Image embedding and user multi-preference modeling for data collection sampling

Anju Jose Tom, Laura Toni, Thomas Maugey

EURASIP Journal on Advances in Signal Processing

RESEARCH Open Access

Image embedding and user multi-preference modeling for data collection sampling

Anju Jose Tom[1*], Laura Toni[2] and Thomas Maugey[1]

*Correspondence:
anjujosetom@gmail.com

[1] Inria Rennes Bretagne Atlantique, Beauliau Campus, Rennes, France
[2] University College London, London WC1E 6BT, UK

Abstract

This work proposes an end-to-end user-centric sampling method aimed at selecting the images from an image collection that are able to maximize the information perceived by a given user. As main contributions, we first introduce novel metrics that assess the amount of *perceived* information retained by the user when experiencing a set of images. Given the actual information present in a set of images, which is the volume spanned by the set in the corresponding latent space, we show how to take into account the user's preferences in such a volume calculation to build a user-centric metric for the perceived information. Finally, we propose a sampling strategy seeking the minimum set of images that maximize the information perceived by a given user. Experiments using the coco dataset show the ability of the proposed approach to accurately integrate user preference while keeping a reasonable diversity in the sampled image set.

Keywords: Perceived information, Representation learning, User preference modeling, ResNeXt, COCO dataset, Image collection sampling

1 Introduction

Image sampling based on user perception has recently increased interest given the unstoppable increase of stored datasets. For example, let's think of the large-scale image collections associated with Instagram and other social media accounts, online retailers, or your own personal photograph gallery saved on the cloud. Storing such large-scale datasets has become a big burden, from an economical as well as sustainability perspective [1]. Traditional image data sampling methods can be understood as a way to alleviate such burden, by retaining only pictures from a dataset with the ideal goal of providing a summary (in the sense of a brief description) of the database. The challenging aspect is, however, to understand which is the best set of images to select (not to delete), especially given the user's preferences. This is highly challenging as the user will perceive the deleted images as lost information. In this work, we aim specifically at minimizing this sense of lost information. Specifically, we propose a large-scale data sampling technique aimed at identifying the key images that should be preserved to maximize the information perceived by a given user.

Compared to existing sampling strategies, the context and concept of the sampling presented for this work are distinct. Consider a user who has free memory on their device and needs to decide whether to keep or erase a photograph folder that he may or may not revisit in the future. Normally the user has two options, to keep or to delete the images. Here, we offer a third option, *i.e.*, a personalized sampling experience that produces a subset from the image folder each time corresponding to the multiple preferences without any loss of pixel information. A subset given out from this sampling scheme does not completely represent the whole dataset instead, it gives the user's preferences at that point of time. Hence, this sampling tool mainly asks for how much data to sample and what semantic information the user needs to retain. The user is finally left with the impression that no pixel information was lost during the sampling.

In order to define such an acceptable sample, the semantic information in the images is more significant than the pixel information. Unlike the conventional similarity search-based sampling approaches, we prefer to use a sampling strategy that preserves the user's perceived information while maintaining image diversity. This is because the way a user perceives an image or an item can be a powerful sign of his likes and dislikes (and decisions), and thus, the amount of perceived information is proven to be a relevant measure of the user's preferences which essentially is a semantic information. However, the representations must be able to model a certain amount of very precise data and easily deal with the newly added items to the data collection. The wide range of features available when considering the feature space of a large-scale image database stretches the dimensionality leading to computational troubles and also bringing down the overall prediction accuracy. Selecting an optimal image subset of preferred features from such a large image collection still exists as an NP-hard problem. To address this, the perceived information (PI)-based sampling introduced in [2] is extended and strengthened by adopting image embeddings and incorporating a user preference model. This leads to the sampling algorithm proposed in this work, named Reinforced Image Collection Sampling (RICS).

With respect to classical video summarization schemes [3–7], which usually preserves the partial information of the original sources to produce a data outline, our work implements the personalized sampling experience even when the original source dataset is completely lost or unavailable. The methods such as [8] combine an attention mechanism to identify the important parts of the video and are further trained unsurprisingly. Image grouping methods [9–11] with image captioning are also in line with combining self-attention mechanism with contrastive feature construction to effectively summarize common information from each image group while capturing discriminative information between them. There exists a variety of minimum redundancy maximum relevance (MRMR)-based schemes [12–15], cosine similarity approaches [16–18] and multi-class feature selection approaches for feature selection strategy [19] designed and implemented for row-wise feature selection, whereas our feature vectors are arranged as columns. How this arrangement happens is explained in Sect. 3. There are also works based on determinantal point processes (DPPs) [20]-based sampling schemes that perform well for cases below the rank of the feature matrix and thus it is useful for the small-scale datasets. We often come across various large image collections, for example, from social media accounts, blogs, stock photographs, etc. This work aims to sample such a

vast image collection and deliver a customized visual search engine by maximizing the presence of user-preferred images while minimizing the inter-image redundancies present in the sampled data collection.

1.1 Contributions

The Reinforced Image Collection Sampling (RICS) is novel when considering the following key contributions.

- We extend the PI-based sampling scheme introduced in [2] to deliver a visual search engine by appropriately plugging a ResNeXt weakly supervised learning (WSL)-based image embedding to it.
- In addition, this work proposes a novel user multi-preference modeling for the image collection sampling to ensure personalized image collection sampling experience for each user.

The remainder of this paper is organized as follows. Section 2 presents the data collection sampling problem including motivation. The detailed description on each subportion of the work, *i.e.*, the image embedding and user multi-preference modeling, are sketched, respectively, in Sects. 3 and 4. The experiments and the evaluation strategies along with quantitative and qualitative results are provided in Sect. 6 Conclusions are outlined in Sect. 7.

1.2 Notations

We represent a data collection with calligraphy letters, e.g.: $\mathcal{X}$. Boldface capital letters are used to represent matrices, e.g.: $\mathbf{B}$. Boldface small letters represent vectors, e.g.: $\mathbf{b}$. Scalars are represented using small letters in italics.

