Accelerating Vision-Language Pretraining with Free Language Modeling

Teng Wang[1,2†], Yixiao Ge[3], Feng Zheng[1,5*], Ran Cheng[1], Ying Shan[3], Xiaohu Qie[4], Ping Luo[2,6]

[1]Southern University of Science and Technology [2]The University of Hong Kong
[3]ARC Lab, [4]Tencent PCG [5]Peng Cheng Laboratory [6]Shanghai AI Laboratory

tengwang@connect.hku.hk {yixiaoge, yingsshan, tigerqie}@tencent.com

f.zheng@ieee.org ranchengcn@gmail.com pluo@cs.hku.hk

Abstract

The state of the arts in vision-language pretraining (VLP) achieves exemplary performance but suffers from high training costs resulting from slow convergence and long training time, especially on large-scale web datasets. An essential obstacle to training efficiency lies in the entangled prediction rate (percentage of tokens for reconstruction) and corruption rate (percentage of corrupted tokens) in masked language modeling (MLM), that is, a proper corruption rate is achieved at the cost of a large portion of output tokens being excluded from prediction loss. To accelerate the convergence of VLP, we propose a new pretraining task, namely, free language modeling (FLM), that enables a 100% prediction rate with arbitrary corruption rates. FLM successfully frees the prediction rate from the tie-up with the corruption rate while allowing the corruption spans to be customized for each token to be predicted. FLM-trained models are encouraged to learn better and faster given the same GPU time by exploiting bidirectional contexts more flexibly. Extensive experiments show FLM could achieve an impressive $2.5\times$ pretraining time reduction in comparison to the MLM-based methods, while keeping competitive performance on both vision-language understanding and generation tasks. Code will be public at https://github.com/TencentARC/FLM.

1. Introduction

Vision-language pretraining (VLP) has recently demonstrated impressive performance on a handful of vision-language tasks [7,10,14,18,19,22], *e.g.*, visual question answering, cross-modal retrieval, and image captioning. Several factors are responsible for the success: the availability of large-scale image-text datasets collected from the web [30], high-capacity model architectures like Trans-
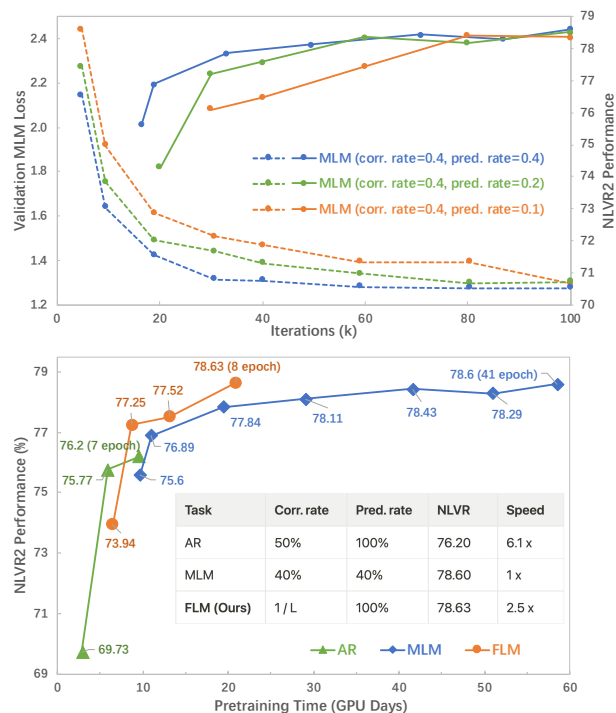
---
† Work done during internship in ARC Lab, Tencent PCG.
∗ Corresponding author

Figure 1. (a) Large prediction rate accelerates training. Given a fixed corruption rate, we vary the prediction rate by randomly selecting a subset of output tokens for prediction loss. The learning rate schedule follows METER [10]. (b) The proposed FLM achieves competitive performance compared with MLM meanwhile significantly accelerating the pretraining stage. The downstream performance on NLVR[2] [32] is reported. We show accuracy curves before convergence for better visualization.

former [34], and effective pretraining objectives for cross-modal learning.

One of the dominant pretraining objectives is masked language modeling (MLM), which was first introduced in natural language processing [9] and has been applied to vision-language areas in recent years [19]. MLM is a generative pretraining task designed to reconstruct a few (usually 40% for VLP) masked text tokens via reasoning

among the context of the remaining texts and the paired image. While effective in capturing cross-modal interactions, MLM-based methods [7,15,21] suffer from slow convergence and long training time, especially for large-scale models and noisy web data.

We argue that the limited prediction rate in MLM impedes the convergence speed of pretraining, since a large portion of tokens accompanied by corruption are excluded from prediction loss. As shown in Fig. 1 (top), under the same corruption rate , a larger prediction rate for MLM results in faster convergence of validation loss and downstream performance. It is intuitive to set a prediction rate of 100% to fully exploit text tokens. However, a paradox emerges where a large prediction rate can only be achieved with a greater corruption rate in MLM, but an extremely large corruption rate leads to an extremely tough pretraining task that may cause training collapse.

Autoregressive language modeling (AR) provides a workable solution to enable a 100% prediction rate. It predicts the next token given the observation of previous tokens. As shown in Fig. 1 (bottom), AR performs favorably in training efficiency against MLM, *i.e.*, $6.1\times$ speed-up for convergence. However, the converged performance by AR is, unfortunately, much inferior to MLM. It is probably caused by the sub-optimal unidirectional corruption pattern, which is insufficient for downstream understanding tasks that usually rely on bidirectional contexts.

