Zero-Shot Content-Based Crossmodal Recommendation System

Expert Systems With Applications 258 (2024) 125108

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Zero-Shot Content-Based Crossmodal Recommendation System

Federico D'Asaro [a,b,*], Sara De Luca [a], Lorenzo Bongiovanni [a], Giuseppe Rizzo [a,b], Symeon Papadopoulos [c], Manos Schinas [c], Christos Koutlis [c]

[a] AI, Data and Space (ADS), LINKS Foundation, 10138, Turin, Italy
[b] Dipartimento di Automatica e Informatica (DAUIN), Politecnico di Torino, 10129, Turin, Italy
[c] Centre for Research and Technology Hellas (CERTH), 57001, Thessaloniki, Greece

ARTICLE INFO

Keywords:
Crossmodal recommendation
Crossmodal retrieval
Zero-shot recommendation
Multimodal recommendation
Content-based recommendation
Similarity search
Agglomerative clustering

ABSTRACT

Information Recommendation (IR) systems are conventionally designed to operate within a single modality at a time, such as Text2Text or Image2Image. However, the concept of cross-modality aims to facilitate a versatile recommendation experience across different modalities, such as Text2Image. In recent years, significant strides have been made in developing neural recommender models that are multimodal and capable of generalizing across a broad spectrum of domains in a zero-shot manner, thanks to the robust representation capabilities of neural networks. These architectures enable the generation of embeddings for assets (i.e., content uploaded on a platform by users), presenting a concise representation of their semantics and allowing for comparisons through similarity ranking. In this paper, we present ZCCR, a Zero-shot Content-based Crossmodal Recommendation System that leverages knowledge from large-scale pretrained Vision-Language Models (VLMs) such as CLIP and ALBEF to redefine the recommendation task as a zero-shot retrieval task, eliminating the need for labeled data or prior knowledge about the recommended content. Furthermore, ZCCR performs crossmodal similarity search on an optimized index, such as FAISS, to improve the speed of recommendation. The goal is to recommend to the user assets that are similar to those they have previously uploaded, commonly referred to as their user profile. Within the user profile, we identify "areas of interest"—groups of assets associated with specific user interests, such as cooking, sports, or cars. To identify these areas of interest and construct the search query for the retrieval operation, ZCCR employs an innovative use of Agglomerative Clustering. This technique groups user past assets by similarity without requiring prior knowledge of the number of clusters. Once the areas of interest, or clusters, are identified, the centroid is utilized as the search seed to find similar assets. Experimental results demonstrate the efficiency of the selected components in terms of search time and retrieval performance on modified MSCOCO and FLICKR30k datasets tailored for the recommendation task. Furthermore, ZCCR outperforms both a baseline tagging system (BT) and a more advanced tag system which utilizes a Large Language Model (LLM) to extract embeddings from tags. The results show that, even compared with the latter, embeddings directly extracted from raw assets yield superior outcomes compared to relying on intermediate tags generated by other tools. The code implementation to reproduce all experiments and results shown in this paper is provided at the following link: ZCCR-experiments.

1. Introduction

Images and videos constitute a massive source of data for indexing and search. Extensive metadata for this content is often not available and a variety of machine learning and deep learning algorithms are being used to interpret and classify these complex, real-world entities. Popular examples include text representation encoders such as word2vec (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013), representations of images by convolutional neural networks (Gong, Wang, Guo, & Lazebnik, 2014; Sharif Razavian, Azizpour, Sullivan, & Carlsson, 2014), and image descriptors for instance search (Gordo, Almazán, Revaud, & Larlus, 2016).

These representations, commonly known as embeddings, typically consist of real-valued, high-dimensional vectors ranging from 50 to over 1000 dimensions. This paper delves into multimodal deep learning within the context of recommendation systems, highlighting the effectiveness of multimodal embeddings in synthesizing semantic information from both images and texts. Multimodal deep learning refers to simultaneously considering and integrating data from various

* Corresponding author at: Dipartimento di Automatica e Informatica (DAUIN), Politecnico di Torino, Italy.
E-mail addresses: federico.dasaro@polito.it (F. D'Asaro), sara.deluca@linksfoundation.com (S. De Luca), lorenzo.bongiovanni@linksfoundation.com (L. Bongiovanni), giuseppe.rizzo@linksfoundation.com (G. Rizzo), papadop@iti.gr (S. Papadopoulos), manosetro@iti.gr (M. Schinas), ckoutlis@iti.gr (C. Koutlis).

https://doi.org/10.1016/j.eswa.2024.125108
Received 11 August 2023; Received in revised form 6 May 2024; Accepted 12 August 2024
Available online 24 August 2024

modalities, such as text and images, to enhance overall understanding and performance. In multimodal deep learning, an embedding denotes the transformation of varied data types, like text and images, into a unified, lower-dimensional vector space. This allows for the representation and understanding of inter-modal relationships, capturing nuanced semantic connections across different modalities.

In the context of Information Retrieval, the search task involves seeking and retrieving relevant information from a dataset. Thanks to the semantics encoded within these embeddings, the search task becomes independent of tags. According to the Pfeiffer report[1], tags are commonly linked with the following challenges: (1) It is impractical to anticipate all the ways individuals might search for something, considering factors like mood, colors, visual perspective, and purpose. Consequently, tagging fails to provide sufficient support for diverse search queries. This leads to cumbersome workflows in the search process when relying solely on keyword-based searches along with tagging tasks. (2) Maintaining discoverability through tagging is time-consuming due to the need for exhaustive content description, involving multiple tagging methods, each associated with an execution time. According to the Pfeiffer report, a multimodal search and recommendation system offers three notable improvements for digital asset management: (a) Companies eliminate the need to spend time tagging for making assets discoverable. (b) Users can now employ descriptive language for searches, precisely locating what they seek. (c) There is an increase in asset usage, as users easily find existing assets, reducing the likelihood of recreating content or settling for subpar assets.

1.1. The proposed system

In this context, we introduce ZCCR, a Zero-Shot Content-Based Crossmodal Recommendation System. ZCCR leverages assets previously uploaded by the target user to recommend new assets, framing the recommendation task as a crossmodal retrieval problem. ZCCR employs a pre-trained Vision-Language Model (VLM), such as CLIP (Radford et al., 2021), to generate multimodal embeddings for images and texts, eliminating the need for tagging systems. It provides recommendations to the target user in a purely zero-shot fashion, requiring no additional training.

In the context of social media platforms, an "asset" is defined as an individual item posted by a user, playing a pivotal role in delineating the user's distinct areas of interest. An "area of interest" is characterized as a collection of homogeneous assets posted by the user, revolving around specific concepts such as animals, cars, cooking, or travel. We designate the term "user profile" to encompass the set of assets that have been previously uploaded by the user. Following the extraction of embeddings from these assets through a pre-trained VLM, we apply Agglomerative Clustering to systematically group together the assets within the user profile, relying on their inherent similarities. Agglomerative Clustering is a hierarchical unsupervised machine learning method that starts with individual data points as separate clusters and progressively merges them based on similarity until a termination condition is met. All assets within the same cluster are presumed to belong to the same area of interest for the user, while assets in different clusters are considered to belong to different areas of interest.

In a content-based recommendation system, the user query is autonomously derived from the user's prior uploads, eliminating the need for direct input. The user query refers to the summarized representation of the user's areas of interest, constructed by condensing identified cluster information into what we now define as a "seed" – a representative of the cluster. This seed operates as a search query, facilitating the recommendation of new assets aligning with the target user's preferences. After identifying the seeds, the recommendation task transforms into a crossmodal retrieval, seeking similar assets relative to each seed. As each identified cluster represents an area of interest, the number of seeds corresponds to the identified clusters, with each seed querying the database for new assets.

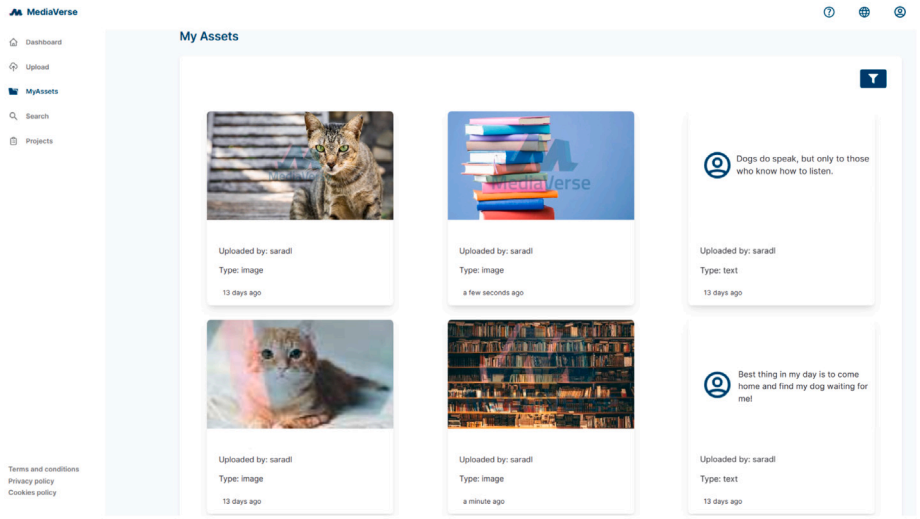
1.2. Modules of ZCCR

ZCCR comprises three primary modules: (1) A large-scale VLM functioning as a multimodal encoder, which includes two parallel encoders for processing images and texts. (2) A FAISS similarity search index (Johnson, Douze, & Jégou, 2019) enhances the efficiency of the retrieval process in terms of recommendation time. In the domain of Multimodal Learning, CLIP (Radford et al., 2021) and ALBEF (Li, Selvaraju et al., 2021) stand out as prominent VLMs pretrained on large-scale image-text corpora, excelling in zero-shot crossmodal retrieval on benchmark datasets such as MSCOCO and FLICKR30k. However, a notable observation is that when these architectures encode image and text features from these datasets, the embeddings from different modalities do not align effectively. Specifically, unimodal pairs like text-text and image-image consistently yield representations associated with higher similarity values than multimodal pairs like image-text, irrespective of their semantic content. This results in an undesired outcome where features tend to cluster in the feature space based on their modality rather than purely on semantic relevance, indicating a lack of modality-invariance. We define the "Search Space" as the collection of retrievable assets during the search operation, and we refer to "noise" as the presence of assets within the Search Space lacking semantic relevance concerning the search query. We argue that the misalignment of multimodal embeddings, referred to here as "Modality Gap" negatively impacts crossmodal retrieval performance. This is particularly evident in scenarios where noise is introduced by irrelevant data of the same modality as the query. This is because they are consistently associated with a higher similarity score than data of different modalities. Consequently, we decide to instantiate as many search indexes as there are modalities—in our case, one index for texts and one for images. This approach ensures that unimodal (e.g., Text2Text, Image2Image) and crossmodal (e.g., Text2Image, Image2Text) search tasks are performed by querying two distinct indexes. This avoids the mixture of embeddings from different modalities in the same search index while still enabling both unimodal and crossmodal retrieval. In unimodal retrieval, we search among assets of the same modality as the query, while in crossmodal retrieval, we search among assets of a different modality compared to the query. We term this operation "decoupling" the unimodal and crossmodal retrieval tasks, as they are accomplished by querying different search indexes. (3) Agglomerative Clustering acts as the semantic grouping mechanism for previously uploaded user assets without requiring the pre-specification of the number of clusters. This is essential as the number of areas of interest, derived from the user's past uploaded assets, is not formally defined and must be discerned from the arrangement of embeddings in the semantic space. Following the identification of user asset clusters, the cluster centroids act as seeds for retrieving similar texts and images.

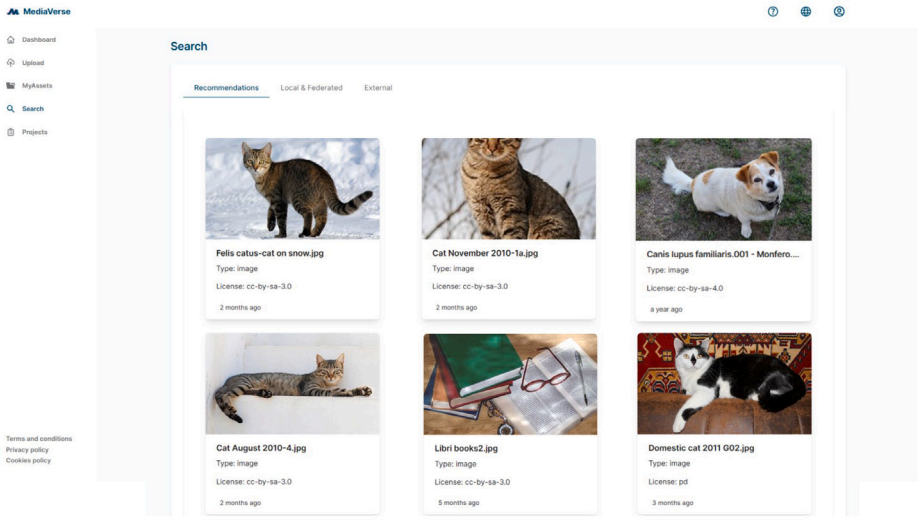
1.3. The MediaVerse use case

ZCCR is developed in the scope of MediaVerse (MV) a H2020-ICT-2020-1 3-years project entitled "A universe of media assets and co-creation opportunities at your fingertips". MV is an answer to the blurred boundaries between professional media houses, prosumers and small creators. It is a decentralized network for intelligent, automated, and accessible digital asset management systems, where traditional stakeholders and other media owners can share, enrich, verify, and monetize multimedia content. Since the speed of communication and publishing is increasing, audiences are seeking more user-driven and accessible multimedia experiences. A semantics-based multimodal

[1] https://www.pfeifferreport.com/wp-content/uploads/2018/11/Creativity_and_AI_Report_INT.pdf?ref=kailua-labs.ghost.io.



(a) User Profile.



(b) Search Space.

Fig. 1. In this example scenario within the MediaVerse Platform, ZCCR recommends images to the user based on their areas of interest (cats, dogs, books) by conducting both Image2Image and Text2Image searches.

Search and Recommendation tool is important for a multimedia content platform populated by a plurality of users.

In Fig. 1, we present an illustrative use case of ZCCR within a prototype version of the MediaVerse platform, which may be subject to modifications compared to the version depicted here. Fig. 1(a) showcases the user profile when it involves assets related to cats, dogs, and books. Specifically, cat and book assets are uploaded in image format, while textual posts related to dogs have been published. Fig. 1(b) displays the assets recommended to the user in (a) based on their interests. Images of dogs, cats, and books are recommended, even though no dog images were initially present in the user's profile. In this case, ZCCR conducted Image2Image searches for cats and books and crossmodal Text2Image searches for dogs.

1.4. Contributions

We investigate both text and image modalities, which are the focus of interest in the MediaVerse project. However, our ZCCR can be expanded to encompass any combination of modalities, as long as there are encoders capable of converting those modalities into embeddings.

Our main contributions are summarized below:

1. We present ZCCR, a Zero-shot Content-Based Crossmodal Recommendation system that transforms the recommendation task into a retrieval task without the need of training. It is crafted as a plug-and-play solution, seamlessly integrating without dependence on domain-specific data, while fully leveraging the rich knowledge encoded in pretrained VLMs. ZCCR combines two prominent research directions: the use of pretrained multimodal architectures into recommendation systems and the creation of a zero-shot recommender.
2. ZCCR leverages Agglomerative Clustering to generate user queries, termed as seeds, that mirror the user's areas of interest. The integration of Agglomerative Clustering strengthens ZCCR's zero-shot capabilities, and to the best of our knowledge, we are the first to use clustering to generate user query in the realm of content-based recommendation.
3. ZCCR addresses the issue of Modality Gap associated with VLMs by decoupling the unimodal and crossmodal retrieval tasks. This is achieved by utilizing distinct search indexes for each of the modalities involved.
4. ZCCR outperforms a baseline tag-based recommender system by efficiently clustering embeddings from multimodal encoders

based on user areas of interest. This highlights the superior semantic information extraction of ZCCR's multimodal encoder, allowing the clustering algorithm to generate homogeneous clusters reflecting the user's areas of interest.

The paper is structured as follows: Section 2 provides an introduction to the context. Section 3 outlines our ZCCR recommendation pipeline, detailing individual components and explaining the rationale behind the decision to instantiate separate search indexes for each modality. Section 4 presents the setup of experiments conducted to justify design choices and compare the performance of ZCCR versus the Tag system. Finally, Section 5 offers comprehensive results for our approach, showcasing concrete improvements over the vanilla tagging approach.