2 Data collection sampling problem

Consider an image data collection $\mathcal{X} = \{X_1, X_2, ..., X_N\}$. Let ω be a sampler defined as, $\omega : \mathcal{X} \rightarrow \mathcal{Y}$, where $\mathcal{Y}$ denotes the sampled image collection. The sampler tries to maximize a given metric denoted by $\mathcal{Q}_u$ that measures how relevant a sampled set $\mathcal{Y}$ for a given user's preference u. This notion of "good" sampled set depends on the application and will be discussed in Sect. 5. In a nutshell, the sampled image collection $\mathcal{Y} = \omega(\mathcal{X})$ should depict as much as possible what a user with preferences u perceives as information in $\mathcal{X}$. The data collection sampling problem can thus be formulated as follows:

$$\begin{aligned} &\max_{\omega} \ \mathcal{Q}_u(\omega(\mathcal{X})) \\ &\text{s.t.} \ \ |\omega(\mathcal{X})| \leq M \end{aligned} \tag{1}$$

where M is the maximum number of images in the sampled collection.

This formulation raises three major issues.

1 How to capture and model what is perceived and retained by a user in a data collection $\mathcal{X}$?
2 How to specifically model individual user's preference u to produce personalized data collection sampling?

3 How to define a metric $\mathcal{Q}_u$ that properly captures the subjectiveness of how a user perceives a data collection $\mathcal{X}$?

In order to tackle these issues, we propose a scheme based on three original contributions. The first one consists of an image embedding algorithm to capture the high-level information present in each image $X_i \in \mathcal{X}$. The estimated features are vectors $\{\mathbf{b}_i\}_i$, each of them describing the semantic information present in X_i. The data collection information is depicted by the matrix $\mathbf{B}$ whose columns are made of each single $\mathbf{b}_i$. This contribution is detailed in Sect. 3. The second part of the overall scheme aims at capturing the user's preference in the feature space. From a simple user's input formulated as high-level preferences (e.g., a user prefers images with portraits to those containing only landscape), our solution builds a set of learned weights giving more or less importance to the features in the embedded space. This user's preference modeling is detailed in Sect. 4. The third part gathers the user preference model and the feature matrix to define a $\mathcal{Q}_u$ metric based on the perceived information metric introduced in our previous work in [2]. This enables us to define a sampler guided by $\mathcal{Q}_u$. This part is detailed in Sect. 5 A general descriptive diagram of the proposed work is illustrated in Fig. 1.

3 Image embedding

It is nowadays accepted that an image is more complex than a simple concatenation of individual pixels. It is therefore meaningless to represent the information of an image by a vectorized version of it, namely $\mathbf{x} \in \mathbb{R}^{H\times W}$ (where H and W are, respectively, the height and width of an image). The pixels, when arranged together, can indeed represent a shape that may have a high-level meaning (e.g., a wheel, a line). The concatenation of several shapes may also lead to higher-level information (e.g., a bike). This image analysis can continue like that toward a global interpretation of the high-level information spanned by the image. This process can be modeled by a function:

$$\begin{aligned} f : \mathbb{R}^{H\times W} &\to \mathbb{R}^{L} \\ \mathbf{x} &\mapsto \mathbf{b}. \end{aligned} \tag{2}$$

The latent vector $\mathbf{b}$ of dimension L describes the *high-level information* in $\mathbf{x}$ and is precisely what we want to capture/model when sampling a data collection. The function f is usually called *image embedding* [21–23] in the literature. Embedding functions usually respect the following two properties. First, it enables to reduction of the dimension:

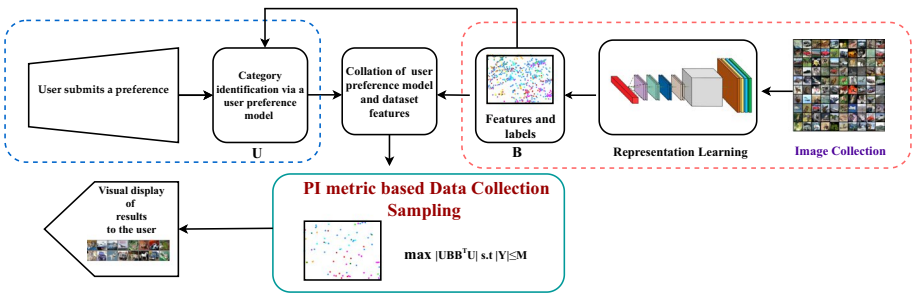

Fig. 1 Proposed data collection sampling pipeline

$D < H \times W$. Second, it enables us to describe the similarity between two images as their angle in the latent space. More formally, if $\mathbf{x}_i$ and $\mathbf{x}_j$ are two images, the amount of common high-level information can be measured by the so-called *cosine-similarity* $\mathbf{b}_i^\top \mathbf{b}_j$, which simply measures the angle between their embedded descriptions in the latent space (assuming that they are normalized). Note that two images could be different in the pixel domain, i.e., $\mathbf{x}_i^\top \mathbf{x}_j \approx 0$, but much correlated in the latent space, i.e., $\mathbf{b}_i^\top \mathbf{b}_j \approx 1$. This is, for example, the case of two images describing bikes of different colors and shapes. This property is really useful for data collection sampling as the sampler aims at reducing the redundancy that resides at the semantic level between the images. This is the reason why the first step of our proposed data collection sampling consists of an image embedding stage.