A natural question arises, *can we accelerate the convergence of VLP by predicting 100% tokens like AR meanwhile achieving competitive performance with MLM?* Towards this end, we introduce a new pretraining task, dubbed free language modeling (FLM), for VLP, that enjoys an extreme 100% prediction rate and flexible bidirectional contextualized representations. We for the first time break up the entanglement between corruption and prediction rates, making the two factors freely determined. Furthermore, for each output token to be predicted, we allow independent and arbitrary-length spans (from one to 100% tokens) as corrupted connections. Rather than the suffix-like corruption pattern as in AR (as well as PrefixLM [37]), the corruption span of FLM is primarily distributed in the middle of the sequence, establishing a flexible perception of bidirectional contexts for better adaptation to VL understanding tasks. The comparison between different pretraining objectives is illustrated in Fig. 2.

To perform VLP with FLM, we propose an encode-corrupt-predict framework, which performs feature encoding once and reconstructs several corrupted versions of the text sequence in parallel. In the encoding step, bidirectional representations are achieved by learning forward and reverse unidirectional representations respectively, the order of which is manipulated by (reverse) casual masks in the same text Transformer. Subsequently, we ensure a 100% prediction rate by customizing corruption-prediction tasks for predicting each input token. In each corruption-prediction task, a span of corruption is randomly sampled and attached to the encoded sequence, followed by a reconstructor to solve the prediction task by reasoning among the remaining contexts. Unlike previous works (*e.g.*, MLM, AR) that adopt pre-encoding corruption, we inject corruptions after one-time feature encoding, encouraging flexible corruption patterns and efficient parallel prediction.

Our contributions are three-fold. (1) A novel pretraining objective for VLP, namely, free language modeling (FLM), is proposed to free the prediction rate from the constraints of corruption rate, enabling an appealing 100% prediction rate for accelerating convergence speed during pretraining. (2) An encode-corrupt-predict framework built upon FLM objective is proposed, allowing efficient and effective learning of a set of prediction tasks by merely conducting feature encoding once. (3) Extensive experiments on VQA, NLVR^2, image captioning, and image-text retrieval demonstrate the effectiveness of our FLM, where comparable performances to MLM are achieved with less than 50% pretraining time.

2. Related Work

Vision-Language Pretraining. Vision-language pretraining tasks can be divided into two categories: (i) *discriminative* tasks, *e.g.*, image-text contrastive (ITC), image-text matching (ITM), and (ii) *generative* tasks, *e.g.*, masked language modeling (MLM) and autoregressive language modeling (AR). *Discriminative* tasks consider the image-text pairs as multi-modal views of the same semantics. Contrastive or multi-view learning is adopted for learning the alignment between multiple modalities. For example, CLIP [29], ALIGN [14], and following works [18, 20, 40] utilize cross-modal contrastive learning by projecting the image and language information into a joint (structured) semantic space. *Generative* tasks aim to reconstruct the corrupted text (image) with the assistance of visual (text) modality. The main body of representative works [2, 7, 19, 22, 27, 35, 42, 43], employ the MLM-like objectives, where input texts(image) are partially masked and then interact with visual (text) tokens to reconstruct the corrupted part. SimVLM [37] introduces a single prefix language modeling (PrefixLM) objective for exploiting large-scale weak supervision in VLP. CoCa [41] further verifies the representation ability of autoregressive language modeling (AR) in the vision-language domain. While most existing methods combine discriminative and generative tasks for better representation learning, BEiT-3 [36] shows a single generative language modeling (*e.g.*, MLM) could handle the vision-language interactions and alignments well with the mixture-of-expert transformer. Although superior performance has been attained, most existing methods based on MLM suffer from low utilization of output tokens and lead to a slow
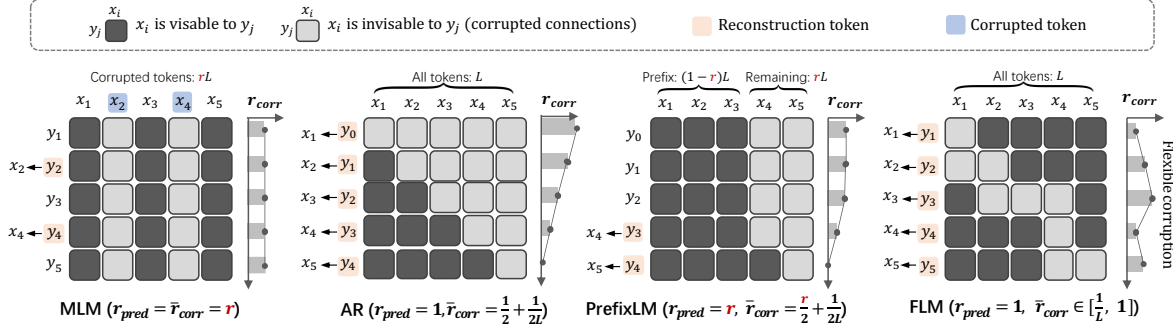
Figure 2. Dependency matrix of different language modeling methods in vision-language pretraining. r_{pred} represents the proportion of output tokens for reconstruction. r_{corr} represents the proportion of corrupted inputs for each output token. $\bar{r}_{\text{corr}}$ is the mean corruption rate of all reconstruction tokens. FLM has distinct advantages compared with others: 1) Different from MLM and PrefixLM that bind r_{pred} and $\bar{r}_{\text{corr}}$ together by r, the unbound prediction rate in FLM could achieve 100% for accelerating training as much as possible. 2) Without relying on a position-aware unidirectional corruption in AR/PrefixLM or fixed corruption across all positions in MLM (see the right-side line graph), the corrupted span in FLM for each output token could be different, and the corrupted rate is independent of the position of the output token, enabling a more flexible corruption pattern for better exploiting the bidirectional context information.

convergence rate. This paper proposes a new generative language modeling method targeting pretraining acceleration.

