A Subspace-Based Analysis of Structured and Unstructured Representations in Image-Text Retrieval

Erica K. Shimomoto[1], Edison Marrese-Taylor[1], Hiroya Takamura[1], Ichiro Kobayashi[1,2], Yusuke Miyao[1,3]
National Institute of Advanced Industrial Science and Technology[1]
Ochanomizu University[2], The University of Tokyo[3]
{kidoshimomoto.e,edison.marrese,takamura.hiroya}@aist.go.jp
koba@is.ocha.ac.jp, yusuke@is.s.u-tokyo.ac.jp

Abstract

In this paper, we specifically look at the image-text retrieval problem. Recent multimodal frameworks have shown that structured inputs and fine-tuning lead to consistent performance improvement. However, this paradigm has been challenged recently with newer Transformer-based models that can reach zero-shot state-of-the-art results despite not explicitly using structured data during pre-training. Since such strategies lead to increased computational resources, we seek to better understand their role in image-text retrieval by analyzing visual and text representations extracted with three multimodal frameworks: SGM, UNITER, and CLIP. To perform such analysis, we represent a single image or text as low-dimensional linear subspaces and perform retrieval based on subspace similarity. We chose this representation as subspaces give us the flexibility to model an entity based on feature sets, allowing us to observe how integrating or reducing information changes the representation of each entity. We analyze the performance of the selected models' features on two standard benchmark datasets. Our results indicate that heavily pre-training models can already lead to features with critical information representing each entity, with zero-shot UNITER features performing consistently better than fine-tuned features. Furthermore, while models can benefit from structured inputs, learning representations for objects and relationships separately, such as in SGM, likely causes a loss of crucial contextual information needed to obtain a compact cluster that can effectively represent a single entity.

1 Introduction

The integration of techniques from Natural Language Processing (NLP) and Computer Vision (CV) has led to the development of multimodal approaches, which have quickly attracted the scientific community's attention. Examples include tasks such as image captioning (Hossain et al., 2019), machine translation (Specia et al., 2016; Elliott et al., 2017), word sense disambiguation (Bevilacqua et al., 2021), and visual question answering (Antol et al., 2015). Great progress in these tasks has been made by using massive amounts of training data with deeper models, leading to rapidly increasing computational costs.

In this paper, we specifically look at the image-text retrieval task, where the goal is to retrieve an image from a text query (image retrieval) or a text from an image query (text retrieval) from a database containing images and texts. In this context, we see a line of works encoding local and global structures to learn representations for both modalities, extracted using object detectors (Qu et al., 2020) and large pre-trained language models (Diao et al., 2021). To further understand the relationship between such structures, several works also encoded visual (Shi et al., 2019) and textual (Wang et al., 2020) scene-graphs or designed their pipelines to learn such graphs (Schroeder and Tripathi, 2020).

A more recent trend has been to use Transformer-based models to learn the representations for each modality and to model their interaction (Chen et al., 2020), also making use of such structured data (Messina et al., 2021; Dong et al., 2022). While these frameworks have resulted in state-of-the-art performance in multiple downstream tasks, including image-text retrieval, the inference is computationally expensive for this task as it requires a forward pass of each image-text pair in the database to perform retrieval.

Although structured inputs and fine-tuning have shown consistent performance improvement across all the aforementioned models, this paradigm has been challenged recently with newer Transformer-based models, such as CLIP (Radford et al., 2021). This model, for example, can not only reduce the computational inference overhead by allowing the images and texts to be processed individually, but

Proceedings of the Workshop on Unimodal and Multimodal Induction of Linguistic Structures (UM-IoS), pages 29 - 44
December 7, 2022 ©2022 Association for Computational Linguistics

it also achieves zero-shot state-of-the-art results for image-text retrieval despite not explicitly using structured data during its pre-training.

In light of these issues, this paper analyzes visual and text representations produced by several multimodal frameworks in the task of image-text retrieval. We are particularly interested in studying the ability of these models in encoding relevant information to perform retrieval in a variety of scenarios, including model fine-tuning versus zero-shot performance for models that require pre-training, as well as how the addition or removal of structure information from images (e.g., scene-graphs) and texts (e.g., semantic triplets), affects such representations. We find it pivotal to understand the role of such strategies as their integration ultimately leads to increased computational resources.

To perform such an analysis, we set a common ground by looking at subspace representations in the context of image-text retrieval. In the subspace setting, the idea is to represent a single entity, e.g., an image or a sentence, as a low-dimensional linear subspace in the original high-dimensional feature space and to perform retrieval based on subspace similarity. Such representation is based on the empirical evidence that patterns of the same entity (e.g., pictures of the same person) tend to cluster in high-dimensional space (Watanabe and Pakvasa, 1973; Iijima et al., 1974). We expect features from the same entity learned by such multimodal frameworks also form these compact clusters, and therefore their distribution can be represented by linear subspaces. Furthermore, as most image-text retrieval frameworks rely on the cosine similarity between feature vectors to compare two entities, the subspace similarity comes in handy as it is equivalent to cosine similarity when we have one-dimensional subspaces (i.e., a single vector representing an entity). Finally, subspaces give us the flexibility to model an entity based on a set of vectors, e.g., a set of object embeddings in an image or set of entities in a sentence, allowing us to observe how integrating more information by fine-tuning or adding structure data, changes the representation of each entity.

This paper focuses on frameworks that explicitly incorporate or capture structured inputs, either from the visual or textual side. Concretely, we evaluate and compare the text-image retrieval performance using the subspace representation of features extracted using three frameworks: SGM (Wang et al., 2020), UNITER (Chen et al., 2020), and CLIP (Radford et al., 2021). We chose these three models based on the distinct way they treat multimodal data: SGM, a scene graph-based model, heavily relies on structured data, generating object-level and relationship-level cross-modal features; UNITER, a pre-trained Transformer-based model that generates joint visual and textual embeddings relying on objects detected on the input images; and CLIP, a pre-trained contrastive model which is trained by simply pairing whole images with complete sentences and without making explicit use of structure, which also allows us to extract of image and text embeddings individually in a zero-shot fashion, overcoming the limitations of previous models such as UNITER.

We analyze the performance of feature subspaces of selected models on two standard benchmark datasets, COCO (Lin et al., 2014) and Flickr30k (Young et al., 2014; Plummer et al., 2015), focusing on the tasks of image-to-text and text-to-image retrieval. Furthermore, we observe how results change when modeling pre-trained and fine-tuned features from UNITER and introducing or removing structure information from SGM and CLIP features. Our results indicate that UNITER's pre-training leads to features with critical information representing each entity during pre-training, with zero-shot features performing consistently better than fine-tuned features. Moreover, we observed that learning representations for objects and relationships separately, such as in SGM, likely causes a loss of crucial contextual information needed to effectively represent a single entity, whereas using only SGM's object representations led to better performance. This result might explain why CLIP features can better characterize entities when features are extracted based on global features, where we observed that explicitly considering local structure information harms retrieval performance.

