Reproducibility Study: Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control

Doruk Tarhan
University of Amsterdam
Amsterdam, Netherlands
doruk.tarhan@student.uva.nl

Alvaro de la Maza
University of Amsterdam
Amsterdam, Netherlands
alvaro.de.la.maza@student.uva.nl

Yigit Can Ozkaya
University of Amsterdam
Amsterdam, Netherlands
yigit.ozkaya@student.uva.nl

Abstract

Recent advancements in multimodal retrieval have introduced innovative techniques for integrating textual and visual data efficiently. This reproducibility study examines and extends the findings of *Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control* [9] by replicating its results on MSCOCO and Flickr30k datasets using BLIP and ALBEF models. Our experiments demonstrate the reliability of the D2S projection layer, closely aligning with the original findings, while also revealing slight computational trade-offs when using newly generated embeddings. To test the generalizability of the approach, we apply it to the TextCaps dataset, observing robust performance with BLIP embeddings but limitations with ALBEF, underscoring the significance of encoder selection. In exploring enhancements, we evaluate alternative encoders (BLIP-2 and CLIP) and additional techniques such as overuse penalties. While BLIP-2 offers competitive results, it does not surpass BLIP due to architectural complexities that hinder retrieval-specific alignment. CLIP's simplicity highlights the challenges of general-purpose models in fine-grained retrieval. Furthermore, the introduction of overuse penalties reveals a trade-off between computational cost and retrieval effectiveness. These findings provide actionable insights for optimizing multimodal sparse retrieval frameworks and lay the groundwork for future research to advance their efficiency and adaptability. Our code is available at https://anonymous.4open.science/r/multimodal-lsr-reproduce-4BDF and the embeddings and model checkpoints at 🤗 huggingface.co/ir2multimodal

ACM Reference Format:
Doruk Tarhan, Alvaro de la Maza, and Yigit Can Ozkaya. 2025. Reproducibility Study: Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control. In . University of Amsterdam, Amsterdam, Netherlands, 7 pages.

1 Introduction

Learned Sparse Retrieval (LSR) [2], can be defined as the projection of dense embeddings constructed by transformer-based encoders to sparse lexical vectors that have the size of the vocabulary and are compatible with the traditional inverted index method [9]. After the establishment of BERT [1] as encoder-based transformers, dense retrieval took the first place in state-of-the-art multimodal retrieval due to its ability to represent different modalities, such as images, which led to the establishment of strong pretrained models such as BLIP and ALBEF [4, 5]. However, LSR has different advantages over dense retrieval in terms of efficiency and interpretability, especially in first-stage retrieval, where the task is to define the top candidates over a large set of documents [9]. Learned Sparse Retrieval (LSR) was first introduced in SNRM (Neural Sparse Representation-based Models for First-Stage Retrieval) [16], and later extended by models like LEXLIP and STAIR [15, 17]. These models offer competitive results with dense retrieval but require extensive multi-step training on large text-image datasets. SPLADE [2] advanced this field by introducing expansion control with FLOPS regularization, making it both effective and efficient. However, there remains a significant gap in research on applying LSR methods to multimodal tasks, which is addressed through probabilistic expansion control in the paper *Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control* [9].

In *Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control*, the authors approach the Multimodal LSR problem by training a Language Model projection head on top of pre-trained dense embeddings extracted from models BLIP and ALBEF. The authors focus on improving sparse retrieval methods for multimodal tasks by addressing two key challenges: semantic deviation and high-dimension coactivation [9]. These issues arise when dense embeddings are converted into sparse representations, leading to irrelevant or overly activated terms that reduce retrieval accuracy. The paper introduces a probabilistic expansion control mechanism, which uses Bernoulli parameters to balance the trade-off between sparsity and informativeness with a two-level approach at word and caption levels. The offered approach effectively generates interpretable sparse representations while maintaining strong retrieval performance. The results demonstrate competitive or state-of-the-art metrics on benchmarks like MSCOCO and Flickr30k, highlighting the approach's effectiveness and efficiency.