2. Related work

2.1. Content-based recommender systems

Content-based recommender systems analyze a set of documents and descriptions (or only either document or description) of items previously rated by a user, and build a model, also called profile, of user areas of interest based on asset embeddings rated by that user (Ko, Lee, Park, & Choi, 2022). The profile is a structured representation of user areas of interest adopted to recommend interesting new items. The fundamental assumption of the content-based methods is that an item is recommended to a user if that user has liked a particular item with characteristics similar to the recommended item. The result of the recommendation task is the level of relevance that a new asset has for the considered user. In our case, we do not exploit the user interaction with other assets (user ratings), we instead rely on their previously uploaded assets assuming that they form a basis on user areas of interest.

Content-based recommender systems were mainly used in various application areas, such as recommendations according to the properties of movies (Ali, Nayak, Lenka, & Barik, 2018), e-commerce recommendations (De, Banerjee, Rath, Swain, & Samant, 2022). According to Ko et al. (2022) Content-Based Filtering is the simplest recommendation model. In the early 2000s, there were many studies using it to present recommendations to users, but due to its disadvantage of recommending only biased items, the number of studies using this model alone has gradually decreased since 2010. However, it is still being studied and utilized continuously in the fields of books and news, which are application fields centered on text information.

Nonetheless, we argue that a purely content-based strategy, employing pretrained VLM as feature encoder and a Hierarchical clustering technique to formulate the user query, is remarkably efficient in addressing the Mediaverse Context. This methodology allows us to create a zero-shot recommendation system without relying on item data, user data, or interaction data.

Using a VLM as a semantic extractor is motivated by the challenge in content-based recommenders, where textual features often face difficulties due to natural language ambiguity. In De Gemmis, Lops, Musto, Narducci, and Semeraro (2015), researchers explore semantic representations to overcome limitations of keyword-based approaches. The survey categorizes semantic methods into top-down (integration of external knowledge) and bottom-up (using lightweight representations). Emphasizing the importance of semantic incorporation in recommender systems for advancing content-based recommendations, ZCCR adopts a bottom-up approach with a pretrained VLM.

Building upon the achievements in multimodal representation learning for crossmodal retrieval (Feng, Wang, & Li, 2014; Peng, Huang, & Qi, 2016; Wang, He, Wang, Wang, & Tan, 2013), this paper employs two extensive pre-trained image-text encoders. The first, CLIP (Contrastive Language–Image Pre-training) (Radford et al., 2021), stands as a state-of-the-art dual-encoder network pre-trained on a dataset comprising 400 million (image, text) pairs collected from various publicly available Internet platforms. CLIP utilizes Contrastive Loss (Sohn, 2016), incorporating paired text and image embeddings as inputs during training. The second encoder, ALign BEfore Fuse (ALBEF) (Li, Selvaraju et al., 2021), adopts a similar methodology to CLIP, independently encoding images and text using a detector-free image encoder and a text encoder. Both architectures undergo pretraining to uncover projections of data items from diverse modalities into a shared, semantics-based feature representation subspace. Within this subspace, a direct assessment of similarity between different modalities becomes achievable.

In the absence of user-specific information, we enhance the utilization of data derived from the user's previously uploaded content to formulate what we now identify as the "seed" for the user query. This is achieved through the application of agglomerative clustering on representations of individual user content. Hence, during the recommendation phase, a particular user is linked to a quantity of queries/seeds equal to the number of semantic concepts expressed in the content previously posted.

Let us proceed with referencing two pivotal aspects within the domain of ZCCR: the application of clustering techniques and the deployment of a pre-trained multimodal architecture within the realm of zero-shot recommendation systems.

2.2. Clustering in recommender systems

Clustering is a technique mainly used to identify user groups similar to users in the Collaborative Filtering model. When clustering is used for Content-Based Filtering, it is mainly used when clustering and analyzing the similarity of location-based data in the recommendation system of the travel field (Dietz, Sen, Roy, & Wörndl, 2020). Clustering techniques have been utilized to enhance the efficacy of recommendation systems.

The work by Berbague et al. (2021) leverages clustering to augment recommendation diversity, a recognized issue in recommendation systems. Addressing this concern has been shown to result in heightened user satisfaction. The authors of Zhang, Lin, Lin, and Liu (2016) proposed a new collaborative filtering algorithm based on clustering user preferences to reduce the impact of data sparsity. Collaborative filtering is widely used by online vendors and review sites to recommend items based on the ratings of many users. However, this method has several problems, and one of them is the presence of attacks aimed at distorting the predicted ratings of specific elements. Zhang (2019) proposed a collaborative filtering technique that reduces the impact of attacks while maintaining or improving prediction accuracy by repeatedly applying clustering to target data and predicting ratings for unrated items within each cluster. In this way the recommendation reliability is improved. Jiang, Zhang, Jiang, Wang, and Pei (2019) present a method of joint filtering based on biclustering and information entropy to eliminate the effect of historical sparseness in user ratings.

As evident, the majority of clustering techniques are employed in collaborative recommendation strategies with the objectives of improving reliability, enhancing diversity, grouping users according to preferences, or mitigating the challenge of data sparsity within the user-item interaction matrix. In this article, we focus on content-based recommenders and utilize hierarchical Agglomerative clustering within the ZCCR to generate user queries, referred to as 'seeds'. Instead of employing more complex deep learning clustering techniques, we chose Agglomerative clustering for its simplicity and lack of additional training requirements (Aljalbout, Golkov, Siddiqui, Strobel, & Cremers, 2018), enabling the ZCCR to provide zero-shot recommendations.

According to Assent (2012), many clustering algorithms face efficiency challenges in higher-dimensional spaces due to the inherent sparsity of data. In such applications, it is common for points to be far apart in at least a few dimensions. One approach is to project points from higher-dimensional to lower-dimensional spaces, assuming that

data can be reasonably approximated with only a small number of dimensions retained. This method, often using techniques like Principal Component Analysis (PCA) (Jain & Dubes, 1988) or Singular Value Decomposition (SVD) (Strang, 2012), can effectively reduce noise and enhance data analysis.

2.3. Large multimodal recommendation models

Inspired by the achievements of Large Language Models (LLMs), the recommender system community began focusing on enhancing the generalization capability and transferability of recommendation models (Li, Zhang, Liu and Chen, 2023). Fueled by the success of Large Language Models (LLMs), some models, like Li, Zhang and Chen (2021, 2023) leverage LLM knowledge for enhanced recommendation interpretability. Others, such as Geng, Liu, Fu, Ge and Zhang (2022) and Xu, Hua, and Zhang (2023) address multiple recommendation tasks concurrently including direct recommendation, sequential recommendation, and explanation generation. With the increasing volume of multimedia content, the research focus has shifted to multimodal recommendation. The common practice involves utilizing multimodal content as supplementary information to aid recommendation decisions (He & McAuley, 2016b; Meng, Feng, He, Gao, & Chua, 2020) or incorporating visual data sources to provide visual explanations to users (Geng, Fu et al., 2022; Hou et al., 2019). Multimedia recommendation systems, which consider extensive multimedia content information for items, have proven successful in various applications like e-commerce, instant video platforms, and social media platforms (Cui, Yu, Wu, Liu, & Wang, 2021; He & McAuley, 2016a; Veit et al., 2015). Taking a step beyond the current trend of pretrained LLMs and the use of multimodal models involves utilizing Vision-Language Models (VLMs) like CLIP. Examples include VIP5 (Geng, Tan, Liu, Fu, & Zhang, 2023), which integrates the CLIP image branch to encode visual data, and e-CLIP (Shin et al., 2022), addressing e-commerce domain-specific data by training a CLIP architecture from scratch using contrastive product image and text keywords. Another instance is MICRO (Zhang et al., 2022) that proposes graph structure learning to uncover latent item relationships based on multimodal features. While these multimodal solutions prove highly effective in their respective domains and tasks, they typically require a training step and the availability of data regarding users, items, and their interactions. A partial exception among these is P5 (Geng, Liu et al., 2022), achieving domain transferability by training the architecture on an auxiliary domain to solve tasks on target domains, where users are known to P5 but the items have never been encountered by the model. In this paper, we aim to achieve top-k ranking of items by integrating CLIP multimodal encoder within a procedure that requires no training. Our pure content-based approach, combined with a foundational clustering technique like Agglomerative Clustering, demonstrates the effectiveness of using multimodal features rather than solely textual ones, thereby contributing to the advancement of research in multimodal recommendation systems.

2.4. Zero-shot recommendation

The performance of recommender systems heavily relies on available training data, yet there are instances, known as zero-shot cases, where historical records are limited. Success in handling such startup cases indicates a commendable generalization ability of recommendation models. One extensively explored challenge in this context is cold-start recommendation, wherein either users or items are new to the system with no prior interaction records. Solutions to this issue either involve learning to model content features (Li et al., 2019; Shi et al., 2019), enabling inference without interaction records, or learning to transfer representations from auxiliary domains (Man, Shen, Jin, & Cheng, 2017; Yuan et al., 2021; Zhu et al., 2021). In our specific case, neither the users nor the items have been encountered by the ZCCR before. Additionally, the ZCCR does not require any training, making it a fitting plug-in solution for both pure content-based and hybrid recommenders.

As evident, the most innovative recommendation systems delve into two emerging research branches: Large Multimodal Recommendation Models and Zero-Shot Recommendation. In this article, we introduce ZCCR, a Zero-Shot Content-Based Crossmodal Recommendation system that leverages the knowledge of large-scale pretrained VLMs to extract multimodal embeddings from assets present on the MediaVerse social media platform. ZCCR operates without the need for any training data, positioning itself as a pure zero-shot recommender. The latter is enhanced by the innovative use of Agglomerative Clustering to generate the user query for retrieving assets most similar to those uploaded by the target user.

3. Method

Our ZCCR leverages asset embeddings of user u profile P_u, generated by multimodal neural encoders, to form clusters that align with the user's areas of interest (e.g., animals, cars, sports). It employs Agglomerative Clustering to generate these clusters. Subsequently, it utilizes the centroids of these clusters as search seeds to identify multimodal assets similar to those previously uploaded by user u. We note that the cluster centroid effectively represents the entire cluster information. ZCCR separates the search operation into unimodal search (txt2txt, img2img) and crossmodal search (txt2img, img2txt) by creating multiple similarity search indexes, one for each modality. We leverage Facebook AI Similarity Search (FAISS) (Johnson et al., 2019) as an index to store text and image encoded embeddings, as it significantly accelerates search times to achieve optimal performance levels.

The ZCCR pipeline is depicted in Fig. 2. The recommendation process begins with the user u's previously uploaded items (the user profile P_u), arranged by modality.

1. Given a user u and its profile P_u consisting of N images $X = [x_1, x_2, \dots, x_N]$ and M texts $Y = [y_1, y_2, \dots, y_M]$, the image encoder E_I and the text encoder E_T project them respectively into a shared embedding space:

$$z_{I,i} = E_I(x_i), \quad z_{I,i} \in \mathbb{R}^D \tag{1}$$

$$z_{T,j} = E_T(y_j), \quad z_{T,j} \in \mathbb{R}^D \tag{2}$$

Here, $Z_I = [z_{I,1}, z_{I,2}, \dots, z_{I,N}]$ and $Z_T = [z_{T,1}, z_{T,2}, \dots, z_{T,M}]$ are real-valued multi-dimensional embeddings of the image and text in the shared D-dimensional embedding space. E_I and E_T are the multimodal encoders that originate from pre-trained CLIP or ALBEF.

2. Separately for each modality, asset embeddings are projected into a lower-dimensional space using PCA:

$$p_{I,i} = \text{PCA}(z_{I,i}), \quad p_{I,i} \in \mathbb{R}^{D'} \tag{3}$$

$$p_{T,j} = \text{PCA}(z_{T,j}), \quad p_{T,j} \in \mathbb{R}^{D'} \tag{4}$$

where $p_{I,i}$, $p_{T,j}$ are the linearly projected image and text features belonging to an embedding space of dimension $D' < D$. Then $P_I = [p_{I,1}, p_{I,2}, \dots, p_{I,N}]$ and $P_T = [p_{T,1}, p_{T,2}, \dots, p_{T,M}]$ are separately clustered through the Agglomerative Clustering technique. Let $C_I = \{C_{I,1}, C_{I,2}, \dots, C_{I,K}\}$ be the clusters formed from images projections P_I, and $C_T = \{C_{T,1}, C_{T,2}, \dots, C_{T,L}\}$ the clusters formed from texts projections P_T, these are obtained:

$$C_I = \text{AgglomerativeClustering}(P_I) \tag{5}$$

$$C_T = \text{AgglomerativeClustering}(P_T) \tag{6}$$

Referring back to full representations Z_I and Z_T, each point is assigned to a cluster such that:

$$C_{I,k} = \{z_{I,i} | z_{I,i} \text{ belongs to cluster } k\} \tag{7}$$

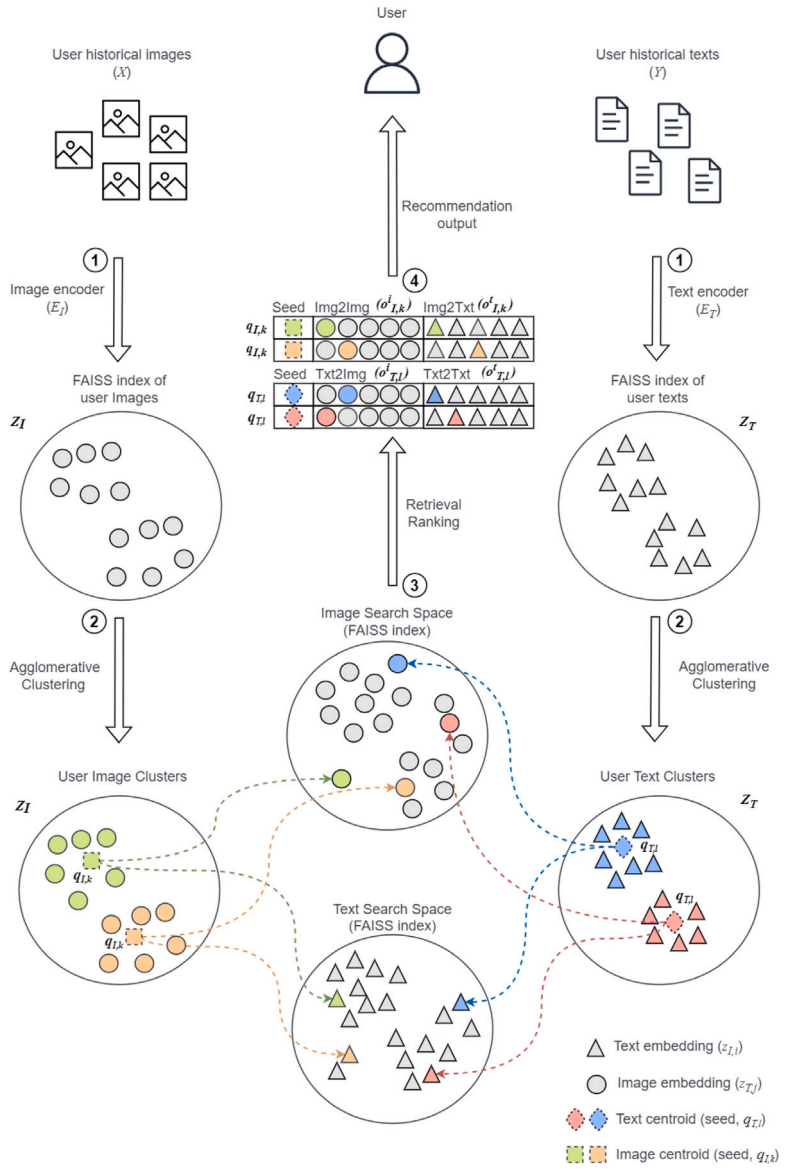


Fig. 2. ZCCR's recommendation process consists of two parallel pipelines initiated with the previously uploaded texts and images of user u. These texts and images are encoded by language and vision encoders, respectively, and then clustered using Agglomerative Clustering. In this scenario, we assume that the user has uploaded assets from two clusters (each representing an area of interest) for both texts and images. The centroid of each cluster is used as a search seed to perform unimodal and crossmodal retrieval on the text and image pools. Finally, for each seed, a ranked top-k list of assets is returned to the user u.

$$C_{T,l} = \{z_{T,j} | z_{T,j} \text{ belongs to cluster } l\} \tag{8}$$

In this way, each user asset is associated with a label indicating the cluster it belongs to. Considering that the embedding space of each modality is semantically consistent, assets of the same cluster are similar to each other, creating the so-called areas of interests. Once each asset has a cluster label, we refer back to the full multi-dimensional embeddings Z_I, Z_T.