2 Background

2.1 Subspace representation

Given a set of entities (i.e., images, sentences) whose representations lie on a rich high-dimensional feature space, subspace-based methods aim to encode a set of features representing a given entity (i.e., CNN features from an image, word vectors from a sentence) by a lower-dimensional linear subspace in the original feature space. While there are several ways to obtain the subspace representation, we focus on the formulation based on principal component analysis (PCA). The reason that leads us to consider this method is that PCA can compactly represent the distribution of the features in a set based on the directions of highest variance. Such characteristics lead to a model that can discard irrelevant information, such as noise, while effectively representing variations, e.g., rotation and illumination in images.

Formally, consider a set of N feature vectors $\{\boldsymbol{x}_i\}_{i=1}^N$ representing an entity, stacked as the columns of the matrix $\boldsymbol{X} \in \mathbb{R}^{p \times N}$, where p is the dimension of the original feature space. We apply PCA without data centering to model a subspace from this set of features. The orthonormal basis vectors of the m-dimensional subspace $\mathcal{Y}$ are obtained as the eigenvectors with the m largest eigenvalues $\{\lambda_l\}_{l=1}^m$ of the matrix $\boldsymbol{R} = \boldsymbol{X}\boldsymbol{X}^\top$. The entity is finally represented as $\boldsymbol{Y} = [\boldsymbol{\Phi}_1 \ldots \boldsymbol{\Phi}_m] \in \mathbb{R}^{p \times m}$, which has the corresponding orthonormal basis vectors as its column vectors. For simplicity, we will refer to the subspaces by their bases matrices. Such basis vectors can be interpreted as the main hidden features representing the distribution of the features in the set.

Though several subspace-based methods have been developed over the course of the past 50 years, mainly for image classification, the most relevant variations for this work are the Subspace Method (SM) and the Mutual Subspace Method (MSM; Maeda, 2010), as they establish two important similarity measures that we need to perform image-text retrieval.

Vector-subspace similarity in SM: Consider we have k reference classes represented as m_i-dimensional subspaces $\{\boldsymbol{Y}_i\}_{i=1}^k$ in a p-dimensional vector space, where $m_i < p$. SM seeks to classify an input entity represented by a single feature vector $\boldsymbol{v}_{in}$ normalized to have norm 1. To measure the similarity between the input feature vector $\boldsymbol{v}_{in}$ and a class reference subspace $\boldsymbol{Y}_i$, defined as $S^{in,i} = \boldsymbol{v}_{in}^\top \boldsymbol{P}_i \boldsymbol{v}_{in}$, where $\boldsymbol{P}_i = \boldsymbol{Y}_i \boldsymbol{Y}_i^\top$ is the projection matrix onto the subspace $\boldsymbol{Y}_i$.

Subspace-subspace similarity in MSM: MSM is a generalization of SM, where both input and references are represented as subspaces. Such an approach has been shown to improve the robustness when applied to image-set classification tasks (Maeda, 2010; Fukui and Maki, 2015).

In MSM, the input is represented by a subspace $\boldsymbol{Y}_{in}$ modeled from a set of feature vectors $\{\boldsymbol{x}_i\}_{i=1}^N$. To perform classification, it is necessary to calculate the similarity between the input subspace $\boldsymbol{Y}_{in}$ and the i-th class subspace $\boldsymbol{Y}_i$. This similarity is measured by using the canonical angles between them (Chatelin, 2012). We can calculate them by using the singular value decomposition (SVD) (Fukui and Yamaguchi, 2005).

Consider two subspaces, $\boldsymbol{Y}_{in} \in \mathbb{R}^{p \times m_{in}}$ and $\boldsymbol{Y}_i \in \mathbb{R}^{p \times m_i}$, with m_{in} and m_i dimensions respectively, and $m_{in} \leq m_i$. We first calculate the SVD $\boldsymbol{Y}_{in}^\top \boldsymbol{Y}_i = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$, where $\boldsymbol{\Sigma} = \mathrm{diag}(\kappa_1, \ldots, \kappa_{m_{in}})$, $\{\kappa_j\}_{j=1}^{m_{in}}$ represents the set of singular values, and $(\kappa_1 \geq \ldots \geq \kappa_{m_{in}})$. The similarity can then be calculated as $S^{in,i}(t) = \frac{1}{t}\sum_{j=1}^{t} \kappa_j^2$, where $1 \leq t \leq m_{in}$. This similarity is equivalent to taking the average of the squared cosine of t canonical angles.

Vector-vector similarity: In the special case where both input and reference subspaces have only one dimension, i.e., $\boldsymbol{Y}_{in} = \boldsymbol{\Phi}_{in} \in \mathbb{R}^{p \times 1}$ and $\boldsymbol{Y}_i = \boldsymbol{\Phi}_i \in \mathbb{R}^{p \times 1}$, the subspace similarity is equivalent to the cosine similarity $S^{in,i} = \boldsymbol{\Phi}_{in}^\top \boldsymbol{\Phi}_i$, where both $\boldsymbol{\Phi}_{in}$ and $\boldsymbol{\Phi}_i$ have norm equal to 1.

2.2 Multimodal retrieval frameworks

We used features obtained from three multimodal frameworks that can generate sets of features representing each entity in each modality. As all of our selected models achieve outstanding performance in image-text retrieval while leveraging different types of information, we are interested in studying how varying such input affects the representation of each entity by assessing their performance when using the subspace representation. We briefly introduce our selected models below, referring the reader to the original papers for more details.

2.2.1 SGM

Wang et al. (2020) proposed a scene-graph matching framework (SGM) for image-text retrieval. Concretely, they encode visual and textual scene-graphs in a joint embedding space, resulting in a representation vector for each object and relationship in both modalities. This framework has four main parts, which we describe below.

Scene-graph parsers: Images are fed to a pre-trained scene-graph generator, such as MSDN (Li et al., 2017) and Neural Motifs (Zellers et al., 2018). The obtained visual scene-graphs contain both object and relationship nodes, and each of them has a text label. On the textual side, scene-graphs also contain object and relationship nodes; In addition, textual scene-graphs also have two types of edges: *Word order edge*, which follows the order of the words in the texts; and *Semantic edge*, which is obtained by parsing semantic triplets using SPICE (Anderson et al., 2016), relating objects by their relationships.