We aimed to reproduce the findings of the paper *Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control* to validate its results on the datasets and models utilized while also expanding the scope with additional experiments. First, we replicated the Learned Sparse Retrieval (LSR) results using pre-executed embeddings available on the Huggingface platform. Additionally, we extracted dense embeddings from scratch using BLIP and ALBEF models on the MSCOCO and Flickr30k datasets [6, 11], successfully reproducing similar results to those reported in the paper. To evaluate the generalizability of the approach, we introduced a new dataset, TextCaps [13], to examine the model's performance on unseen data.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Our experiments confirmed most of the findings presented in the original paper. We achieved comparable scores using both the preloaded embeddings and those extracted from scratch. However, when testing the models on the TextCaps dataset, we observed a significant performance drop, particularly with the ALBEF model. This indicates that while the frozen embeddings perform well on the original datasets, their generalizability to new datasets remains a limitation.

To further explore the limitations mentioned, we experimented with different encoder models for extracting pre-trained embeddings from the MSCOCO and Flickr30k datasets. Recognizing the direct impact of embedding quality on the semantic deviation problem in the LSR head, we tested BLIP-2 and CLIP encoders. These models were selected for their ability to provide well-aligned text and image representations while maintaining compatibility with the LSR head. Although the results did not exceed the original model's performance, we believe BLIP-2 has potential for improvement, as our experimentation was constrained by limited time. Additionally, we introduced a new regularizer, overuse loss, to address the co-activation problem in the LSR head. This resulted in increased sparsity, as expected while maintaining similar overall performance. We anticipate that further tuning of hyperparameters could enhance the effectiveness of the overuse loss and exceed the original paper's results.

2 Overview of the Paper

The paper addresses two main challenges during the LSR training process: **semantic deviation** and **high-dimension co-activation**. The high-dimension co-activation problem occurs when an excessive number of similar dimensions in the lexical sparse representations of text and image embeddings are activated, leading to the creation of a sub-dense space in the vocabulary [9]. This issue is quantified using FLOPs, which represents a smooth relaxation of the average number of floating-point operations required to compute the score of a document. FLOPs are directly related to retrieval time and serve as a metric for measuring computational efficiency [2].

Semantic deviation. refers to the disparity between the semantic information in the visual or textual query and the semantic meaning of the sparse output terms [9]. It is evaluated using two metrics introduced in the original paper, *Semantic@k* and *Exact@k*, which measure the similarity and exactness of the activated dimensions in the lexical representations of the input captions and the top-k highest weighted output terms.

high-dimension co-activation. primarily impacts the efficiency of retrieval, semantic deviation affects its effectiveness. The primary objective of this paper, and of LSR methods in general, is to strike a balance between these two factors to achieve both efficient and effective retrieval.

2.1 FLOPs vs Performance

The paper explores the trade-off between efficiency and effectiveness in LSR by leveraging FLOPs and probabilistic expansion control. Lower regularization levels generally result in higher FLOPs, which correspond to increased activation across dimensions. While this leads to better retrieval performance due to the broader semantic coverage, it comes at the cost of reduced computational efficiency. FLOPs regularization is introduced to decrease this inefficiency by penalizing excessive co-activation and encouraging sparsity in the representations. This results in a model that balances the retrieval accuracy and efficiency, ensuring scalability for large datasets.

The results from the original paper demonstrate this balance: higher regularization reduces FLOPs significantly, improving retrieval speed, albeit with a slight decrease in performance metrics such as Recall@K and MRR. By optimizing this trade-off, the proposed method achieves competitive results without sacrificing computational efficiency, making it suitable for practical retrieval scenarios.

2.2 Probabilistic Expansion Control

The **probabilistic expansion control** mechanism, implemented through Bernoulli parameters, plays a key role in mitigating the semantic deviation problem by balancing sparsity and informativeness in lexical representations. This mechanism progressively activates terms based on the expansion parameter, and its effectiveness is evaluated under different configurations:

(1) **Expansion = 0 (Zero Expansion):** In this configuration, no new words are activated beyond those present in the original caption. This approach minimizes FLOPs and results in extremely sparse representations. However, it significantly underperforms on metrics such as Recall@K, as it fails to capture broader semantic alignments.
(2) **Expansion = 1 (Full Expansion):** This configuration activates all possible terms, providing dense-like semantic coverage. While it achieves performance metrics comparable to dense embeddings, it incurs very high FLOPs, making it computationally expensive and inefficient.
(3) **Controlled Expansion (Expansion = C):** A middle ground where expansion is gradually increased, selectively activating relevant terms. This ensures a balance between sparsity and semantic alignment, leading to significant improvements in retrieval performance while maintaining lower FLOPs.
(4) **Controlled Expansion with Word-Level Adjustments (Expansion = C + W):** This configuration combines caption-level expansion with word-level fine-tuning for additional flexibility. It achieves performance nearly identical to full expansion but with significantly reduced FLOPs, offering the best trade-off between efficiency and effectiveness.

