Playing Lottery Tickets with Vision and Language

Zhe Gan,[1] Yen-Chun Chen,[1] Linjie Li,[1] Tianlong Chen,[2]
Yu Cheng,[1] Shuohang Wang,[1] Jingjing Liu,[3] Lijuan Wang,[1] Zicheng Liu[1]

[1]Microsoft Corporation, [2]University of Texas at Austin, [3]Tsinghua University
{zhe.gan, yen-chun.chen, lindsey.li, yu.cheng, shuowa, lijuanw, zliu}@microsoft.com
tianlong.chen@utexas.edu, JJLiu@air.tsinghua.edu.cn

Abstract

Large-scale pre-training has recently revolutionized vision-and-language (VL) research. Models such as LXMERT and UNITER have significantly lifted the state of the art over a wide range of VL tasks. However, the large number of parameters in such models hinders their application in practice. In parallel, work on the lottery ticket hypothesis (LTH) has shown that deep neural networks contain small matching subnetworks that can achieve on par or even better performance than the dense networks when trained in isolation. In this work, we perform the first empirical study to assess whether such trainable subnetworks also exist in pre-trained VL models. We use UNITER as the main testbed (also test on LXMERT and ViLT), and consolidate 7 representative VL tasks for experiments, including visual question answering, visual commonsense reasoning, visual entailment, referring expression comprehension, image-text retrieval, GQA, and NLVR^2. Through comprehensive analysis, we summarize our main findings as follows. (i) It is difficult to find subnetworks that strictly match the performance of the full model. However, we can find "relaxed" winning tickets at 50%-70% sparsity that maintain 99% of the full accuracy. (ii) Subnetworks found by task-specific pruning transfer reasonably well to the other tasks, while those found on the pre-training tasks at 60%/70% sparsity transfer universally, matching 98%/96% of the full accuracy on average over all the tasks. (iii) Besides UNITER, other models such as LXMERT and ViLT can also play lottery tickets. However, the highest sparsity we can achieve for ViLT is far lower than LXMERT and UNITER (30% vs. 70%). (iv) LTH also remains relevant when using other training methods (*e.g.*, adversarial training).

Introduction

Inspired by the success of BERT (Devlin et al. 2019), vision-and-language pre-training (VLP) has becoming an increasingly central paradigm for vision-and-language (VL) research. Models such as LXMERT (Tan and Bansal 2019), ViLBERT (Lu et al. 2019a) and UNITER (Chen et al. 2020d), have achieved state-of-the-art performance across a wide range of VL tasks, such as visual question answering (VQA) (Antol et al. 2015; Goyal et al. 2017), visual commonsense reasoning (VCR) (Zellers et al. 2019), and image-text retrieval (Lee et al. 2018). Despite its empirical success, the memory and computation footprint of these pre-trained models is huge because of their large number of parameters, making it infeasible to use them in resource-constrained scenarios. A natural question that came to our mind: *Can we prune a large pre-trained VL model while preserving its performance and transferability?*

In this work, we aim to answer this question via the lens of *lottery ticket hypothesis* (LTH) (Frankle and Carbin 2019), which states that there exist matching subnetworks in dense neural networks that can be trained in isolation from initialization to reach a comparable accuracy to the

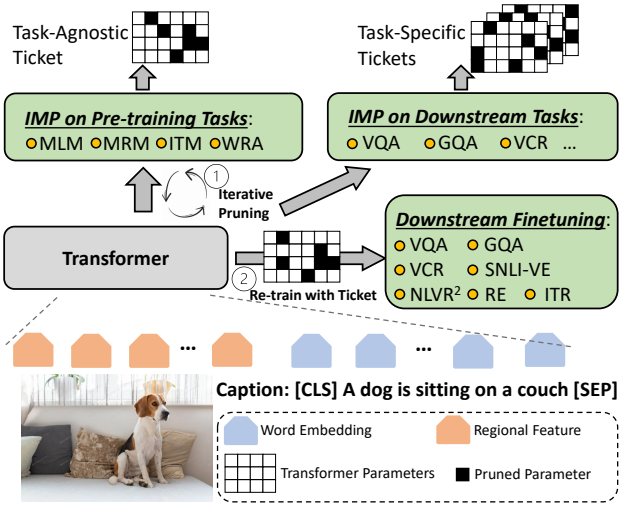


Figure 1: Overview of our training paradigm for *playing lottery tickets with vision and language*. Matching subnetworks (or winning tickets) can be found by Iterative Magnitude-based Pruning (IMP). We then re-train the found ticket with the original parameter initialization to verify the downstream performance. Not only *task-specific* winning tickets can be found when running IMP on each downstream task separately, a *task-agnostic* winning ticket is also discovered via IMP on joint pre-training. The task-agnostic ticket results in *universally transferable* subnetworks at 60%/70% sparsity that matches 98%/96% of the full accuracy averaged over all the tasks considered.