Visual graph encoder: Visual features are extracted by encoding the image regions into feature vectors by using a Faster-RCNN. The feature vectors from **object nodes** are extracted from its corresponding image region, and the feature vectors from **relationship nodes** are extracted from the union of the image region of the two object nodes that are connected by the relationship node. Then, these visual features are fused with the word embedding corresponding to the node's label through a multimodal fusion layer. Finally, this graph is encoded by a Graph Convolutional Network, generating one feature vector for each object and each relationship nodes. This results in the feature sets $\boldsymbol{O} = \{\boldsymbol{h}_{o_i}\}_{i=1}^{N_o} \in \mathbb{R}^{1024\times N_o}$, and $\boldsymbol{P} = \{\boldsymbol{h}_{p_i}\}_{i=1}^{N_p} \in \mathbb{R}^{1024\times N_p}$.

Textual graph encoder: It consists of a word embedding layer, a word-level bi-GRU encoder, and a path-level bi-GRU. The word-level bi-GRU processes the nodes following the word order in the caption, while the path-level processes the nodes following the semantic paths. The final feature vector for each node is obtained by averaging the representation given by both bi-GRUs, resulting in the feature sets $\boldsymbol{W} = \{\boldsymbol{h}_{w_t}\}_{i=1}^{N_w} \in \mathbb{R}^{1024\times N_w}$, and $\boldsymbol{R} = \{\boldsymbol{h}_{r_i}\}_{i=1}^{N_r} \in \mathbb{R}^{1024\times N_r}$.

Similarity calculation: Images and texts are compared based on two similarities: Between the visual and textual object nodes (S^o) and between the visual and textual relationship nodes (S^r), defined in Equations 1 and 2. The final graph-based similarity is obtained by summing S^o and S^r.

$$S^o = \frac{1}{N_w}\sum_{t=1}^{N_w} \max_{i\in[1,N_o]} \boldsymbol{h}_{w_t}^T \boldsymbol{h}_{o_i} \tag{1}$$

$$S^r = \frac{1}{N_p}\sum_{t=1}^{N_p} \max_{i\in[1,N_r]} \boldsymbol{h}_{p_t}^T \boldsymbol{h}_{r_i} \tag{2}$$

2.2.2 UNITER

UNiversal Image-TExt Representation (Chen et al., 2020) (UNITER) is a Transformer-based large-scale pre-trained model for joint multimodal embedding. UNITER first goes through a designed pre-training task and learns generalizable contextualized embeddings for each region in an image and each word in an input text, and can be further fine-tuned for image-text retrieval. The model contains mainly two parts: image and text embedders and the transformer module.

Image and text embedders: For images, they first use Faster R-CNN (Ren et al., 2015) to extract visual features for each image region. Next, they encode this information along with the location of the features through a fully-connected layer and then project them into the joint embedding space. For text, they tokenize following BERT (Devlin et al., 2019). Finally, they sum the word embedding and position embedding to generate the final text representation on the joint embedding space.

Transformer module: A transformer module further processes both image and text embeddings, learning generalizable contextualized embeddings for each region and word. In our experiments, we use the output from this module to represent images and texts.

2.2.3 CLIP

Contrastive Language–Image Pre-training (Radford et al., 2021) is also a Transformer-based model which uses a simple contrastive pre-training to predict which caption matches a given caption. In this manner, the model can efficiently construct image and text representations. Natural language supervision is later used to ask the model to name learned visual concepts (or describe new ones), allowing zero-shot transfer to downstream tasks with state-of-the-art performance in many cases.

3 Subspace-based image-text retrieval

The goal of image-text retrieval is to find an image based on a text query (image retrieval) or a text passage based on an image query (text retrieval) from a database containing images and texts. Formally, given a query entity q in one modality, we seek to find the most similar entity e in the other modality.

In this paper, we represent entities and queries by the sets of features extracted from the multimodal frameworks described in the previous section and perform retrieval using subspace-based similarities. In doing so, we assume that the entities in the database are represented as subspaces $\{\boldsymbol{Y}_d\}_{d=1}^{N_d}$

modeled from each entity's feature set, and that the query entity is represented by a set of feature vectors $\{\boldsymbol{q}_i\}_{i=1}^{N_q}$, or by its subspace. Then, we compare the query and each database entity subspace using subspace similarity. We highlight that such setting is equivalent to comparing two feature vectors based on the cosine similarity when we only have one feature vector representing each entity.

We explore the two fundamental subspace similarities described in Section 2.1, performing image-text retrieval in two different ways: Retrieval based on SM and based on MSM.

3.1 SM-based retrieval

In this case, we use the vector-subspace similarity. Since SM assumes single vector inputs, we propose a modification so that the similarity between a set of features and a subspace is defined by the mean similarity between the query features $\{\boldsymbol{q}_i\}_{i=1}^{N_q}$ and the database entity subspace $\boldsymbol{Y}_d$:

$$S^{q,d} = \frac{1}{N_q} \sum_{j=1}^{N_q} \boldsymbol{q}_j^\top \boldsymbol{P}_d \boldsymbol{q}_j, \tag{3}$$

where $\boldsymbol{P}_d = \boldsymbol{Y}_d \boldsymbol{Y}_d^\top$ is the projection matrix onto the subspace of entity d in the database.

When using this similarity, we assume each feature vector of the query is equally important for retrieval.

3.2 MSM-based retrieval

In this case, we use the subspace-subspace similarity. First, we model the query subspace $\boldsymbol{Y}_q$ from its set of features. Then, we perform the search based on the subspace similarity defined in section 2.1.

Using this similarity, we find the closest hidden features in each subspace and measure the angles between them, i.e., the canonical angles. As PCA is used to model the subspaces, features that do not contribute to representing each entity vector set are considered less important to perform retrieval.

4 Experimental Framework

We experimented with image-text retrieval on two datasets, Flickr30k (Young et al., 2014; Plummer et al., 2015) (FLICKR30K) and COCO (Young et al., 2014; Lin et al., 2014) (COCO). Both datasets contain 5 captions (i.e., text passages) for each image. However, they differ in one order of magnitude regarding the number of examples (approx. 300K images on COCOand approx. 30K on FLICKR30K). Because of this reason, in order to keep computational costs within our budget, we used FLICKR30K to extensively study multiple settings and selected only the best configurations for our experiments with COCO.

In all cases, each image and caption is represented by a single or several feature vectors, and retrieval is performed using SM and MSM as defined earlier.

Our evaluation is performed based on the R@k metric, the percentage of queries whose ground-truth is ranked within the top k, which is the standard for the task. We experimented using different subspaces' dimensions and report the best results. Below, we give details about how our multi-modal features are extracted for each model.