The results presented in the original paper demonstrate that both **controlled expansion (C)** and **controlled expansion with word-level adjustments (C + W)** provide the desired trade-off. They achieve retrieval performance as strong as full expansion while keeping FLOPs significantly lower. These configurations address both the semantic deviation and high-dimension co-activation challenges, ensuring efficient and effective multimodal retrieval.

3 Experimental Setup

3.1 Datasets

- **MSCOCO and Flickr30k** : Specially benchmarked for text-image retrieval they contain 164k and 158k text-image pairs

Table 1: Comparison of results after reproducing different configurations

Model	MSCOCO (5k)				Flickr30k (1k)			
	R@1 ↑	R@5 ↑	MRR@10 ↑	FLOPS ↓	R@1 ↑	R@5 ↑	MRR@10 ↑	FLOPS ↓
Original paper								
D2S (ALBEF, $\eta = 1e-3$)	49.6	77.7	61.4	18.7	74.2	93.8	82.6	21.7
D2S (ALBEF, $\eta = 1e-5$)	50.7	78.2	62.4	74.2	75.4	94.3	83.6	64.3
D2S (BLIP, $\eta = 1e-3$)	51.8	79.3	63.4	**11.5**	77.1	94.6	84.6	**9.9**
D2S (BLIP, $\eta = 1e-5$)	54.5	**80.6**	65.6	78.4	79.8	95.9	86.7	39.5
Using original dense embeddings								
D2S (ALBEF, $\eta = 1e-3$)	49.5	76.7	61.5	20.7	74.3	92.9	81.9	22.3
D2S (ALBEF, $\eta = 1e-5$)	51.0	78.1	62.3	74.8	76.1	93.3	84.1	66.3
D2S (BLIP, $\eta = 1e-3$)	51.6	79.1	63.5	13.3	77.7	94.9	85.0	11.6
D2S (BLIP, $\eta = 1e-5$)	**54.8**	**80.7**	65.8	79.7	**80.4**	95.9	**87.1**	41.3
Using our BLIP embeddings								
D2S (BLIP, $\eta = 1e-3$)	51.8	79.4	63.6	12.52	75.7	94.0	83.7	20.11
D2S (BLIP, $\eta = 1e-5$)	54.6	**80.6**	**66.4**	80.4	78.8	**96.0**	86.7	42.3

respectively. ALBEF and BLIP, as well as the original D2S projection layer proposed by the authors, were trained on these datasets[6, 11].

- **TextCaps**: Generalizability is tested on this dataset which focuses on image captioning with reading comprehension. It comprises 145k captions for 28k images extracted form OpenImages [13].

3.2 Models

- **BLIP**: BLIP is a vision-language model that uses bootstrapped learning to combine text and picture comprehension. To handle noisy online data, it uses contrastive learning, image-text matching, and caption synthesis, and it uses pseudo-labeling to improve its performance [4].
- **ALBEF**:For multimodal tasks, ALBEF prioritizes textual and visual feature alignment before combining them. It makes use of cross-modal attention in transformer layers for feature fusion and contrastive learning for alignment [5].
- **BLIP-2**: By utilizing lightweight adapters to link visual encoders with large language models, BLIP-2 expands BLIP's architecture. It reduces processing and the requirement for intensive fine-tuning by aligning visual characteristics with frozen LLMs [3].
- **CLIP**: This model is a basic approach that uses contrastive loss to learn a common embedding space for text and pictures. Large-scale noisy datasets are used for training, allowing for significant generalization in zero-shot learning for tasks including multimodal reasoning [12].

Metrics. To compare the different model instances three main different evaluation metrics are used. Recall at k, with k={1,5} (R@1 and R@5), are used to address which proportion of relevant items are retrieved. Mean Reciprocal Rank (MRR) is used to assess the quality of the ranking. In addition to these retrieval metrics FLOPs are calculated to compare the efficiency of the algorithms independently of the hardware used for training.