Efficient Pretraining. While early VLP methods [7,22,33, 43] rely on time-consuming pretrained object detectors for visual representation, PiexlBERT [13] and ViLT [15] directly apply grid/patch-level visual features to reduce computation complexity of the object-level visual encoder. Beyond the design of efficient model architecture, a few research focuses on data-efficient training. Bitton et al. [3] propose an alternative masking strategy that better focuses on visually-related physical words to improve VLP in low-resource settings. DeCLIP [23] enhances the CLIP by exploring more supervision signals, such as self-supervision within a single modality or multi-view supervision across different modalities. The most relevant work to this paper is GRIT-VLP [5], which assigns a larger mask rate for MLM and performs grouped in-batch negative sampling for ITC to accelerate the convergence. However, only half of the output tokens are assigned for the reconstruction task, where the under-used output tokens impede a further speed-up of pretraining. Our method decouples the corruption and reconstruction rate, making them freely chosen for a better combination between performance and efficiency.

Language Modeling. In NLP, MLM [9,12,25] and AR [4, 8] have been the two most popular generative pretraining objectives. AR aims to estimate the probability distribution of a given text sequence using the product rule by an auto-regressive model. However, unidirectional encoding may not be suitable for language understanding tasks that prefer bidirectional context information. MLM enables bidirectional contexts for language understanding tasks but can not be directly adopted into language generation tasks. Some works [11, 39] unify MLM and AR for better performance on both language understanding and generation tasks. Wettig et al. [38] study the choice of the mask ratio

in MLM from the perspective of both corruption and prediction. However, among previous methods, little attention has ever been devoted to the issue of training efficiency. We target accelerating vision-language pretraining meanwhile keeping decent performances on vision-language understanding and generation tasks.

3. Method

In this section, we first recap the representative language modeling methods for VLP from a corruption-prediction view in Sec. 3.1. Then we propose the new language modeling method FLM to decouple the prediction rate from the corruption rate in Sec. 3.2. Finally, we introduce FLM into VLP and propose a novel encode-corrupt-predict framework for accelerating VLP in Sec. 3.3.

3.1. Language Modeling as Corruption-Prediction

Given an input sequence $\mathbf{x} = \{x_1, ..., x_L\}$, **MLM** aims to learn a deep bidirectional representation by randomly replacing part of input tokens with a special mask token, and then maximize the probability of reconstructing those masked tokens $P(\mathbf{x_m}|\mathbf{x_{\backslash m}})$, where $\mathbf{x_m}$ represents corrupted tokens. **AR** uses a left-to-right autoregressive factorization to model the density distribution of the sequence $\sum_{i=1:L} \log P(x_i|x_{<i})$. **PrefixLM** enables bidirectional perception between prefix tokens and left-to-right autoregressive factorization to model the density distribution of the remaining sequence $\sum_{i=L_p:L} \log P(x_i|x_{[L_p,i]}, x_{<L_p})$ where L_p represents the prefix length.

Note that all the above methods could be interpreted as a corruption-prediction problem because each prediction token only has a partial observation of the input data, *i.e.*, the interactions between output tokens and some input tokens are corrupted. Therefore, their pretraining objectives could

be unified as maximizing the reconstruction probability:

$$\mathbb{E}_{\mathbf{M} \sim B(r)} \sum_{i=1:L} \mathbb{1}_{m_{ii}=0} \log P(x_i | \{x_j | m_{ij}=1\}), \quad (1)$$

where $\mathbf{M} = [m_{ij}]_{1 \leq i \leq L, 1 \leq j \leq L}$ represents the dependency matrix between the input and the prediction target, $m_{ij} = 1/0$ represents that x_j is visible/invisible when predicting x_i. $B(r)$ represents a distribution parameterized by r, which is customized by specific models. The dependency matrices $\mathbf{M}$ of different language modeling methods are as follows (also illustrated in Fig. 2):

- For MLM, $\mathbf{m}_{1,:} = \cdots = \mathbf{m}_{L,:} = \mathbf{p} \sim \text{Binomial}(r_{\text{mask}})$. The corruption for predicting all x_i is the same and it is sampled from a Binomial distribution.

- For LM, $m_{ij} = \mathbb{1}_{j<i}$. The corruption for predicting x_i depends on the position i, which gets shorter with a larger i.

- For PrefixLM, $m_{ij} = \mathbb{1}_{j < \max(i, L_{\text{p}})}$, where prefix length $L_{\text{p}} = (1 - r_{\text{span}}) \cdot L$ and $r_{\text{span}} \sim \text{Uniform}(0, 1)$ represents the length ratio of the corrupted span.

3.2. Free Language Modeling (FLM)

From the above analysis, the representative MLM or AR/PrefixLM methods have limited freedom of the dependency matrix, which is prone to the following issues: 1) the tie-up between the prediction and corruption rate in MLM may lead to a low convergence speed during training; 2) inflexible and non-customizable contexts for each prediction target result in sub-optimal context learning. For example, the suffix-like corruption in AR disables the bidirectional context modeling which is essential for downstream understanding tasks. Moreover, the autoregressive prior in AR results in uneven distribution of corruption rate. Latter tokens are always assigned with a smaller corruption rate, thus being easy-predictable compared with former ones. The position-related distribution of the corruption rate may cause a sub-optimal difficulty degree of pretraining tasks.

