Subsampling of Frequent Words in Text for Pre-training a Vision-Language Model

Mingliang Liang
Radboud University
Nijmegen, Netherlands
m.liang@cs.ru.nl

Martha Larson
Radboud University
Nijmegen, Netherlands
m.larson@cs.ru.nl

ABSTRACT

In this paper, we introduce Subsampling of frequent Words for Contrastive Language-Image Pre-training (SW-CLIP), a novel approach for the training Vision-Language Models (VLMs). SW-CLIP uses frequency-based subsampling of words that has been previously proposed to train skip-gram models in natural language processing and applies it to the textual training data of VLMs. We report on experiments that demonstrate the ability of frequency-based subsampling to speed up training and also to deliver a substantial improvement in accuracy in a number of downstream zero-shot (i.e., transfer) classification tasks. We notice that the classification test sets on which SW-CLIP seems to be particularly effective are those in which the labels of the classes occur infrequently as words in the training data, and thus have a high probability of being retained during frequency-based subsampling of the model training data. Overall, the advantages of SW-CLIP demonstrated in this paper serves to motivated further future work in text subsampling for the training of VLMs. Our code and pre-trained weights are available at https://github.com/Anastasiais-ml/sw_clip.git

CCS CONCEPTS

• **Information systems** → **Multimedia and multimodal retrieval**.

KEYWORDS

Vision-language model; subsampling; frequent words; zero-shot image Classification.

ACM Reference Format:
Mingliang Liang and Martha Larson. 2023. Subsampling of Frequent Words in Text for Pre-training a Vision-Language Model. In *Proceedings of the 1st Workshop on Large Generative Models Meet Multimodal Applications (LGM3A '23), November 2, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3607827.3616843

1 INTRODUCTION

Vision-language Models (VLMs), which learn visual-semantic embedding from large-scale natural language supervision, show strong zero-shot transferability [12, 13, 18, 21]. Pre-trained VLMs have provided representations generated by image and text encoders to obtain new state-of-the-art performance on different tasks, such text-to-image generation [22, 23]. In this research, we build upon CLIP, a pioneering VLM model, that captures the connection between visual content and textual descriptions, with a particular focus on advancing zero-shot classification performance.

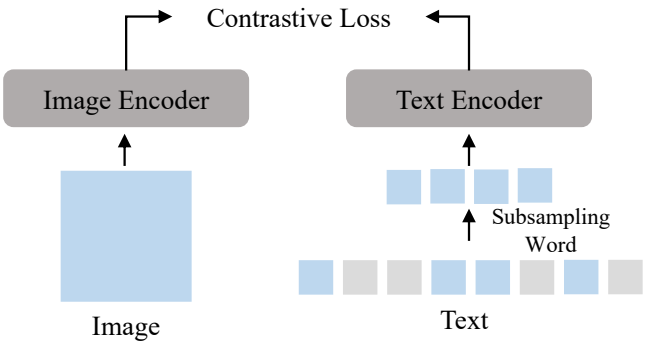

Figure 1: The architecture of SW-CLIP involves subsampling the frequent words from the text during the pre-training phase and then training the model using contrastive loss

In this paper, we propose Subsampling of frequent Words for Contrastive Language-Image Pre-training (SW-CLIP). The idea is inspired by the use of subsampling of frequent words in the text domain in skip-gram language models [17]. Subsampling of frequent words balances the ratio of high frequency and low frequency words. In [17], it is shown that subsampling frequent words can speed up training and substantially improve the ability the model to represent infrequent words. The novelty of our SW-CLIP approach is moving this method from a text model to a VLM and demonstrating that it yields benefits there.

When applying SW-CLIP, we remove more than half of the running words in the training text. The subsampling makes the training process more efficient since it reduces the computational burden associated with processing and updating the representations of these words. The SW-CLIP is also motivated by the idea that subsampling will make the model less biased towards frequent words and allow it to focus on capturing, faster and more accurately, the relationship between images and the infrequent words.