3. To condense cluster $C_{I,k}$ and $C_{T,l}$ information into a singular vector, we calculate their centroids and utilize them as a search seed to query both text and image databases. The query for each cluster is given by:

$$q_{I,k} = \frac{1}{|C_{I,k}|} \sum_{z_{I,i} \in C_{I,k}} z_{I,i} \tag{9}$$

$$q_{T,l} = \frac{1}{|C_{T,l}|} \sum_{z_{T,j} \in C_{T,l}} z_{T,j} \tag{10}$$

The goal is to identify the pertinent assets associated with the user's initial clusters, which summarize their interests.

4. For each seed $q_{I,k}$, $q_{T,l}$ (each representative of an area of interest), we retrieve the top-k assets for both unimodal and crossmodal searches. Specifically, a single seed $q_{I,k}$ queries both the Image Search Space and the Text Search Space separately, resulting in two top-k assets, namely $o^t_{I,k} = [y_1, y_2, \ldots, y_k]$ and $o^i_{I,k} = [x_1, x_2, \ldots, x_k]$ respectively associated to recommended texts and images associated to the seed $q_{I,k}$ (area of interest emerging from image assets). The same goes for $q_{T,l}$ which results in two top-k assets, namely $o^t_{T,l} = [y_1, y_2, \ldots, y_k]$ and $o^i_{T,l} = [x_1, x_2, \ldots, x_k]$ respectively associated to recommended texts and images associated to the seed $q_{T,l}$ (area of interest emerging from text assets).

Since image and text features (Z_I, Z_T) are stored in two separate search indexes, the entire recommendation process involves two parallel pipelines—one for images and one for texts. This results in the separation of unimodal and crossmodal searches, as each seed $q_{I,k}$ and $q_{T,l}$ performs queries on both the image database and the text database. For instance, a seed generated from images ($q_{I,k}$), previously uploaded

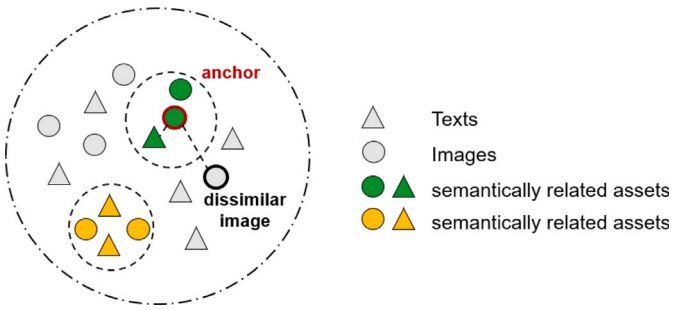


Fig. 3. Desired multimodal embeddings arrangement in the common feature space. Similar assets that share common semantics are close to each other, regardless of their modality.

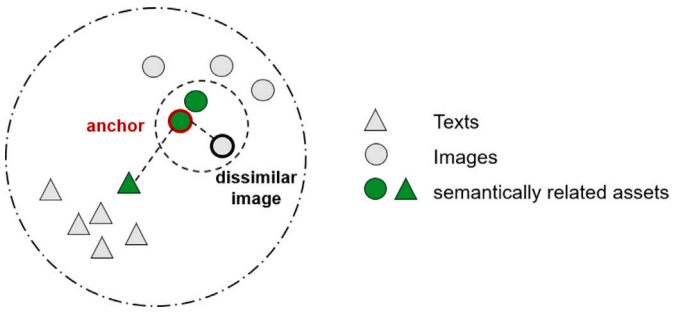


Fig. 4. Undesired multimodal embeddings arrangement in the common feature space. Similar assets of different modalities that share common semantics are further apart than dissimilar ones of the same modality.

by user u, is used to conduct unimodal search among other images and crossmodal search of texts by querying their respective indexes. The decision to avoid instantiating a single index (a single database) containing embeddings from all modalities (images and texts) stems from the misalignment in embedding similarities between unimodal and multimodal pairs, defined as Modality Gap (see Section 3.1.3).

In the following of this section, we further describe ZCCR implementation details.

3.1. Multimodal encoder

We adopt a dual-encoder (text and image) approach as (Radford et al., 2021) in parallel pre-trained to feed a contrastive loss (Sohn, 2016). We assess CLIP and ALBEF as alternative image-text encoders due to their representation of the cutting edge in Visual Language Models (VLMs). However, another VLM alternative can be seamlessly integrated into the ZCCR pipeline. Here, we provide an outline of the specifications of the pre-trained architectures utilized.

3.1.1. CLIP

As regards the text encoder, it is a Transformer (Vaswani et al., 2017). As a base size we use a 63M-parameter 12-layer 512-wide model with 8 attention heads. The transformer operates on a lower-cased byte pair encoding (BPE) representation of the text with a 49,152-vocab size. For computational efficiency, the max sequence length was capped at 76. For the visual encoder we opted for ViT-B/32 (Dosovitskiy et al., 2021), the ViT base variant with 32 × 32 input patch size. The final text and image representations are 512-d embeddings, $z_I, z_T \in \mathbb{R}^{512}$, where z_I, z_T are text and image features respectively.

3.1.2. ALBEF

The Text Encoder is a 6-layer transformer initialized using the first 6 layers of the BERTbase (Devlin, Chang, Lee, & Toutanova, 2018) (123.7M parameters). The Vision Encoder is a 12-layer visual transformer ViT-B/16 (85.8M parameters). The final text and image representations are 256-d embeddings, $z_I, z_T \in \mathbb{R}^{256}$.

3.1.3. Modality gap

Features originating from different modalities often exhibit inconsistent distribution and representation, leading to a gap that needs to be addressed. Models like CLIP and ALBEF aim to discover (i.e., learn) projections of data items from various modalities into a shared embedding space, allowing for direct assessment of similarity between them. As depicted in Fig. 3, the desired representation space is expected to be primarily organized based on asset semantics rather than being arranged in a modality-aware fashion, such as modality-specific clusters shown in Fig. 4.

However, recent studies (Liang, Zhang, Kwon, Yeung, & Zou, 2022; Shi, Welle, Björkman, & Kragic, 2023) have highlighted a misalignment in embeddings derived from Vision-Language Models (VLMs) such as CLIP and ALBEF, often referred to as "Modality Gap". This misalignment indicates that image and text embeddings tend to concentrate in distinct subregions within the full embedding space, as quantified by the difference between the centroids of image and text embeddings:

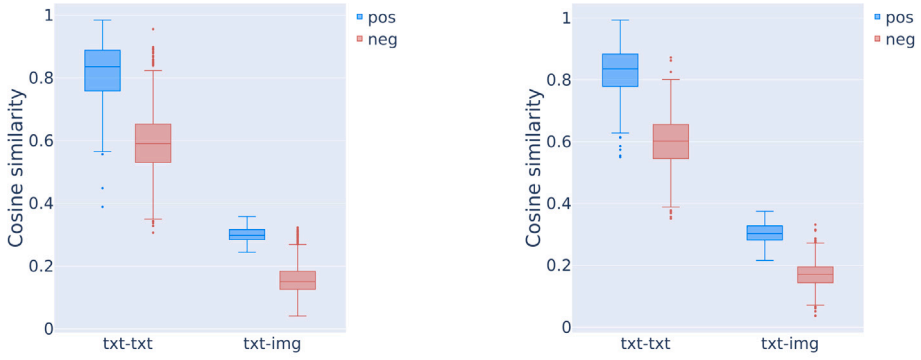
$$\Delta_{\text{gap}} = \frac{1}{n}\sum_{i=1}^{n} z_{I,i} - \frac{1}{n}\sum_{i=1}^{n} z_{T,i} \tag{11}$$

Here, z_I and z_T denote the L2-normalized image and text embeddings, respectively. The cause of this misalignment remains an ongoing research question, with initial findings suggesting that it may stem from the dual-encoder architecture of VLMs, making them sensitive to random weight initialization and the adopted contrastive learning procedure. While a comprehensive investigation into the causes and potential solutions to this phenomenon exceeds the scope of this article, several factors warrant exploration: (1) the level of detail in the obtained embeddings, where finer-grained encodings at the word or image patch level may yield more aligned embeddings, (2) the complexity and size of the training data distribution, and (3) the potential benefits of utilizing a VLM with a single shared encoder among modalities to mitigate the problem of modality gap.

Before discussing our approach to tackle modality gap, we further examine this phenomenon by analyzing cosine similarity distribution on MSCOCO and FLICKR30k datasets.

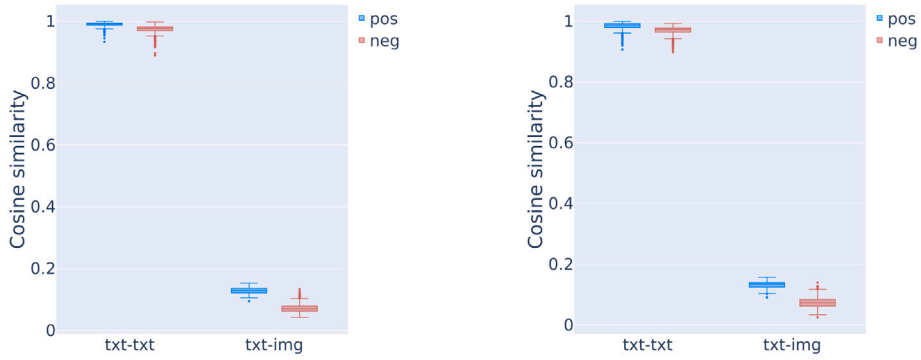
To evaluate the challenge of modality gap, we randomly sample 1000 text-image pairs from the validation sets of MSCOCO and FLICKR30k datasets. Subsequently, we compute the cosine similarity matrix among their features. A pair is considered positive when both elements share a common semantics (similar pair), and negative when they do not (dissimilar pair). In this matrix, the diagonal entries represent the similarity values between positive image-text pairs, while the off-diagonal entries correspond to values between negative pairs. Additionally, we calculate the similarity for both positive and negative text-text pairs. The cosine similarity for negative text-text pairs is determined by extracting the upper-diagonal elements from the similarity matrix constructed using only the textual features of text-image pairs. Conversely, for positive text-text pairs, we leverage the presence of multiple captions associated with the same image. As these captions share a common semantic context, extracting the upper-diagonal values from their similarity matrix provides us with the cosine similarity values for similar text-text pairs.

We observe that representations of unimodal positive text-text pairs are consistently more aligned than multimodal positive image-text pairs. Starting from CLIP, Fig. 5 indicates that both positive and negative multimodal pairs are associated with lower cosine similarities compared to unimodal ones. More significantly, even unimodal negative text-text pairs exhibit higher similarity scores than positive image-text pairs. Specifically, positive image-text pairs display cosine similarities around 0.3, in contrast to both positive text-text pairs (around 0.8)



(a) MSCOCO. (b) FLICKR30k.

Fig. 5. Cosine similarity distribution of **CLIP** image and text encoders applied to MSCOCO (a) and FLICKR30k (b) image-text pairs. x axis specifies if it is an unimodal pair of texts or a text-image multimodal one. Blue boxes show positive pairs, i.e. cosine similarity computed between similar assets. While the red boxes indicate negative pairs, i.e. pairs that do not share the same semantics. In both datasets, we notice a severe misalignment between unimodal and multimodal pairs.



(a) MSCOCO. (b) FLICKR30k.

Fig. 6. Cosine similarity distribution of **ALBEF** image and text encoders applied to MSCOCO (a) and FLICKR30k (b) image-text pairs. As for MSCOCO in Fig. 5, between unimodal and multimodal pairs we notice a severe misalignment that generates problems in the retrieval phase when all modalities reside in a common Search Space.

and negative text-text pairs (around 0.5). This discrepancy hinders crossmodal retrieval when irrelevant data from the same modality as the query is introduced into the Search Space. Similar observations hold for ALBEF representations (Fig. 6), where cosine similarity is more compactly distributed. These distributions suggest that embeddings are arranged in a modality-aware fashion that compromises semantic relationships, as depicted in Fig. 4.

Another evidence of the modality-aware arrangement of embeddings in the space is given by the bi-dimensional t-Distributed Stochastic Neighbor Embedding (t-SNE) plot of text and image features coming from MSCOCO and FLICKR30k. Fig. 7 shows that CLIP features are grouped by modality rather than only with respect to semantics.

In the development of ZCCR, we have devised an implementation strategy that circumvents the problem of modality gap by decoupling the FAISS indexes associated with images and texts. This way, we avoid both embedding modalities residing in the same Search Space because otherwise, assets of the same modality as the query would introduce noise in the crossmodal search for assets of a different modality (effectively making the crossmodal retrieval task impossible).

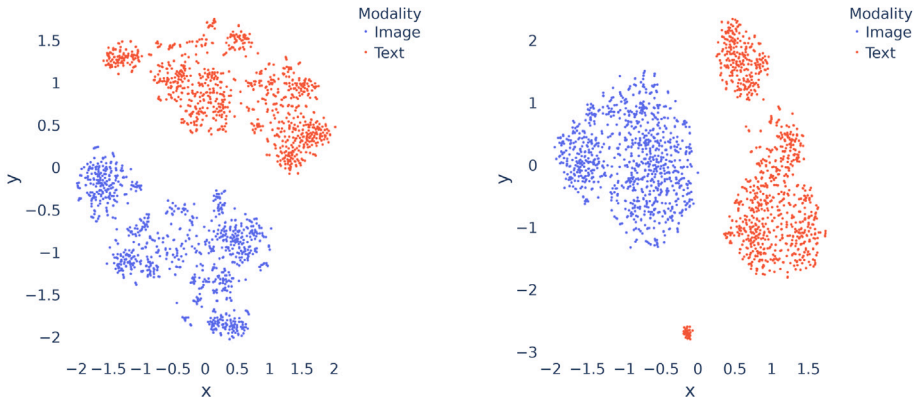
Hence, each index is filled with assets of the same modality. This ensures the generation of multiple top-k rankings, each specific to a particular modality, resulting in distinct queries for unimodal or crossmodal retrieval. Fig. 8 illustrates the multi-index structure in the general case with M modalities. The use of separate indexes guarantees that the retrieval process remains unaffected by noise from unimodal assets (with respect to the input query) that may have higher similarity scores, as each index node generates its own top-k ranking.

To better highlight how the proposed strategy of decoupling indexes circumvents the problem of modality gap, we compare the crossmodal retrieval performance in two different retrieval scenarios, here defined as *Unimodal Search Space* and *Multimodal Search Space*.

We define crossmodal retrieval tasks as follows: given an input query of one modality m_q (e.g., text), we retrieve relevant (semantically related) assets of a different modality m_r, where $m_r \neq m_q$ (e.g., $image \neq text$). This retrieval occurs within the Search Space of assets, as defined in Section 1. To emphasize the effect of modality gap on crossmodal retrieval from a single Search index, we distinguish between two cases (Fig. 9) based on the assets populating the *Search Space*:

- *Unimodal Search Space Retrieval*: The *Search Space* comprises data from a single modality (m_r), distinct from the query modality (m_q). Specifically, for the MSCOCO 1k validation split, the Search Space consists of 1000 texts for Text Retrieval and 1000 images for Image Retrieval.
- *Multimodal Search Space Retrieval*: In this setting, the Search Space includes data from all modalities (text and images). This configuration is designed to simulate the presence of noise introduced by data of the same modality as the query within the Search Space. In the case of the MSCOCO 1k val split, it contains 500 texts with 500 images for both Text and Image Retrieval.

Tables 3 and 4 in Section 5.1 display Recall@k (k = 1,5,10) values in the two Search Spaces for both Image Retrieval (Txt2Img) and Text Retrieval (Img2Txt) tasks. The results clearly indicate that, due to the modality gap, performing crossmodal retrieval in the Multimodal



(a) MSCOCO. (b) FLICKR30k.

Fig. 7. T-SNE vector arrangement on the plane of **CLIP** image and text embeddings taken from MSCOCO (a) and FLICKR30k (b) image-text pairs.