The goal of FLM is to disentangle the prediction and corruption rate for fully utilizing training signals to accelerate VLP. The model after disengagement has a more flexible corruption pattern that benefits bidirectional contextual representation learning. Following the unified formulation of the corruption-prediction problem in Eqn. 1, we introduce the dependency matrix of FLM:

$$m_{ij} = 1 \text{ if } m_{ij} \notin \text{span}_i, \text{ otherwise } 0. \quad (2)$$

where span_i is random span corruption with length L_{span}^i that satisfies $i \in \text{span}_i$. The starting position and length of span_i could be customized or randomly sampled from a distribution. In our implementation, we sample $L_{\text{span}}^i \sim \text{Bernoulli}(L, r_{\text{corr}})$ for each i, r_{corr} is the hyperparameter indicating the expected corruption rate.
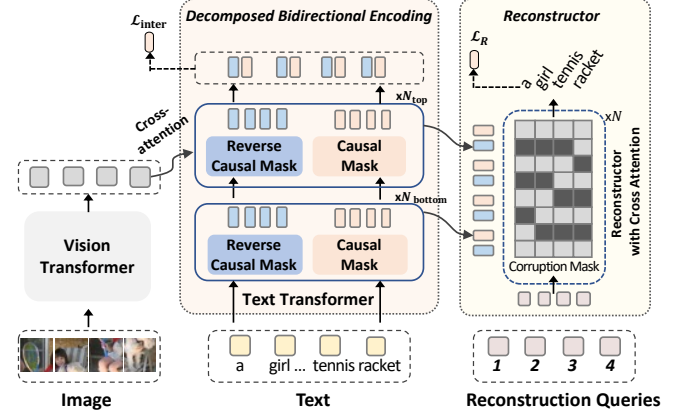

Figure 3. Overview of the proposed VLP framework with free language modeling (FLM). First, the image is patchified and encoded by a vision transformer into a sequence of vision tokens. Then, the text transformer performs uni-modal text feature encoding in the bottom layers and multimodal fusion between visual and text features in the top layers. Bidirectional multimodal representations are achieved by learning forward and reverse unidirectional representations, respectively, the order of which is manipulated by (reverse) causal masks in the same text transformer. After feature encoding, we construct a set of independent corruption-prediction tasks. For each task, we inject a random span corruption into the multimodal representation and then introduce a reconstruction query that gathers informative contexts from the corrupted features for reconstructing a single target token. Benefiting from the flexibility of post-encoding corruption, 100% text tokens could be efficiently reconstructed in parallel.

Note that the corrupted span in FLM could differ for different predictions, hopefully increasing the flexibility of bidirectional contextualized interactions and helping optimization. Since the choice of span_i does not interfere with each other for different i, the prediction rate could increase to 100% which allows all input tokens to be reconstructed. As for the corruption rate, we note that the minimal corruption rate is $1/L$, since at least one token should be corrupted to reconstruct itself to avoid information leakage.

3.3. Vision-Language Pretraining with FLM

Built upon FLM, a new encode-corrupt-predict pretraining framework is proposed for efficient pretraining with decoupled prediction and corruption rates. Given the input sequence $\mathbf{x} = \{x_1, ..., x_L\}$, we formulate the reconstruction of all input tokens as L independent corruption-prediction tasks. For i-th task, the learning objective is to maximize $P(x_i | \{x_j | m_{ij} = 1\})$ by reasoning upon uncorrupted bidirectional tokens. Fig. 3 depicts the pipeline of the model.

Decomposed Bidirectional Encoding. Since FLM establishes a customized corruption span for each prediction task, a naive solution of repeating the MLM-style feature encoding (transformer with fully-visible attention mask) for

each task becomes time-consuming. Instead, we propose to share the intermediate features of different tasks for efficient inference by decomposing the token representation into two complementary views, left-to-right, and right-to-left features. The former adopted a text transformer with a casual mask, enforcing the latter tokens attending on previous tokens. The latter adopted the same transformer with a reverse causal mask for a right-to-left context flow.

Specifically, we first encode image features by a CLIP image transformer. For the text input, the N_{bottom} bottom layers of the text transformer perform the decomposed bidirectional encoding with the language input only, while N_{top} top layers further receive image features and fuse the multimodal features by cross-attention layers. After encoding, we obtain the bidirectional representation in text transformer, denoted as $E^n = \{e_1^n, \cdots, e_L^n\}$, where $e_i = \{e_i^{\text{l2r},n}, e_i^{\text{r2l},n}\}$ is the token representation for x_i comprised of features from forward and reverse flow at the n-th layer.

Reconstructor. Following the description in Sec. 3.2, we sample a dependency matrix $\mathbf{M}$ to construct several corruption-prediction tasks. As a consequence of the span corruption, some elements in E^n that rely on corrupted inputs need to be neglected to avoid information leakage. To reconstruct x_i, we gather context from uncorrupted features in E by cross-attention:

$$
\begin{aligned}
q_i^{n+1} &= \text{CrossAttention}(q_i^n, E_i^n) \\
E_i^n &= \{e_j^{\text{l2r},n} | m_{ij}=1, j<i\} \cup \{e_j^{\text{r2l},n} | m_{ij}=1, j>i\},
\end{aligned}
\tag{3}
$$

where E_i^n represents all uncorrupted elements in E^n given M. q_i is a learnable reconstruction query, which is initialized as the i-th positional embedding in the first layer. The selection process from E^n to E_i^n is implemented as a specific attention mask in cross-attention layers, as illustrated in Fig. 3. By forwarding on stacked cross-attention layers, q_i^n aggregates deep bidirectional contexts for effective reconstruction of x_i. The output features of the last layer in the reconstructor are input into an MLP for final prediction.

Note that q_i^n works independently from each other, making a flexible addition or removal of some tasks. By sharing feature encoding, all reconstruction tasks run in parallel with low computation consumption.

Pretraining Objectives. The reconstruction objective is to minimize the negative log-likelihood of predicted tokens:

$$
\mathcal{L}_{\text{R}} = -\mathbb{E}_{\mathbf{M} \sim B(r)} \sum_{i=1:L} \log P(x_i | \{x_j | m_{ij}=1\}),
\tag{4}
$$

To further enhance the model's representability, we introduce an intermediate prediction loss upon the E^N that improves the local temporal dependency between words, where N represents the last layer of the text transformer. We supervised the forward/reverse sequence $e_i^{\text{l2r},N}/e_i^{\text{r2l},N}$ by their next/previous tokens. The intermediate loss

is the summation of two unidirectional prediction problems: $\mathcal{L}_{\text{inter}} = \mathcal{L}_{\text{l2r}} + \mathcal{L}_{\text{r2l}} = \sum_{i=1:L} \log P(x_i|x_{<i}) + \sum_{i=1:L} \log P(x_i|x_{>i})$. The overall pretraining objective of FLM is calculated by $\mathcal{L}_{\text{FLM}} = \mathcal{L}_{\text{R}} + \mathcal{L}_{\text{inter}}$.