Image embedding has been intensively studied in the literature for various tasks [24–26]. In this work, we use the weakly supervised learning (WSL) of the ResNeXt model [27, 28], which is an evolved version of the widely adopted ResNet model [29, 30] in which a so-called cardinality dimension is added. Moreover, the weakly supervised learning (WSL) version from the Facebook AI Research (FAIR) group became established as it was trained on 3.5 billion public Instagram pictures in order to predict around 8000 hashtags sourced by the users. The mini version of ResNeXt-WSL (32x8d) is chosen for the proposed work since it is still accurate enough on the public benchmarks with less memory requirement and computational reliability. ResNeXt also has its clear influence here as a powerful pre-trained network to extract embedding vectors that can accurately describe any kind of picture in an abstract latent feature space on account of the in-depth pre-training done on ImageNet dataset [31] using a wide range of about 940 million public images and 1.5k hashtags sourced by the users. Thus, ResNeXt-WSL becomes the well-suited representation learner here for the proposed work. An illustration of our representation learning section is illustrated in Fig. 2. We focus on the recently released WSL model [28], *i.e.*, the simplified version that was released in the PyTorch hub (bottleneck width = 8) which uses a $32 \times 8d$ architecture. Leaving aside the in-depth architecture, we are interested in the final section of the average pooling layer in the ResNeXt which combines all the filters into a single 2048-dimensional vector. This vector was then meant to predict the 1000 classes of the ImageNet dataset [32] after the softmax layer. Since the class predictions are not a point of interest for the proposed work, we proceed with the 2048-dimensional feature array from the average pool layer as the embedding features for our image collection. The pooling in turn is significant in reducing the dimensionality of the feature maps [29] making it computationally economic. The N feature vectors of dimension L=2048 received from the ResNeXt last layer are combined as columns to form a broader feature matrix $\mathbf{B} \in \mathbb{R}^{L \times N}$ describing

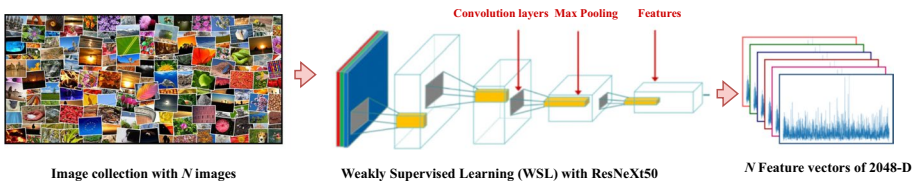

Fig. 2 The representation learning section of the proposed work

the dataset's high-level information. In this work, each item of a data collection is represented with a feature vector. All the feature vectors corresponding to each item of the data collection are aggregated in a matrix **B**. Therefore, a row of the matrix corresponds to one feature describing each item of the data collection. And, a column of the matrix corresponds to the complete feature description of a single item. The problem of data collection sampling tackled in this paper then consists of choosing which columns of **B** to keep. The rows remain unchanged.

4 User multi-preference acquisition and modeling

While embedding function pushes the information modeling toward the interpretation level, its subjectiveness naturally increases. In neuro-science, it is well-known that everyone "sees" (i.e., understands) what he/she wants to see in an image. On top of this cognitive bias, the user's tastes may also influence the definition of what is important in the information present in an image. Therefore, users may have different levels of priority between the different information to keep or delete in a sampling process. These preferences could be expressed in a semantic forms, e.g., a user may prefer to keep the portrait while another rather prefers the landscapes. However, translating this information to a mathematical form is not an easy task. In practice, the semantic expression of a preference should be expressed in the same latent space $\mathbb{R}^L$ as the function f. Let us assume that a user is able to give an information $\mathbf{c} \in \mathbb{R}^C$. We consider C predefined classes and $\mathbf{c}$ describe the user's scores for each of them. This score depicts how much a user likes the corresponding class. This score can be either given by the user directly or learned by algorithm [33]. From a vector $\mathbf{c}$, the goal is to find a latent vector weighting the importance of each feature:

$$\begin{aligned} \mathbb{R}^C &\to \mathbb{R}^L \\ \mathbf{c} &\mapsto \mathbf{u}, \end{aligned} \tag{3}$$

In this paper, we propose user modeling. The user queries are treated such that the categories of the multiple preferences are identified (e.g., the category trees and animals for a user who submits the query, I love trees and animals). Keeping this in mind, we derive a vector $\mathbf{u}$ containing weights that represent the energy of each feature in a particular category. These vectors are derived for all C categories that we want to consider. Later, the weights form the diagonals of the user preference model matrix $\mathbf{U}$ for every category in the image collection. This diagonal matrix can thus be used to weight a feature vector by simply calculating $\mathbf{Ub}$. The proposed user's preference modeling is described in Algorithm 1. In other words Algorithm 1 takes the user's preference $\mathbf{c}$ and computes the popularity of each feature. The matrix is then built as $\mathbf{U} = diag(\mathbf{u})$ and is thus diagonal.

5 Sampling scheme

This section explains how we related and adopted an appropriate $\mathcal{Q}$ metric that satisfied the following requirements that we expected.

- The quality metric ($\mathcal{Q}$) should describe the volume of information globally spanned by a picture collection. Adding a new picture to a collection should only

increase (or keep constant) this volume (and not reduce it). For this reason, the metric should be computed in the feature's space domain.

- The sampling scheme shall proceed in accordance with the user-perceived information which is obvious from the user multi-preferences while preserving the features quality and features diversity.

A preferable choice satisfying these is the perceived information (PI) metric introduced in [2], and it is well compatible with the intended $\mathcal{Q}$ metric for the data collection sampling.

This can be justified in view of the fact that the PI metric reflects the volume spanned by the covariance matrix of the features, i.e., information in the latent space. The non-relying nature on the source dataset size is one special characteristic of this covariance matrix which makes it compatible with sampling tasks. Moreover, the properties of the PI [2] match absolutely with the objectives of the image collection sampling scheme.

Considering PI as a quality metric in the proposed scheme, we first model the feature description of every image in the dataset. We assume the existence of a kernel function that maps every item in $\mathcal{X}$ to an L-dimensional feature space, as the function f in Eq. (2). Thus, all the N feature vectors representing each image in the collection are now gathered as the columns of a matrix $\mathbf{B} \in \mathbb{R}^{L\times N}$. In addition, we assume the presence of a covariance matrix $\Sigma^u_{\mathcal{Y}}$ that matches the user-preferred sampled subset of features in the definition of PI [2]. This is termed each user's weighted personalized covariance matrix (WPCM) and can be formulated straightforward as follows [2]:

$$\Sigma^u_{\mathcal{Y}} = \mathbf{U}\mathbf{B}_{\mathcal{Y}}\mathbf{B}^T_{\mathcal{Y}}\mathbf{U} \tag{4}$$

where $\mathbf{B}_{\mathcal{Y}} \in \mathbb{R}^{L\times M}$ is the submatrix of $\mathbf{B}$ whose columns are indexed by $\mathcal{Y}$ and $\mathbf{U}$ is the user's preference matrix introduced in Sect. 4. This throws light on the degree to which the user prefers each feature. In fact, each element in the WPCM describes the covariance among the features and is emphasized by the user weights. The PI metric is mathematically expressed as:

$$\pi_u(\mathcal{Y}) = \frac{L}{2}\log_2(2\pi e|\Sigma^u_{\mathcal{Y}}|) = \frac{L}{2}\log_2(2\pi e|\mathbf{U}\mathbf{B}_{\mathcal{Y}}\mathbf{B}^T_{\mathcal{Y}}\mathbf{U}|) \tag{5}$$