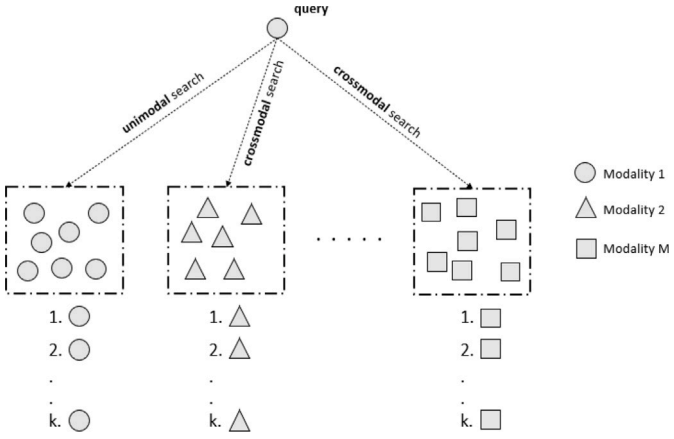


Fig. 8. In the retrieval process employing the multi-index structure, the input query serves as a seed to search for similar assets in each modality-specific index. For each of these indexes, the top-k similar assets are then retrieved.

Search Space is not viable. This highlights the infeasibility of relying on a single FAISS index instance that contains assets from various modalities, preventing the search for assets of a different modality than the query. This rationale justifies the choice to create multiple FAISS indexes, each tailored to a specific modality, and each functioning within the Unimodal Search Space scenario.

3.2. Agglomerative clustering

To distinguish among different areas of interest, we apply Agglomerative Clustering on the user u previously uploaded items. Specifically, as shown in Fig. 10, we apply PCA to linearly reduce the input embedding dimension. We take PCA's minimum number of components that retain 99% of total variance. The projected embeddings (P_I, P_T) are then grouped according to the Agglomerative Clustering algorithm. We adopt Ward's linkage due to its effectiveness in dealing with datasets where clusters have different sizes (i.e., a varying number of assets over different areas of interest of the user) and its lower sensitivity to noise and outliers compared to other linkage methods such as single-linkage or complete-linkage. Points belonging to the same cluster share a common semantics, i.e. the same area of interest. Being each item assigned with a cluster label, we go back to the full dimensional space (Z_I, Z_T) and compute the centroid of embeddings sharing a common label ($q_{I,k}$, $q_{T,l}$). Each centroid is then used as search seed to query text and image databases.

In Section 5.2, we conducted an ablation study of the different components that can be part of the Agglomerative Clustering module. Among these, we assessed whether the retrieval performance of the seed varies when generated from the centroid or medoid of the associated cluster. Medoids emerge as an appropriate substitute for centroids, especially when dealing with non-convex clusters. Unlike centroids, which hinge on the average of data points, medoids denote the data point that minimizes the sum of dissimilarities to all other points within the cluster. This difference underscores a significant drawback of centroids, as they are prone to problems like sensitivity to outliers and difficulties in managing non-convex shapes. According to our experiments, there are no systematic differences between the two. Due to the simplicity and linear time complexity of centroid calculations, as opposed to the potentially higher computational cost associated with medoid calculations, we chose to use the centroid as the cluster seed for executing the search query.

A challenge faced by many clustering algorithms is efficiency in higher-dimensional spaces due to the inherent sparsity of data (Assent, 2012). When dealing with assets uploaded by a user, ZCCR generates 512-dimensional embeddings from the pretrained VLM, which faces difficulties in being appropriately clustered. To address this issue, we choose to project points from higher-dimensional to lower-dimensional space using the PCA technique. In Section 5.2, we demonstrate how our strategy of first linearly projecting the embeddings with PCA and then clustering them using Agglomerative Clustering leads to better recommendation performance compared to using the original 512-dimensional embeddings to generate clusters (without projection). Additionally, specifically in the context of ZCCR, we inquire whether the number of user-associated areas of interest and, consequently, the number of clusters that the Agglomerative algorithm must identify, along with the number of assets associated with each area (i.e., how many assets the user has previously uploaded in that area of interest, the cardinality), impact performance. In Section 5.2, we showcase the robustness of Agglomerative Clustering to variations in both the number and cardinality of areas of interest, highlighting its resilience to fluctuations in user preference scenarios. Through a comparative analysis with another clustering algorithm, such as HDBSCAN (McInnes, Healy, & Astels, 2017), we provide evidence of the superior performance of Agglomerative Clustering.

3.3. Search index

We choose the Facebook AI Similarity Search (FAISS) index (Johnson et al., 2019) for its specialized optimizations tailored for high-dimensional embedding searches. FAISS accelerates search times through GPU support, optionally employing quantization and compression techniques. It leverages parallel processing and multi-threading,

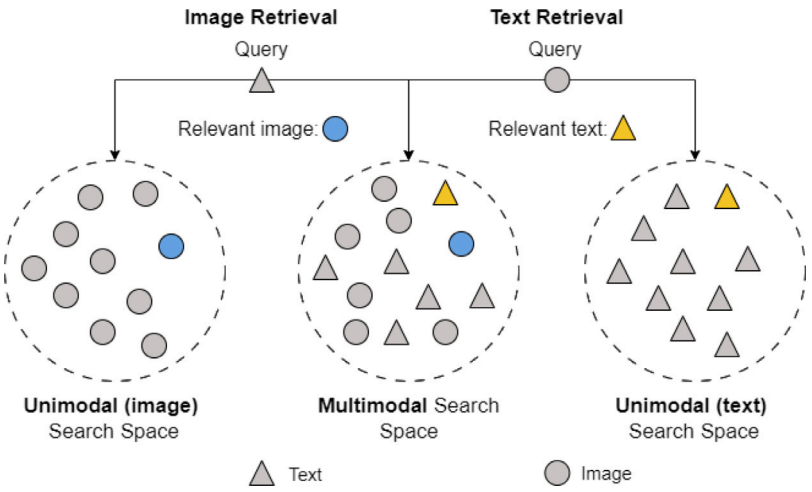


Fig. 9. Image and Text Retrieval are conducted in both Unimodal and Multimodal Search Spaces. In *Image Retrieval*, a text query (triangle) is used to search in both the Unimodal Search Space, which contains only images (circle), and the Multimodal Search Space, which includes both images and texts. The goal of this search query is to identify the relevant paired image (blue). Similarly, for the *Text Retrieval* task, a query image (circle) is used to search in both the Unimodal Search Space, which contains only texts (triangle), and the Multimodal Search Space, which includes both texts and images. The aim is to retrieve the relevant paired text (yellow).

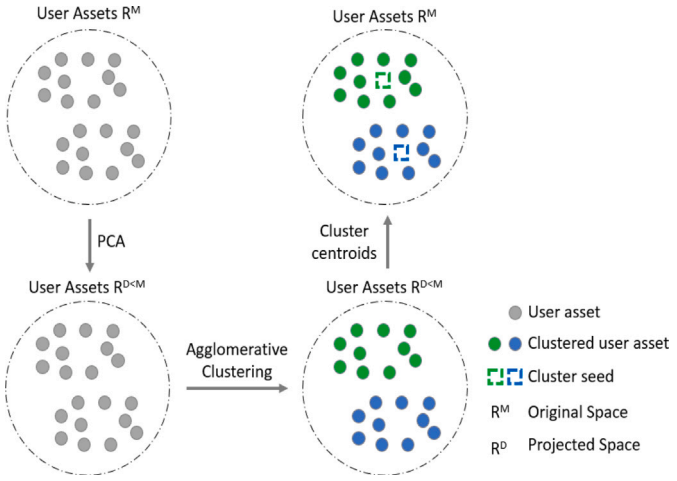


Fig. 10. Pipeline of the Agglomerative Clustering of user assets.

making it adaptable to modern hardware architectures. FAISS is versatile, supporting various similarity measures, and its scalability and efficiency in handling large datasets make it a popular choice for improving search times in machine learning applications. We utilize the FAISS index without quantization or compression, relying on cosine similarity for embedding comparisons. This choice prioritizes simplicity and information preservation, as the approach with raw embeddings is flexible and interpretable, avoiding potential information loss associated with compression. However, it underscores the importance of considering efficiency in computational aspects.

In Table 6, we compare the search time performance of the FAISS index with a vanilla linear search. In this vanilla scenario, raw embeddings are stored without additional optimizations. The FAISS index, designed for high-dimensional vector searches, demonstrates improved efficiency in similarity searches compared to the simple storage of raw embeddings while still maintaining comparable performance in terms of recall (refer to Table 7). In the case of the CLIP encoder, we achieve a significant speedup in search time, reducing it from 129 ms with the linear vanilla approach to 0.72 ms using the FAISS index. This improvement is observed with only a minor decrease in recall, approximately 0.2% for image retrieval and 0.14% for text retrieval.

Due to the observed modality gap (refer to Section 3.1.3), we create a separate index for each modality involved. In our scenario, dealing with text and image modalities, we instantiate two indexes (see Fig. 11). All asset embeddings from all users are stored within these two indexes. Since FAISS does not store any embedding metadata, we maintain an additional container data structure that associates FAISS indexes with the embeddings of the assets uploaded by each user. This allows us to select only the embeddings of the user targeted by the recommendation and use embeddings from other users to build the Search Space.

In practice, when generating a top-k recommendation for user u, we select $k + N_u$ assets, where N_u is the number of assets uploaded by user u, and then filter out those belonging to user u. This approach ensures that we avoid suggesting assets to user u that they have generated themselves. Instead, we recommend assets from the same areas of interest as user u but created by other users.

The problem of modality gap led us to instantiate two FAISS indexes. However, we are concerned about the impact this may have on retrieval performance in terms of search times compared to the scenario where only one FAISS index is instantiated. In Table 1, we present the mean search time and standard deviation (std dev) in milliseconds required to retrieve k = 1000 assets in a Search Space of size 1 000 000. We computed these values using a GPU NVIDIA GeForce RTX 2080 Ti. For each setting, we calculated search times over different numbers of FAISS indexes (1,2) by distributing the size of the Search Space across the various indexes. Specifically, if there are 2 indexes, each is populated with 500k assets. The search times are obtained by averaging over 1000 queries.

It is observed that in both scenarios, the search time increases with the number of indexes, suggesting that a single global index with all assets from all modalities, constituting a unified Search Space, may be a better solution in terms of retrieval time. The mean values of the two cases were compared using a two-sample t-test to determine if there is a significant difference between the means or if they occurred by chance. We obtain a p-value of approximately 0. This suggests that the pursuit of modality-invariance also has an impact on the efficiency of retrieval times, and we encourage further exploration in this direction.

In summary, ZCCR stands out as an innovative zero-shot content-based crossmodal recommendation system, leveraging knowledge from large-scale pretrained VLM to create multimodal embeddings for images and texts. Positioned at the forefront of large multimodal recommendation systems, ZCCR also explores the innovative domain of Zero-Shot Recommenders. This is enhanced by employing Agglomerative Clustering to formulate the user query. This innovative approach, incorporating clustering within a recommendation system, casts the

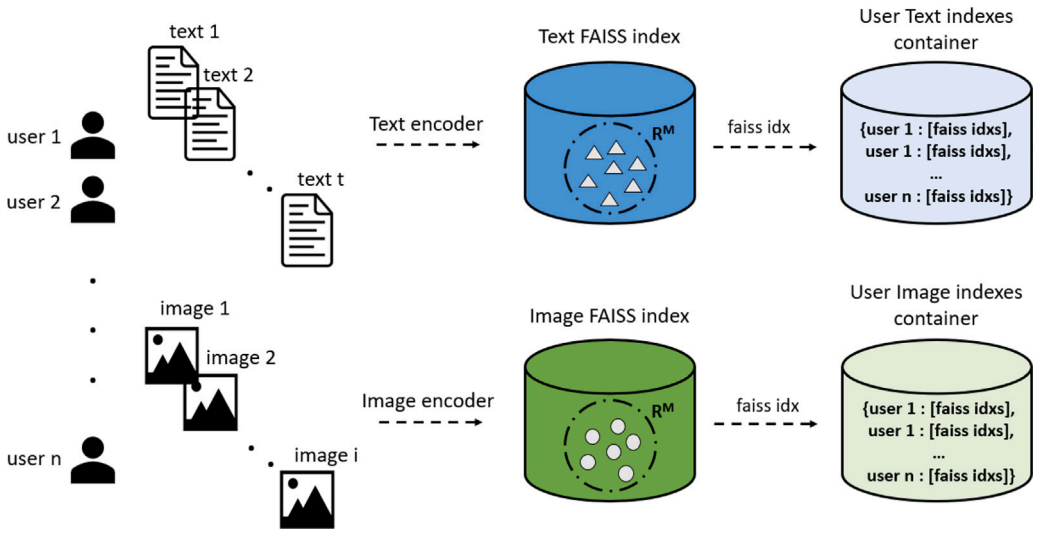


Fig. 11. The FAISS index structure consists of separate search indexes for texts and images. Additionally, for each index, we maintain an associated container that maps each user to the assets they uploaded.

Table 1
Mean and standard deviation of search time over 1000 queries for different numbers of FAISS indexes (1, 2). The low p-value confirms that, with an equal total Search Space size, two indexes are associated with higher search times; hence, a single index instance is more efficient than two instances.

n. FAISS indexes	Emb. size	k	Search Space size	Mean time (ms)	Std Dev	p-value
1	512	1000	1 000 000	13.811	0.340	≈ 0
2	512	1000	1 000 000	14.490	0.433	

typical recommendation task into a retrieval task. It recommends new assets to the user based on their previously uploaded assets. As far as we know, we are the pioneers in employing a clustering technique for user query generation. The combination of a large-scale pretrained architecture and Agglomerative Clustering for query generation empowers ZCCR to operate without the need for training procedures or domain-specific data, addressing potential gaps in certain contexts. This adaptability positions ZCCR as a plug-and-play solution suitable for integration into more intricate systems employing collaborative logic. Finally, the utilization of the FAISS index for storing embeddings and accelerating the similarity search of new assets has been adjusted following the observation of the modality gap phenomenon. In the multimodal context, ZCCR proposes instantiating two separate FAISS indexes—one for images and another for texts. This ensures that the retrieval task is not hindered by noise generated by assets of the same modality as the query.

ZCCR also faces challenges typical of content-based recommendation systems, including the need for diversification in recommendation output to offer users serendipity and the avoidance of redundancy to prevent information overload from assets in the same area of interest. As discussed in Section 2.2, clustering has been employed to enhance asset diversity and can be implemented in ZCCR at the output level for this purpose. Regarding the prevention of redundancy, in ZCCR, it heavily relies on the number of recommended assets originating from a specific seed related to a user's area of interest. Decreasing this number (k) and thus limiting recommendations to a short top-k ranking can help alleviate the redundancy of semantically related assets.

4. Experimental setup

In this section, we detail the datasets employed and outline the experimental setups necessary for conducting the subsequent studies, including the rationale behind the choice of ZCCR clustering components and the comparison of ZCCR's performance with the Baseline Tag system.

4.1. Dataset

The experimental setups described in the following sections are based on MSCOCO validation split (Lin et al., 2014) (40,504 images) and FLICKR30k (Young, Lai, Hodosh, & Hockenmaier, 2014) (30,000 images and 150K descriptive captions). Since, in MediaVerse, we are unable to conduct any quantitative tests due to the absence of users, we rely not only on MSCOCO but also on a retrieval benchmark dataset like FLICKR30k. It is a combination of rich multimedia assets, user-generated images, and annotations, where each user has different interests and produces quality assets. This diversity in user preferences and asset themes makes it an ideal platform to assess recommendation tools that need to cater to a broad spectrum of user areas of interest and asset types. It is a publicly available dataset, making it a valuable resource for evaluating recommendation tools designed for social media platforms.