Recently, [13] has proposed Fast Language-Image Pre-training (FLIP), a method that randomly masks a large portion of patches in the training images when pre-training a VLM. This work is related to our SW-CLIP because it studies subsampling for VLMs. FLIP can speed up training up to 3.7 times, while maintaining the same

LGM3A '23, November 2, 2023, Ottawa, ON, Canada.

ACM ISBN 979-8-4007-0283-9/23/11.
https://doi.org/10.1145/3607827.3616843

level of performance. This substantial speedup demonstrates the effectiveness of the masking method in training vision-language models.

Interestingly, the authors of [13] found that word subsampling deteriorates performance. However, upon closer consideration we conclude that their experiment should not be taken as conclusive evidence that word subsampling does not work. Closer consideration of [13] reveals that the experiment actually combined image subsampling with text subsampling and did not test text subsampling alone. Further, the number of pre-training epochs is not kept constant. Our work is motivated by the idea that with more systematic testing the performance deterioration can be addressed and the benefits of word sub-sampling for VLMs can be realized.

SW-CLIP is also different from FLIP in that it adopts frequency-based masking instead of random masking, which can be considered a more intelligent approach. Frequency-based masking selectively masks words that frequently appear in the training dataset, such as common articles like "the," "a,", prepositions like "of." and nouns like "person". SW-CLIP aims to build on the success of [17], employing a similar intelligent subsampling strategy can be employed to pre-train VLMs. However, we note that random masking is also interesting, and we will return to comment on this topic at the end of this paper.

With this paper, we make two contributions:

- We show that SW-CLIP frequency-based word subsampling can speed up training of a VLM.
- We show that SW-CLIP can also improve the accuracy of the model with experiments on 26 downstream classification tasks.

The larger aim of this paper is to open the idea that word-subsampling in the training data, if properly realized, can improve VLMs.

2 RELATED WORK

Learn From Natural Language Supervision Recently, visual-semantic models learned from large-scale natural language supervision, such as CLIP, ALIGN, LiT and FLIP [12, 13, 21, 31], have become popular due to their excellent zero-shot transfer capabilities. These models are pre-trained by contrastive learning, which pushes positive image-text pairs closer to each other and separates negative image-text pairs. OpenCLIP is an open source implementation of OpenAI's CLIP [11, 21], which is trained on a large-scacle public LAION dataset [24]. The results of their experiments clearly show that observing a larger number of samples in a larger data set produces better results than repeating viewing the same number of samples in a smaller data set. Consequently, it becomes evident that focusing on duplicate data does not contribute to performance enhancement for the model; instead, it incurs additional computational costs [11, 13].

Image and Text Masking Fast Language-Image Pre-training (FLIP) randomly removes 50% or 75% of the image patches in the image training data. Surprisingly, this approach not only speeds up the training time but also attains better accuracy than unmasked models [13]. As previously mentioned, the paper also explores the random masking and removal of text. However, in the text subsampling experiment, in which 50% of the text is removed, a 2.2% decrease was observed in zero-shot classification accuracy on the ImageNet dataset [6]. The strategy of random masks could potentially remove words that are important for learning the relationship between images and text, so on one hand this result is not surprising. In our paper, the investigation of text subsampling is more systematic than in [13] and we conclude that the text subsampling results reported in [13] should not be taken as grounds to abandon the idea of text subsampling for VLMs.

Subsampling of Frequent Words Skip-gram models are a type of language modeling approach that aims to learn high-quality word vector representations by predicting words within a specified context window surrounding a given target word [16]. This technique has been widely used in natural language processing and has proven to be effective in capturing semantic relationships between words. The original skip-gram model was extended to incorporate subsampling of frequent words during training [17], leading to a notable increase in training speed and enhancing the accuracy of word representations, particularly for less frequent words. It was demonstrated that after training on millions of examples, the representation of frequent words remains stable [17]. We show that that subsampling frequent words is also effective to speedup training and significantly improve the representation of infrequent words in VLMs.