full model within similar training iterations. LTH has been shown great success in various fields (Yu et al. 2020; Renda, Frankle, and Carbin 2020; Chen et al. 2020b), and its properties have been widely studied (Malach et al. 2020; Pensia et al. 2020; Frankle et al. 2020). However, LTH has not been introduced to VL tasks yet, it could be a powerful tool to understand the parameter redundancy in the current prevailing VLP models. To start, we use UNITER (Chen et al. 2020d) as the main testbed, and consider 7 representative VL tasks for experiments, including VQA (Goyal et al. 2017), VCR (Zellers et al. 2019), GQA (Hudson and Manning 2019), NLVR^2 (Suhr et al. 2018), visual entailment (Xie et al. 2019), referring expression comprehension (Yu et al. 2016), and image-text retrieval (Lee et al. 2018). In our context, a *ticket* means a VLP subnetwork, and a *winning ticket* means a subnetwork that can match the performance of the original full VLP model. Based upon this, we ask the following three questions:

- ***Existence***: Can we draw winning tickets successfully for various VL tasks?
- ***Transferability***: Can we find tickets that transfer universally to all downstream VL tasks?
- ***Compatibility***: Do the LTH observations still hold when switching to different backbones (*e.g.*, LXMERT (Tan and Bansal 2019), ViLT (Kim, Son, and Kim 2021)), and training strategies (*e.g.*, adversarial training)?

First, *can we draw VL winning tickets?* To answer this, we use the pre-trained weights as our model initialization for task-specific finetuning, and use Iterative Magnitude-based Pruning (IMP) (Han, Mao, and Dally 2015) to draw the tickets for each VL task. However, finding tickets through iterative and repeated train-prune-retrain cycles for each task is very time-consuming, primarily when a large pre-trained model is used here. Then, it becomes critical to ask: *how can we find subnetworks that transfer universally?* If this can be achieved, the extraordinary cost of finding a winning ticket can be amortized by transferring it to a range of downstream tasks. Inspired by Chen et al. (2020b), a natural idea is to perform IMP on the pre-training tasks using the pre-training data, and assess whether such learned tickets are transferable or not, since pre-training can be considered as task-agnostic. Besides this, we further comprehensively analyze the transfer behavior among all the downstream tasks to better understand the found task-specific winning tickets.

The above analysis is conducted on UNITER, which is a one-stream model and uses an object detection module to first extract visual features offline. To study the compatibility of LTH, we also experiment on LXMERT (a two-stream model instead), and ViLT (directly taking image patches and word tokens as model inputs). Moreover, instead of cross-entropy training, we further test LTH under adversarial training (Gan et al. 2020) to investigate its corresponding training behaviors. Through comprehensive analysis, we summarize our main findings as follows.

- ***VLP can play lottery tickets too***: It is difficult to find UNITER subnetworks that *strictly* match the full performance, even with rewinding. However, it is encouraging that *"relaxed"* winning tickets that match 99% of the full accuracy can be found at 50%-70% sparsity across all the VL tasks considered.
- ***One ticket to win them all***: Matching subnetworks found via IMP on pre-training tasks transfer universally. Interestingly, matching subnetworks found on each downstream task also transfer to other tasks well, indicating that the learned task-specific subnetworks do not aggressively overfit to one specific task.
- ***Different VLP models behave differently***: Though all the VLP models can play lottery tickets, we also observe that the highest sparsity we can achieve for ViLT is far lower than LXMERT and UNITER (30% vs. 70%).
- ***Playing lottery tickets adversarially***: Compared with standard cross-entropy training, we observe that sparse winning tickets can also be identified with adversarial training, with enhanced performance.

We conclude that the primary LTH observations found in computer vision, NLP, and other areas also hold in the context of vision and language.

Related Work

Vision-and-Language Pre-training (VLP). The past two years have witnessed a boom of VLP methods. By adopting transformer (Vaswani et al. 2017) as the building block, early approaches use a two-stream architecture for multi-modal fusion (Lu et al. 2019a; Tan and Bansal 2019; Lu et al. 2019b), while single-stream architecture has gained popularity later on (Su et al. 2019; Li et al. 2019b,a; Chen et al. 2020d; Zhou et al. 2019; Gan et al. 2020; Li et al. 2020; Zhang et al. 2021). While most of these methods rely on an object detection module to extract visual features offline, recently, end-to-end VLP methods (Huang et al. 2020, 2021; Kim, Son, and Kim 2021; Xue et al. 2021; Li et al. 2021; Dou et al. 2021) are becoming increasingly popular.