To evaluate ZCCR, we need to simulate a scenario where users have uploaded assets coming from specific areas of interest. From all the assets associated with a detected area of interest, a seed is extracted and used to perform retrieval operations over other assets uploaded by different users. To achieve this, we initiate the evaluation process using the validation splits of MSCOCO and FLICKR30k. These two datasets consist of images, each of which is paired with five descriptive captions. They do not allow us to track the semantic relationship between text-to-text and image-to-image pairs, but only between image-to-text pairs. For this reason, we design a setup in which we assign a common semantic label to each image-text pair that allows us to pair texts that share a common semantics (same for images). In practice, for each image we sample one of the five associated caption, to obtain the same number of images and texts. Then, we label each image with ResNet50 (He, Zhang, Ren, & Sun, 2016), pretrained on ImageNet (Deng et al., 2009) and able to classify an image according to one of 1000 classes. This way, each image-text pair is associated with a single label. This means that the text shares the same label as its paired image, creating an "image, text, label" triplet structure. Now, we can group together all the images and texts that share the same label, indicating that they have the same high-level semantics defined as the area of interest in the context of ZCCR. To enhance the semantic relevance among assets of the same area of interest, we retain only the pairs whose images are associated

with a classification confidence higher than 90%. In this manner, we obtain two datasets with the triplet ⟨text, image, ImageNetClass⟩ that we designate as **MSCOCO classified** and **FLICKR30k classified**.

In 4.3, we define the performance evaluation of ZCCR as the extent to which the unsupervised clusters identified by the Agglomerative Clustering technique align with the nominal areas of interest of a given user, which correspond to the assigned ImageNet label. Quantitatively, this is assessed by computing retrieval Recall starting from the seed generated from the identified clusters.

4.2. ZCCR retrieval task

We evaluate crossmodal retrieval using Recall@k, where k denotes the number of retrieved assets. For the Txt2Img (Img2Txt) task, we take a randomly sampled image-text pair, using the text (image) component as a query to retrieve the image (text) component in a Search Space of 1000 retrievable images (texts). We calculate averages over 100 queries.

4.3. ZCCR recommendation task

In the construction of the experimental setup for ZCCR, the first step involves creating what we refer to as a user u profile. This profile comprises a collection of assets associated with the user, specifically those assets previously uploaded by the user. The profile is generated by randomly selecting N assets from M randomly chosen ImageNet classes within the datasets detailed in 4.1 (MSCOCO classified and FLICKR30k classified). The parameter M signifies the number of distinct areas of interest for user u, which are derived from the uploaded assets. Once the user profile is established, we employ ZCCR to cluster the asset embeddings based on their semantics, uncovering areas of interest for the user u. It is crucial to emphasize that the clusters identified by ZCCR rely solely on the semantics of the assets and lack information about the ImageNet classes from which the assets were originally sourced, representing the ground truth. A well-performing clustering procedure is expected to yield clusters predominantly composed of assets from a single ImageNet class (area of interest), with only a few assets from different areas of interest, as illustrated in Fig. 12. After the clusters are identified, their centroid (or medoid for comparison) is calculated and used as a seed, serving as a search query to retrieve assets belonging to the same area of interest. These assets are assigned to the same ImageNet class but were uploaded by other users and are present in the Search Space. The Search Space is constructed by randomly selecting 1000 assets from the datasets outlined in 4.1. Among these, we incorporate a single relevant asset for each of the M ImageNet classes utilized in the earlier step to construct the user u profile. The objective is for ZCCR to be capable of recommending, from the Search Space, the asset associated with each of the M classes that constitute the user u profile.

We evaluate the performance of ZCCR in relation to Recall@10 across Txt2Img, Img2Txt, Txt2Txt, and Img2Img, considering various configurations influenced by the factors outlined in Table 2: (1) The number of ImageNet classes used to sample assets for the user u profile (number of ImageNet classes; 1, 2, 5), indicating the number of areas of interest the user u has at recommendation time. (2) The number of assets drawn for each ImageNet class (number of points per ImageNet class; 5, 10, 20, 30), representing the quantity of assets associated with an area of interest. We expect in some cases the low number of embeddings per class label to be a problem in discriminating from different clusters. (3) If the cluster centroid or medoid is used as search seed for recommendation. (4) Which projection techniques is aplied before Clustering: No projection applied before clustering (None), linear projection of the embeddings (PCA), and non-linear projection (TSNE). (5) The choice of Hierarchical Clustering technique: Agglomerative Clustering or HDBSCAN.

Table 2
ZCCR configurations are specified, and the components crucial for clustering, denoted in bold, serve as ablative components. The selection of these components is contingent upon the variation in Recall@10 values obtained, corresponding to changes in both the number of ImageNet classes and the number of points per ImageNet class.

Name	Values
Number of ImageNet classes	1,2,5
Number of points per ImageNet class	5,10,20,30
Projection technique	None, PCA, TSNE
Clustering algorithm	Agglomerative clustering, HDBSCAN
Representative cluster point	Centroid, Medoid

For each configuration, we calculate the Recall@10 values by averaging across 100 queries. These queries are generated by initially randomly selecting ImageNet classes and subsequently sampling assets for each class based on the specified number of assets per class in the configuration.

Fig. 12 illustrates the configuration for recommendations when user u has previously uploaded assets associated with three areas of interest (three ImageNet classes). The ZCCR algorithm organizes the points coherently based on their semantics and generates a seed for each cluster to query the Search Space. Within our setup, a single relevant point in the Search Space corresponds to each ImageNet class. Our goal is to have the output of ZCCR include the three relevant assets associated with the three ImageNet classes used to construct the user u profile. The successful retrieval of these three relevant assets indicates the effectiveness of the Clustering algorithm in grouping points according to their ImageNet class. The Recall@10 for the user u recommendation is determined by averaging the partial Recall@10 values obtained from the queries, each associated with a seed. To address the possibility of multiple queries retrieving the same relevant asset (as user u has as many relevant assets as the ImageNet classes assigned during profile creation), we maintain a list of already retrieved relevant assets within the scope of the user u recommendation. This approach ensures that even if the same relevant asset is retrieved from multiple seeds, it is counted only once for Recall@10 purposes. Without this consideration, the same relevant asset could be counted multiple times for the same recommendation.

For a clearer visualization of the diverse settings we assess, Fig. 13 illustrates some scenarios based on the number of ImageNet classes and the number of points per ImageNet class.

4.4. Baseline tagger

In comparing ZCCR with traditional tag systems, we make use of annotation models developed within the context of the MediaVerse project to enrich both images and texts with descriptive tags. This comprehensive approach encompasses models designed for image captioning, object detection, and action recognition. The image captioning model, utilizing the advanced OFA (Wang et al., 2022), generates descriptive text for each image, providing an avenue to enhance the retrieval of visual content. In terms of object detection, we employ the Yolov8 model by ultralytics [2] to identify objects within images, presenting confidence scores and bounding box information. For action recognition, our choice is the SlowFast R50 model (Feichtenhofer, Fan, Malik, & He, 2019), which has been trained on the Kinetics400 dataset (Carreira, Noland, Hillier, & Zisserman, 2019).

Concerning the text associated with each image, we utilize Part of Speech (POS) tagging and Lemmatization to preserve only the root form of nouns and verbs within the caption. The generated list of these words functions as the tags associated with the text, forming the basis for comparison with the visual tags. Both POS tagging and Lemmatization are applied to the visual caption generated by the image

[2] https://github.com/ultralytics/ultralytics.

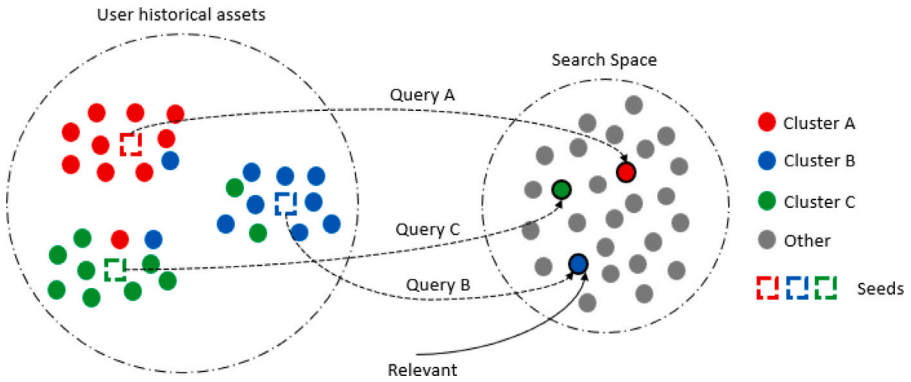


Fig. 12. An illustration of a recommendation scenario begins with user u having a profile comprising images associated with three distinct areas of interest. The accompanying figure depicts clusters identified by ZCCR's clustering module. It is important to note that the labels assigned by the clustering algorithm may not perfectly align with the ImageNet classes assigned to the points; rather, they are influenced by the capabilities of the clustering algorithm. A seed is constructed from all points within the same cluster to encapsulate cluster information. This seed is then used as a query to retrieve similar assets from the Search Space.

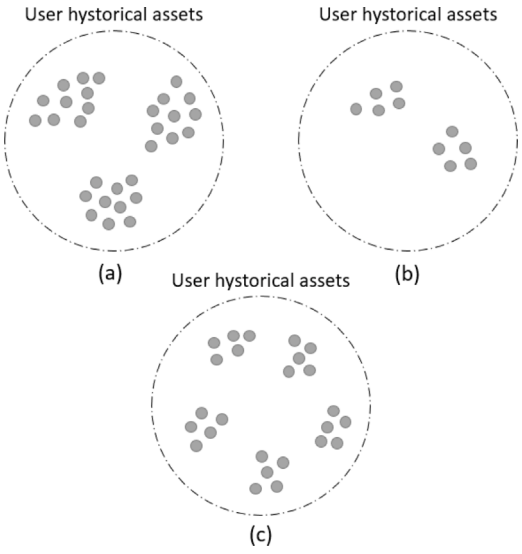


Fig. 13. Some scenarios featuring different numbers of ImageNet classes and points per ImageNet class. In (a), the user generated 10 assets distributed across three areas of interest (3 ImageNet classes). In (b), two ImageNet classes are represented, each comprising 5 assets. Finally, in (c), there are 5 classes, each containing 5 assets.

captioning model. However, for the text output of object detection and action recognition, we exclusively apply Lemmatization. To ensure comparability between the outcomes of the tag system and our ZCCR, we calculate Recall@10 across various configurations. These configurations depend on the number of ImageNet classes (indicative of the areas of interest linked to user u's previously uploaded assets) and the number of assets per ImageNet class. This assessment spans the tasks of Txt2Txt, Txt2Img, Img2Txt, and Img2Img 5.3.

For each configuration, Recall@10 values are computed by averaging across 100 queries. These queries are generated by first randomly selecting ImageNet classes and then sampling assets for each class based on the specified number of assets per class in the configuration.

We evaluate two versions of the tagger:

1. Baseline Tagger (BT): In this version, assets are evaluated through an exact match of tags. Specifically, among all the tags associated with the assets in user u's profile, we keep the 33% most frequently occurring ones. These selected tags serve as search keys to retrieve the top 10 assets with the highest number of matches in a Search Space comprising 1000 assets. The limitation of this strategy lies in the heterogeneity of the tags originating from multiple assets, which can belong to different areas of interest.
2. Baseline Tagger + BERT embeddings (Devlin et al., 2018) + Agglomerative Clustering (BTBA): In this version, each asset, whether text or image, is linked to embeddings obtained by encoding the tags using BERT. The comprehensive asset embedding is derived by averaging its tag embeddings. Subsequently, following the same procedure as ZCCR described in Section 3, the assets of user u are clustered utilizing their PCA projections, and the resulting centroids are employed as search queries to recommend similar assets in the Search Space. This Search Space is constructed similarly to ZCCR but with tag embeddings.

5. Results

5.1. Retrieval results

Tables 3 and 4 illustrate the Recall@k for crossmodal retrieval of CLIP and ALBEF on both the MSCOCO and FLICKR30k 1k validation sets under the scenarios of Unimodal Search Space and Multimodal Search Space as defined in Section 3.1.3. Across both datasets, ALBEF demonstrates superior performance to CLIP in Txt2Img and Img2Txt tasks. In investigating the reasons behind ALBEF's outperformance over CLIP, several distinctive elements of ALBEF come into play. These include the use of momentum distillation self-training, which involves learning from pseudo targets generated by the model itself, cross-attention to enhance language and vision alignment, and the use of contrastive learning with hard negative samples. Apart from architectural differences, we believe that training data also contributes to performance variations. Specifically, ALBEF was trained using two web datasets (Conceptual Captions (Sharma, Ding, Goodman, & Soricut, 2018), SBU Captions (Ordonez, Kulkarni, & Berg, 2011)) and two in-domain datasets (COCO (Lin et al., 2014) and Visual Genome (Krishna et al., 2017)), totaling 14.1 million images. The presence of COCO in the training datasets contributes to enhancing ALBEF's performance on this dataset, allowing it to outperform CLIP, which was trained on a substantially larger dataset consisting of 400 million image-text pairs. In the scenario of Multimodal Search Space, the Recall values at 0.0 underscore the challenge of conducting crossmodal retrieval, mainly due to the misalignment in multimodal embeddings. This difficulty is attributed to the noise introduced by assets sharing the same modality as the query.

Moreover, Table 6 demonstrates the substantial benefits of FAISS compared to vanilla exhaustive search in terms of search time, with measurements conducted on a 64-core AMD Epyc CPU with 1TB of RAM and one NVIDIA A100 GPU with 80 GB. Vanilla cosine denotes a similarity search performed across the entire Search Space without

Table 3
Values of **Recall@k** computed on 1k validation set of **MSCOCO**. The recall values are reported in two scenarios: Unimodal Search Space and Multimodal Search Space. In the latter, it can be observed how the modality gap makes crossmodal retrieval impossible in the presence of noise generated by assets of the same modality as the query.

Model/Recall	Unimodal Search Space						Multimodal Search Space					
	Txt2Img			Img2Txt			Txt2Img			Img2Txt		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP	0.51	0.81	0.90	0.53	0.80	0.89	0.0	0.0	0.0	0.0	0.0	0.0
ALBEF	**0.67**	**0.93**	**0.97**	**0.69**	**0.92**	**0.97**	0.0	0.0	0.0	0.0	0.0	0.0

Table 4
Values of **Recall@k** computed on 1k validation set of **FLICKR30k**. The recall values are reported in two scenarios: Unimodal Search Space and Multimodal Search Space. In the latter, it can be observed how the modality gap makes crossmodal retrieval impossible in the presence of noise generated by assets of the same modality as the query.

Model/Recall	Unimodal Search Space						Multimodal Search Space					
	Txt2Img			Img2Txt			Txt2Img			Img2Txt		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP	0.60	0.85	0.90	0.58	0.85	0.91	0.0	0.0	0.0	0.0	0.0	0.0
ALBEF	**0.61**	**0.86**	**0.92**	**0.63**	**0.87**	**0.92**	0.0	0.0	0.0	0.0	0.0	0.0

Table 5
Encoding times of CLIP and ALBEF.

	Text encoding time (ms)	Image encoding time (ms)
CLIP	9.10	17.43
ALBEF	**8.15**	**15.71**

Table 6
Search time of vanilla exhaustive search and FAISS applied on CLIP and ALBEF embeddings.

	n. assets	Embeddings size	K	Run time (ms)
Vanilla cosine	5000	512	5000	129.31
FAISS (CLIP)	5000	512	5000	0.72
FAISS (ALBEF)	5000	256	5000	**0.56**

any filtering. In contrast, FAISS optimizes this search by narrowing the Search Space to areas where there is a high probability of finding assets associated with higher similarity scores concerning the input query. For both VLM encoders, there are noteworthy improvements in search time, decreasing from 129.31 ms to 0.72 and 0.56 for CLIP and ALBEF, respectively, without compromising recall values on the 1k MSCOCO validation set (refer to Table 7). In the case of ALBEF, there is only a marginal decrease of 0.24% and 0.18% for Image Retrieval and Text Retrieval, respectively. It is important to highlight that ALBEF, which maps assets into 256-dimensional embeddings instead of 512-dimensional ones, is associated with search times shorter by about 0.2 ms compared to CLIP. In a large-scale scenario, this, along with a shorter encoding time, can lead to a significant advantage in terms of retrieval time (see Table 5).