Training Configuration. The models were trained with the same hyperparameters as the original paper. They are trained on a single A100 GPU from the Snellius Dutch computer cluster [14] with a batch size of 512 for 200 epochs. The temperature τ is set to 0.001.

4 Results and Discussion

The reproducibility study aimed to reproduce the results from the authors of the original paper. The authors provided all the necessary code [7], in addition, the dense embeddings from MSCOCO and Flickr30k using pre-trained BLIP and ALBEF were also provided in a HuggingFace repository [8].

We first reproduced the results using the provided dense embeddings by the authors and only trained the D2S projection layer. The next step was creating our own MSCOCO and Flickr30k dense embedding using a different version of the pre-trained BLIP.

In the last step, we test the generalizability of the authors' approach by testing it on the TextCaps dataset. This is especially interesting as BLIP and ALBEF were trained on MSCOCO and Flickr30k, and therefore even though the authors used these encoders frozen, they were already trained on that data. We use BLIP and ALBEF pre-trained on COCO to obtain our TextCaps dense embedding and later train the D2S projection layer on these embeddings to obtain the sparse embeddings.

4.1 Reproducing LSR-Multimodal

MSCOCO and Flickr30k.: The results, in Table 1, show that D2S with our own BLIP embeddings and a learning rate of $\eta = 1e-5$ slightly outperforms other configurations, achieving the highest values in R@5 (80.6) and MRR@10 (66.4) on MSCOCO and the highest R@5 (96.0) and MRR@10 (86.7) on Flickr30k, though at a higher computational cost (80.4 FLOPS). The results when reproducing using the authors' dense embeddings are similar to the original

Table 2: Performance Metrics from D2S Model under TextCaps dataset

Model	TextCaps (22k)			
	R@1 ↑	R@5 ↑	MRR@10 ↑	FLOPS ↓
D2S(ALBEF,$\eta = 1e-3$)	15.5	22.9	19.2	**23.23**
D2S(ALBEF,$\eta = 1e-5$)	21.7	30.1	30.1	72.81
D2S(BLIP,$\eta = 1e-3$)	48.0	57.4	57.4	33.22
D2S(BLIP,$\eta = 1e-5$)	**50.8**	**60.5**	**55.7**	86.17

ones at a slightly higher computational cost (e.g. R@1 for ALBEF with $\eta = 1e-3$ is 49.6 in the original and 49.5 in the reproduced).

TextCaps.: While reproducing the method on a new dataset we can observe , in Table 2, that training the D2S layer with dense embeddings computed with BLIP gives a decent performance with R@1 of 50.8 similar to the original R@1 of 54.5 obtained on MSCOCO (both with $\eta = 1e-5$).

4.1.1 Reproducibility Discussion. Reproducing the results on the original datasets, Flickr30k and MSCOCO, showed that our solution closely matches the results the authors described. The replicated findings displayed minor differences, such as a tiny change in R@1 for BLIP with $\eta = 1e-5$ (51.8 in the original study versus 51.6 recreated), while using the original dense embeddings from the HuggingFace repository. Notably, we obtained somewhat better performance at a greater computational cost when we used our own BLIP embeddings, addressing that our embeddings were slightly better but less efficient. For example, on MSCOCO, the D2S model with BLIP embeddings and $\eta = 1e-5$ produced the greatest R@5 (80.6) and MRR@10 (66.4), whereas on Flickr30k, it produced the highest R@5 (96.0) and MRR@10 (86.7). These findings highlight the D2S projection layer is reproducible on the same datasets using the code and dense embeddings provided by the authors in the paper.

4.1.2 Generalizability Discussion. To assess the generalizability of the D2S framework, we evaluated its performance on the TextCaps dataset. The findings indicated a satisfactory transfer of capabilities when utilizing BLIP embeddings but unsuccessfully when using embedding computed with the ALBEF model. The D2S model, which was trained using BLIP embeddings with a learning rate of $\eta = 1e-5$, achieved a Recall at 1 (R@1) score of 50.8, which is comparable to the original R@1 score of 54.5 reported for the MSCOCO dataset. Despite the inherent differences in the characteristics of the datasets, the D2S approach maintained competitive performance, with R@5 and Mean Reciprocal Rank at 10 (MRR@10) scores of 60.5 and 55.7, respectively. However, the FLOPs indicate that training the D2S model on this new dataset is more computationally expensive than with the original datasets. While using embeddings computed with ALBEF the results are far behind the performance of the framework in any configuration. These imply that ALBEF fails to generalize well when used on a different dataset as a pre-trained model. In general, these results imply that the D2S methodology applies to datasets featuring textual captions beyond those initially examined only when using embeddings from BLIP.