3 METHOD

In this section, we present our SW-CLIP approach, which subsamples of frequent words of the training text when we are pre-training a VLM. Subsampling of frequent words allows us to use shorter text lengths in the text encoder, which helps speed up pre-training at each steps. Moreover, by reducing the proportion of frequent words, VLMs can learn image and text representations more rapidly compared to models that do not employ subsampling throughout the entire training duration.

Subsampling of Frequent Words Subsampling of frequent words was introduced by [17] in the natural language processing (NLP) domain in order to accelerate the learning process for word representations. The goal was to address the issue of imbalance between frequent and infrequent words. In our work, we use the formula (Formula 1) introduced in [17] in order to to determine the probability of keeping a word during training.

$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}} \tag{1}$$

Here, $f(w_i)$ is the frequency of word w_i and t is the threshold used to control the probability of a word being discarded [17]. Subsampling according to Equation 1 subsamples words with frequencies greater than t and at the same time maintains the ranking of the words by frequency. During subsampling, rare words have a higher probability to be retained than frequent words.

Contrastive Learning In VLMs, contrastive learning is used to learn representations of images and text that are output by image and text encoders [8, 12, 13, 18, 21]. Specifically, CLIP linearly projects and normalises representations of image and text into a shared embedding space. Noise-Contrastive Estimation (InfoNCE)

loss [27] with temperature parameter τ [3, 21] is shown in Equation 2.

$$\mathcal{L}_{\text{InfoNCE}} = -\log\left(\frac{\exp(\text{sim}(I_i, T_i)/\tau)}{\sum_{j=1}^{N}\exp(\text{sim}(I_i, T_j)/\tau)}\right) \quad (2)$$

Here, *sim* is the similarity score between image I_i and text T_i. InfoNCE is used to push positive image-text pairs closer together and pull negative image-text pairs further apart in order to drive the model to learn the relationship between images and text.

After pre-training the model by subsampling frequent words, we fine-tune the model for one epoch without subsampling with the aim of eliminating the distribution gap that could have been caused by subsampling. The distribution gap would be expected to impact the words with the highest probability of being removed during the subsampling process. For low-frequency words with frequencies less than t, the subsampling probability is 0.

4 EXPERIMENTS

4.1 Implementation Details

In this section, we present the details of the training and testing. We train our model with small visual models (ResNet50, ViT-B/16) and a small dataset (CC3M) to evaluate our method, which allows us to make the most of our limited computing resources.

Dataset. We utilize Conceptual Captions 3M (CC3M), which extracts and filters image-caption annotations from the internet [26] to pre-train our model. In this dataset, each image has an associated text. Because a portion of the URLs included in the training set have expired or are no longer accessible, so we have successfully downloaded 2.7M data items.

Architecture We follow the method of the OpenCLIP implementation [11]. For image encoder, we use ResNet-50 (RN50) [10] and ViT-B/16 [7] architectures, which contains over 23M and 86M parameters. The input size of the images is 224. For the text encoder, we use a Transformer-based model [28] that contains 53M parameters and uses byte-pair encoding with a 49K token vocabulary. The maximum context length is 32 when training with full-text and 16 when training with subsampled text. We calculate the cosine similarity between embeddings of images and text that have the same dimensionality and are projected by a linear layer follow the class token embedding. We use the InfoNCE loss [27] with a learnable temperature parameter τ [3, 21], which scales the cosine similarity of the embeddings to pre-train the model.

Training We use the same setup as OpenCLIP [11] to pre-train our model. The details of pre-training configuration is shown in Table 1.

Zero-shot transfer. Following work on CLIP, SLIP, and FLIP [13, 18, 21], we use a variety of classification benchmarks to evaluate SW-CLIP's zero-shot transfer capability. These benchmarks test the model's ability to perform image classification on classes it has not been explicitly trained on. The benchmarks include ImageNet1K [6], which is a well-known dataset commonly used in computer vision for image classification tasks. It includes 50k validation samples and 1K classes.