5.2. ZCCR clustering components

As introduced in Section 4.3, we test different projection techniques (None, PCA, TSNE), clustering techniques (Hierarchical Agglomerative Clustering, HDBSCAN) and which choice of cluster representative (centroid, medoid) lead to better performance in terms of Recall@10. We focus on ALBEF multimodal encoder applied to *MSCOCO classified* and *FLICKR30k classified* as defined in Section 4.1. Figs. 14, 15, 16, and 17 display Recall@10 values across varying numbers of ImageNet classes assigned to the user u (column-wise), reflecting the respective number of areas of interest (1, 5, 10) associated with user u at recommendation time. The rows exhibit different projection techniques, including None (no projection), linear PCA, and non-linear TSNE. In each cell, Recall@10 is graphically represented over various numbers of points per ImageNet class (5, 10, 20, 30). This refers to the count of assets uploaded by user u from a particular area of interest at recommendation time. The plots distinguish between Agglomerative Clustering (in red) and HDBSCAN (in blue), with further distinction based on the use of centroid (solid line) or medoid (dashed line) as the cluster representative (seed). We observe that: (1) Not projecting data before clustering points leads to poor results. As regards Agglomerative Clustering, the capabilities to discriminate among clusters in a high-dimensional space becomes difficult as the number of nominal ImageNet classes increases (i.e. the number of user u interests). It leads to representatives (centroids, medoids) that are not effective in retrieving relevant information since they come from non-cohesive clusters. This challenge is mitigated by implementing linear or non-linear projection just before clustering points, indicating that reducing the dimensionality facilitates clustering algorithms in discerning between distinct groups. (2) The ability of clustering algorithms to effectively group similar points and distinguish dissimilar ones improves with an increase in the number of points per ImageNet class. This effect is particularly notable for HDBSCAN, whereas Agglomerative Clustering is less influenced by the number of assets originating from a specific area of interest. Consequently, Agglomerative Clustering demonstrates greater robustness across diverse scenarios. (3) Agglomerative Clustering exhibits higher Recall@10 values compared to HDBSCAN. It excels at identifying well-separated clusters, particularly when preceded by a dimensionality reduction technique. From these clusters, Agglomerative Clustering generates seeds that prove effective in retrieving similar assets within the Search Space. (4) As outlined in Section 3.2, there are no systematic differences in terms of retrieval Recall between centroid and medoid. Given the simplicity and linear time complexity of centroid calculations, in contrast to the potentially higher computational cost associated with medoid calculations, we opt to utilize the centroid as the cluster seed for executing the search query. (5) Implementing PCA projection followed by Agglomerative Clustering produces the highest Recall@10 values. Furthermore, this approach exhibits reduced sensitivity to fluctuations in the number of areas of interest and their respective number of assets.

The observations remain consistent across all tasks, including Txt2Img, Txt2Txt, Img2Txt, Img2Img, and datasets MSCOCO and FLICKR30k. To illustrate this, Fig. 18 serves as a further example, reaffirming the earlier findings, even when CLIP is employed to encode MSCOCO data. Specifically, we focus on instances where PCA projection is applied across the four tasks on MSCOCO.

The decision to use Agglomerative Clustering, preceded by a dimensionality reduction of embeddings using PCA, enables ZCCR not only to attain favorable retrieval results but also to exhibit flexibility across diverse scenarios. This is particularly evident in cases where users possess varying numbers of areas of interest and upload different volumes of assets. Such an approach reinforces the zero-shot characteristics of ZCCR, making it well-suited for integration without the need for specific training to adapt to particular contexts.

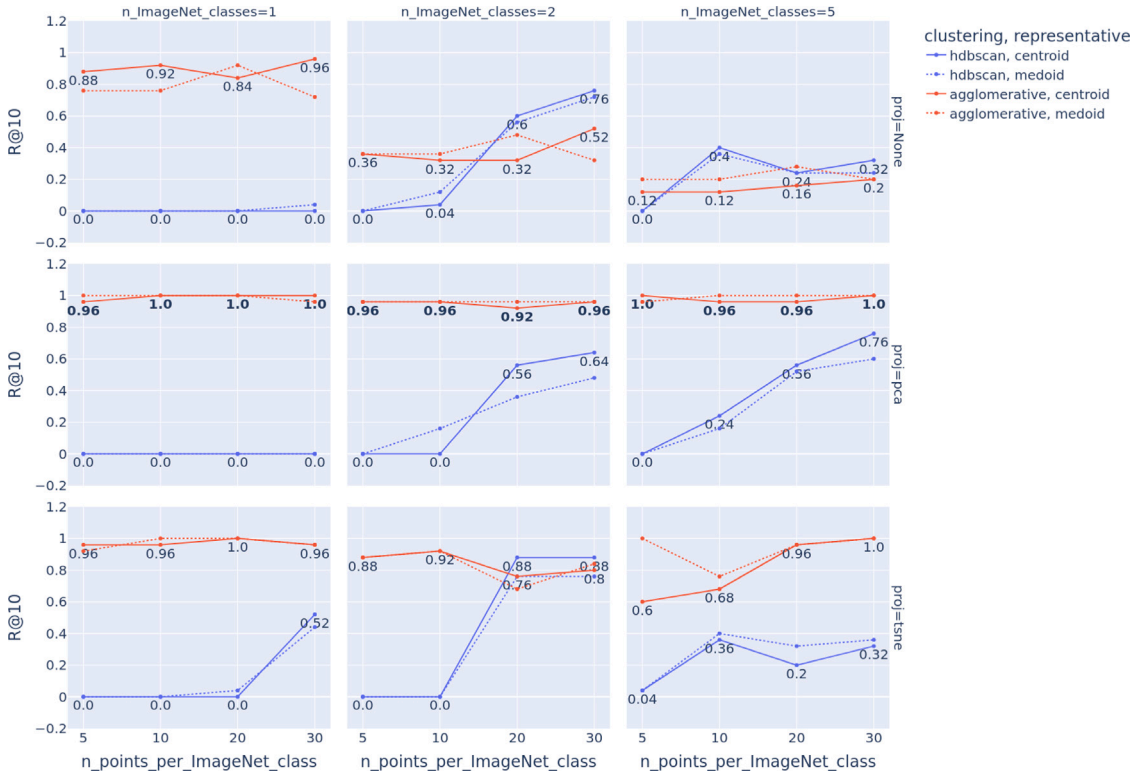


Fig. 14. **MSCOCO** classified 1k validation set - **Txt2Img** Recall@10. The values are obtained using the **ALBEF** multimodal encoder. Each row in the grid represents the projection technique (None, PCA, T-SNE) applied before employing either Agglomerative Clustering or HDBSCAN. Each column denotes the true number of areas of interest (ImageNet classes) existing in the user u historical asset space. Within a cell, Recall@10 is influenced by the number of assets within each ImageNet class. Red and blue lines depict the values for Agglomerative Clustering and HDBSCAN, respectively. A solid line indicates the use of the centroid as the cluster representative, while a dashed line represents the medoid.

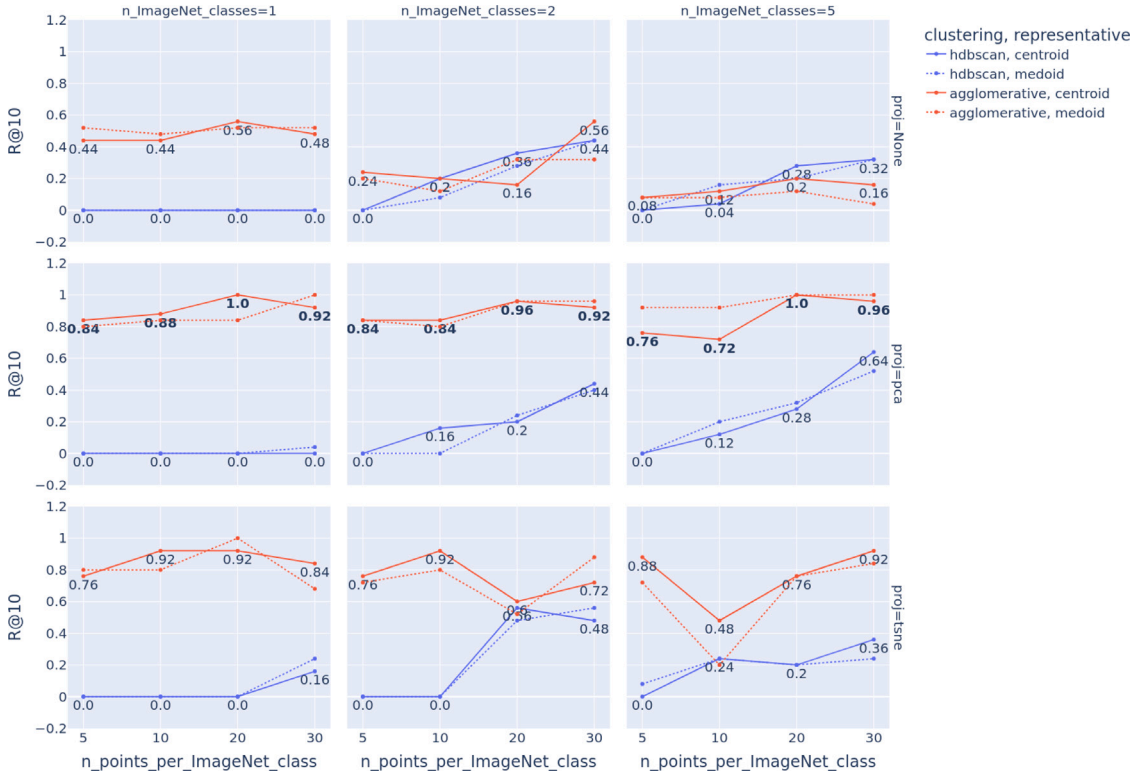


Fig. 15. **MSCOCO** classified 1k validation set - **Txt2Txt** Recall@10. The values are obtained using the **ALBEF** multimodal encoder. Each row in the grid represents the projection technique (None, PCA, T-SNE) applied before employing either Agglomerative Clustering or HDBSCAN. Each column denotes the true number of areas of interest (ImageNet classes) existing in the user u historical asset space. Within a cell, Recall@10 is influenced by the number of assets within each ImageNet class. Red and blue lines depict the values for Agglomerative Clustering and HDBSCAN, respectively. A solid line indicates the use of the centroid as the cluster representative, while a dashed line represents the medoid.

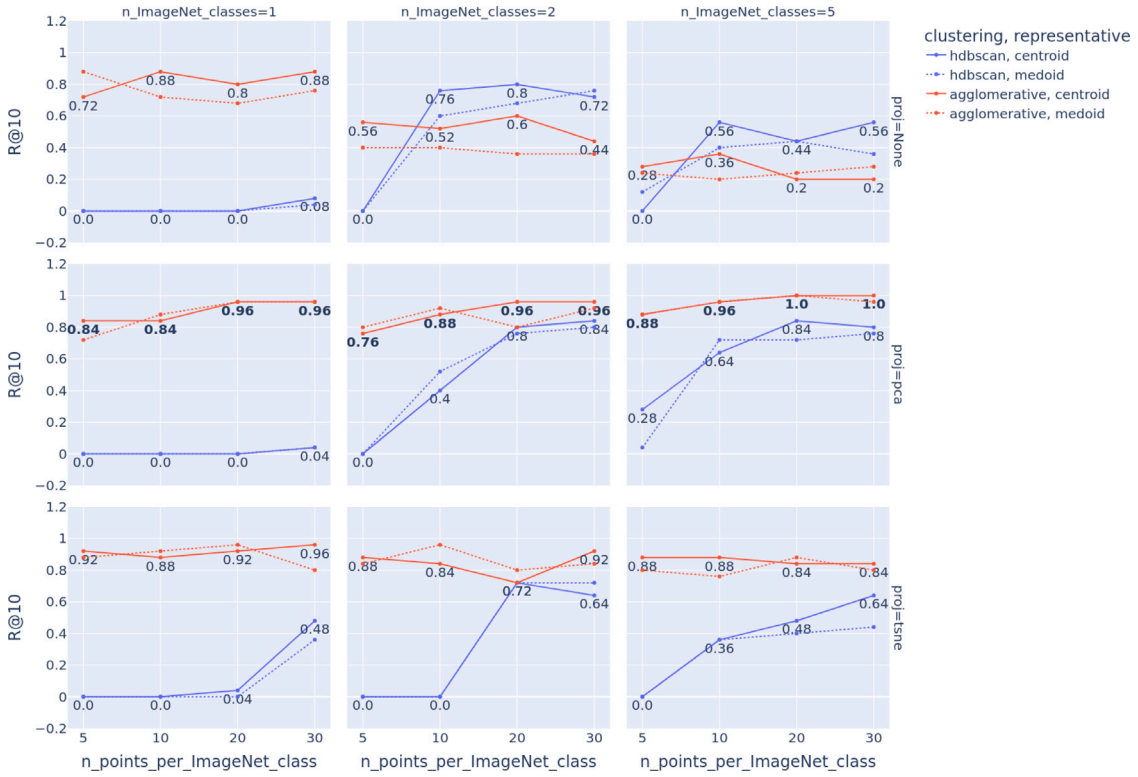


Fig. 16. **FLICKR30k** classified 1k validation set - **Img2Txt** Recall@10. The values are obtained using the **ALBEF** multimodal encoder. Each row in the grid represents the projection technique (None, PCA, T-SNE) applied before employing either Agglomerative Clustering or HDBSCAN. Each column denotes the true number of areas of interest (ImageNet classes) existing in the user u historical asset space. Within a cell, Recall@10 is influenced by the number of assets within each ImageNet class. Red and blue lines depict the values for Agglomerative Clustering and HDBSCAN, respectively. A solid line indicates the use of the centroid as the cluster representative, while a dashed line represents the medoid.

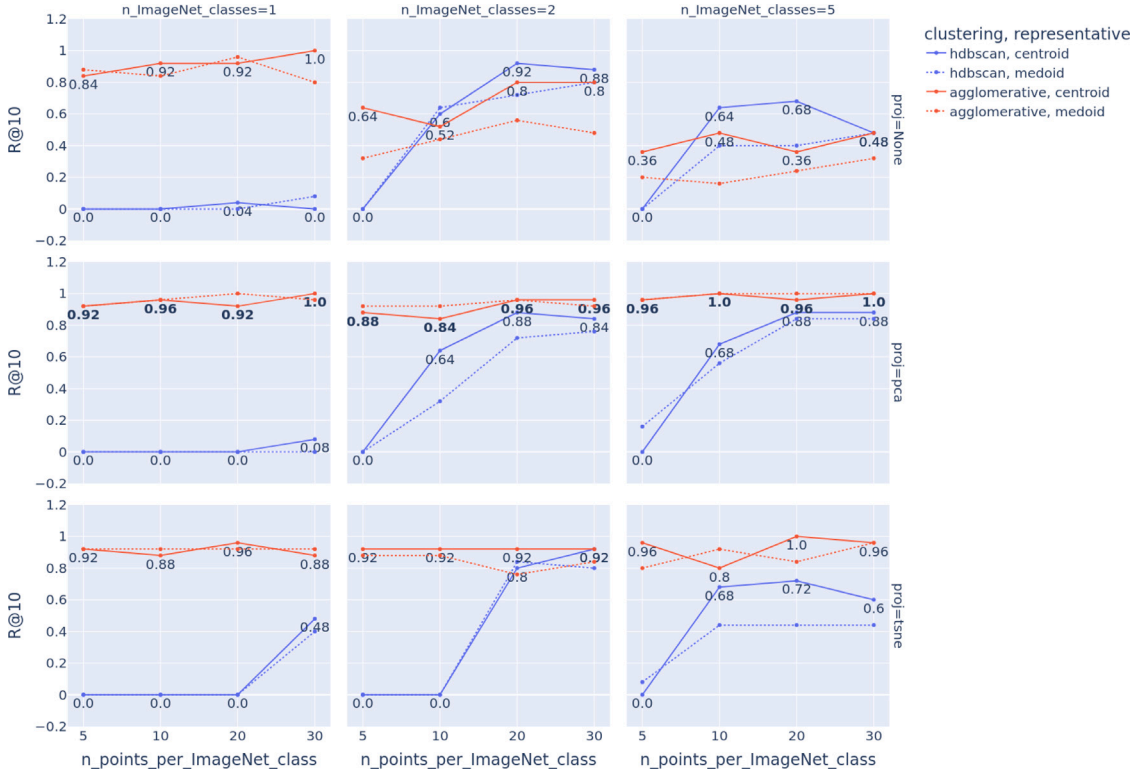


Fig. 17. **FLICKR30k** classified 1k validation set - **Img2Img** Recall@10. The values are obtained using the **ALBEF** multimodal encoder. Each row in the grid represents the projection technique (None, PCA, T-SNE) applied before employing either Agglomerative Clustering or HDBSCAN. Each column denotes the true number of areas of interest (ImageNet classes) existing in the user u historical asset space. Within a cell, Recall@10 is influenced by the number of assets within each ImageNet class. Red and blue lines depict the values for Agglomerative Clustering and HDBSCAN, respectively. A solid line indicates the use of the centroid as the cluster representative, while a dashed line represents the medoid.