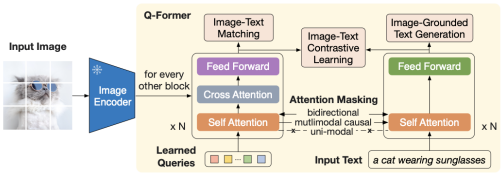

Figure 1: BLIP2 Architecture [3]

4.2 Extension of Encoders for Dense Vector

In the original paper, it was explicitly stated that the quality of the frozen encoders serves as a direct limitation to the final results produced by the LSR head. To further investigate this issue, we extracted dense embeddings from different pretrained models and compared the results to those obtained from the original models, BLIP and ALBEF, on the same datasets used in the original study, MSCOCO and Flickr30k [6, 11]. However, selecting new encoders posed significant challenges, as many state-of-the-art (SOTA) multimodal models are designed primarily for generative tasks, utilizing visual encoders and text decoders. These models do not align image and text embeddings into the same feature space, which was a requirement for our study.

To address this, we selected CLIP [12] and BLIP-2 [3], as both models are trained to align text and image embeddings in the same feature space using contrastive learning. This alignment strategy makes them suitable alternatives to the original BLIP and ALBEF models.

4.2.1 CLIP. One of the first models designed to align image and text embeddings into a shared feature space through contrastive learning was CLIP . It achieves this by training on a large dataset of image-caption pairs to maximize the similarity between matching pairs and minimize it for mismatched ones. This design enables CLIP to perform image-text alignment tasks effectively, making it a strong candidate for this study. Additionally, its straightforward architecture allows efficient computation, aligning well with the requirements of the LSR pipeline.

4.2.2 BLIP-2. This model extends the architecture of BLIP by introducing lightweight adapters to bridge vision encoders and large language models (LLMs) . While BLIP-2's primary purpose is multimodal generation, its training includes contrastive learning to align image and text embeddings into a common feature space, making it suitable for retrieval tasks like LSR.1

The architecture of BLIP-2 differs significantly from BLIP in that the image embeddings generated by its vision encoder are processed through a transformer block with cross-attention layers. This block uses 32 pretrained query embeddings as inputs, each attending to distinct aspects of the image. The output consists of 32 distinct image embeddings, which are compared to the corresponding text embedding during training. To train the Image-Text Contrastive (ITC) head, BLIP-2 measures the cosine similarity between each of the 32 image embeddings and the corresponding text embedding, selecting the highest similarity for loss calculation. However, our approach required a single image embedding. To address this, we experimented with two strategies for combining the 32 embeddings:

Table 3: Comparison of new D2S Models with New Encoders

Model	MSCOCO (5k)				Flickr30k (1k)			
	R@1 ↑	R@5 ↑	MRR@10 ↑	FLOPS ↓	R@1 ↑	R@5 ↑	MRR@10 ↑	FLOPS ↓
Original Results								
D2S (BLIP, $\eta = 1e-3$)	51.8†	79.3†	63.4†	**11.5**	77.1†	94.6†	84.6†	**9.9**
D2S (BLIP, $\eta = 1e-5$)	**54.5**†	**80.6**†	**65.6**†	78.4	**79.8**†	**95.9**†	**86.7**†	39.5
New Encoder Trials								
D2S (BLIP2avg, $\eta = 1e-3$)	45.9	73.3	57.5	20.9	70.2	91.9	79.5	26.7
D2S (BLIP2max, $\eta = 1e-3$)	50.5	76.8	61.7	**17.3**	74.0	92.0	81.7	**20.0**
D2S (BLIP2avg, $\eta = 1e-5$)	47.3	73.7	58.6	118.6	70.8	**92.2**	80.0	87.4
D2S (BLIP2max, $\eta = 1e-5$)	**51.9**	**77.5**	**62.8**	100.3	**74.9**	91.8	**82.3**	79.5
D2S (CLIP, $\eta = 1e-3$)	-	-	-	-	55.0	81.8	66.3	41.8
D2S (CLIP, $\eta = 1e-5$)	-	-	-	-	55.3	81.2	66.5	64.5

- **Max Cosine Similarity Selection**: Selecting the single embedding with the highest cosine similarity to the text embedding.
- **Average Pooling**: Averaging all 32 embeddings into a single representation.