4. Experiments

4.1. Experimental Setting

Pretraining Data. Following previous work [10, 18], the pretraining data comes from four commonly used datasets, including COCO [24], Visual Genome [16], SBU Captions [28], and Conceptual Captions 3M [30], totally 4M images. An enlarged version of ∼13M images is further used to boost performance by including Conceptual Caption 12M [6] [1].

Downstream Tasks. During finetuning, we append a special [CLS] token into the reconstructor and use its output features as the global cross-modal representation. We follow [7] to adapt the pre-trained model to four downstream vision-language understanding tasks, visual question answering (VQA) [1], natural language for visual reasoning (NLVR^2) [32], image-text retrieval (TR), text-image retrieval (IR). We also test the performance on vision-language generation tasks, *i.e.*, image captioning [24]. For image captioning, we drop the reconstructor and use the text transformer with a causal mask for sequence generation. More details can be found in supplementary materials.

Pretraining Details. Following ALBEF [18] and METER [10], the visual transformer is initialized by CLIP-ViT [29] pretrained on 400M noisy image-text pairs. The visual transformer with ViT-B/32 is used as our base architecture for ablation study, and the one with ViT-L/14 is for scaling up to compare with other methods. We denote models with ViT-B image encoder as Ours and ViT-L as Ours$_{\text{LARGE}}$. The bottom six layers of the text transformer are initialized by the bottom six layers of RoBERTa$_{\text{BASE}}$ [25]. The reconstructor is implemented by a 12-layer ($N_{\text{bottom}} = N_{\text{top}} = 6$) transformer decoder (remove self-attention layers and only keep cross-attention layers) with a hidden size of 768 and a head number of 12. The default corruption rate of FLM is $1/L$, *i.e.*, in each corruption-prediction task, only a single token is corrupted and then reconstructed from their contexts. While minimal corruption achieves decent performance, we further explore the choice of corruption rates in Sec. 4.3.

We pretrain the model for a maximum of 30k steps, with a total batch size of 4096 on 16 TITAN V100 GPUs by AdamW [26] optimizer and gradient accumulation. Mixed-precision training is used to reduce memory consumption and accelerate training. We use a 5% warm-up schedule with a maximum learning rate of 4e-4. Following [10], we

---

[1] Only 8.7M images of CC12M are accessible due to expired URLs.

Method	r_{corr}	r_{pred}	VQA test-dev	NLVR^2 dev	test	Retrieval (Flickr30K) IR@1	TR@1	COCO Captioning BLEU	METER	CIDEr	GPU Days (speed-up)
AR	50%	100%	72.85	75.79	76.29	66.59	84.10	<u>35.70</u>	<u>28.86</u>	<u>120.6</u>	9.6 (6.1×)
PrefixLM	25%	50%	72.64	75.73	76.17	66.21	82.70	35.50	28.79	119.4	10.0 (5.9×)
MLM	15%	15%	73.52	77.46	78.28	71.33	<u>88.40</u>	34.90	28.50	117.5	58.7 (1×)
MLM	40%	40%	**73.95**	<u>77.62</u>	<u>78.60</u>	**73.41**	**89.20**	35.50	28.79	120.3	58.7 (1×)
FLM (Ours)	$1/L$	100%	<u>73.85</u>	**77.99**	**78.63**	<u>72.81</u>	87.40	**36.68**	**29.17**	**123.0**	22.7 (2.5×)

Table 1. Performance Comparison between different language modeling methods. r_{corr} and r_{pred} refer to the corruption and prediction rates. All models are based on CLIP-B/32 image encoder and a text transformer initialized by RoBERTa. Note that the default r_{corr} of FLM is set to $1/L$ for better efficiency, while FLM's performance will be further improved by an optimal r_{corr}, as indicated in Table 3d.

Method	VQA test-dev	NLVR^2 dev	test	Captioning CIDEr	GPU Days
CLIP-B/32 on 13M data					
AR	73.46	76.60	77.21	121.5	21.3 (5.4×)
MLM	74.25	78.63	79.19	122.6	116.0 (1×)
FLM	**74.28**	**78.73**	**79.52**	**122.6**	32.0 (3.6×)
CLIP-B/16 on 4M data					
AR	75.05	77.38	78.79	126.0	12.3 (5.0×)
MLM	75.76	**79.93**	79.83	125.4	61.4 (1×)
FLM	**75.95**	79.02	**80.03**	**126.5**	16.1 (3.8×)

Table 2. Performance comparison of different pretraining objectives with a larger data scale (from 4M to 13M) or a larger number of patches (patch size from 32 to 16). For 13M data, we extend the training iteration of MLM to 200k.

assign a lower learning rate of 8e-5 for all pretrained layers. The text sequence length is limited to 50 subwords. More details are in the supplementary materials.

Baselines. We also pretrain some generative language modeling methods for comparison, including MLM, AR, and PrefixLM [37]. Specifically, we directly input the (corrupted) text sequence into the text transformer and then build an MLP layer upon the last layer of the text transformer for reconstructing the original input. For AR and PrefixLM, we follow the same learning rate schedule as FLM. For MLM, we follow [10] to train the model for 100k iterations with a maximum learning rate of 5e-5 and a warm-up rate of 10%. To compare the convergence speed of different methods, we report the GPU days when reaching the best validation performance (*i.e.*, reconstruction accuracy on the COCO validation set).