Different from these efforts on making VLP models larger and stronger, we focus on a different direction, making VLP models *smaller*. Note that two recent works, MiniVLM (Wang et al. 2020a) and DistilVLM (Fang et al. 2021), have also attempted to train a smaller VLP model; however, our focus is different from theirs. Specifically, MiniVLM directly adopts MiniLM (Wang et al. 2020b) for the transformer module, while spending a larger portion of efforts on designing a compact image feature extractor; DistilVLM focuses on knowledge distillation. Here, we study the over-parameterization of VLP models via the lens of *lottery ticket hypothesis*, a popular concept in deep learning nowadays, but not introduced to VL research yet.

Lottery Ticket Hypothesis (LTH). LTH (Frankle and Carbin 2019) claims the existence of sparse, separate trainable subnetworks that are able to match or even surpass the performance of the original dense network. Though originally working only on small networks, later on, rewinding is found to be a useful technique to scale up LTH to large networks (Renda, Frankle, and Carbin 2020; Frankle et al. 2020). Since its birth, LTH has received wide attention and becomes an emerging subfield in deep learning. The properties of LTH are widely studied for image

classification (Liu et al. 2019; Evci et al. 2019; Frankle, Schwab, and Morcos 2020; Savarese, Silva, and Maire 2020; Wang, Zhang, and Grosse 2020; You et al. 2020; Ma et al. 2021). Recently, LTH has also been evidenced across other fields, such as NLP (Gale, Elsen, and Hooker 2019; Yu et al. 2020; Prasanna, Rogers, and Rumshisky 2020; Chen et al. 2020b,c), object detection (Girish et al. 2020), generative adversarial networks (Chen et al. 2021d; Kalibhat, Balaji, and Feizi 2020; Chen et al. 2021a), graph neural networks (Chen et al. 2021b), reinforcement learning (Yu et al. 2020), and life-long learning (Chen et al. 2021c).

Recent work has also started to investigate the existence of winning tickets in self-supervised pre-training of visual encoders (Chen et al. 2020a) and language models (Chen et al. 2020b,c). However, to the best of our knowledge, the study of lottery tickets in VLP remains untouched. As VLP becomes increasingly popular, it is critical to understand the parameter redundancy in such models, potentially making them small without sacrificing the performance.

Preliminaries

In this section, we detail the techniques we use to identify winning tickets, and present our setup for empirical study.

Backbones. We use UNITER (Chen et al. 2020d) as an example to introduce the backbone, which shares the same structure as BERT, except that the input is a mixed sequence of two modalities. Specifically, given a dataset that consists of image-text pairs $\boldsymbol{x} = (\boldsymbol{x}_{img}, \boldsymbol{x}_{txt})$, UNITER first encodes the corresponding image regions and textual tokens into low-dimensional feature vectors $\boldsymbol{z}_{img} = g_{bu}(\boldsymbol{x}_{img})$ and $\boldsymbol{z}_{txt} = g_{emb}(\boldsymbol{x}_{txt})$, where $g_{bu}(\cdot)$ is the fixed bottom-up image feature extractor (Anderson et al. 2018), $g_{emb}(\cdot)$ is a learnable word embedding function. Then, a transformer is applied on top to obtain contextualized representations: $\tilde{\boldsymbol{z}}_{img}, \tilde{\boldsymbol{z}}_{txt}, \tilde{\boldsymbol{z}}_{cls} = f_1(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt}; \boldsymbol{\theta})$, where a special `[CLS]` token is employed whose embedding $\tilde{\boldsymbol{z}}_{cls}$ is considered as the joint multimodal representation. $\boldsymbol{\theta} \in \mathbb{R}^{d_1}$ includes all the trainable parameters. For a particular downstream task, we add a final, task-specific classification layer on top of $\tilde{\boldsymbol{z}}_{cls}$ to obtain the output logit vector $f_2(\tilde{\boldsymbol{z}}_{cls}; \boldsymbol{\phi})$, where $\boldsymbol{\phi} \in \mathbb{R}^{d_2}$ denotes task-specific parameters. The whole UNITER network is abbreviated as $f(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi})$ that absorbs both $f_1(\cdot, \cdot)$ and $f_2(\cdot)$. For LXMERT (Tan and Bansal 2019), it takes the same image features from object detection as model input, but adopts a two-stream model architecture instead. For ViLT (Kim, Son, and Kim 2021), it uses the same one-stream architecture, but directly takes image patches and word tokens as inputs, and models all the intra- and inter-modality interaction via a single unified transformer.

Given the task-specific supervision signal $\boldsymbol{y}$ (typically a label in VL tasks), model training can be summarized as:

$$\min_{\boldsymbol{\theta}, \boldsymbol{\phi}} \mathbb{E}_{(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}} [L(f(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}), \boldsymbol{y})] \,, \tag{1}$$

where $L(\cdot)$ is the cross-entropy loss, and $\mathcal{D}$ denotes the dataset for a downstream task. We use the official UNITER/LXMERT/ViLT code bases for experiments.