Table 7
Comparison of retrieval recall between vanilla cosine and the FAISS search index for both CLIP and ALBEF encodings of the MSCOCO 1k validation set. The FAISS index results in a very low decrease in Recall for both encoders, approximately 0.2% and 0.14% for image and text retrieval for CLIP, and 0.24% and 0.18% for ALBEF.

		Txt2Img				Img2Txt			
		R@1	R@5	R@10	Avg. Delta	R@1	R@5	R@10	Avg. Delta
CLIP	Vanilla cosine	0.514	0.818	0.906	+0%	0.529	0.806	0.889	+0%
	FAISS	0.513	0.816	0.904	−0.208%	0.528	0.805	0.888	−0.138%
ALBEF	Vanilla cosine	0.670	0.931	0.971	+0%	0.692	0.920	0.971	+0%
	FAISS	0.668	0.929	0.969	−0.244%	0.691	0.918	0.969	−0.182%

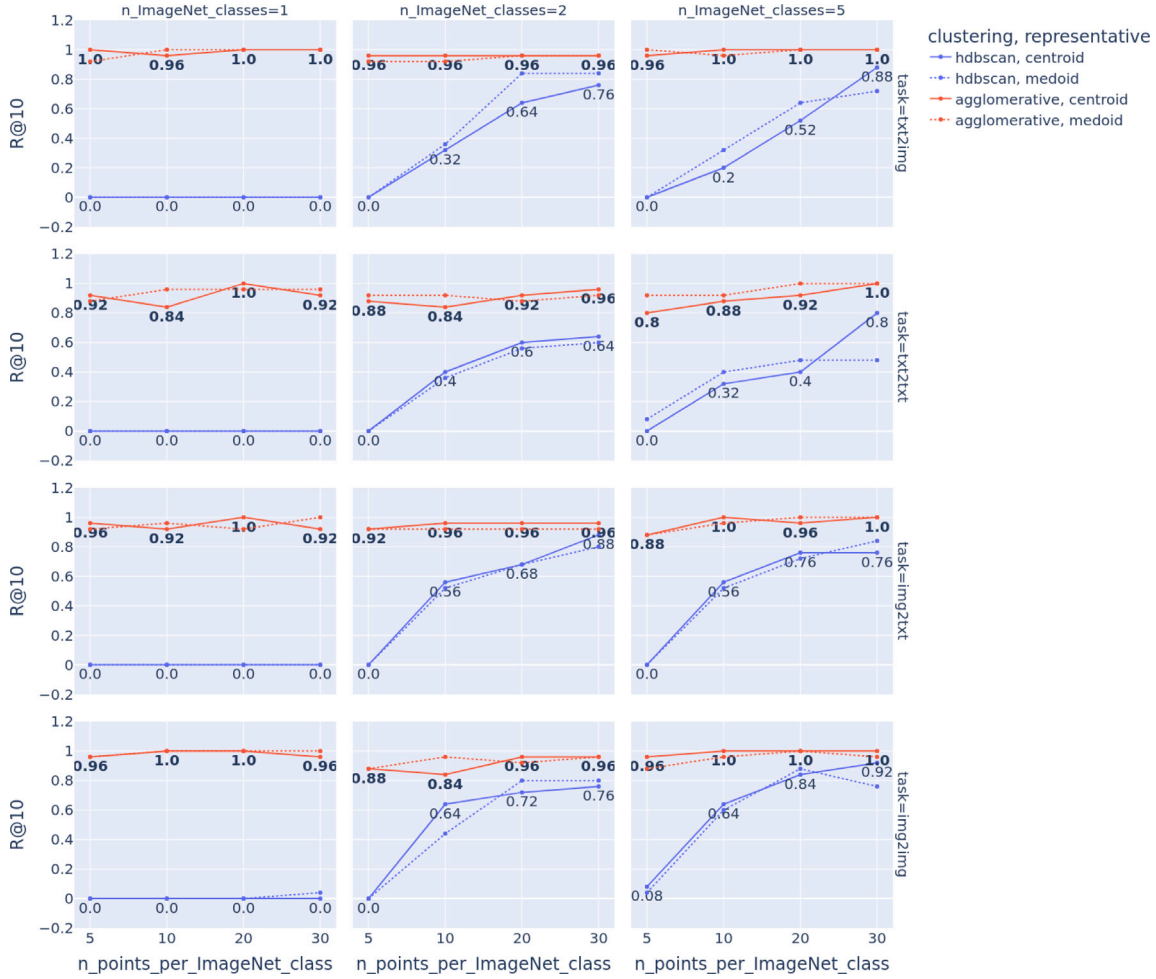


Fig. 18. MSCOCO classified val 1k recommendation evaluated in terms of Recall@10. Values are obtained by using **CLIP** multimodal encoder followed by **PCA** projection to cluster the embeddings by means of Agglomerative Clustering or HDBSCAN. Each row in the grid shows the task. Columns indicate the true number of areas of interest (ImageNet classes) that exist in the user u historical asset space. Inside a cell, Recall@10 is a function of how many assets each ImageNet class contains. Red and blue lines show the value of Agglomerative Clustering and HDBSCAN respectively. Solid line specifies the use of centroid as cluster representative, while dashed line the medoid.

5.3. Comparison between ZCCR and baseline tagging

Figs. 19 and 20 present a comparative analysis between our ZCCR and the Baseline Tag systems, focusing on Recall@10. The evaluation is conducted on the *MSCOCO classified* and *FLICKR30k classified* datasets (refer to Section 4.1). Similar to the structure outlined in 5.2, Recall@10 values are reported across varying numbers of ImageNet classes assigned to the user u (column-wise), reflecting the corresponding number of areas of interest (1, 5, 10) associated with user u at recommendation time. The rows represent different retrieval tasks, specifically Txt2Img, Txt2Txt, Img2Txt, and Img2Img. In each cell, Recall@10 is visually depicted across different numbers of points per ImageNet class (5, 10, 20, 30). These values correspond to the count of assets uploaded by user u from a specific area of interest at the time of recommendation. The Recall@10 values for our ZCCR best configuration, involving PCA projection followed by Agglomerative Clustering with the centroid as the cluster representative, are depicted by the Red and Pink lines. These configurations employ ALBEF and CLIP as multimodal feature encoders, respectively. The Blue line corresponds to the Baseline Tagging (BT) system, which compare different assets through tag matching, as outlined in 4.4. The Green line represents BTBA (Baseline Tagging Bert Agglomerative), which relies on asset tags and shares components with ZCCR. BTBA uses BERT to extract embeddings from asset-associated tags, followed by Agglomerative Clustering to identify user u areas of interest. Additionally, a FAISS index is employed to search for other assets, following the same index decoupling structure as defined in 3.3. The key distinction between ZCCR and BTBA lies in the encoder utilized for extracting asset embeddings: the former employs a Vision-Language Model (VLM) for encoding raw assets, while the latter uses BERT for encoding tags associated with an asset. This

comparative analysis aims to demonstrate the superior effectiveness of ZCCR's VLM multimodal encoding over tag language encoding.

Across both datasets and all four tasks, it is observed that the Recall@10 values associated with BT are consistently the lowest, and these values tend to decrease as the number of ImageNet classes increases. This phenomenon can be attributed to the heterogeneity of tags derived from assets uploaded by user u, which compose the search query. The diversity in these tags prevents a focused search for relevant assets in the Search Space. The excessive generality of the query is effectively addressed by clustering embeddings derived from the tags of assets uploaded by user u. In contrast to BT, BTBA not only outperforms BT due to the enhanced semantic value of tag Bert embeddings but also maintains high Recall@10 values even as the number of ImageNet classes increases (indicative of more areas of interest associated with user u). This highlights the effectiveness of clustering embeddings in mitigating the challenges posed by overly general queries and contributes to improved performance in asset retrieval tasks.

Finally, the results clearly show how ZCCR, whether using ALBEF or CLIP as the VLM encoder, is associated with higher Recall@10 values than BTBA. Therefore, it exhibits more efficient recommendation performance. Considering that the sole difference between the two alternatives lies in the encoder used, this suggests that a pretrained VLM encoder is more effective than a unimodal language encoder in extracting embeddings in a multimodal context, such as that of a social media platform in MediaVerse.

6. Discussion

The suggested ZCCR method, standing for Zero-shot Content-based Crossmodal Recommendation system, functions as a plug-and-play solution that does not necessitate any training or domain-specific data. It utilizes CLIP or ALBEF as Vision-Language Models (VLM) to encode images and texts, producing embeddings that we have observed to be linked with the challenge of misalignment in multimodal embeddings. This misalignment involves inconsistencies in similarities between unimodal and crossmodal pairs. The examination of the causes and potential solutions for this issue extends beyond the limits of this article, requiring specialized efforts to achieve modality-invariance in multimodal embeddings. Modality-invariance denotes the consistency of embeddings across diverse modalities, depending exclusively on the semantic content of the encoded asset.

We observed that the modality gap adversely affects retrieval performance when both text and image embeddings are integrated into the same Search Space (Tables 3, 4). To tackle this challenge, we suggest an approach that includes creating two FAISS indexes – one dedicated to text and another to images. The FAISS index plays a dual role by storing embeddings and expediting the search for similarity during recommendation, resulting in improved efficiency (Table 6).

The separation of indexes proved effective in recommending across various input and output modalities (Text2Text, Image2Image, Text2Image, and Image2Text), Figs. 19, 20. ZCCR converts the recommendation task into a retrieval one by creating user profiles through clustering uploaded assets based on latent areas of interest. We noted the efficacy of Agglomerative Clustering, preceded by PCA dimensionality reduction, in identifying clusters (and hence areas of interest) without prior knowledge of their number 14. The centroid of these clusters serves as a search seed for retrieval tasks on the Search Space containing all assets associated with other users on the platform.

Experiments on MSCOCO and FLICKR30k validate these findings, establishing ZCCR's superiority over a baseline tagger that matches queries and assets in the Search Space based on their tags. Furthermore, it outperforms a more advanced system named BTBA, which employs a Language Model (LLM) like BERT for encoding tags. Although BTBA shares the same structure as ZCCR, the only distinction lies in the encoder employed (LLM vs. VLM). Results indicate that directly extracting embeddings from the asset using VLM is more effective than depending on tags, resulting in more semantically consistent and compact embeddings within a latent area of interest, Figs. 19, 20.

In the following, we refer to two limitations in our experimental setup. (1) It is important to note a limitation in using Flickr for evaluating recommendation systems. The dataset's focus on photography and high-quality images may not fully represent the broader asset landscape found on some social media platforms. Researchers should consider this limitation when generalizing findings to recommendation scenarios that involve a more varied asset mix. (2) Another limitation of the described experimental setup lies in the assumption that semantically similar assets can be grouped into the same user area of interest. This is not necessarily true because assets with similar embeddings might be associated with different areas of interest. Consider two images, Image A and Image B, both portraying beach scenes. These images are deemed semantically similar based on the embeddings or features derived from them, as they exhibit shared visual characteristics associated with beach scenes. Now, suppose a user has two distinct areas: "Tropical Beaches" and "Surfing" that come from the previous uploaded assets. However, due to the nature of the embeddings, both Image A and Image B might be linked to the same or similar embeddings, leading to the assumption that they pertain to the same user area of interest. In reality, the user may have intended Image A for the "Tropical Beaches" category and Image B for the "Surfing" category. Therefore, the assumption that semantically similar assets are inherently in the same user area of interest may not hold true in all cases. This limitation underscores the need for careful consideration and validation when using embedding-based approaches for grouping and recommending user interests.

These two limitations restrict the full generalizability of the system to all scenarios, paving the way for new experimental setups.

7. Conclusion and future work

In this paper, we introduced ZCCR, a Zero-shot Content-based Crossmodal Recommendation System that utilizes a pretrained Vision-Language Model (VLM) to extract embeddings from texts and images. ZCCR employs Agglomerative Clustering to identify a user's areas of interest and constructs a user query for searching similar assets in the Search Space. Additionally, it addresses the challenge of modality gap associated with VLMs by using two FAISS indexes—one for texts and another for images. ZCCR proves to be highly effective in recommending assets similar to a user's profile, transforming the recommendation task into a retrieval task and enhancing search efficiency through FAISS similarity search indexes. ZCCR outperforms both a baseline tagging system (BT) and a more sophisticated system named BTBA, which uses a Large Language Model (LLM) to extract embeddings from tags. The results demonstrate that even in the latter case, embeddings extracted directly from raw assets yield better outcomes than relying on intermediate tags generated by other tools. Consequently, ZCCR emerges as a Zero-shot recommendation solution that can be seamlessly integrated without requiring training or domain-specific data, encompassing both text and image modalities.

ZCCR has its limitations, primarily stemming from its focus on the modalities of images and text, as it relies on pre-trained VLMs in these specific domains. Another constraint is associated with the extensive variety of assets that can potentially be recommended to users. Specifically, with n_{task} representing the number of tasks (which is equal to 4: Text2Image, Text2Text, Image2Text, and Image2Image) and m indicating the number of areas of interest linked to a user profile, the system's output comprises $m \times n_{\text{task}}$ top-k rankings. This quantity can become considerable when the number of areas of interest is substantial. Therefore, it is advisable to maintain a small value for k to ensure that each top-k ranking provides a concise list.

Future work involves extending ZCCR to handle additional modalities, such as video, 3D, 360-degree content, and audio. Efforts should be

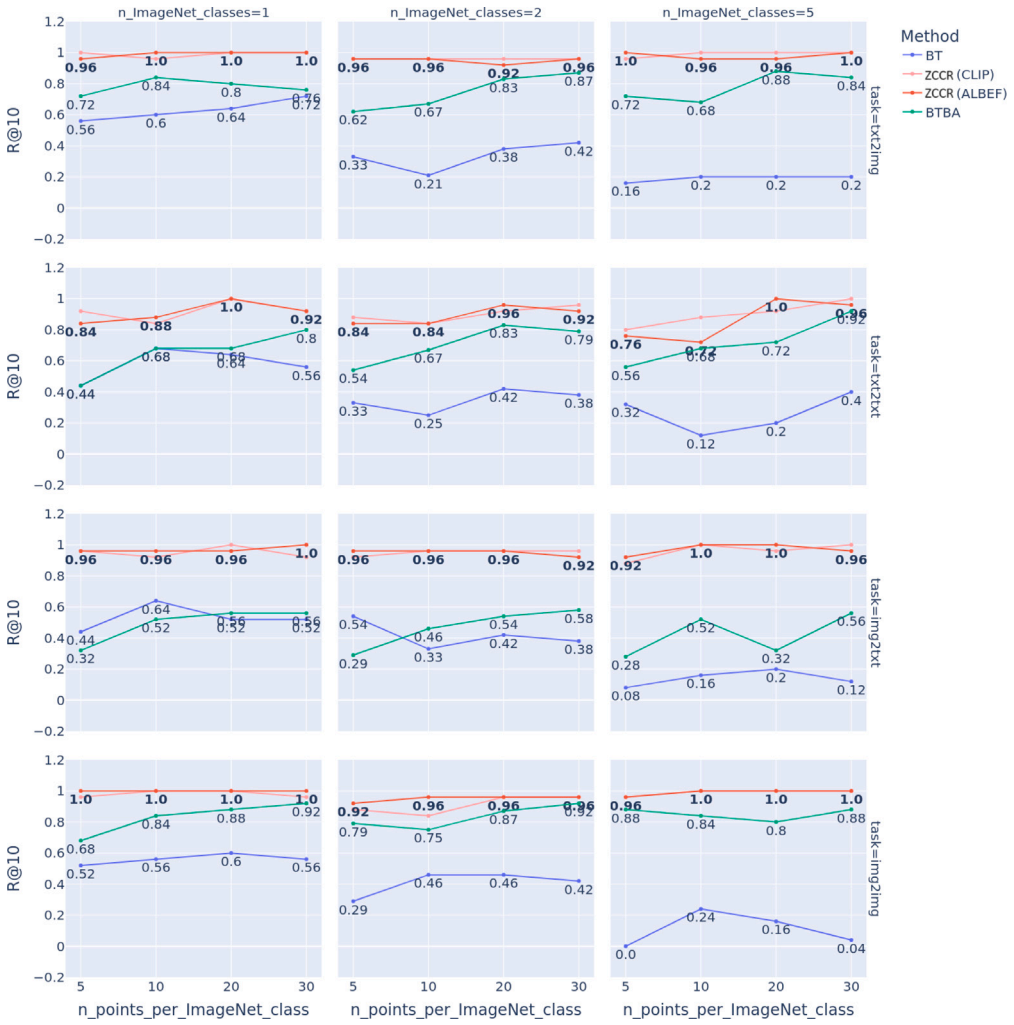


Fig. 19. The Recall@10 values for ZCCR are represented in the following manner: red for ALBEF and pink for CLIP. Baseline Tagger (BT) is denoted in blue, while Baseline Tagger + Bert embeddings + Agglomerative Clustering (BTBA) is in green. The evaluation is performed on **MSCOCO classified**. The values are plotted against the number of ImageNet classes (columns) and the number of points per ImageNet class (x-axis). The tasks Txt2Img, Txt2Txt, Img2Txt, and Img2Img are differentiated by rows. Text numbers on the chart are displayed for all curves except ZCCR (CLIP) to maintain clarity in visualization. The best values associated with ZCCR (ALBEF) are highlighted in bold.

also directed towards investigating the causes and potential solutions for the modality gap issue. Resolving this problem, specifically achieving modality-invariant embeddings, would enable ZCCR to expedite search times by using a single index for all modalities, as seen in the instance of a single index being more advantageous than multiple indexes. Furthermore, having a single index for all modalities would present users with a clearer and more concise recommendation result. Achieving modality-invariance in embeddings would remove the necessity to distinguish tasks based on the query and Search Space modalities, consolidating them into a single index. This unified index would then execute m queries, each associated with an area of interest containing semantically similar texts and images. Consequently, providing a seed generated by both images and texts of the same area of interest would yield similar texts and images by querying a single index. Moreover, this approach would impact the clustering algorithm, reducing its execution frequency to once instead of multiple times for each involved modality. Concluding, an alternative avenue for exploration involves evaluating other clustering algorithms that do not require prior knowledge of the number of clusters, which in ZCCR corresponds to the number of areas of interest during the recommendation phase. These approaches have been investigated in the domain of Evolutionary Computation, with a specific emphasis on employing Genetic Algorithms, which have demonstrated their effectiveness in addressing NP-Complete problems such as clustering.