4.2. Comparison with Language Modeling Methods

As shown in Table 1, compared with MLM, the proposed FLM achieves a 2.5×speed-up while keeping the comparable performance on VL understanding tasks and superior performance on VL generation tasks.

AR achieves decent performance on image captioning generation but inferior performance on other VL understanding tasks, due to the lack of ability to capture bidirectional interactions among sequences. Moreover, AR has a faster convergence rate and high training efficiency.

Although PrefixLM enables bidirectional interactions between partial inputs, which is beneficial for VL classification tasks, the performance of PrefixLM is similar to AR. However, the reconstruction target mainly falls on the right side of the sequence, where the uneven distribution may push the learned representation towards an unsatisfactory language prior. For MLM, we found a corruption rate of 40% achieves the best VQA performance among $\{10\%, 20\%, ..., 80\%\}$, indicating an appropriate corruption rate is essential to control the task difficulty. However, the convergence rate of MLM is slow, making larger training steps necessary to achieve decent performance. Our method FLM surpasses MLM on NLVR^2 and image captioning, also showing an impressive 2.5×speed-up of training time.

In Table 2, we show that the superiority of the proposed FLM consistently holds with a larger data scale or with more powerful visual features. The reason may be that FLM learns bidirectional context patterns by encoding the text sequence once and densely predicting the 100% input in parallel, while MLM usually needs more pretraining steps to see such diverse patterns. Therefore, FLM is a friendly pretext task for accelerating training under low-resource scenarios, which to some extent, enjoys the high efficiency of AR/PrefixLM and the high performance of MLM.

As for underperformed retrieval performance compared with MLM, we conjecture that FLM with span corruptions has fewer corruption variants than MLM with random corruptions but focuses more on local semantics, which favors fine-grained tasks like VQA/captioning more than retrieval.

4.3. Ablation Studies

FLM Loss. The ablation study for the loss terms in FLM is shown in Table 3a. With merely reconstruction loss $\mathcal{L}_R$, our model achieves better performance (73.04 on VQA) compared with AR (72.85) or PrefixLM (72.64). When further introducing left-to-right or right-to-left intermediate caption loss, the model gains consistent improvements over two downstream tasks. Note that left-to-right loss shows non-trivial superiority to the right-to-left one, verifying the effectiveness of the causal relationships between words. By combing bidirectional caption loss, the model achieves 0.81/1.45 absolute gains over the model with merely recon-

Loss	VQA	NLVR^2
L_R	73.04	77.18
$L_R + L_{\text{r2l}}$	73.38	77.59
$L_R + L_{\text{l2r}}$	73.67	78.00
$L_R + L_{\text{inter}}$	**73.85**	**78.63**

(a) Loss term. Intermediate losses are effective and complementary to FLM loss.

Text Encoder	VQA	NLVR^2
Unshared text encoder	73.46	77.51
Shared text encoder	**73.85**	**78.63**

(b) Parameter sharing. Sharing two unidirectional encoders is effective and efficient.

Prediction rate	VQA	NLVR^2
50%	73.74	77.47
75%	73.89	77.65
90%	**74.00**	78.17
100%	73.85	**78.63**

(c) Prediction rate. FLM with a larger prediction rate improves performance.

Corruption	VQA	NLVR^2
span corruption ($1/L$)	73.85	78.63
span corruption (30%)	73.96	**78.83**
span corruption (40%)	**74.04**	78.82
span corruption (50%)	74.01	77.84
random corruption (15%)	73.93	78.38
random corruption (30%)	73.69	77.74

(d) Corruption Rate. FLM enables a flexible choice of the corruption rate.

Bottom	Top	VQA	NLVR^2
×	1	73.46	77.49
×	3	73.62	78.20
×	6	73.74	78.14
3	6	73.69	78.20
6	6	**73.85**	**78.63**

(e) Number of reconstruction Layers. FLM benefits from a deeper reconstructor.

Table 3. FLM ablation experiments with ViT-B/32 pretrained on 4M data. We report the finetuned accuracy (%) on the VQA test-dev and NLVR^2 test set. Default settings are marked in `gray`.

struction loss.

Parameter Sharing. During decomposed bidirectional encoding, parameter sharing is used in the text transformer for two unimodal encodings with different attention masks. Table 3b shows that the shared text transformer clearly surpasses the unshared one, indicating that the two unidirectional representations could implicitly benefit each other by sharing the same feature space.

Number of Reconstruction Layers. The reconstructor aims to construct several corrupted sequences upon the high-level representations and reconstruct the corrupted information in parallel. Table 3e shows that a deep structure of the reconstructor helps the downstream tasks. The multi-layer reconstructor gathers text and multimodal features from low to high levels, promising to enhance the representation ability.

Prediction Rate. We test the converged performance of the pretrained model with different prediction rates. To this end, we randomly mask a subset of output tokens from loss calculation. As shown in Table 3c, a lower prediction rate tends to achieve poor performance both on VQA and NLVR^2, probably suggesting that the prediction loss containing a larger number of tokens helps the optimization.

Corruption Rate. The corruption rate determines how much context should be used for predicting the corrupted tokens. It controls the difficulty of the reconstruction problem and closely affects model performance. We study the influence of corruption strategies in FLM. As shown in Fig. 3d, First, we test the length of span corruption. With the growth of span length, the VQA and NLVR^2 performance steadily reach their maximum values at the 30%~40% corruption rate. Our method keeps a 100% prediction rate while allowing a customized corruption rate, which is hopeful to serve

as a replacement for the widely-used MLM to improve convergence speed.

Besides the span corruption which occurs after feature encoding, we also test the influence of pre-encoding corruption. We assign random corruptions to each token of the input sequence and then perform FLM to reconstruct all input tokens. With a 15% corruption rate, random corruption could slightly increase the VQA score. But unfortunately, the NLVR^2 hurts with a larger corruption rate. We found that the optimal corruption rate may differ for different corruption methods. How to effectively fuse different types of corruption may be a promising direction to increase the diversity of contexts further.