SGM With this model, we are particularly interested in understanding how considering objects and their relationships affects retrieval. To extract the features, we use the model checkpoints trained on both datasets provided by the authors. Each image is represented by one set of visual object features $\boldsymbol{O} \in \mathbb{R}^{1024 \times N_o}$, and one set of visual relation features $\boldsymbol{P} \in \mathbb{R}^{1024 \times N_p}$. Each caption is represented by one set of textual object features $\boldsymbol{W} \in \mathbb{R}^{1024 \times N_w}$ and one set of textual relation features $\boldsymbol{R} \in \mathbb{R}^{1024 \times N_r}$.

Considering we have two sets of features representing each visual and textual entity, we followed the same strategy taken by SGM when performing retrieval by calculating S^o and S^r based on subspace similarity, and then summing both to achieve the final similarity for the pair $S^{o,r}$. To better understand the role of each set of features in representing an entity, we also performed retrieval based only on S^o, only on S^r, and on S^g, which represents each entity by the concatenation of the object and relation features.

UNITER As UNITER's excellent performance is due mostly to its extensive pre-training and fine-tuning, we are interested in comparing the retrieval performance of pre-trained features versus fine-tuned features in image-text retrieval. We feed positive image-caption pairs through the model to obtain their joint representations (i.e., sequence of vectors). We split each sequence to obtain one set of features $\boldsymbol{I} \in \mathbb{R}^{768 \times N_i}$ for each image, and one set of features $\boldsymbol{C} \in \mathbb{R}^{768 \times N_c}$ for each caption. For the captions, we disregarded the representation for the [SEP] token.

While we understand that processing only positive image-caption pairs is not the ideal approach to perform retrieval, we reckon this is a limitation of UNITER, as it requires an image and a text passage to be fed simultaneously. Ideally, we would like to be able to forward each image and text only once and perform a ranking on top of the obtained representations. We performed preliminary experiments feeding only captions and only images, but the results showed that this approach does not create meaningful representations. Therefore, since we want to observe the effects of fine-tuning on the multi-modal representations, we primarily focus on the performance difference between them rather than the actual numbers.

We use the pre-trained UNITER released by the authors and test it on three different settings: zero-shot (ZS) where we directly use the pre-trained UNITER to extract our representations; Fine-tuned (FT), where we further train the pre-trained model on the downstream dataset with the default sampling strategy; and another fine-tuned model where the final training is performed using an improved technique for hard negative example mining (FT$_{HN}$). We note that the latter strategy has resulted in the best retrieval performance for the original model.

CLIP: We use the pre-trained model released by OpenAI. Different from SGM, CLIP does not explicitly use structured inputs and represents each image and text as a single feature vector $\boldsymbol{h} \in \mathbb{R}^{512}$.

In this scenario, we seek to understand if processing structured information with CLIP could help improve retrieval performance. To verify this point, we use the co-reference chains and manually annotated bounding boxes for each of the images and captions in the FLICKR30K dataset provided by Plummer et al. (2017) to input structured information and verify how the resulting features perform in contrast with the original CLIP features.

We follow the standard CLIP pipeline and extract an image vector $\boldsymbol{v}_{img} \in \mathbb{R}^{512}$ for each image (Img$_{\text{G}}$), and a caption vector $\boldsymbol{v}_{cap} \in \mathbb{R}^{512}$ for each caption (Text$_{\text{G}}$). Retrieval, in this case, is performed by using simple cosine similarity. We further crop the images following the annotated bounding boxes and process each cropped portion, which results in a set of vectors $\boldsymbol{I} \in \mathbb{R}^{512\times N_i}$ with N_i representations of local objects for each image (Img$_{\text{L}}$). Analogously, we use the annotated entities in the captions to obtain a set of features $\boldsymbol{C} \in \mathbb{R}^{512\times N_c}$ with N_c textual entities representations (Text$_{\text{L}}$). In this case, retrieval is performed based on subspace similarity. We evaluate the performance by using both global (G) and local (L) features, as well as their combination.

Method	Sim	Dim	Mean	R@1	R@5	R@10
			Text Retrieval			
SGM	S^o	-	85.96	70.40	92.10	95.40
	S^r	-	2.43	0.40	2.40	4.50
	$S^{o,r}$	-	**86.33**	71.80	91.70	95.50
SM	S^g	5	**40.40**	20.40	45.10	55.70
	S^o		38.20	18.80	42.10	53.70
	S^r	5	0.33	0.10	0.40	0.50
	$S^{o,r}$		33.80	16.00	37.00	48.40
MSM	S^g	10	59.03	40.20	63.60	73.30
	S^o		**60.53**	40.60	65.70	75.30
	S^r	5	0.80	0.10	0.90	1.13
	$S^{o,r}$		20.80	11.10	22.20	29.10
			Image Retrieval			
SGM	S^o	-	72.54	52.72	78.92	86.00
	S^r	-	1.74	0.40	1.76	3.08
	$S^{o,r}$	-	**73.20**	53.52	79.62	86.46
SM	S^g	5	**39.90**	18.44	44.12	57.14
	S^o		38.48	17.52	42.70	55.24
	S^r	5	1.20	0.28	1.20	2.12
	$S^{o,r}$		36.51	16.30	40.34	52.90
MSM	S^g	5	46.08	26.40	50.60	61.24
	S^o		**47.21**	27.70	51.82	61.10
	S^r	5	1.13	0.20	1.20	2.00
	$S^{o,r}$		42.00	23.68	46.30	56.02

Table 1: Results with SGM-Subspace on the Flickr30k dataset. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces in SM and MSM. Results for the baseline were taken from our reproduction of the original model.

4.1 Choice of subspace dimension

In general, for single modality problems, it is possible to get an idea of the suitable subspace dimension by observing the variance contribution ratio with each additional dimension.

The amount of variance retained by the basis vectors of the subspace can be determined by using the cumulative contribution rate $\mu(m)$. Considering that we want to keep a minimum of μ_{min} of the text variance, we can determine m by ensuring that $\mu(m)_d \geq \mu_{min}$, where $\mu(m)_d = \sum_{l=1}^{m}(\lambda_l)/\sum_{l=1}^{p}(\lambda_l)$. However, preliminary experiments showed us that this metric alone is not suitable to choose the dimension of subspaces modeled from artificially generated multimodal fea-

Method	Sim	Dim	Mean	R@1	R@5	R@10
			Text Retrieval			
SGM	$S^{o,r}$	-	58.56	35.30	64.90	75.50
SM	S^g	5	21.20	7.20	21.20.	35.20
	S^o		**26.40**	9.60	27.6	42.00
	S^r	1	0.40	0.00	0.40	0.80
	$S^{o,r}$		16.70	4.80	16.80	28.40
MSM	S^g	5	42.90	24.40	46.40	58.00
	S^o		**44.90**	24.00	50.40	60.40
	S^r	5	0.10	0.00	0.00	0.40
	$S^{o,r}$		17.10	10.00	18.00	23.20
			Image Retrieval			
SGM	$S^{o,r}$	-	58.90	35.30	64.90	76.50
SM	S^g	5	19.30	4.20	20.20	33.40
	S^o		**21.10**	7.50	22.00	33.80
	S^r	5	0.00	0.00	0.00	0.00
	$S^{o,r}$		21.10	7.70	21.80	33.80
MSM	S^g	5	34.30	16.70	37.10	49.00
	S^o		**35.10**	17.40	38.50	49.40
	S^r	5	0.00	0.00	0.00	0.00
	$S^{o,r}$		34.50	17.70	37.00	48.60

Table 2: Results with SGM-Subspace on the COCO dataset. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces in SM and MSM. Results for the baseline were taken from the original SGM paper.

tures. Therefore, in this work we performed a grid search by assessing the image-text retrieval performance with different subspace dimensions, reporting the best results. We refer the readers to the supplementary material for results with all tested dimensions.