Subnetworks. A subnetwork of $f(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi})$ means a network $f(\boldsymbol{x}; \boldsymbol{m} \odot \boldsymbol{\theta}, \boldsymbol{\phi})$ with a binary pruning mask $\boldsymbol{m} \in \{0, 1\}^{d_1}$ indicating which part of the parameters are set to 0, and $\odot$ is the element-wise product. Following Frankle and Carbin (2019), we define a *matching subnetwork* as a subnetwork that can be trained to the full accuracy of the dense network within similar training iterations. A *winning ticket* is defined as a matching subnetwork $f(\boldsymbol{x}; \boldsymbol{m} \odot \boldsymbol{\theta}_0, \cdot)$ where $\boldsymbol{\theta} = \boldsymbol{\theta}_0$, which is typically is a random weight initialization. However, in our context, $\boldsymbol{\theta}_0$ represents the pre-trained model weights. We also define a "*relaxed*" winning ticket as one that matches $p\%$ of the the full accuracy, where p is set to a large number close to 100 (such as 99).

Finding Subnetworks. As used in many lottery ticket papers, we use Iterative Magnitude-based Pruning (IMP) (Han, Mao, and Dally 2015) to find the subnetwork. Specifically, the pruning mask $\boldsymbol{m}$ is determined by training the unpruned network to completion on a downstream task, then pruning individual weights with the lowest magnitudes globally throughout the network. The weights are then reset to the pre-trained initialization $\boldsymbol{\theta}_0$ (or, $\boldsymbol{\theta}_i$ for a specific *rewinding* step i in training), and only the learned mask $\boldsymbol{m}$ is stored. We prune a certain amount (*e.g.*, 10%) of non-zero weights after completion, and re-train the network several times to meet the sparsity requirement. The full IMP procedure is provided in the Appendix.

We consider finding subnetworks via both (i) task-specific finetuning and (ii) task-agnostic pre-training,[1] hoping that universal transferable subnetworks can be identified. For UNITER pre-training, we use all the pre-training tasks to learn the mask, including Masked Language Modeling, Masked Region Modeling, Image-Text Matching, and Word-Region Alignment. See Chen et al. (2020d) for details of these tasks. As our model is initialized by pre-trained UNITER, we further pre-train only 10% of original training steps in each pruning round (we prune 9 rounds in total). Therefore, the total time spent for a full IMP process roughly equals the time used for pre-training UNITER from scratch.

Evaluation of Subnetworks. For a particular downstream task, after obtaining a subnetwork $f(\boldsymbol{x}; \boldsymbol{m} \odot \boldsymbol{\theta}, \cdot)$, we reset the weights to $\boldsymbol{\theta}_0$ or $\boldsymbol{\theta}_i$ (if rewinding is used), and then completely re-train the subnetwork to test whether the final subnetworks can still achieve the original accuracy. For pre-training, since the performance of the pre-training tasks validation loss does not correlate to the task-specific performance (Chen et al. 2020d), we finetune and test the identified subnetworks on all the downstream tasks. We use both the in-domain and out-of-domain image-text datasets for IMP-based pre-training, including COCO (Lin et al. 2014), Visual Genome (Krishna et al. 2017), Conceptual Captions (Sharma et al. 2018), and SBU Captions (Ordonez, Kulkarni, and Berg 2011).

Downstream Tasks. We consider 7 VL tasks for experiments. (i) For VQA (Goyal et al. 2017), GQA (Hudson and

[1] We only perform pre-training on UNITER, since pre-training is heavy; we perform finetuning for UNITER, LXMERT, and ViLT.

#	Dataset / Sparsity	VQA mini-dev† 70%	GQA test-dev 70%	VCR Q→AR val 50%	NLVR^2 dev 60%	SNLI-VE val 60%	RefCOCO+ val^d 70%	Flickr30k IR R@1 60%	Flickr30k TR R@1 60%
1	UNITER$_{\text{B}}$ (paper)	70.75	–	54.94	77.18	78.59	75.31	72.52	85.90
2	UNITER$_{\text{B}}$ (reimp.)	70.64±0.06	59.64±0.15	54.37±0.31‡	76.75±0.19	78.47±0.10	74.73±0.06	71.25±0.11⋆	84.63±1.02⋆
3	×99%	69.93	59.04	53.83	75.98	77.69	73.98	70.54	83.78
4	$f(\boldsymbol{x}; \boldsymbol{m}_{\text{IMP}} \cdot \boldsymbol{\theta}_0)$	69.98±0.05	59.26±0.09	53.15±1.02	76.32±0.41	77.69±0.07	74.06±0.27	70.15±0.71	83.77±0.76
5	$f(\boldsymbol{x}; \boldsymbol{m}_{\text{RP}} \cdot \boldsymbol{\theta}_0)$	60.45	55.95	25.35	52.42	71.30	72.95	61.44	76.80
6	$f(\boldsymbol{x}; \boldsymbol{m}_{\text{IMP}} \cdot \boldsymbol{\theta}'_0)$	67.98	58.45	50.39	54.15	76.45	71.09	63.38	79.30
7	$f(\boldsymbol{x}; \boldsymbol{m}_{\text{IMP}} \cdot \boldsymbol{\theta}''_0)$	60.46	47.49	6.25	51.52	69.32	67.34	38.94	48.00