4.4. Comparsion with State-of-the-Arts

The comparisons on VQA, NLVR^2 and image captioning are shown in Table 4, without using complicated pretraining tasks like ITM and ITC, our method achieves competitive performance by merely using FLM as the pretraining task. Compared with prior arts, our method has appealing advantages regarding pretraining time: First, the proposed FLM helps the convergence speed by enabling 100% token prediction. Second, we leverage FLM as the single pretraining objective, without relying on additional time-consuming pretraining objectives like ITM. Third, we use the patch-level image features instead of a heavy object detection used in [22, 43].

The performance on cross-modal retrieval is shown in Table 5. Our FLM-trained model performs poorly if directly fine-tuned on target downstream datasets. Note that retrieval is heavily required for cross-modal alignment learning (e.g., ITM or ITC) on large-scale datasets since negative samples are essential to learning discriminative features.

Model	Pretrain. Task	Pretrain. Time (GPU Days)	VQAv2 test-dev	VQAv2 test-std	NLVR² dev	NLVR² test	BLEU4	METEOR	CIDEr	SPICE
Pre-trained with <10M images										
UNITER_LARGE [7]	MLM, ITM, MVM, WRA	152 (V100)	73.82	74.02	79.12	79.98	-	-		
UNIMO_LARGE [21]	MLM, MVM, ITC	640 (V100)	75.06	75.27	-	-	-	-		
OSCAR	MLM, ITM	220 (V100)	73.61	73.82	79.12	80.37	37.4	**30.7**	127.8	23.5
VinVL_BASE [43]	MLM, ITM	320 (V100)	75.95	76.12	82.05	83.08	38.2	30.3	129.3	23.6
VinVL_LARGE [43]	MLM, ITM	320 (V100)	76.52	76.60	<u>82.67</u>	**83.98**	38.5	<u>30.4</u>	130.8	23.4
PixelBERT [13]	MLM, ITM	-	74.45	74.55	76.5	77.2	-	-		
CLIP-ViL [31]	MLM, ITM, VQA	40 (A100)	76.48	76.70	-	-	40.2*	29.7*	134.2*	23.8*
ViLT [43]	MLM, ITM, WRA	192 (V100)	71.26	-	75.70	76.13	-	-		
ALBEF (4M) [18]	MLM, ITM	28 (A100)	71.40	-	-	77.51	-	-		
ALBEF (4M) [18]	MLM, ITM, ITC	28 (A100)	74.54	74.70	80.24	80.50	-	-		
METER_BASE [10]	MLM, ITM	64 (A100)	77.68	77.64	82.33	83.05	<u>38.8</u>	30.0	128.2	23.0
Ours_LARGE (4M)	FLM	18 (V100)	<u>77.80</u>	<u>77.84</u>	81.77	81.83	38.3	30.2	<u>130.9</u>	-
Pre-trained with 10M∼100M images										
ALBEF (14M) [18]	MLM, ITM, ITC	140 (A100)	75.84	76.04	82.55	83.14	-	-		
BLIP (14M) [17]	AR, ITM, ITC	112 (A100)	77.54	77.62	82.67	82.30	38.6	-	129.7	-
Ours_LARGE (13M)	FLM	75 (V100)	**78.18**	**78.24**	**82.90**	<u>83.86</u>	**39.1**	30.3	**132.7**	-
Pre-trained with >100M images										
SimVLM_BASE (1.8B) [37]	PrefixLM	-	77.87	78.14	81.72	81.77	39.0	32.9	134.8	24.0
SimVLM_HUGE (1.8B) [37]	PrefixLM	-	80.03	80.34	84.53	85.15	40.6	33.7	143.3	25.4
LEMON (400M)	MLM	-	-	-	-	-	40.3	30.2	133.3	23.3

Table 4. Comparisons with models on visual question answering, visual reasoning, and image captioning tasks. The best scores are in **bold**, and the second best scores are in <u>underlined</u>. MVM, ITC, ITM, and WRA represent masked vision modeling, image-text contrast, image-text matching, and word-region alignment, respectively. Ours_LARGE are trained with 30k/100k steps on 4M/13M data, respectively.

Model	Pretrain. Task	Pretrain. Time (GPU Days)	Flickr30k IR@1	IR@5	IR@10	TR@1	TR@5	TR@10	COCO IR@1	IR@5	IR@10	TR@1	TR@5	TR@10
Pre-trained with <10M images														
UNITER_LARGE [7]	MLM, ITM, MVM, WRA	152 (V100)	75.56	94.08	96.76	87.30	98.00	99.20	52.93	79.93	87.95	65.68	88.56	93.76
UNIMO_LARGE [21]	MLM, MVM, ITC	640 (V100)	78.04	94.24	97.12	89.40	98.90	<u>99.80</u>	-	-	-	-	-	-
VinVL_LARGE [43]	MLM, ITM	320 (V100)	-	-	-	-	-	-	**58.8**	**83.5**	**90.3**	<u>75.4</u>	<u>92.9</u>	<u>96.2</u>
PixelBERT [13]	MLM, ITM	-	71.5	92.1	95.8	87.0	98.9	99.5	50.1	77.6	86.2	63.6	87.5	93.6
ViLT [43]	MLM, ITM, WRA	192 (V100)	64.4	88.7	93.8	83.5	96.7	98.6	42.7	72.9	83.1	61.5	86.3	92.7
ALBEF (4M) [18]	MLM, ITM, ITC	28 (A100)	<u>82.8</u>	<u>96.7</u>	<u>98.4</u>	<u>94.3</u>	99.4	99.8	56.8	81.5	89.2	73.1	91.4	96.0
METER_BASE [10]	MLM, ITM	64 (A100)	82.22	96.34	98.36	<u>94.30</u>	**99.60**	**99.90**	<u>57.08</u>	<u>82.66</u>	<u>90.07</u>	**76.16**	**93.16**	**96.82**
Ours_LARGE (4M)	FLM	18 (V100)	74.53	93.96	97.26	88.10	98.30	99.60	46.46	75.43	85.09	62.84	86.64	93.00
Ours_LARGE (4M)	FLM, ITM	57 (V100)	**83.40**	**97.04**	**98.72**	**95.00**	<u>99.50</u>	**99.90**	56.55	82.02	89.63	73.52	91.95	95.97
Pre-trained with >10M images														
ALBEF (14M) [18]	MLM, ITM, ITC	60 (A100)	85.6	97.5	98.9	95.9	99.8	100.0	60.7	84.3	90.5	77.6	94.3	97.2
BLIP (14M)	AR, ITM, ITC	112 (A100)	87.2	97.5	98.8	96.6	99.8	100.0	63.1	85.3	91.1	80.6	95.2	97.6