For the text embeddings, we compute the average of the caption embeddings for each class across the prompt templates, which are provided by CLIP [21] and SLIP [18]. We evaluate our model using the standard metric of top-1 accuracy for ImageNet1K zero-shot classification tasks, which measures the percentage of cases in which the most likely predicted category is correct. The other datasets are evaluated slightly differently: we follow the same settings as SLIP (refer to SLIP for details) [18].

To measure the speedup time between SW-CLIP and CLIP, we compare their respective training times on a common hardware setup consisting of 8 RTX A5000 GPUs. The speedup is calculated as the ratio of the training time for CLIP to the training time for SW-CLIP. This comparison allows us to quantify the efficiency gain achieved by SW-CLIP in terms of training time reduction.

config	value
optimizer	AdamW [15]
learning rate	1e-3
weight decay	0.1
optimizer momentum	$\beta 1, \beta 2 = 0.9, 0.999$ [2]
learning rate schedule	cosine decay [14]
warmup steps	10k
epoch	30
t	10^{-4}
numerical precision	Automatic mixed precision
augmentation	RandomResizedCrop

Table 1: The details of pre-training setup.

method	text mask	text len	time	ImageNet1K
CLIP	original, 100.00%	32	1.00×	16.9%
SW-CLIP	SW, 42.30%	16	0.86×	17.2%

Table 2: Comparison of SW-CLIP and CLIP for zero-shot classification on ImageNet1K. The only difference between them is whether or not the text is subsampled during pre-training. The backbone of image encoder is RN50, and the model pre-trained on CC3M for 30 epochs. The first column specifies the method. The second column is the percentage of the words remaining in the text. The third column is the text length. The fourth column is the time of training the model: the training time of standard CLIP is defined 1.00×. The fifth column shows the top-1 accuracy of zero-shot classification on ImageNet1K .[6].

4.2 Evaluation

Efficiency In our experiment, we use RN50 as the image encoder and the Transformer-base as the text encoder. Table 2 shows that SW-CLIP reduces the computational cost of the text encoder by 50% without compromising performance. This reduction is achieved through the decrease in text length. In our experiment, benefiting from subsampling, SW-CLIP can reduce training time by a total of approximately 14% based on a rough estimation.

No-subsampling tuning Table 3 presents results regarding the final tuning stage that uses the full text that has not been subsampled. We fine-tuned the model for one epoch with the full text (i.e., without subsampling) using the same settings as for pre-training except for the warmup steps, which was 10% of the total steps. With

	mask 42.3%
baseline	16.8%
+no-subsampling tuning	17.2%
+early stop	17.7%

Table 3: No-subsampling tuning. The first row represents the performance of the model without any no-subsampling tuning. In second row, we observe the performance after applying no-subsampling tuning on 29th epoch. The third row show that if we fine-tune the model on an earlier epoch (20th), we can achieve a better performance. We report the performance of the zero-shot classification on ImageNet1k.

	t	before	after
text length	10^{-4}	10.31 ± 4.7	4.26 ± 2.66

Table 4: After subsampling, the length of the text become shorter.

our subsampling approach, the model pre-trained on CC3M obtained 15.8% zero-shot ImageNet1K classification on the 29th epoch. Table 3 show the results fine-tuning the model on the unmasked text for the final epoch, SW-CLIP obtained a better performance than CLIP, 17.2%, on zero-shot ImageNet1K classification.

Early stopping We observed that the SW-CLIP model, pre-trained on RN50 as an image encoder, tended to overfit sooner compared to the CLIP model pre-trained on the full-text. Specifically, when pre-training the models on the CC3M dataset, the SWCLIP model trained on RN50 achieved maximum performance (16.8%) on the ImageNet1K dataset within only 20 epochs, while the CLIP model, which pre-trained the full-text, continued to improve throughout the training process. Furthermore, fine-tuning the SW-CLIP model for 1 additional epoch based on the 20th epoch obtained even better performance (17.7%) compared to the model pre-trained on the full-text, which required 30 epochs in the Table 2. The SW-CLIP, with its specific pre-training approach and shorter training and fine-tuning period, outperforms the CLIP model trained on the full-text.