Table 1: Performance of subnetworks at the highest sparsity for which IMP finds "relaxed" winning tickets that maintains 99% of the full accuracy on each task. Entries with ± are the average across three runs. IMP: Iterative Magnitude Pruning; RP: Random Pruning; $\boldsymbol{\theta}_0$: pre-trained UNITER weights; $\boldsymbol{\theta}'_0$: pre-trained BERT weights; $\boldsymbol{\theta}''_0$: randomly shuffled pre-trained UNITER weights. (†) To avoid submitting results to the VQA test server too frequently, instead of reporting results on test-dev/-std sets, we use a mini-dev set for comparison. The same min-dev set was also used in UNITER. (‡) For fair comparison on transfer learning, we did not perform 2-nd stage pre-training for VCR task as in UNITER. (⋆) To rule out other factors that may influence results besides pruning, we did not use hard negative mining as in UNITER.

Manning 2019) and VCR (Zellers et al. 2019), given an image and an input question, the model selects an answer from a candidate pool. (*ii*) For NLVR^2 (Suhr et al. 2018), given a pair of images and a natural language description, the model judges the correctness of the description based on the input image pair. For Visual Entailment (Xie et al. 2019), the model predicts whether a given image entails a given sentence. (*iii*) For Referring Expression (RE) Comprehension, we evaluate on RefCOCO+ (Yu et al. 2016), where given a text description, the model selects the described region from a set of image region proposals. (*iv*) For Image-Text Retrieval (ITR), we consider both image retrieval and text retrieval on Flickr30k dataset.

For VCR, 2nd-stage pre-training was found useful in UNITER finetuning. For simplicity and ease of study of transfer learning, we do not use 2nd-stage pre-training. For ITR, hard negative mining is necessary to boost performance. We do not use this as it is computationally heavy, and we aim to study LTH rather than chasing state-of-the-art performance. For VQA, we mainly report results on an internal mini-dev set for faster evaluation of the found tickets, and avoid submitting results to the VQA test server too frequently. This same mini-dev set is also used in UNITER (Chen et al. 2020d).

Experiments

In this section, we perform extensive experiments to examine the LTH in the context of vision and language.

VLP Can Play Lottery Tickets Too

First, we evaluate whether winning tickets exist in UNITER. In particular, we answer the following questions.