From Eq. (5), it is obvious that the determinant of the WPCM in the high level demonstrates the volume spanned by the sampled images in the latent space [34]. To be more specific, when the weighted features are highly correlated, the volume of the latent space remains small. As we intend to obtain diverse features with due consideration to user preference and popularity, volume maximization will result in maximizing the quality of the data collection sampling. As a matter of fact, maximizing the PI involves lifting the user-preferred features with high values. However, this maximization of the determinant of the WPCM also accounts for penalization when it encounters multiple sources appealing for particular features. Thus, maximizing this personalized PI, in turn, minimizes the inter-item redundancies present in the data collection and renders a visual search engine depending on the user's preference.

5.1 Data collection sampling

The PI-based image collection sampling problem is detailed in this section. The sampling problem has to tackle the constraints on the user-preferred categories and sampling rate restriction as well (i.e., size of the subset). Reformulation of the data collection sampling problem in Eq. (1) in terms of PI metric can be done as follows:

$$\begin{aligned} &\max_{\omega_u} \ \pi_u(\omega(\mathcal{X})) \\ &\text{s.t.} \ \ |\omega(\mathcal{X})| \leq M \end{aligned} \tag{6}$$

Since the sampled collection can be expressed as, $\omega_u(\mathcal{X}) = \mathcal{Y}$, Eq. (6) can be converged to the PI-based compression problem [2] as follows.

$$\begin{aligned} \max_{\mathcal{Y}\subseteq\mathcal{X}} \ \frac{L}{2}\log_2(2\pi e|\mathbf{U}\mathbf{B}_{\mathcal{Y}}\mathbf{B}_{\mathcal{Y}}^T\mathbf{U}|) \\ \text{s.t.} \ \ |\mathcal{Y}| \leq M \end{aligned} \tag{7}$$

Equation (7) calls for the maximization of the determinant of the WPCM, *i.e.,* $|\mathbf{U}\mathbf{B}_{\mathcal{Y}}\mathbf{B}_{\mathcal{Y}}^T\mathbf{U}|$. To achieve this, the sampling has to progress in the direction such that WPCM carries the highest possible diagonal entries while ensuring just the essential (least possible values for) off-diagonal entries. This is where the term *popularity* of an image with feature b given by $\|\mathbf{U}\mathbf{b}\|_2^2$ has its significance. With the popularity held high (by ensuring high values for user-preferred ones), the diagonal entries get the maximum values possible. This also assures a sampled set $\mathcal{Y}$ that bears all the popular images. At the same time, the off-diagonal entries play a role here to characterize the similarity between the selected images in the sampled set (*i.e.,* between the rows of $B_{\mathcal{Y}}$). While the operation of the basic PI-based scheme [2] is by random selection of *items*, the reinforced algorithm operates by selecting *images* randomly from an image collection for a given popularity considering the *user's multiple preferences*. The probability of each image is decided as the trade-off established between popularity and dissimilarity. This PI sampling scheme

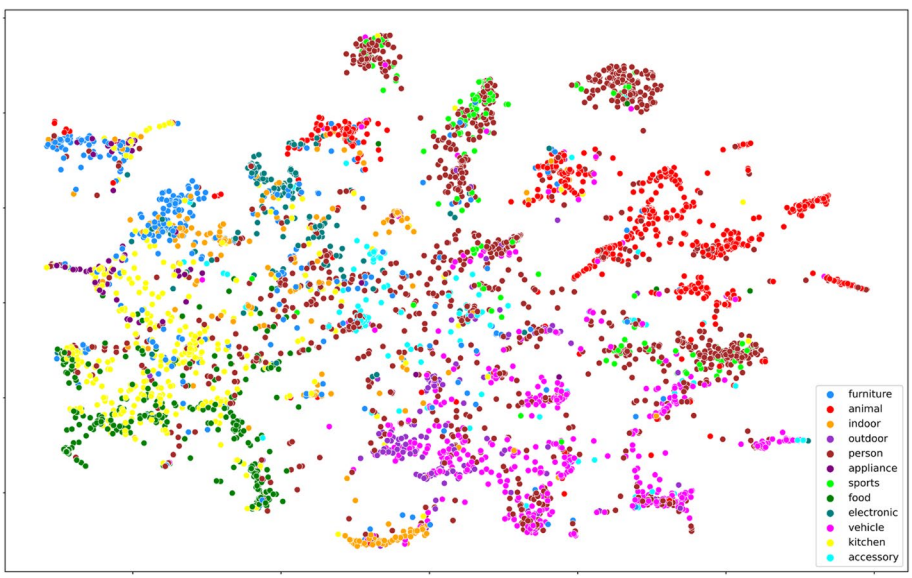

Fig. 3 The different supercategories of coco dataset embedded on t-SNE

with image embedding and user preference modeling is hence called Reinforced Image Collection Sampling (RICS) and is briefed in Algorithm 1 and Algorithm 2.