Initial experiments on a small set of images showed that average pooling sometimes produced higher cosine similarities than selecting a single embedding. As a result, we extracted embeddings using both methods to evaluate their performance comprehensively.

4.3 Results and Discussion for New Encoders

4.3.1 Results. The results of using new encoders, BLIP-2 and CLIP, compared to the original models BLIP and ALBEF, are summarized in Table 3. While we could not achieve better performance than the original results, the experiments provide valuable insights into the potential and limitations of these new encoders for Learned Sparse Retrieval (LSR).

BLIP-2 Performance: BLIP-2 embeddings were tested using two pooling strategies: **max cosine similarity (BLIP2Max)** and **average pooling (BLIP2Avg)**. BLIP2Max achieved the best results among new encoders, with MSCOCO scores of R@1 = 51.9 and MRR@10 = 62.8 at $\eta = 1e-5$, coming closer to the original BLIP embeddings (R@1 = 54.5, MRR@10 = 65.6). On Flickr30k, BLIP2Max achieved R@1 = 74.9 and MRR@10 = 82.3, showing competitive performance but still lower than the original BLIP scores. BLIP2Avg, on the other hand, consistently underperformed, achieving R@1 = 47.3 on MSCOCO at $\eta = 1e-5$, with higher FLOPs, indicating less efficiency.

The results suggest that BLIP2Max benefits from selecting the most relevant embedding from its 32 output embeddings, but BLIP-2's design primarily optimized for generative tasks limits its performance in retrieval settings. Further optimization, such as training a projection head to consolidate the 32 embeddings into a single representation, could improve its performance and potentially exceed the original BLIP results.

CLIP Performance: CLIP's embeddings, designed for general-purpose zero-shot tasks, struggled to match the performance of models tailored for LSR. On Flickr30k, CLIP achieved moderate results, with R@1 = 55.3 at $\eta = 1e-5$, significantly lower than the original BLIP embeddings (R@1 = 79.8). While CLIP provides computational efficiency, its training objective does not prioritize fine-grained alignment between text and images, making it less effective for tasks requiring precise retrieval.

Key Insights: The results confirm the importance of specialized encoders like BLIP for high retrieval performance. BLIP2Max shows potential for improvement, especially with further fine-tuning. CLIP, while effective in general multimodal tasks, remains less suited for the fine-grained alignment required in LSR pipelines.

4.3.2 Discussion. The experiments with BLIP-2 and CLIP as alternative encoders for dense embeddings revealed valuable insights into their capabilities and limitations. While BLIP-2 embeddings, particularly with the max cosine similarity pooling strategy (BLIP2Max), demonstrated competitive performance on MSCOCO and Flickr30k, they did not surpass the results obtained with the original BLIP model. The best results for BLIP2Max on MSCOCO were R@1 = 51.9 and MRR@10 = 62.8, while on Flickr30k, R@1 reached 74.9 with MRR@10 at 82.3. These metrics, though close, fell short of BLIP's original scores, even though they had higher FLOPs. On the other hand, CLIP, despite its simplicity and zero-shot generalization capabilities, struggled to match the performance of both BLIP and BLIP-2, particularly on fine-grained retrieval tasks, with Flickr30k R@1 only reaching 55.3.

The results highlight several challenges and opportunities for improvement. The results for CLIP were expected as it is the most primitive model for this task. Furthermore, BLIP-2's architecture, designed primarily for generative tasks, introduces complexities in aligning embeddings for retrieval. The model's reliance on 32 output embeddings, each attending to distinct aspects of the image, complicates the extraction of a single, representative embedding. Although our pooling strategies (average pooling and max cosine similarity) provided a workaround, they are not fully optimized for retrieval tasks. Developing a dedicated projection head to combine these 32 embeddings into a unified representation could enhance alignment quality and improve performance.

The findings also underscore the importance of selecting encoders aligned with the specific objectives of the LSR framework.

Table 4: Performance metrics for D2S with overuse penalty on Flickr30k. The baseline model is included for reference.