4.3 Masking Analysis

Table 4 shows the average length of text before and after subsampling are 10.31 ± 4.7 and 4.26 ± 2.6. About 67.7% of words are removed from the text, allowing SW-CLIP to use a shorter token length.

Table 5 presents the potential text resulting from the subsampling technique. In each epoch, the text varies as words are subjected to the possibility of retention or removal, guided by their frequency. This particular approach serves two primary purposes: enhancing text diversity and mitigating the risk of overfitting to frequent words.

4.4 Models

To better understand the performance of SW-CLIP, we compare the zero-shot accuracy of SW-CLIP and CLIP in the ImageNet1K classification task by using different image encoders, which are

before	woman sending text messages while sitting on a sofa
case 1	sending messages sofa
case 2	sending text messages sofa
case 3	sending messages while sofa
case 4	woman sending messages sofa
case 5	sending text messages while sitting sofa

Table 5: The cases before and after subsampling frequent words in the CC3M dataset. The first row is the original text, the other rows are potential texts resulting after subsampling. The value of t is set to 10^{-4}.

method	data	epochs	model	text len	zero-shot
CLIP	CC3M	30	RN50	32	16.9%
SW-CLIP	CC3M	30	RN50	16	**17.2%**
CLIP	CC3M	30	ViT-B/16	32	14.9%
SW-CLIP	CC3M	30	ViT-B/16	16	**15.2%**

Table 6: Zero-shot accuracy on ImageNet1K classification. We pre-train SW-CLIP with different image encoders and our approach obtains better performance and requires less training time.

method	data	epochs	model	text len	zero-shot
CLIP	CC3M	30	RN50	32	16.9%
SW-CLIP	CC3M	**21**	RN50	16	**17.7%**
CLIP	CC3M	30	ViT-B/16	32	14.9%
SW-CLIP	CC3M	**21**	ViT-B/16	16	**16.2%**

Table 7: SW-CLIP achieves higher zero-shot accuracy on ImageNet1K classification with fewer epochs, when fine-tuning on the 20th epoch for RN50 and ViT-B/16 for an additional epoch.

RN50 and ViT-B/16. These models were pre-trained 30 epochs on the CC3M dataset with text lengths of 32 and 16 for CLIP and SW-CLIP respectively. In Table 6, with the same number of epochs, the accuracy of our method on RN50 (17.2% *vs.* 16.9%) and ViT-B/16 (15.2% *vs.* 14.9%) is improved by 0.3.

By diminishing the influence of high-frequency words, SW-CLIP achieves a better balance between frequent and infrequent words during the training process. Therefore, SW-CLIP learning is more effective than CLIP which does not subsample frequent words. Rows 2 and 4 of the Table 7 show that our method achieves higher accuracy on ImageNet1K zero-shot classification at 21 epochs, for which the models are pre-trained on CC3M by RN50 (17.7% *vs.* 16.9%) and ViT-B/16 (16.2% *vs.* 14.9%) image encoders with subsampling frequent words for 20 epochs and fine-tuning for 1 epoch without subsampling.

In summary, SW-CLIP requires less training time or fewer epochs compared to the CLIP, indicating that SW-CLIP is more efficient in terms of training time while achieving better performance.