5 Results and Discussions

SGM-subspace: Tables 1 and 2 show the results when using SGM features. In this case, the best subspace performance was achieved by MSM for both tasks, which indicates that leveraging the distribution of the features for both input and references leads to more robust representations.

Furthermore, we can see that while SGM benefits from considering both S^o and S^r with $S^{o,r}$, the subspace-based methods performed better when considering only the objects (S^o) or when considering both globally (S^g), where the information from relationships helped improve results over $S^{o,r}$. Such contrast in results could be due to how SGM calculates the similarity between two entities: It leverages vector-vector relationships, possibly leading the model to focus on local structures and ignore the global context. However, such contextual information is crucial for the subspaces to effectively represent the features from the entity, thus leading to degraded performance.

Method	Type	Dim	Mean	R@1	R@5	R@10
			Text Retrieval			
UNITER	ZS*	-	91.43	80.70	95.70	98.00
	ZS	-	91.50	80.80	95.70	98.00
	FT$^\star_{HN}$	-	**93.93**	85.90	97.10	98.80
	FT$_{HN}$	-	93.36	83.10	95.50	98.50
SM	ZS	20	**91.60**	86.10	93.30	95.40
	FT	20	80.80	69.70	84.50	88.30
	FT$_{HN}$	20	44.20	27.70	48.90	56.10
MSM	ZS	1	**76.00**	63.10	80.10	84.90
	FT	5	56.80	0.40	80.60	89.50
	FT$_{HN}$	5	56.20	1.60	79.00	87.90
			Image Retrieval			
UNITER	ZS*	-	-	66.16	88.40	92.94
	ZS	-	-	66.14	88.36	92.94
	FT$^\star_{HN}$	-	**84.17**	75.52	92.36	96.08
	FT$_{HN}$	-	-	68.02	89.54	94.54
SM	ZS	1	**48.00**	35.00	51.60	57.40
	FT	5	47.40	34.50	51.20	56.50
	FT$_{HN}$	5	28.40	17.10	31.10	37.10
MSM	ZS	1	**75.00**	63.70	78.60	82.70
	FT	15	53.60	32.40	60.50	67.90
	FT$_{HN}$	15	55.30	42.90	58.90	64.10

Table 3: Results with UNITER on FLICKR30K. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces in SM and MSS, and $\star$ denotes results taken from Chen et al. (2020).

Method	Type	Dim	Mean	R@1	R@5	R@10
			Text Retrieval			
UNITER	ZS	-	**81.71**	64.10	87.74	93.30
	FT$^\star_{HN}$	-	81.62	64.40	87.40	93.08
SM	ZS	10	**68.30**	51.30	73.60	79.90
MSM	ZS	5	58.20	38.60	63.80	72.20
			Image Retrieval			
UNITER	ZS	-	70.45	48.79	76.72	85.84
	FT$_{HN}$	-	72.00	50.33	78.52	87.16
SM	ZS	1	24.50	17.00	26.30	30.00
MSM	ZS	1	**38.00**	31.20	40.00	42.80

Table 4: Results with UNITER on COCO, on the full 5k images test set. Mean denotes the mean of the R@1, R@5, and R@10, Dim denotes the dimensions of the subspaces in SM and MSM, and indicates results taken from Chen et al. (2020).

Features	Method	Dim	Text retrieval				Image Retrieval			
			Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
Text$_{\mathrm{G}}$ & Img$_{\mathrm{G}}$	CLIP	-	**90.30**	77.80	95.00	98.10	**76.60**	58.10	82.50	89.40
Text$_{\mathrm{L}}$ & Img$_{\mathrm{L}}$	SM	1	**47.90**	27.30	52.30	64.10	**36.80**	19.00	40.60	50.70
	MSM	1	47.30	27.00	51.70	63.10	35.40	19.90	38.30	47.90
Text$_{\mathrm{L}}$ & Img$_{\mathrm{G}}$	SM	1	61.40	37.40	68.10	78.60	42.00	24.60	45.90	55.50
Text$_{\mathrm{G}}$ & Img$_{\mathrm{L}}$		5	75.70	59.10	81.00	87.00	67.70	46.10	74.50	82.40
Text$_{\mathrm{G}}$ & Img$_{\mathrm{G+L}}$		5	**83.70**	69.90	87.90	93.30	**74.90**	54.50	81.30	88.80
Text$_{\mathrm{G+L}}$ & Img$_{\mathrm{G}}$		5	83.40	67.00	89.20	93.90	70.40	49.90	76.30	84.80
Text$_{\mathrm{G+L}}$ & Img$_{\mathrm{G+L}}$	SM	5	**70.00**	50.30	75.50	84.20	**61.90**	38.40	68.60	78.70
	MSM	1	63.90	44.30	69.70	77.80	61.10	41.10	66.40	75.90

Table 5: Results of our experiments with for CLIP-subspace on FLICKR30K, where the sub-indices G and L indicate the use of global and local features to represent each image and/or caption.

UNITER-subspace: Tables 3 and 4 show the best results for retrieval when using UNITER features. The best subspace performance was achieved using SM in caption retrieval and MSM in image retrieval. We can observe that while the performance of the original UNITER increases after fine-tuning, our best results were achieved using ZS UNITER features, performing about 33.70% and 19.65% better in caption and image retrieval, respectively, in terms of mean R@k compared to hard-negative features in the FLICKR30K dataset.

We can also observe that the best results for both FLICKR30K and COCO were achieved using subspaces with dimensions ranging from 1 to 20, much smaller than the original 768-dimensional feature space, even when ZS features are used. Such low-dimensional subspaces could indicate that the UNITER has already compressed critical information to represent each entity during pre-training.

CLIP-subspace: Table 5 shows the best retrieval results when using CLIP features. Out of the three chosen models, the original CLIP is the closest to the subspace-based retrieval, as it is equivalent to using one-dimensional subspaces of the global G features and, therefore, direct comparison with the subspace-based retrieval is adequate.