Model	R@1 ↑	R@5 ↑	MRR@10 ↑	FLOPs ↓
D2S (BLIP, $\eta = 1e-3$)	77.1	94.6	84.6	**9.9**
Loss with overuse penalty $\lambda = 0.1$				
D2S (BLIP, $\eta = 1e-3$, *batch size* = 64)	73.2	92.8	81.6	3.4
D2S (BLIP, $\eta = 1e-3$, *batch size* = 512)	77.6	95.0	86.1	8.1
D2S (BLIP, $\eta = 1e-3$, *batch size* = 2048)	79.1	**95.5**	86.1	27.9
Loss with overuse penalty $\lambda = 0.001$				
D2S (BLIP, $\eta = 1e-3$, *batch size* = 64)	73.9	93.1	82.2	3.8
D2S (BLIP, $\eta = 1e-3$, *batch size* = 512)	77.7	94.7	**85.0**	10.4
D2S (BLIP, $\eta = 1e-3$, *batch size* = 2048)	**79.1**	95.1	86.1	40.8

BLIP-2's controlled architecture and CLIP's generalization capabilities offer unique strengths, but their performance in retrieval tasks depends on adaptations that align their embeddings more closely with the requirements of sparse retrieval.

4.4 Extension of Regularization Penalty for Sparse Vector

The overuse penalty [10] is an extension applied to the loss function to mitigate the high-dimension co-activation problem. If we consider the overuse penalty as a specific regularisation term, it serves to improve the sparsity of the representations by discouraging the overactivation of dimensions in the sparse lexical vectors. Specifically, the overuse penalty term is added to the loss function, as follows:

$$L = \ell_{t2i} + \ell_{i2t} + \lambda_I \ell^I_{\text{overuse}} + \lambda_T \ell^T_{\text{overuse}}$$

Here, ℓ_{t2i} and ℓ_{i2t} are the standard bidirectional alignment losses for text-to-image and image-to-text tasks, while ℓ^I_{overuse} and ℓ^T_{overuse} are the overuse penalty terms for the encoded vectors from the text and image encoders, respectively. λ_I and λ_T control the weight of the overuse penalty for each modality.

The overuse penalty for each vector is defined as:

$$\ell_{\text{overuse}} = V \sum_{j=1}^{V} \frac{\bar{s}_{\cdot,j}}{\sum_{k=1}^{V} \bar{s}_{\cdot,k}} \cdot \bar{s}^2_{\cdot,j} - \left(NV \sum_{j=1}^{V} \left(\sum_{i=1}^{N} \frac{s_{i,j}}{N} \right)^3 \Big/ \sum_{j=1}^{V} \sum_{i=1}^{N} s_{i,j} \right)$$

where $s_{i,j}$ is the weight of the j-th dimension in the sparse vector for the i-th instance, $\bar{s}_{\cdot,j}$ is the average weight for the j-th dimension across all instances, V is the vocabulary size, and N is the batch size.

To investigate the impact of the overuse penalty, experiments were conducted with varying λ values ($\lambda = 0.1$ and $\lambda = 0.001$) and different batch sizes (64, 512, and 2048). These configurations were evaluated using the Flickr30k dataset with BLIP ($\eta = 1e-3$) as the backbone.

4.5 Results and Discussion for Overuse Penalty

4.5.1 Results. The table 4 illustrates that for smaller batch sizes (e.g., 64 or 512), the overuse penalty decreases FLOPs compared to the baseline model, but the impact on metrics remains minimal. For larger batch sizes (e.g., 2048), there is a slight improvement. For instance, with $\lambda = 0.1$, the model achieves the highest R@1 (79.1) and R@5 (95.5) when using a batch size of 2048. This improvement, however, comes at the cost of increased FLOPs due to the computational demands of larger batches.

4.5.2 Discussion. For overuse penalty, when we look at the results (Table 4), we concluded that it slightly decreases the FLOPs but has no significant impact on other metrics. The choice of λ does not appear to affect evaluation metrics. However, it effectively reduces FLOPs, as seen in smaller batch size configurations.

When larger batch sizes are used, the metrics show a slight improvement, particularly with $\lambda = 0.1$. For instance, with a batch size of 2048 and $\lambda = 0.1$, R@1 increases to 79.1, which is higher than the baseline. This enhancement, however, comes at the cost of increased FLOPs due to the computational demands of larger batches. These results suggest that the overuse penalty can be leveraged to reduce computational cost (FLOPs) while maintaining retrieval effectiveness, but the trade-offs between λ, batch size, and computational cost must be carefully balanced.