Algorithm 1 User Multi-Preference Modeling algorithm

Input: 1: The user's multi- category preference submission $c_1, c_2..c_v$ where $v \subseteq$, number of categories. 2: Feature vector matrix **B.**
Results: User multi-preference weight matrix **U.**
Initializations: weight, $\mathfrak{w} = 0$, counter, $\mathfrak{c} = 0$
Defining weights for features of categories with preferences
for i in all category labels **do**
  if $i \in \{c_1 \cup c_2, ..., c_v\}$ **then**
    for $j = 1$ to N, i.e all images **do**

$$\mathfrak{w} = \begin{cases} \mathfrak{w} + \mathbf{B}[:, j] & \text{if } z[j] \in \{c_1 \cup c_2, ..., c_v\} \\ \mathfrak{w} & \text{if } z[j] \notin \{c_1 \cup c_2, ..., c_v\} \end{cases}$$

    end for
  end if
end for
c=c+1
$\mathfrak{w} = \mathfrak{w}/\mathfrak{c}$
Build user preference matrix **U**=diag($\mathfrak{w}$)

Algorithm 2 Reinforced Image Collection Sampling (RICS) algorithm

Data: The user's query or multi- category preference submission $c_1, c_2..c_v$ where $v \subseteq$, number of categories , the sampling rate M
Result : The sampled images visually displayed to the user *i.e* sampled set $\mathcal{Y}$ mathematically
Initialization : sampled set $\mathcal{Y} = 0 \in \mathbb{R}^{L \times M}$, similarity vector $s = 0 \in \mathbb{R}^N$, trade-off parameter $\lambda = 0.001$
Build user preference model **U** using **Algorithm 1**
Assemble matrix $\mathbf{B} \in \mathbb{R}^{L \times N}$ built from representation learning method outlined in section 3
Popularity assessment ($p \in \mathbb{R}^N$) of the all N images
$p = \text{diag}(\mathbf{B}^T U U \mathbf{B})$
for m=1 to M **do**
  Average probability estimation with a set popularity-dissimilarity trade-off,
  $\pi = \sqrt{p} - \lambda s$,
  $\pi = \frac{\pi}{\Sigma_{j=1}^{N} \pi_j}$
  Random choice of images to the subset $\mathcal{Y}$
  $l =$ Randomchoice($\mathcal{X} \backslash \mathcal{Y} | \pi_{\mathcal{X} \backslash \mathcal{Y}}$), $\mathcal{Y} = \mathcal{Y} \cup \{l\}$
  Popularity-similarity update $s = \frac{1}{M} \left(\Sigma_{i=1}^{M} \mathbf{B}_{\mathcal{Y}}(i)\right)^T \mathbf{B}$
end for
Visualisation (display) of the sampled images $\mathbf{B}[:, \mathcal{Y}]$

6 Experimental results and analysis

This section gives an idea about how the implementation and visualization are carried out followed by a detailed evaluation of the quantitative and qualitative results.

The software implementation was carried out using Pycharm version 3.8 on a single PC with CPU specifications as Intel i7-8665U CPU (1.90GHz, 2112 MHz with 4 Core(s), 8 Logical Processor(s)), along with 32 GB RAM and 64-bit operating system.

The Microsoft Common Objects in COntext (COCO) dataset [35] is used to evaluate the performance of the proposed sampling scheme. The details of this dataset are delivered in the next subsection.

6.1 Common Objects in COntext (COCO) dataset

Common Objects in COntext (COCO) [35, 36] is a large-scale object detection, segmentation and captioning dataset, widely used as a benchmark for many machine learning tasks. We chose the Detection 2017 dataset (validation fold) for evaluating the proposed data collection sampling. This dataset consists of 5,000 realistic images with different shapes and resolutions from diverse contexts with multiple annotations on each image. As it covers several disorganized scenes with various backgrounds, overlapping objects, etc., it is possible to train models on objects and people in realistic settings. Moreover, there are few data loaders and libraries for COCO already implemented in Python and PyTorch.

The 5000 pictures are annotated and classified into 80 categories grouped in 12 super categories. We chose the 12 supercategories (*i.e.*, person, animal, furniture, food, appliance, accessory, sports, electronic, indoor, outdoor, vehicle and kitchen) as our classes/categories while implementing the proposed sampling. We assume that the user's interest can be expressed as a preference score for each of these categories (see Sect. 4).

6.2 Evaluation strategy

In this section, we detail the adopted twofold evaluation strategy. The proposed work is a large-scale image collection sampling. The performance evaluation has to ensure the accuracy of the method by considering how much it could achieve with respect to each of the following criteria.

- How accurate is the user multi-preference modeling?
- How satisfying is the proposed sampling in terms of the following trade-off: fit with the user's preference (popularity) vs diversity among the sampled items (diversity)

6.3 User's preference modeling

We first evaluate the ability of the algorithm proposed in Sect. 4 to capture the user's preference. For that, we assume that a user explicitly gives its preferred categories. Thanks to our proposed algorithm, we transfer his preference into the latent space by building the matrix $\mathbf{U}$. The popularity of each item i can then be calculate as

$\mathbf{U}\mathbf{b}_i^\top\mathbf{b}_i\mathbf{U}$. Figure 3 shows the organization of the whole dataset and the associated supercategories. For a visual comparison of this estimated popularity and the true category, we use the t-Distributed Stochastic Neighbor Embedding (t-SNE) representation [37]. t-SNE is used to represent high-dimensional datasets or data points in reduced (*e.g.,* 2 or 3) dimensional space. It enables the interpretability of the data in the lower dimension. It is implemented by applying a nonlinear dimensionality reduction technique where the focus is on keeping very similar data points close together in lower-dimensional space. Thus, it can be adopted as a visualization method for high-dimensional datasets such as *coco* with 5000 images. Note that for images that contain more than one supercategory inside, the t-SNE plots a point corresponding to the first supercategory appearing in an image in the *coco* dataset. Figure 4 compares, in the t-SNE representation, the estimated popularity and the actual user's category for different examples of user's preferences. We can see, for example, that the calculated values of popularity undoubtedly emphasize the category *animal* (with high values to the features of category *animal*). In other words, the images belonging to the category *animal* are more popular. Similarly, 3 popular categories are demonstrated in row 2 of Fig. 4 when 3 preferences, i.e., *animal*, *furniture* and *food* are submitted.

In addition to this, the proposed user multi-preference model was tested by comparing the PI-based sampling driven by this proposed user model and a random sampling driven by an Oracle user model. In this oracle model, we assume that the true category is known and the popularity of the user-preferred category (for example, *animal*) is set to 1 and the popularity of the other categories is set to 0.1. From the comparison illustrated in Fig. 5, we can first see that the proposed sampling method indeed gives more importance to the preferred category. However, compared to the random sampling, the diversity in the dataset is preserved.