CRediT authorship contribution statement

Federico D'Asaro: Conceptualization, Methodology, Software, Validation, Investigation, Writing — original draft, Writing — review & editing. **Sara De Luca:** Conceptualization, Methodology, Investigation, Visualization, Writing — original draft. **Lorenzo Bongiovanni:** Conceptualization, Writing — review & editing. **Giuseppe Rizzo:** Resources, Validation, Supervision. **Symeon Papadopoulos:** Data Curation, Supervision. **Manos Schinas:** Supervision. **Christos Koutlis:** Data Curation, Validation.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Federico D'Asaro reports financial support was provided by European Unions Horizon 2020 Research and Innovation Programme under grant agreement No 957252, MediaVerse project.

Data availability

I've shared the link to the code in the article abstract.

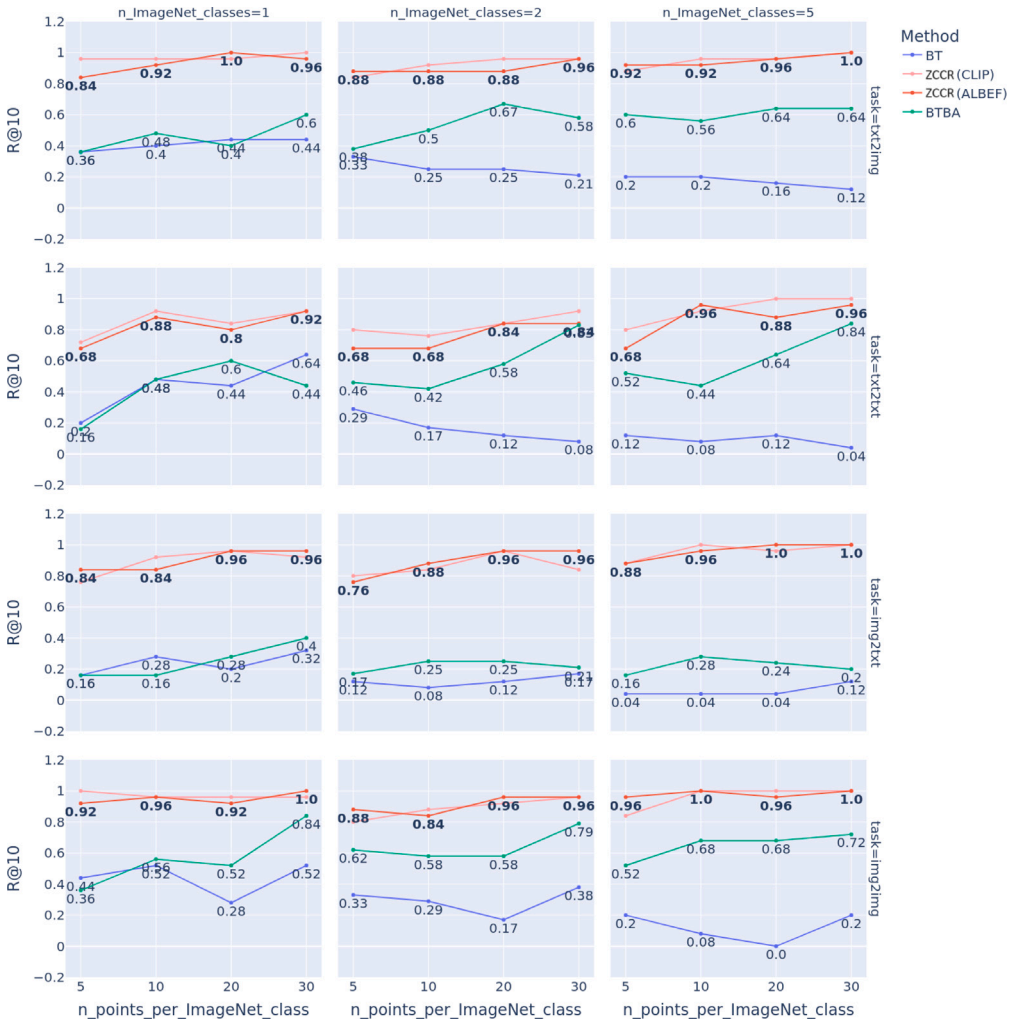


Fig. 20. The Recall@10 values for ZCCR are represented in the following manner: red for ALBEF and pink for CLIP. Baseline Tagger (BT) is denoted in blue, while Baseline Tagger + Bert embeddings + Agglomerative Clustering (BTBA) is in green. The evaluation is performed on **FLICKR30k classified**. The values are plotted against the number of ImageNet classes (columns) and the number of points per ImageNet class (x-axis). The tasks Txt2Img, Txt2Txt, Img2Txt, and Img2Img are differentiated by rows. Text numbers on the chart are displayed for all curves except ZCCR (CLIP) to maintain clarity in visualization. The best values associated with ZCCR (ALBEF) are highlighted in bold.

Acknowledgments

The authors acknowledge the funding received from the European Unions Horizon 2020 Research and Innovation Programme under grant agreement No 957252 "Empowering next-generation media creation, enrichment and distribution", Call: H2020-ICT-2018-20, Topic: ICT-44-2020 - Next Generation Media.

References

Ali, S. M., Nayak, G. K., Lenka, R. K., & Barik, R. K. (2018). Movie recommendation system using genome tags and content-based filtering. In *Advances in data and information sciences: proceedings of ICDIS-2017, volume 1* (pp. 85–94). Springer.

Aljalbout, E., Golkov, V., Siddiqui, Y., Strobel, M., & Cremers, D. (2018). Clustering with deep learning: Taxonomy and new methods. arXiv preprint arXiv:1801.07648.

Assent, I. (2012). Clustering high dimensional data. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *2*(4), 340–350.

Berbague, C. E., Karabadji, N. E.-i., Seridi, H., Symeonidis, P., Manolopoulos, Y., & Dhifli, W. (2021). An overlapping clustering approach for precision, diversity and novelty-aware recommendations. *Expert Systems with Applications*, *177*, Article 114917.

Carreira, J., Noland, E., Hillier, C., & Zisserman, A. (2019). A short note on the kinetics-700 human action dataset. arXiv preprint arXiv:1907.06987.

Cui, Z., Yu, F., Wu, S., Liu, Q., & Wang, L. (2021). Disentangled item representation for recommender systems. *ACM Transactions on Intelligent Systems and Technology*, *12*(2), 1–20.

De, U. C., Banerjee, S., Rath, M. K., Swain, T., & Samant, T. (2022). Content based apparel recommendation for E-commerce stores. In *2022 3rd international conference for emerging technology* INCET, (pp. 1–6). IEEE.

De Gemmis, M., Lops, P., Musto, C., Narducci, F., & Semeraro, G. (2015). Semantics-aware content-based recommender systems. *Recommender Systems Handbook*, 119–159.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248–255). Ieee.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Dietz, L. W., Sen, A., Roy, R., & Wörndl, W. (2020). Mining trips from location-based social networks for clustering travelers and destinations. *Information Technology & Tourism*, *22*, 131–166.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, arXiv 2020.

Feichtenhofer, C., Fan, H., Malik, J., & He, K. (2019). Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 6202–6211).

Feng, F., Wang, X., & Li, R. (2014). Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the 22nd ACM international conference on multimedia* (pp. 7–16).

Geng, S., Fu, Z., Ge, Y., Li, L., De Melo, G., & Zhang, Y. (2022). Improving personalized explanation generation through visualization. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)* (pp. 244–255).

Geng, S., Liu, S., Fu, Z., Ge, Y., & Zhang, Y. (2022). Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM conference on recommender systems* (pp. 299–315).

Geng, S., Tan, J., Liu, S., Fu, Z., & Zhang, Y. (2023). VIP5: Towards multimodal foundation models for recommendation. arXiv preprint arXiv:2305.14302.

Gong, Y., Wang, L., Guo, R., & Lazebnik, S. (2014). Multi-scale orderless pooling of deep convolutional activation features. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VII 13* (pp. 392–407). Springer.

Gordo, A., Almazán, J., Revaud, J., & Larlus, D. (2016). Deep image retrieval: Learning global representations for image search. In *Computer vision–ECCV 2016: 14th European conference, amsterdam, the netherlands, October 11-14, 2016, proceedings, part VI 14* (pp. 241–257). Springer.

He, R., & McAuley, J. (2016a). Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th international conference on world wide web* (pp. 507–517).

He, R., & McAuley, J. (2016b). VBPR: visual bayesian personalized ranking from implicit feedback. *Vol. 30*, In *Proceedings of the AAAI conference on artificial intelligence*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Hou, M., Wu, L., Chen, E., Li, Z., Zheng, V. W., & Liu, Q. (2019). Explainable fashion recommendation: A semantic attribute region guided approach. arXiv preprint arXiv:1905.12862.

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc..

Jiang, M., Zhang, Z., Jiang, J., Wang, Q., & Pei, Z. (2019). A collaborative filtering recommendation algorithm based on information theory and bi-clustering. *Neural Computing and Applications*, *31*, 8279–8287.

Johnson, J., Douze, M., & Jégou, H. (2019). Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, *7*(3), 535–547.

Ko, H., Lee, S., Park, Y., & Choi, A. (2022). A survey of recommendation systems: recommendation models, techniques, and application fields. *Electronics*, *11*(1), 141.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., et al. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, *123*, 32–73.

Li, J., Jing, M., Lu, K., Zhu, L., Yang, Y., & Huang, Z. (2019). From zero-shot learning to cold-start recommendation. *Vol. 33*, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 4189–4196).

Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., & Hoi, S. C. H. (2021). Align before fuse: Vision and language representation learning with momentum distillation. *Advances in Neural Information Processing Systems*, *34*, 9694–9705.

Li, L., Zhang, Y., & Chen, L. (2021). Personalized transformer for explainable recommendation. arXiv preprint arXiv:2105.11601.

Li, L., Zhang, Y., & Chen, L. (2023). Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems*, *41*(4), 1–26.

Li, L., Zhang, Y., Liu, D., & Chen, L. (2023). Large language models for generative recommendation: A survey and visionary discussions. arXiv preprint arXiv:2309.01157.

Liang, V. W., Zhang, Y., Kwon, Y., Yeung, S., & Zou, J. Y. (2022). Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, *35*, 17612–17625.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13* (pp. 740–755). Springer.

Man, T., Shen, H., Jin, X., & Cheng, X. (2017). Cross-domain recommendation: An embedding and mapping approach. *vol. 17*, In *IJCAI* (pp. 2464–2470).

McInnes, L., Healy, J., & Astels, S. (2017). Hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, *2*(11), 205.

Meng, L., Feng, F., He, X., Gao, X., & Chua, T.-S. (2020). Heterogeneous fusion of semantic and collaborative information for visually-aware food recommendation. In *Proceedings of the 28th ACM international conference on multimedia* (pp. 3460–3468).

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, *26*.

Ordonez, V., Kulkarni, G., & Berg, T. (2011). Im2text: Describing images using 1 million captioned photographs. *Advances in Neural Information Processing Systems*, *24*.

Peng, Y., Huang, X., & Qi, J. (2016). Cross-media shared representation by hierarchical learning with multiple deep networks. *vol. 3846*, In *IJCAI* (p. 3853).

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748–8763). PMLR.

Sharif Razavian, A., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). 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).

Sharma, P., Ding, N., Goodman, S., & Soricut, R. (2018). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: long papers)* (pp. 2556–2565).

Shi, P., Welle, M. C., Björkman, M., & Kragic, D. (2023). Towards understanding the modality gap in CLIP. In *ICLR 2023 workshop on multimodal representation learning: perks and pitfalls*.

Shi, S., Zhang, M., Yu, X., Zhang, Y., Hao, B., Liu, Y., et al. (2019). Adaptive feature sampling for recommendation with missing content feature values. In *Proceedings of the 28th ACM international conference on information and knowledge management* (pp. 1451–1460).

Shin, W., Park, J., Woo, T., Cho, Y., Oh, K., & Song, H. (2022). E-clip: Large-scale vision-language representation learning in e-commerce. In *Proceedings of the 31st ACM international conference on information & knowledge management* (pp. 3484–3494).

Sohn, K. (2016). Improved deep metric learning with multi-class n-pair loss objective. *Advances in Neural Information Processing Systems*, *29*.

Strang, G. (2012). Linear algebra and its applications.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*.

Veit, A., Kovacs, B., Bell, S., McAuley, J., Bala, K., & Belongie, S. (2015). Learning visual clothing style with heterogeneous dyadic co-occurrences. In *Proceedings of the IEEE international conference on computer vision* (pp. 4642–4650).

Wang, K., He, R., Wang, W., Wang, L., & Tan, T. (2013). Learning coupled feature spaces for cross-modal matching. In *Proceedings of the IEEE international conference on computer vision* (pp. 2088–2095).

Wang, P., Yang, A., Men, R., Lin, J., Bai, S., Li, Z., et al. (2022). Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International conference on machine learning* (pp. 23318–23340). PMLR.

Xu, S., Hua, W., & Zhang, Y. (2023). OpenP5: Benchmarking foundation models for recommendation. arXiv preprint arXiv:2306.11134.

Young, P., Lai, A., Hodosh, M., & Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, *2*, 67–78.

Yuan, F., Zhang, G., Karatzoglou, A., Jose, J., Kong, B., & Li, Y. (2021). One person, one model, one world: Learning continual user representation without forgetting. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 696–705).

Zhang, J. (2019). Robust collaborative filtering based on multiple clustering. In *2019 IEEE 7th international conference on computer science and network technology* ICCSNT, (pp. 174–178). http://dx.doi.org/10.1109/ICCSNT47585.2019.8962505.

Zhang, J., Lin, Y., Lin, M., & Liu, J. (2016). An effective collaborative filtering algorithm based on user preference clustering. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, *45*, 230–240.

Zhang, J., Zhu, Y., Liu, Q., Zhang, M., Wu, S., & Wang, L. (2022). Latent structure mining with contrastive modality fusion for multimedia recommendation. *IEEE Transactions on Knowledge and Data Engineering*.

Zhu, F., Wang, Y., Chen, C., Zhou, J., Li, L., & Liu, G. (2021). Cross-domain recommendation: challenges, progress, and prospects. arXiv preprint arXiv:2103.01696.