We can see that using only G features, i.e., CLIP's original performance, leads to the best results. On the other hand, using only local L features leads to the worst performance. However, we can observe that image representation can better benefit from L features than the captions, leading to the best subspace performance when both G and L features are used to represent images. While considering the structure information does not lead to better performance, this result indicates that G image features are better aligned with the L image features than text features. This result could be explained by the fact that processing isolated textual entities could lead to a loss of context as the subspace representation cannot handle word order.

6 Conclusions and Future Work

The main goal of this paper was to better understand the role of structured inputs and fine-tuning in image-text retrieval. We analyzed visual and text representations extracted with SGM, UNITER, and CLIP by representing a single image or text as low-dimensional linear subspaces and performing retrieval based on subspace similarity. We analyzed how the performance of the selected models' features changed when considering fine-tuning versus zero-shot performance for models that require pretraining, as well as the addition or removal of structure information from images (e.g., scene-graphs) and texts (e.g., semantic triplets).

Our results indicate that UNITER's pre-training leads to features with critical information representing each entity during pre-training, with zero-shot features performing consistently better than fine-tuned features. Moreover, we observed that using only SGM's object representations led to better performance than when considering the relationship representations. Finally, considering structure information with CLIP does not improve the retrieval results. However, we could observe that global information from the text side seems more critical than text local information.

A natural progression of this work is to analyze these features from a geometrical perspective, using the well-established literature on subspace representation.

Acknowledgements

This paper is based on results obtained from a project JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO). For experiments, computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used.

References

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*, pages 382–398. Springer.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Michele Bevilacqua, Tommaso Pasini, Alessandro Raganato, Roberto Navigli, et al. 2021. Recent trends in word sense disambiguation: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*. International Joint Conference on Artificial Intelligence, Inc.

Françoise Chatelin. 2012. *Eigenvalues of Matrices: Revised Edition*. SIAM.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Haiwen Diao, Ying Zhang, Lin Ma, and Huchuan Lu. 2021. Similarity reasoning and filtration for image-text matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1218–1226.

Xinfeng Dong, Huaxiang Zhang, Lei Zhu, Liqiang Nie, and Li Liu. 2022. Hierarchical feature aggregation based on transformer for image-text matching. *IEEE Transactions on Circuits and Systems for Video Technology*.

Desmond Elliott, Stella Frank, Loïc Barrault, Fethi Bougares, and Lucia Specia. 2017. Findings of the second shared task on multimodal machine translation and multilingual image description. In *Proceedings of the Second Conference on Machine Translation*, pages 215–233.

Kazuhiro Fukui and Atsuto Maki. 2015. Difference subspace and its generalization for subspace-based methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(11):2164–2177.

Kazuhiro Fukui and Osamu Yamaguchi. 2005. Face recognition using multi-viewpoint patterns for robot vision. *Robotics Research, The Eleventh International Symposium, ISRR*, pages 192–201.

MD Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. 2019. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CsUR)*, 51(6):1–36.

Taizo Iijima, Hiroshi Genchi, and Ken-ichi Mori. 1974. A theory of character recognition by pattern matching method. In *Learning systems and intelligent robots*, pages 437–450. Springer.

Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. 2017. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE international conference on computer vision*, pages 1261–1270.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, Lecture Notes in Computer Science, pages 740–755, Cham. Springer International Publishing.

Ken-ichi Maeda. 2010. From the subspace methods to the mutual subspace method. In *Computer Vision*, pages 135–156. Springer.

Nicola Messina, Fabrizio Falchi, Andrea Esuli, and Giuseppe Amato. 2021. Transformer reasoning network for image-text matching and retrieval. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5222–5229. IEEE.

Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2641–2649.

Bryan A. Plummer, Liwei Wang, Christopher M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2017. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *IJCV*, 123(1):74–93.

Leigang Qu, Meng Liu, Da Cao, Liqiang Nie, and Qi Tian. 2020. Context-aware multi-view summarization network for image-text matching. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1047–1055.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

Brigit Schroeder and Subarna Tripathi. 2020. Structured query-based image retrieval using scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 178–179.

Botian Shi, Lei Ji, Pan Lu, Zhendong Niu, and Nan Duan. 2019. Knowledge aware semantic concept expansion for image-text matching. In *IJCAI*, volume 1, page 2.

Lucia Specia, Stella Frank, Khalil Sima'An, and Desmond Elliott. 2016. A shared task on multimodal machine translation and crosslingual image description. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 543–553.

Sijin Wang, Ruiping Wang, Ziwei Yao, Shiguang Shan, and Xilin Chen. 2020. Cross-modal scene graph matching for relationship-aware image-text retrieval. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1508–1517.

Satosi Watanabe and Nikhil Pakvasa. 1973. Subspace method of pattern recognition. In *Proc. 1st. IJCPR*, pages 25–32.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 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.

Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5831–5840.

A Hardware specifications

For all the experiments conducted in this paper, we used three different machines:

1. For fine-tuning and extracting features from UNITER, we used a server machine with an Intel Xeon E5-2630 CPU, and two NVIDIA RTX-2080 (Driver 418.56, CUDA 10.1) GPUs, running Ubuntu 20.04.

2. For extracting features from SGM and running the experiments with UNITER and SGM features, we used a machine with an Intel Core i7-6800K CPU, with one NVIDIA GeForce GTX 1070 (Driver 471.41, CUDA 11.4), running Ubuntu 18.04 on Windows Subsystem for Linux version 2.

3. For extracting and running experiments with CLIP features, we used a node on large cluster equipped with a 16-GB NVIDIA V100 GPU (CUDA 11.3).

However, we highlight that all experiments using the subspace-based methods can be performed using the second machine listed above.

B Results using different subspace dimensions

In Tables 6 to 12, we show the results with varying subspace dimensions for all three models.

C Replication of original models' results

In Tables 13 to 14, we show our reproduction of UNITER and CLIP's results.

		Text retrieval				Image Retrieval			
Feature	Dim	Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
ZS	1	76.00	63.10	80.00	84.90	**48.00**	35.00	51.60	57.40
	5	84.10	74.60	87.20	90.50	44.20	31.60	47.60	53.50
	10	90.00	84.10	92.00	94.00	42.20	29.20	45.50	52.00
	20	**91.60**	86.10	93.30	95.40	41.30	28.10	44.60	51.30
Ft	1	43.70	29.10	47.20	54.80	39.60	28.30	42.50	48.10
	5	71.00	56.80	74.70	81.60	**47.40**	34.50	51.20	56.50
	10	77.40	66.00	80.90	85.40	45.90	33.70	49.20	54.90
	20	**80.80**	69.70	84.50	88.30	44.90	32.40	48.50	53.80
Ft$_{HN}$	1	22.40	12.90	24.20	30.10	13.10	6.10	14.40	18.90
	5	43.50	26.50	47.10	56.80	**28.40**	17.10	31.10	37.10
	10	42.50	23.40	46.70	57.50	27.70	16.50	30.10	36.40
	20	**44.20**	27.70	48.90	56.10	28.10	17.00	30.60	36.70

Table 6: Results with UNITER-subspace on the Flickr30k dataset using SM. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces in SM.