6.4 User-driven data collection sampling

We now evaluate how this user's preference can be effectively taken into account by the proposed sampler. We compare the proposed strategy with a simple random sampling. Indeed, we have previously demonstrated in [2] the efficiency of the proposed algorithm with respect to the baseline algorithm. We have noticed that the latter are well-performing for a low number of samples (i.e., when the number of samples is lower than the features space's dimension). For larger values of samples, they were equivalent to random sampling. The comparison with random sampling thus shows how the modeling of user preference can improve the quality of the sampling. The results of the proposed sampling and a comparison with random sampling are shown in Fig. 6.

Looking at the first two rows, which give the sampled results for single user preference *i.e animal*, RICS (in row 1) has very well acknowledged the user preferences compared to random sampling (row 2). The number of red points is higher for RICS than random sampling for the sampling size $M = 500$, 1000 and 1500 as illustrated in columns 1–3. Similar conclusions can be drawn for the case of multiple preferences.

We then use the PI metric introduced in [2] and also presented in Eq. (5) to evaluate the sampling quality. The higher the PI, the better the sampling scheme. The PI metric analysis of different image collection sampling methods for different categories of

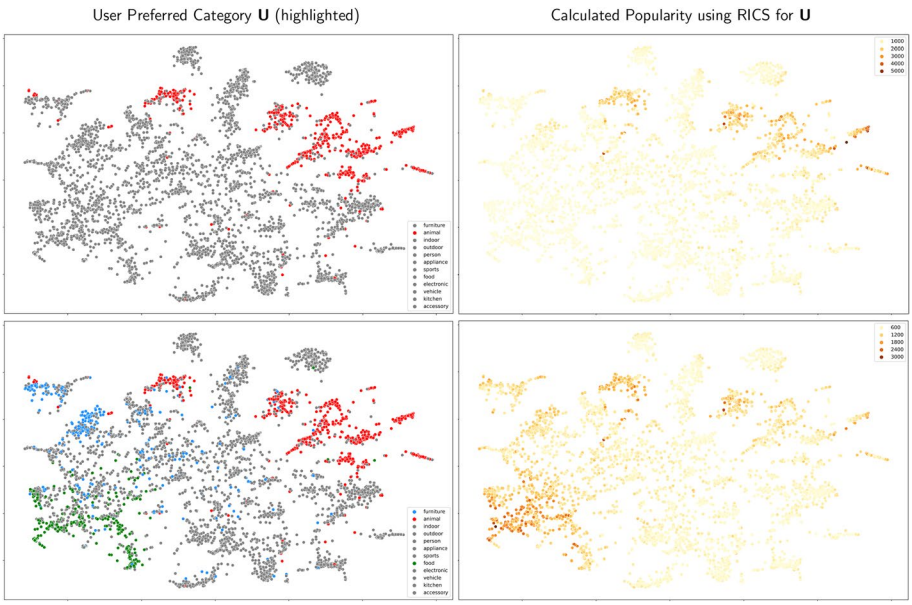

Fig. 4 Columns 1–2 represent, respectively, the user-preferred category in the coco dataset, the t-SNE visualization for the popularity values in accordance with the user preference. While row 1 illustrates single user preference, *i.e.*, category *animal* colored in red, Row 2 displays the popularity for 3 user preferences, *i.e.*, *animal*, *furniture* and *food*, respectively, using colors red, blue and green. The other categories that are of no interest to the user are colored in gray

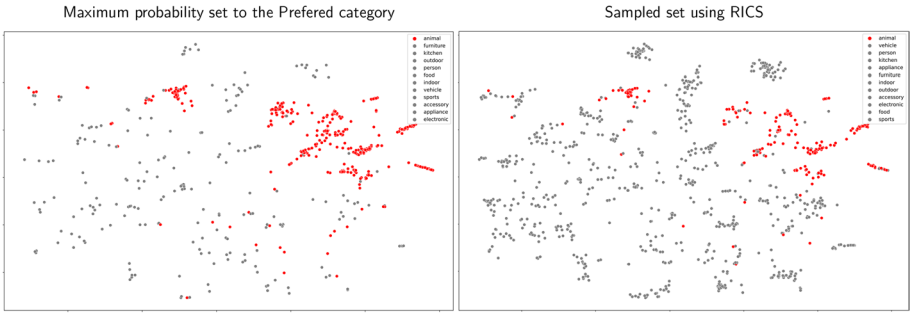

Fig. 5 Columns 1–2 represent, respectively, the t-SNE visualization for the user-preferred category, i.e., *animal* for maximum probability (oracle) user model driven random sampling and column 2 displays the sampled set with RICS with the proposed user preference model, both for a sample size of 1000 images. The other categories that are of no interest to the user are colored in gray

user preferences (using the coco dataset) is shown in Table 1. The PI values obtained while sampling with sample sizes 500, 1000 and 1500 for different user categories are shown. The baseline method used here to compare is the random sampling scheme. RICS due to its consideration for user preference has a higher performance outperforming the random sampling method. In addition, the PI values exhibit a drastic increment than random sampling method due to the right choice of the trade-off parameter λ. Both the methods uphold features diversity, while RICS wisely retains user-preferred samples making it more popular. The achieved sampled subsets with high PI are marked as bold in the table. Analyzing the observed outcomes in detail, it