5 Conclusion

This reproducibility study successfully validated most of the key findings from the original paper *Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control*. By replicating the results on MSCOCO and Flickr30k, we confirmed the robustness of the proposed D2S projection layer using both pre-executed embeddings and embeddings extracted from scratch. Minor deviations in metrics highlighted the potential for variability in computational setups, but the overall trends aligned closely with the original findings, underscoring the reliability of the method.

In our exploration of generalizability, we tested the framework on a novel dataset, TextCaps, which provided insights into the adaptability of the D2S layer. While BLIP embeddings demonstrated satisfactory transfer capabilities, ALBEF struggled to maintain performance, highlighting the importance of encoder quality and dataset characteristics in extending the method to new domains.

Our experiments with new encoders, BLIP-2 and CLIP, provided valuable perspectives on expanding the framework. BLIP-2, particularly with the max cosine similarity pooling strategy, showed competitive performance, though it could not surpass the original BLIP embeddings. The results suggest that further fine-tuning, such as designing a projection head to consolidate its multiple embeddings, may unlock its full potential. On the other hand, CLIP's zero-shot capabilities fell short in this fine-grained retrieval task, emphasizing the need for task-specific adaptations in general-purpose models.

Overall, this study highlights the robustness and flexibility of the probabilistic expansion control framework while providing actionable insights for its enhancement. By addressing the identified limitations, future research can further bridge the gap between dense and sparse retrieval, advancing the applicability of LSR methods in multimodal retrieval.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805* (2019).

[2] Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. SPLADE: Sparse Lexical and Expansion Model for First Stage Ranking. *arXiv preprint arXiv:2107.05720* (2021).

[3] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. arXiv:2301.12597 [cs.CV] https://arxiv.org/abs/2301.12597

[4] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. arXiv:2201.12086 [cs.CV] https://arxiv.org/abs/2201.12086

[5] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. 2021. Align before Fuse: Vision and Language Representation Learning with Momentum Distillation. arXiv:2107.07651 [cs.CV] https://arxiv.org/abs/2107.07651

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft COCO: Common Objects in Context. arXiv:1405.0312 [cs.CV] https://arxiv.org/abs/1405.0312

[7] Thong Nguyen. 2024. GitHub Repository: lsr-multimodal. https://github.com/thongnt99/lsr-multimodal Accessed: 2024-12-21.

[8] Thong Nguyen. 2024. Hugging Face Repository: lsr42. https://huggingface.co/lsr42 Accessed: 2024-12-21.

[9] Thong Nguyen, Mariya Hendriksen, Andrew Yates, and Maarten de Rijke. 2024. Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control. arXiv:2402.17535 [cs.IR] https://arxiv.org/abs/2402.17535

[10] Thong Nguyen, Sean MacAvaney, and Andrew Yates. 2023. A unified framework for learned sparse retrieval. In *European Conference on Information Retrieval*. Springer, 101–116.

[11] Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2016. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. arXiv:1505.04870 [cs.CV] https://arxiv.org/abs/1505.04870

[12] 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. arXiv:2103.00020 [cs.CV] https://arxiv.org/abs/2103.00020

[13] Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. 2020. TextCaps: a Dataset for Image Captioning with Reading Comprehension. arXiv:2003.12462 [cs.CV] https://arxiv.org/abs/2003.12462

[14] SURF. 2024. Snellius: The National Supercomputer. https://www.surf.nl/en/services/snellius-the-national-supercomputer Accessed: 2024-12-21.

[15] Kai Tang, Ye Liu, Chenyan Xiong, and Jimmy Lin. 2020. LEXLIP: Learned Sparse Lexical Representations for First-Stage Ranking. *Proceedings of the ACM International Conference on Research and Development in Information Retrieval (SIGIR)* (2020).

[16] Hamed Zamani, Mostafa Dehghani, W Bruce Croft, Erik Learned-Miller, and Jaap Kamps. 2018. Neural Sparse Representation-based Models for First-Stage Retrieval. *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)* (2018).

[17] Tiancheng Zhao, Xiaopeng Lu, and Kyusong Lee. 2020. STAIR: Sparse Transformer Matching Retrieval for Open-Domain Question Answering. *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)* (2020).