method	encoder	Food-101	CIFAR-10	CIFAR-100	CUB200	SUN397	Cars	Aircraft	DTD	Oxford Pets	Caltech-101	Kinetics700	Flowers102	MNIST	FER-2013	STL10	EuroSAT	Resisc45	GTSRB	KITTI	Country211	PCAM	UCF101	CLEVR	HatefulMemes	SST2	ImageNet	Average
CLIP	RN50	10.66	29.86	12.49	3.35	26.62	1.04	1.43	10.96	11.28	40.94	38.23	2.42	9.96	15.73	73.26	17.04	20.59	3.87	25.00	0.69	46.15	20.46	14.40	44.37	50.08	16.90	18.75
SW-CLIP	RN50	11.20	34.01	14.66	2.87	32.95	0.88	1.34	14.63	15.45	39.48	38.23	2.75	12.30	17.33	76.05	13.28	20.03	5.49	41.18	0.71	52.57	20.41	12.28	45.85	50.08	17.23	20.52
CLIP	ViT-B/16	10.01	35.14	16.02	2.52	28.31	0.90	1.40	9.95	9.94	39.23	47.85	0.67	13.14	22.57	68.49	20.64	21.02	7.51	23.86	0.55	46.85	20.46	10.14	49.03	47.06	14.86	19.48
SW-CLIP	ViT-B/16	11.08	37.72	16.86	3.64	35.05	0.91	1.05	12.66	13.01	43.20	39.87	1.82	11.79	17.07	80.03	20.28	20.37	5.82	35.49	0.77	48.66	23.05	12.23	45.59	49.75	16.25	27.03

Table 8: Zero-shot accuracy on more classification datasets. The image encoders are RN50 and ViT-B/16, the pre-trained dataset is CC3M. The value of t is set to 10^{-4}.

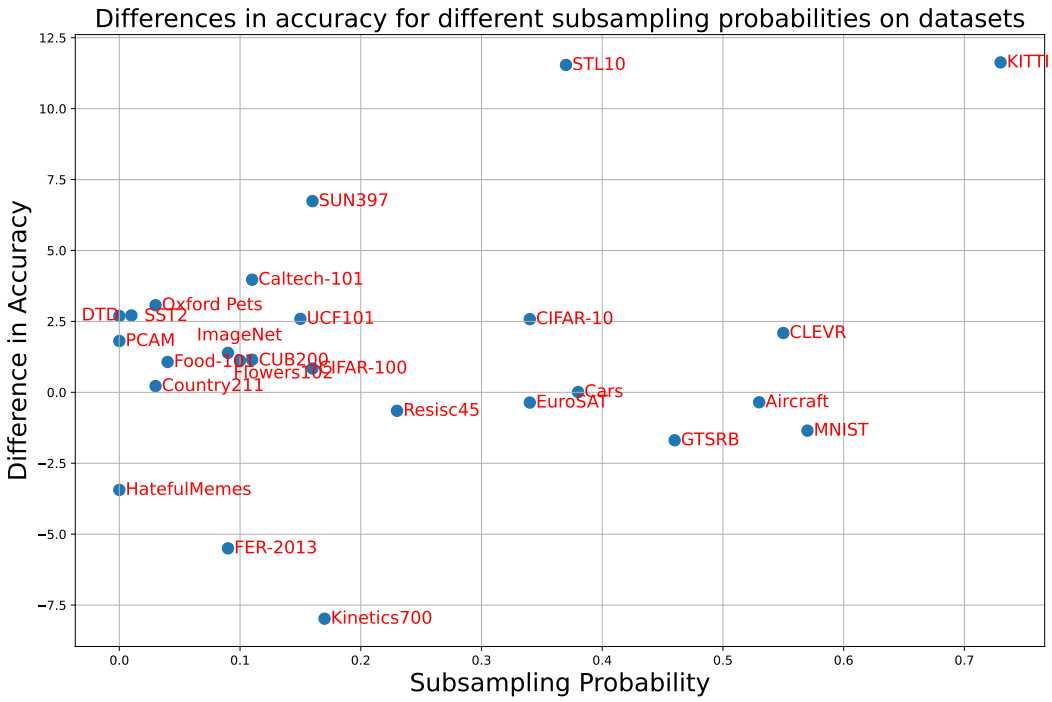

Figure 2: By subsampling frequent words, SW-CLIP achieves performance improvement on different datasets. This enhancement is attributed to the model's ability to improve the representation of infrequent words, striking a favorable balance between frequent and infrequent words in the training process. The image encoder is ViT-B/16 and the pre-trained dataset is CC3M.

