis Research Group at PUC Chile.

Christoph Trattner is currently working as an Associate Professor at the University of Bergen in the Information Science Department. Previously to that, he was an Asst. Prof. at MODUL University Vienna in the New Media Technology Department and an area manager at the Know-Center, Austria's research competence for data-driven business and Big Data analytics where he founded and led the Social Computing area. He holds a Ph.D. (with distinction), an MSc (with distinction) and a B.Sc. in Computer Science and Telematics from Graz University of Technology (Austria). His research interests include Data Science and Recommender Systems. Since 2010, he published two books and over 80 scientific articles, some of them in top CORE A* ranked conferences and journals.

Alvaro Soto received his Ph.D. in Computer Science from Carnegie Mellon University in 2002. Afterwards, he joined the Computer Science Department at Pontificia Universidad Católica de Chile, PUC, where he is currently an Associate Professor. At PUC, he is also part of IA-Lab, one of the leading research groups dedicated to the development of machine intelligence in Latin America. His main research interests include machine learning, visual recognition, and cognitive robotics.