Table 5. Performance comparisons with models pre-trained on Flickr30k and COCO image retrieval (IR) and text retrieval (TR) tasks in the finetuning setting. The best scores are in **bold**, and the second best scores are in <u>underlined</u>.

Therefore, we jointly use ITM and FLM to conduct pre-training to facilitate cross-modal alignments. By doing so, we obtain considerable performance gain and reach superior performance on Flickr30K and competitive performance on COCO over prior arts, suggesting the complementarity of FLM and other alignment-oriented objectives.

5. Conclusion

In this paper, we propose free language modeling (FLM), a new pretraining objective for accelerating vision-language pretraining. Different from previous language modeling methods, such as MLM and AR, FLM seamlessly disentangles the prediction rate from the tie-up with the corruption rate, meanwhile allowing a flexible corruption pattern for each prediction target. Experiments verify the effectiveness of the proposed FLM both in accuracy and efficiency. Our model could converge faster with a decent reduction of training time compared to MLM, while achieving comparable performance on multiple multimodal downstream tasks.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *International Conference on Computer Vision (ICCV)*, 2015. 5

[2] Hangbo Bao, Wenhui Wang, Li Dong, and Furu Wei. Vl-beit: Generative vision-language pretraining. *arXiv preprint arXiv:2206.01127*, 2022. 2

[3] Yonatan Bitton, Gabriel Stanovsky, Michael Elhadad, and Roy Schwartz. Data efficient masked language modeling for vision and language. *arXiv preprint arXiv:2109.02040*, 2021. 3

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

[5] Jaeseok Byun, Taebaek Hwang, Jianlong Fu, and Taesup Moon. Grit-vlp: Grouped mini-batch sampling for efficient vision and language pre-training. *arXiv preprint arXiv:2208.04060*, 2022. 3

[6] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 5

[7] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. UNITER: Universal image-text representation learning. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 5, 8

[8] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022. 3

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019. 1, 3

[10] Zi-Yi Dou, Yichong Xu, Zhe Gan, Jianfeng Wang, Shuohang Wang, Lijuan Wang, Chenguang Zhu, Pengchuan Zhang, Lu Yuan, Nanyun Peng, et al. An empirical study of training end-to-end vision-and-language transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18166–18176, 2022. 1, 5, 6, 8

[11] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335, 2022. 3

[12] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced bert with disentangled attention. *arXiv preprint*, 2020. 3

[13] Zhicheng Huang, Zhaoyang Zeng, Bei Liu, Dongmei Fu, and Jianlong Fu. Pixel-BERT: Aligning image pixels with text by deep multi-modal transformers. *arXiv preprint*, 2020. 3, 8

[14] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. *arXiv preprint*, 2021. 1, 2

[15] Wonjae Kim, Bokyung Son, and Ildoo Kim. ViLT: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning (ICML)*, 2021. 2, 3

[16] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision (IJCV)*, 2017. 5

[17] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022. 8

[18] Junnan Li, Ramprasaath R Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2021. 1, 2, 5, 8

[19] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. VisualBERT: A simple and performant baseline for vision and language. *arXiv preprint*, 2019. 1, 2

[20] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022. 2

[21] Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. Unimo: Towards unified-modal understanding and generation via cross-modal contrastive learning. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2021. 2, 8

[22] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 7

[23] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*, 2021. 3

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014. 5

[25] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettle-

moyer, and Veselin Stoyanov. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint*, 2019. 3, 5

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[27] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019. 2

[28] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2011. 5

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021. 2, 5

[30] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018. 1, 5

[31] Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. How much can clip benefit vision-and-language tasks? *arXiv preprint*, 2021. 8

[32] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019. 1, 5

[33] Hao Tan and Mohit Bansal. LXMERT: Learning cross-modality encoder representations from transformers. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019. 3

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017. 1

[35] Teng Wang, Wenhao Jiang, Zhichao Lu, Feng Zheng, Ran Cheng, Chengguo Yin, and Ping Luo. Vlmixer: Unpaired vision-language pre-training via cross-modal cutmix. In *International Conference on Machine Learning*, pages 22680–22690. PMLR, 2022. 2

[36] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022. 2

[37] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint*, 2021. 2, 6, 8

[38] Alexander Wettig, Tianyu Gao, Zexuan Zhong, and Danqi Chen. Should you mask 15% in masked language modeling? *arXiv preprint arXiv:2202.08005*, 2022. 3

[39] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019. 3

[40] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*, 2021. 2

[41] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 2

[42] Yan Zeng, Xinsong Zhang, and Hang Li. Multi-grained vision language pre-training: Aligning texts with visual concepts. *arXiv preprint arXiv:2111.08276*, 2021. 2

[43] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. VinVL: Revisiting visual representations in vision-language models. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3, 7, 8