4.5 Zero-shot Classification on More Datasets

In Table 8, we present the evaluation results of our method following the approach of CLIP [18, 21] on additional datasets. We adopted the same templates, class names, and metrics as SLIP [18] to ensure consistency in reporting our results. Our method, SW-CLIP, demonstrates superior performance compared to CLIP without subsampling frequent words.

We observed that SW-CLIP, particularly improved performance on datasets where the class names include infrequent words. As shown in Table 8, datasets such as DTD, Oxford Pets, Flower102, Food101, and PCAM, SUN397, among others [1, 4, 19, 20, 29, 30], benefited from this subsampling technique. We provide the average subsampling probabilities for each dataset class name in Figure 2. To calculate the average subsampling probability for a specific class within a dataset, we begin by determining the number of the classes, denoted as K. Next, we employ the formula given by Equation 1 to compute the subsampling probability for the word corresponding for each class. The average subsampling probability of each dataset (p) can be expressed as follows:

$$p = \frac{\sum_{i=1}^{K} \sum_{j=1}^{N} P(w_{i,j})}{K} \tag{3}$$

where N is the number of words in each class name.

Figure 2 shows that performance is improved for 13/16 datasets where class names are infrequent words. We define class names with average subsampling probability less than 0.2 as infrequent. On the left in Figure 2, just above 0, there is an improvement in accuracy for a cluster of datasets with a small subsampling probability. Further to the right in Figure 2, there is another cluster above and below 0, representing datasets with larger subsampling probabilities. This indicates that SW-CLIP enhances the performance of infrequent words without compromising the performance of frequent words. Note that, the STL10 [5] and KITTI [9] datasets show

method	text mask	text len	time	ImageNet1K
CLIP	random, 50.00%	16	0.86×	18.3%
SW-CLIP	SW, 42.30%	16	0.86×	17.2%

Table 9: Comparing the performance of frequency-based subsampling and random subsampling on zero-shot classification in ImageNet1K. The backbone of image encoder is RN50, and the model pre-trained on CC3M for 30 epochs. The first column is the methods. The second column is the percentage of the words remaining in the text after subsampling. The third column is the text length. The fourth column is the time of training the model, the training time of standard CLIP is defined 1.00×. The fifth column shows the top-1 accuracy of zero-shot classification on ImageNet1K .[6].

substantial performance improvements of 11.5% and 11.6%, respectively. The four class names of KITTI are sentences that describing the distance of the cars [9], and STL10 contains 10 classes which include animal (bear, rabbit, etc.) and vehicle (train, bus, etc.) [5]. Overall, these findings highlight the effectiveness of our method, SW-CLIP, and it can significantly the generalization ability of the VLMs by improving the representation of infrequent words.

5 CONCLUSION AND OUTLOOK

In this paper, we have introduced an approach for subsampling frequent words aimed at enhancing the training process and overall performance of Vision-Language Models (VLMs). The core idea behind SW-CLIP is to subsample frequent words within the text during the pre-training phase. Subsampling frequent words balances the focus of the model during training between frequent words and infrequent words to improve the representation of words in the VLMs. By selectively reducing the frequent words, we are able to streamline the training process and improve its efficiency. This approach offers dual benefits, as it not only enhances training speed but also improves the zero-shot classification performance of VLM. In the zero-shot transfer evaluation, SW-CLIP demonstrates a notable ability to rapidly learn image and text representations when compared to a model that pre-trains on the full-text. SW-CLIP achieves these results while only requiring approximately 86% of the time typically needed for pre-training. SW-CLIP shows enhanced performance, with 0.8% (RN50 backbone) and 1.3% (ViT-B/16 backbone) improvements in zero-shot classification on ImageNet1K. Moreover, SW-CLIP leads to substantial average accuracy gains of 1.77% (RN50 backbone) and 7.55% (ViT-B/16 backbone) across 26 datasets in zero-shot classification tasks. We noticed a trend that SW-CLIP tends to yield performance improvement for test datasets in which the labels (names) of the classes occur only infrequently as words in the training data.