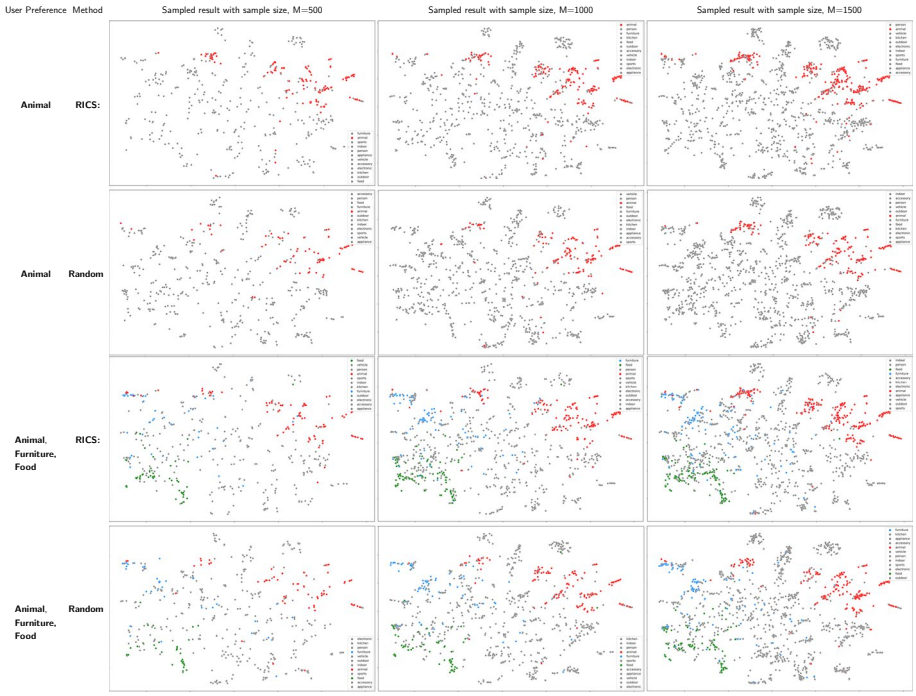

Fig. 6 Columns 1–5 represent, respectively, the user-preferred category in the coco dataset, the t-SNE visualization for all images embedded in the feature space, sampled results for sample size, M=500, sampled results for sample size, M=1000 and sampled results for sample size, M=1500 for the proposed method in row 1, 3 and 5 and for the random sampling method in rows 2, 4 and 5

Table 1 Analysis of different image collection sampling methods for different categories of user preferences (using coco dataset) (BEST: BOLD)

User Preference(s)	Method	PI -METRIC		
		M=500	M=1000	M=1500
Animal	RICS	**5516.11**	**7519.41**	**8321.48**
	Random	5341.94	7400.07	8247.13
Indoor	RICS	**6813.07**	**8857.62**	**9619.54**
	Random	6511.50	8608.57	9354.77
Furniture	RICS	**6485.41**	**8527.52**	**9266.70**
	Random	6141.91	8337.55	9063.85
Person	RICS	**6841.81**	**9057.34**	**9752.64**
	Random	6616.65	8917.81	9626.77
Animal, furniture	RICS	**6193.87**	**8367.34**	**9112.09**
	Random	5970.34	8256.54	9057.631

is implicit that RICS has a significant increment in the PI values and it outperforms random sampling by about 3%.

We finally show the quality of the sampled data set in Fig. 7. This can be regarded as a zoomed version of t-SNE. Here, we plot the 15 nearest neighbors of an arbitrary image (based on cosine similarity). In column 1 of Fig. 7, we take an image of the popular category *animal* with *coco image id* 46804 and searched for the 15 semantically similar images via KNN. The degree to which an image is similar to another semantically is

evaluated by cosine distance the top 15 semantically similar images just contained the category *animal* and there are no images from a different category when we look at the KNN results before sampling as illustrated in column 2 of row 1 of Fig. 7. The RICS sampled version of the KNN results for *animal* is illustrated in row 2 with the popular and non-popular categories for a sample size of 150. In column 1 of row 2, the user prefers the categories, *animal*, *food* and *vehicle*. Thus *animal* falls in the popular category (highly scored by the user). Among the 15 neighbors, 15 images contain the category *animal*. In addition, there are images that include the other preferred categories, *i.e., food* and *vehicle* in them as highlighted by the color labels. On the contrary, in column 1 of row 3, the neighbors of the random sampling scheme (which does not consider a user preference model) are illustrated. There are no neighbors that have the preferred images. Again, if we go for a non-popular image search, (*i.e* if the image is not preferred by the

Fig. 7 The figure illustrates the KNN visualization for 15 nearest neighbors of the query image shown, case 1: when the category *animal* is popular, i.e., the user prefers categories *animal*, *food* and *vehicle*, case 2: when the category *animal* is non-popular, i.e., when the user preferences are *accessory*, *outdoor* and *furniture*. Row 1 represents, respectively, the query image with the color labels, the KNN visualization for 15 neighbors of an image from the category *animal* before sampling, Row 2 shows the KNN of *animal* after RICS and Row 3 gives the KNN after random sampling for sample size 150 samples out of 5000 images

user), the 15 neighbors are illustrated in column 2 of rows 2 and 3 in Fig. 7. Here, the user prefers categories *accessory*, *outdoor* and *furniture*. We now search for the neighbors of the same query image with *coco image id* 46804 which belongs to category *animal*. The RICS sampling (row 2 of column 2) now gives 8 neighbor images that belong to the non-popular category *animal*, and it includes the user-preferred categories such as *accessory*, *outdoor* and *furniture* into the KNN. At the same time, random sampling (row 3 of column 2) is not efficient here to consider the user-preferred categories in the sampling scheme.

7 Conclusions and future scope

This work presented an extension of the PI metric-based scheme to large-scale image collection sampling by strengthening it with an image embedding and user multi-preference modeling. The novelty of this approach lies in the modeling of a mighty mite user multi-preference modeling scheme to ensure personalized sampling experience for each user in accordance with the user's preferences at that point of time. The work also realized an image embedding scheme and tested it with the *coco* dataset. The excellence of the work is demonstrated by both the quantitative and qualitative evaluation. The extensive t-SNE-based visual evaluation and KNN-based neighbors plots prove to be significant in visually establishing the superiority of the proposed work. In the future, this can be extended with an alternative sampling approach and extensions can be done to realize user models from a bigger description (for example, a paragraph) of what the user prefers.

Availability of data and materials

The datasets used and/or analyzed during the current study are public datasets available online.

Declarations

Competing interests

The authors have no competing interests to declare.