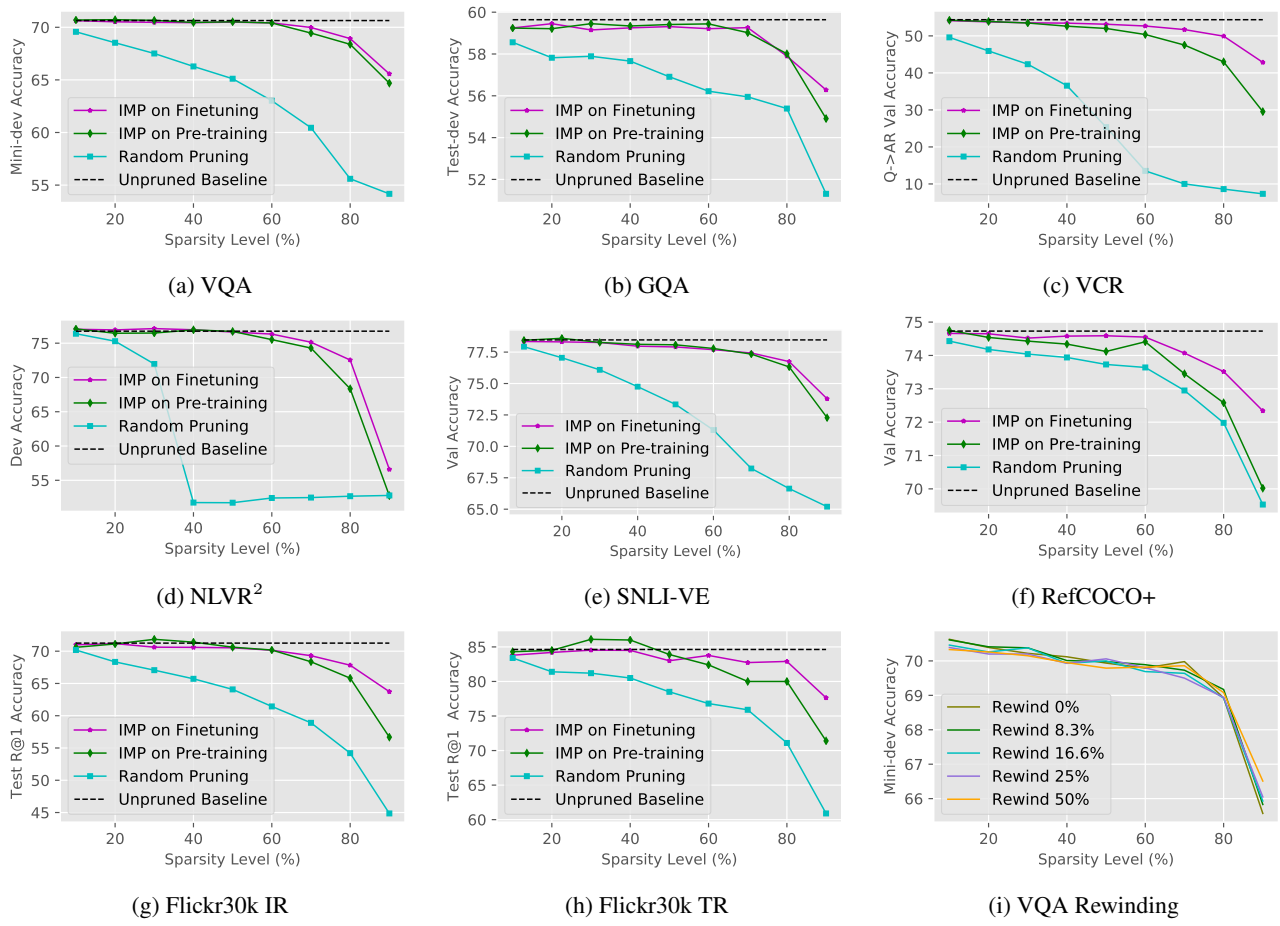
Q1: Are there winning tickets in UNITER? To answer this, we first run IMP on a downstream task $\mathcal{T}$ to obtain a sparsity pattern $\boldsymbol{m}^{\mathcal{T}}_{\text{IMP}}$. This produces a subnetwork $f(\boldsymbol{x}; \boldsymbol{m}^{\mathcal{T}}_{\text{IMP}} \odot \boldsymbol{\theta}_0, \cdot)$. We then train this subnetwork again on task $\mathcal{T}$ to evaluate whether this is a winning ticket.

Results across all the sparsity levels (10% to 90%) on all the downstream tasks are shown in Figure 2 (magenta curves). For tasks of image-text retrieval and NLVR^2, matching subnetworks with sparsity 40% can be identified. However, it is generally challenging to find subnetworks that "strictly" match the performance of the full accuracy on the other tasks. Therefore, we define "*relaxed*" winning tickets as the ones that can match 99% of the full accuracy. It will still be encouraging if such subnetworks can be found.

Results are summarized in Table 1. Row #1 reports the full UNITER$_{\text{B}}$ performance reported in the UNITER paper (Chen et al. 2020d). Row #2 reports the results of our re-implementation, where different random seeds are used to account for fluctuations. We use the default hyperparameters provided in the UNITER code base without any tuning. Row #3 calculates 99% of the full accuracy on each task for reference. As can be seen from Row #4, on all VL tasks, "relaxed" winning tickets can be found. The highest sparsities range from 50% (*e.g.*, VCR) to 70% (*e.g.*, VQA). For VCR, it is challenging to find high-sparsity subnetworks. We hypothesize that commonsense knowledge is harder to learn, and smaller weights also play essential roles in improving model's commonsense reasoning abilities, making the subnetwork for VCR harder to prune.

Q2: Are winning tickets sparser than randomly pruned or initialized subnetworks? Previous work has shown that both the specific learned sparse mask and the specific initialization are necessary for finding winning tickets (Frankle and Carbin 2019). To assess the importance of the learned mask in the context of UNITER, we compare with a random pruning baseline, and report results in Row #5 of Table 1. That is, we finetune a randomly pruned UNITER model on each downstream task. Interestingly, for some tasks (*e.g.*, GQA and RefCOCO+), random pruning achieves pretty strong performance. However, by comparing performance across the board, it is also clear that random pruning performs far worse than the identified winning tickets. In Figure 2, we further compare IMP and random pruning across all sparsities. Again, random pruning achieves far lower performance, confirming that the sparse structure found by IMP is crucial for the good performance of subnetworks.