The most important topic for future work is to understand the relationship between frequency-based subsampling and random subsampling. We trained a model with the same settings as our SW-CLIP model, but randomly removed 50% of the works rather than carrying out frequency-based subsampling. As with our SW-CLIP approach in Table 9, this brought the average length of the texts in the trainind set down to 16 words. This approach achieved an accuracy of 18.3% on ImageNet1K compared to the 17.2% achieved by SW-CLIP. The speed up was 0.86x, the same as for SW-CLIP. Further experiments are necessary to determine if this difference is due to the fact that random removal removed overall less words (50% of the original words remained in the training data vs. 42% for SW-CLIP) or if the impact of frequency-based removal is indeed less important for VLMs than it is for textual models, i.e., in [17]. Recall from Section 2, that FLIP did not achieve a gain with random sampling of words. Instead, randomly subsampling 50% of the text decreases the accuracy by 2.2%, using ViT-L/16 as the backbone for 6.4 epochs of pre-training on the LAION-400M. [13, 25]. We find that this result should be reproduced controlling for the number of epochs and testing text subsampling alone, rather than in combination with image subsampling. We carried out an additional set of exploratory experiments to determine the contribution of SW-CLIP when the training data set is limited in size. Under this condition, frequency-based sampling outperformed random sampling, confirming the importance of SW-CLIP.

Future work should also test SW-CLIP using a large-scale training set. Recall that due to limited computing resources, we conducted an evaluation of our SW-CLIP subsampling frequent words approach on a relatively small dataset. However, we see no intrinsic reason why the advantages of SW-CLIP should not extend to large data sets, in which the frequency of the most frequent words can be expected to be even larger than for small datasets. Evaluating subsampling frequent words approach on different datasets is necessary to understand its generalizability and effectiveness across various corpora.

REFERENCES

[1] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101 – Mining Discriminative Components with Random Forests. In *European Conference on Computer Vision*.

[2] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. 2020. Generative pretraining from pixels. In *International conference on machine learning*.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning*.

[4] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. 2014. Describing Textures in the Wild. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[5] Adam Coates, Andrew Ng, and Honglak Lee. 2011. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In *International Conference on Artificial Intelligence and Statistics*.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

[8] Andrea Frome, Gregory S. Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomás Mikolov. 2013. DeViSE: A Deep Visual-Semantic Embedding Model. In *Advances in Neural Information Processing Systems*.

[9] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. 2013. Vision meets Robotics: The KITTI Dataset. *International Journal of Robotics Research* (2013).

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition*.

[11] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. 2021. *OpenCLIP*.

[12] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*.

[13] Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. 2023. Scaling language-image pre-training via masking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[14] Ilya Loshchilov and Frank Hutter. 2017. SGDR: Stochastic Gradient Descent with Warm Restarts. In *International Conference on Learning Representations*.

[15] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

[16] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *International Conference on Learning Representations*.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*.

[18] Norman Mu, Alexander Kirillov, David A. Wagner, and Saining Xie. 2022. SLIP: Self-supervision Meets Language-Image Pre-training. In *1European conference on computer vision*.

[19] Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated Flower Classification over a Large Number of Classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*.

[20] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. 2012. Cats and Dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[21] 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 *International Conference on Machine Learning*.

[22] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical Text-Conditional Image Generation with CLIP Latents. arXiv:2204.06125

[23] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-Resolution Image Synthesis With Latent Diffusion Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[24] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. LAION-5B: An open large-scale dataset for training next generation image-text models. In *Conference on Neural Information Processing Systems, Datasets and Benchmarks Track*.

[25] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs. arXiv:2111.02114

[26] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. In *Annual Meeting of the Association for Computational Linguistics*.

[27] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation Learning with Contrastive Predictive Coding. arXiv:1807.03748

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*.

[29] Bastiaan S. Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. 2018. Rotation Equivariant CNNs for Digital Pathology. In *Medical Image Computing and Computer Assisted Intervention*.

[30] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. 2010. SUN database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

[31] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. 2022. Lit: Zero-shot transfer with locked-image text tuning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*.