		Text retrieval				Image Retrieval			
Feature	Dim	Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
ZS	1	**76.00**	63.10	80.10	84.90	**75.00**	63.70	78.60	82.70
	5	54.40	0.10	75.80	87.40	39.80	23.70	43.50	52.20
	10	1.80	0.10	0.80	4.40	60.40	43.70	64.70	72.70
	15	1.80	0.10	0.90	4.30	57.20	32.90	64.80	73.90
Ft	1	43.70	29.10	47.20	54.80	43.10	32.30	45.70	51.20
	5	**56.80**	0.40	80.60	89.50	18.10	9.60	19.50	25.30
	10	2.50	0.10	1.10	6.30	50.10	36.00	54.10	60.10
	15	1.90	0.10	1.30	4.30	**53.60**	32.40	60.50	67.90
Ft$_{HN}$	1	22.40	12.90	24.20	30.10	13.60	7.30	14.90	18.70
	5	**56.20**	1.60	79.00	87.90	22.40	14.60	24.00	28.60
	10	1.60	0.10	0.90	3.90	49.30	36.70	52.50	58.50
	15	1.80	0.10	1.20	4.00	**55.30**	42.90	58.90	64.10

Table 7: Results with UNITER-subspace on the Flickr30k dataset using MSM. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces MSM.

		Text retrieval				Image Retrieval			
Feature	Dim	Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
ZS	1	52.80	35.30	57.50	65.60	**24.50**	17.00	26.30	30.00
	5	61.10	43.10	66.50	73.70	23.40	16.10	25.10	28.90
	10	**68.30**	51.30	73.60	79.90	22.50	15.40	24.20	28.00

Table 8: Results with UNITER-subspace on the MSCOCO dataset using SM, using all 5k test images. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces in SM.

		Text retrieval				Image Retrieval			
Feature	Dim	Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
ZS	1	52.80	35.30	57.50	65.70	**38.00**	31.20	40.00	42.80
	5	**58.20**	38.60	63.80	72.20	24.90	15.90	26.90	31.90
	10	50.70	28.20	56.70	67.10	32.80	22.30	35.40	40.70

Table 9: Results with UNITER-subspace on the MSCOCO dataset using MSM, using all 5k test images. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces MSM.

			Text retrieval				Image Retrieval			
Method	Dim	Sim	Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
SM	1	S^g	12.03	4.70	13.20	18.20	31.90	13.08	34.84	47.78
		S^o	17.47	6.10	19.10	27.20	35.04	14.74	38.72	51.68
		S^r	1.27	0.40	1.10	2.30	2.28	0.62	2.28	3.96
		$S^{o,r}$	13.00	5.50	13.40	20.10	35.53	14.74	39.16	52.70
	5	S^g	**40.40**	20.40	45.10	55.70	**39.90**	18.44	44.12	57.14
		S^o	38.20	18.80	42.10	53.70	38.48	17.52	42.70	55.24
		S^r	0.33	0.10	0.40	0.50	1.20	0.28	1.20	2.12
		$S^{o,r}$	33.80	16.00	37.00	48.40	36.51	16.30	40.34	52.90
	10	S^g	**29.23**	13.00	32.10	42.60	**31.42**	13.08	34.86	56.34
		S^o	32.03	14.80	35.40	45.90	30.36	12.62	33.38	45.08
		S^r	0.40	0.00	0.30	0.90	1.19	0.26	1.20	2.12
		$S^{o,r}$	27.03	12.40	29.80	38.90	28.80	11.56	31.58	43.26
MSM	1	S^g	0.63	0.20	0.60	1.10	1.21	0.24	1.18	2.22
		S^o	**17.57**	6.60	18.80	27.30	**31.77**	15.54	34.70	45.06
		S^r	1.23	0.40	1.10	2.20	1.29	0.32	1.36	2.20
		$S^{o,r}$	15.50	5.90	16.20	24.40	31.60	15.30	34.60	44.90
	5	S^g	58.03	37.20	63.60	75.30	46.08	26.40	50.60	61.24
		S^o	**60.53**	40.60	65.70	75.30	**47.21**	27.70	51.82	62.10
		S^r	0.80	0.10	0.90	1.40	1.13	0.20	1.20	2.00
		$S^{o,r}$	20.80	11.10	22.20	29.10	42.00	23.68	46.30	56.02
	10	S^g	**59.03**	40.20	63.60	73.30	**41.71**	23.50	45.72	55.92
		S^o	52.20	31.60	57.10	67.90	41.44	23.26	45.52	55.54
		S^r	0.87	0.10	1.00	1.50	0.97	0.24	0.84	1.82
		$S^{o,r}$	12.00	6.60	13.20	16.20	29.39	14.26	32.38	41.54

Table 10: Results with SGM-subspace on the Flickr30k dataset using SM and MSM. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces in SM and MSM.

			Text retrieval				Image Retrieval			
Method	Dim	Sim	Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
SM	1	S^g	13.10	4.40	14.00	20.80	10.60	0.40	8.50	23.00
		S^o	**26.40**	9.60	27.6	42.00	20.70	7.40	22.40	32.20
		S^r	0.40	0.00	0.40	0.80	0.00	0.00	0.00	0.00
		$S^{o,r}$	16.70	4.80	16.80	28.40	20.70	7.50	22.30	32.20
	5	S^g	**21.20**	7.20	21.20	35.20	19.30	4.20	20.20	33.40
		S^o	21.10	7.70	21.80	33.80	**21.10**	7.50	22.00	33.80
		S^r	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		$S^{o,r}$	16.10	5.20	15.60	27.60	**21.10**	7.70	21.80	33.80
	10	S^g	12.90	5.20	14.00	19.60	17.00	1.80	17.60	31.80
		S^o	**13.30**	5.60	12.80	21.60	20.90	7.80	21.80	33.10
		S^r	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		$S^{o,r}$	10.70	4.40	10.00	17.60	**21.00**	7.70	21.80	33.50
MSM	1	S^g	19.20	6.80	20.00	30.80	21.20	9.10	22.50	32.00
		S^o	**26.10**	9.60	27.60	41.20	**29.20**	13.40	31.40	42.80
		S^r	0.40	0.00	0.40	0.80	0.00	0.00	0.00	0.00
		$S^{o,r}$	19.60	4.80	20.80	33.20	**29.20**	13.40	31.40	42.90
	5	S^g	42.90	24.40	46.40	58.00	34.30	16.70	37.10	49.00
		S^o	**44.90**	24.00	50.40	60.40	**35.10**	17.40	38.50	49.40
		S^r	0.10	0.00	0.00	0.40	0.00	0.00	0.00	0.00
		$S^{o,r}$	17.10	10.00	18.00	23.20	34.50	17.70	37.00	48.60
	10	S^g	**42.90**	22.00	48.40	58.40	**33.70**	16.70	37.40	47.00
		S^o	39.70	20.40	44.80	54.00	32.80	15.10	36.40	46.80
		S^r	0.10	0.00	0.00	0.40	0.20	0.00	0.20	0.30
		$S^{o,r}$	6.40	4.00	6.00	9.20	31.80	14.50	35.40	45.60