To assess the importance of the initialization, we con-



(a) VQA (b) GQA (c) VCR

(d) NLVR^2 (e) SNLI-VE (f) RefCOCO+

(g) Flickr30k IR (h) Flickr30k TR (i) VQA Rewinding

Figure 2: Comparison among (i) IMP performed on task-specific finetuning, (ii) IMP performed on task-agnostic pre-training, and (iii) random pruning on task-specific finetuning across sparsities for all the tasks. We also report rewinding in the VQA task in sub-figure (i).

sider two different initializations with the learned mask unchanged: (i) using pre-trained BERT weights $\boldsymbol{\theta}_0'$ as initialization, and (ii) shuffling the UNITER pre-trained weights within each layer to obtain a new initialization $\boldsymbol{\theta}_0''$. Results of these two baselines are summarized in Row #6 and #7 of Table 1, respectively. Clearly, training from $\boldsymbol{\theta}_0''$ achieves far lower performance than training from $\boldsymbol{\theta}_0$. However, it is also interesting to observe that training from $\boldsymbol{\theta}_0'$ achieves much more reasonable performance, though still lagging behind training from $\boldsymbol{\theta}_0$, indicating the importance of the specific initialization. We hypothesize the good performance of $\boldsymbol{\theta}_0'$ is partially due to that $\boldsymbol{\theta}_0'$ is used as the initialization to pre-train UNITER; therefore, the structure of the UNITER weights may be partially inherited from BERT.

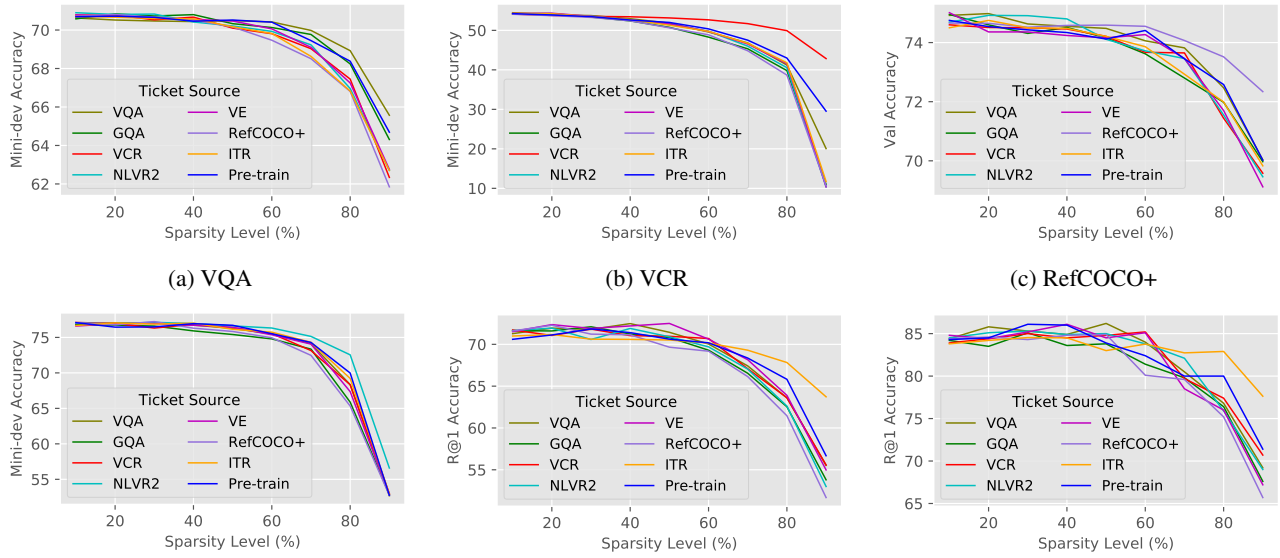
Q3: Does rewinding improve performance? For large networks, *rewinding* is found to be necessary to identify winning tickets (Renda, Frankle, and Carbin 2020). After obtaining the masks, instead of resetting the weights to $\boldsymbol{\theta}_0$, one should rewind the weights to $\boldsymbol{\theta}_i$, the weights after i steps of training. To examine whether rewinding is helpful in the context of UNITER, we run experiments at different rewinding ratios using VQA as the representative task. Results are shown in Figure 2(i). Rewinding does not have a notable effect on the VQA performance, with only minor performance improvement observed at high-sparsity ratio (90%). Similar observations are also found on other downstream tasks.

One Ticket to Win Them All

Finding winning tickets on each downstream task separately is time-consuming, as each time when IMP is performed, it has to go through the full train-prune-retrain cycle multiple times. In this section, we aim to identify subnetworks that transfer well across all the VL tasks. In particular, we answer the following questions.

Q4: Do winning tickets found on pre-training tasks transfer? Pre-training is believed to learn *universal* VL representations. As shown in Cao et al. (2020), the pre-trained weights indeed have captured rich visual coreference and visual relation knowledge. This naturally leads to our hypothesis: can the subnetwork identified by the pre-training tasks on the pre-training data also transfer universally?