Received: 11 May 2023 Accepted: 11 October 2023
Published online: 18 October 2023

References

1. M. Chen, S. Mao, Y. Zhang, V.C.M. Leung, Big data: Related technologies, challenges and future prospects. (2014). https://api.semanticscholar.org/CorpusID:195649387
2. T. Maugey, L. Toni, Large database compression based on perceived information. IEEE Signal Process. Lett. **27**, 1735–1739 (2020)
3. Y. Saquil, D. Chen, Y. He, C. Li, Y.-L. Yang, Multiple pairwise ranking networks for personalized video summarization, *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 1718–1727
4. A. Sabha, A. Selwal, HAVS: human action-based video summarization, taxonomy, challenges, and future perspectives, in *2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)* (IEEE, 2021), pp. 1–9
5. J. Wu, S.-H. Zhong, Y. Liu, Dynamic graph convolutional network for multi-video summarization. Pattern Recognit. **107**, 107382 (2020)
6. Y. Li, B. Merialdo, Multi-video summarization based on video-MMR, in *11th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS 10* (2010, IEEE), pp. 1–4
7. E. Apostolidis, E. Adamantidou, A.I. Metsai, V. Mezaris, I. Patras, Video summarization using deep neural networks: a survey. Proc. IEEE **109**(11), 1838–1863 (2021)
8. E. Apostolidis, E. Adamantidou, A.I. Metsai, V. Mezaris, I. Patras, Unsupervised video summarization via attention-driven adversarial learning, in *International Conference on Multimedia Modeling* (Springer, 2020), pp. 492–504

9. Z. Li, Q. Tran, L. Mai, Z. Lin, A.L. Yuille, Context-aware group captioning via self-attention and contrastive features, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 3440–3450
10. P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, L. Zhang, Bottom-up and top-down attention for image captioning and visual question answering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 6077–6086
11. V. Sharma, A. Kumar, N. Agrawal, P. Singh, R. Kulshreshtha, Image summarization using topic modelling, *2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)* (IEEE, 2015), pp. 226–231
12. H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell. **27**(8), 1226–1238 (2005)
13. Y. Zhang, Y. Ma, X. Yang, Multi-label feature selection based on mutual information, in *2018 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)* (IEEE, 2018), pp. 1379–1386
14. S.M. Lajevardi, Z.M. Hussain, Feature selection for facial expression recognition based on optimization algorithm, in *2009 2nd International Workshop on Nonlinear Dynamics and Synchronization* (IEEE, 2009), pp. 182–185
15. L. Wang, S. Jiang, S. Jiang, A feature selection method via analysis of relevance, redundancy, and interaction. Expert Syst. Appl. **183**, 115365 (2021)
16. S. Saha, M. Ghosh, S. Ghosh, S. Sen, P.K. Singh, Z.W. Geem, R. Sarkar, Feature selection for facial emotion recognition using cosine similarity-based harmony search algorithm. Appl. Sci. **10**(8), 2816 (2020)
17. D. Kumar et al., Feature selection for face recognition using DCT-PCA and bat algorithm. Int. J. Inf. Technol. **9**(4), 411–423 (2017)
18. M. Iqbal, M.S.I. Sameem, N. Naqvi, S. Kanwal, Z. Ye, A deep learning approach for face recognition based on angularly discriminative features. Pattern Recognit. Lett. **128**, 414–419 (2019)
19. L. Zini, N. Noceti, G. Fusco, F. Odone, Structured multi-class feature selection with an application to face recognition. Pattern Recognit. Lett. **55**, 35–41 (2015)
20. A. Kulesza, B. Taskar et al., Determinantal point processes for machine learning. Found. Trends® Mach. Learn. **5**(2–3), 123–286 (2012)
21. L. Yu, V.O. Yazici, X. Liu, J.V.D. Weijer, Y. Cheng, A. Ramisa, Learning metrics from teachers: compact networks for image embedding, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 2907–2916
22. M. Berman, H. Jégou, A. Vedaldi, I. Kokkinos, M. Douze, Multigrain: a unified image embedding for classes and instances. arXiv preprint arXiv:1902.05509 (2019)
23. Z. Akata, F. Perronnin, Z. Harchaoui, C. Schmid, Label-embedding for image classification. IEEE Trans. Pattern Anal. Mach. Intell. **38**(7), 1425–1438 (2015)
24. D. Kiela, L. Bottou, Learning image embeddings using convolutional neural networks for improved multi-modal semantics, in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2014), pp. 36–45
25. M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 1717–1724
26. Z. Li, J. Tang, T. Mei, Deep collaborative embedding for social image understanding. IEEE Trans. Pattern Anal. Mach. Intell. **41**(9), 2070–2083 (2018)
27. V. Gupta, A. Saw, P. Nokhiz, P. Netrapalli, P. Rai, P. Talukdar, P-SIF: document embeddings using partition averaging, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol 34 (2020), pp. 7863–7870
28. [online:] Gianmario Spacagna: Extracting Rich Embedding Features from COCO Pictures Using PyTorch and ResNeXt-WSL
29. S. Liu, G. Tian, Y. Xu, A novel scene classification model combining ResNet based transfer learning and data augmentation with a filter. Neurocomputing **338**, 191–206 (2019)
30. Z. Wu, C. Shen, A. Van Den Hengel, Wider or deeper: revisiting the ResNet model for visual recognition. Pattern Recognit. **90**, 119–133 (2019)
31. A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in *Advances in Neural Information Processing Systems*, vol. 25 (2012)
32. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: a Large-Scale Hierarchical Image Database, in *CVPR09* (2009)
33. P. Lv, J. Fan, X. Nie, W. Dong, X. Jiang, B. Zhou, M. Xu, C. Xu, User-guided personalized image aesthetic assessment based on deep reinforcement learning. IEEE Trans. Multimed. (2021). https://doi.org/10.1109/TMM.2021.3130752
34. A. Kulesza, B. Taskar, k-DPPs: fixed-size determinantal point processes, in *ICML* (2011)
35. T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft coco: Common objects in context, in *European Conference on Computer Vision* (2014) pp. 740–755
36. [online:]: Coco: Common Objects in Context. https://cocodataset.org Accessed (2014)
37. L. Van der Maaten, G. Hinton, Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(11), 2579–2605 (2008)

Publisher's Note