Table 11: Results with SGM-subspace on the MSCOCO dataset using SM and MSM, using all 5k test images. Best results for each method are shown in bold. Mean denotes the mean of the R@1, R@5, and R@10, and Dim denotes the dimensions of the subspaces in SM and MSM.

Table 12: Results of our experiments with for CLIP-subspace on FLICKR30K, where the sub-indices G and L indicate the use of global and local features to represent each image and/or caption.

				Text retrieval				Image Retrieval		
Features	Method	Dim	Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
Text$_{\mathrm{G}}$ & Img$_{\mathrm{G}}$	CLIP	-	**90.30**	77.80	95.00	98.10	**76.60**	58.10	82.50	89.40
Text$_{\mathrm{L}}$ & Img$_{\mathrm{G}}$		-	54.70	29.30	60.40	74.50	42.40	24.80	46.40	55.90
Text$_{\mathrm{G}}$ & Img$_{\mathrm{L}}$		-	71.80	55.10	76.10	84.10	57.70	34.20	63.80	75.20
Text$_{\mathrm{G}}$& Img$_{\mathrm{G+L}}$		-	79.10	62.20	84.20	90.90	63.80	40.20	70.50	90.70
Text$_{\mathrm{G+L}}$ & Img$_{\mathrm{G}}$		-	74.40	52.10	81.00	90.00	65.80	45.40	71.90	80.20
Text$_{\mathrm{L}}$ & Img$_{\mathrm{L}}$	SM	1	**47.90**	27.30	52.30	64.10	**36.80**	19.00	40.60	50.70
		5	47.20	21.90	54.30	65.40	35.90	17.40	39.10	51.30
		10	39.60	20.00	43.10	55.70	35.90	17.60	39.00	51.10
	MSM	1	**47.30**	27.00	51.70	63.10	**35.40**	19.90	38.30	47.90
		5	23.40	12.00	24.70	33.60	31.30	14.10	34.70	45.20
		10	26.40	12.60	27.30	39.30	24.10	11.60	25.90	34.80
Text$_{\mathrm{L}}$ & Img$_{\mathrm{G}}$	SM	1	**61.40**	37.40	68.10	78.60	**42.00**	24.60	45.90	55.50
		5	42.10	23.70	45.10	57.50	35.10	17.40	38.20	49.80
		10	39.40	21.40	42.30	54.40	34.90	17.30	38.00	49.50
Text$_{\mathrm{G}}$ & Img$_{\mathrm{L}}$		1	70.40	54.00	74.00	83.20	62.00	40.60	68.20	77.00
		5	**75.70**	59.10	81.00	87.00	**67.70**	46.10	74.50	82.40
		10	72.40	55.20	77.00	85.10	59.30	37.50	64.90	75.50
Text$_{\mathrm{G}}$ & Img$_{\mathrm{G+L}}$		1	77.90	60.90	82.90	89.90	68.20	47.50	74.60	82.60
		5	**83.70**	69.90	87.90	93.30	**74.90**	54.50	81.30	88.80
		10	78.50	61.90	83.60	90.10	66.70	44.40	73.30	82.50
Text$_{\mathrm{G+L}}$ & Img$_{\mathrm{G}}$		1	77.30	57.00	83.80	91.10	63.80	43.40	69.70	78.40
		5	**83.40**	67.00	89.20	93.90	**70.40**	49.90	76.30	84.80
		10	78.50	58.20	85.60	91.60	69.80	49.20	75.90	84.30
Text$_{\mathrm{G+L}}$ & Img$_{\mathrm{G+L}}$	SM	1	64.70	44.70	70.70	78.80	59.80	37.20	65.70	76.40
		5	**70.00**	50.30	75.50	84.20	**61.90**	38.40	68.60	78.70
		10	60.50	40.00	65.80	75.70	61.20	37.60	67.90	78.20
	MSM	1	**63.90**	44.30	69.70	77.80	**61.10**	41.10	66.40	75.90
		5	62.10	38.50	67.50	80.40	56.20	34.80	61.70	72.20
		10	56.40	34.20	62.00	73.00	37.70	21.30	40.80	50.90

		Text retrieval			Image Retrieval		
Dataset	Model	R@1	R@5	R@10	R@1	R@5	R@10
Flickr30k	ZS	80.70	95.70	98.00	66.16	88.40	92.94
	ZS (ours)	80.80	95.70	98.00	66.14	88.36	92.94
	Ft*	-	-	-	-	-	-
	Ft (ours)	76.40	92.00	96.20	63.00	86.62	91.98
	Ft-HN	85.90	97.10	98.80	72.52	92.36	96.08
	Ft-HN (ours)	83.10	95.50	98.50	68.02	89.54	94.54
COCO	ZS*	-	-	-	-	-	-
	ZS (ours)	64.10	87.74	93.30	48.79	76.72	85.84
	Ft*	-	-	-	-	-	-
	Ft (ours)	54.22	81.30	88.86	42.97	72.26	82.17
	Ft-HN	64.40	87.40	93.08	50.33	78.52	87.16
	Ft-HN (ours)	60.64	84.68	91.70	46.42	74.78	84.40

Table 13: Results of our replication of UNITER on the Flickr30k and COCO datasets, where $*$ indicates results not reported by the original paper.

		Text retrieval				Image Retrieval			
Method	Dim	Mean	R@1	R@5	R@10	Mean	R@1	R@5	R@10
Reported	-	-	88.0	98.7	99.4	-	68.7	90.6	95.2
Ours	-	90.6	78.8	94.9	98.2	77.4	58.8	83.5	90.0

Table 14: Results of CLIP retrieval on the Flickr30k dataset. Reported indicates the result reported in the original paper, and Ours indicates our replication. Mean denotes the mean of the R@1, R@5, and R@10.