To study this, we first identify a subnetwork $f(\boldsymbol{x}; \boldsymbol{m}_{\text{IMP}}^{\mathcal{P}} \cdot$



(a) VQA (b) VCR (c) RefCOCO+

(d) NLVR^2 (e) Flickr30k IR (f) Flickr30k TR

Figure 3: Transferring winning tickets across tasks. Winning ticket performance on target tasks: (a) VQA, (b) VCR, (c) RefCOCO+, (d) NLVR^2, (e) Flickr30k IR, (f) Flickr30k TR. Within each plot, each line represents a different source task for the winning ticket. Better zoomed in and viewed in color. Additional curves are in the Appendix.

Sparisty	VQA	GQA	VCR	NLVR^2	SNLI-VE	RefCOCO+	Flickr30k IR	Flickr30k TR	Ave. Perf. Drop (%)	
	mini-dev	test-dev	Q→AR val	dev	val	val^d	R@1	R@1	All	w/o VCR
0%	70.64	59.64	54.37	76.75	78.47	74.73	71.25	84.63	–	–
50%	70.52	59.41	52.01	76.71	78.08	74.12	70.62	83.90	1.00	0.52
60%	70.41	59.44	50.37	75.52	77.79	74.41	70.18	82.40	1.88	1.10
70%	69.45	59.02	47.52	74.29	77.34	73.45	68.36	80.00	3.90	2.66
80%	68.38	58.01	42.99	69.98	76.32	72.58	65.82	80.00	6.80	4.78

Table 2: Performance of the universal transferable subnetwork found on pre-training at specified sparsities.

$\boldsymbol{\theta}_0, \cdot)$ on the pre-training tasks $\mathcal{T}$, and then train it on all the downstream tasks to evaluate its performance. Results are summarized in Figure 2 (green curves). Interestingly, though pre-training never obtains the supervision signal in the downstream tasks, the found subnetwork transfers pretty *universally*; only when the sparsity is high (*e.g.*, 80%, 90%), the found subnetwork performs worse than the ones found by task-specific IMP, indicating that the pre-training tasks are strong signals for learning how to prune.

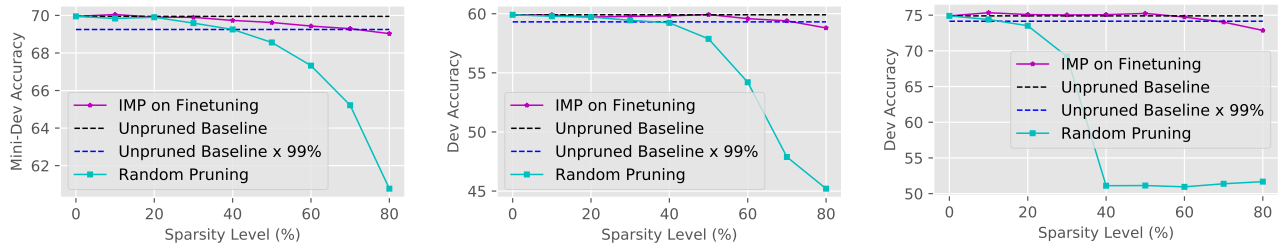
Q5: Do winning tickets found on downstream tasks transfer? One would also wonder whether such transfer learning behavior also exists among the downstream tasks themselves, *i.e.*, whether the found subnetwork on a source task $\mathcal{S}$ transfers to a target task $\mathcal{T}$. We perform a systematic study in Figure 3, where within each plot, 8 ticket sources are considered. There are several key observations. (i) The subnetworks found by task-specific signals typically perform the best, especially on the high-sparsity regime. (ii) Surprisingly, all the individual subnetworks found by downstream tasks transfer well, indicating that models on all the tasks have learned some shared essential knowledge. (iii) The subnetwork from pre-training generally performs better than those from other tasks (*e.g.*, 0.71%-2.69% better than other source tickets at 70% sparsity on the VCR task), indicating its universal transferability. By taking a closer look at Figure 3(a), excluding VQA itself, the best source ticket is from pre-training and GQA, as the task nature of VQA and GQA is similar. From Figure 3(e) and (f), we can see the best source ticket for image-text retrieval is from pre-training. This is because the image-text matching task used in pre-training is similar to the downstream task itself. In Appendix, we also compare the similarity of sparsity patterns found on each downstream task.

Since subnetworks found on pre-training performs the best, we further compare their performance with the full model in more detail, and summarize results in Table 2. The universal subnetwork at 60%/70% sparsity matches 98%/96%[2] of the full accuracy over all the tasks considered, effectively serving as a task-agnositic compressed model.

Additional Study

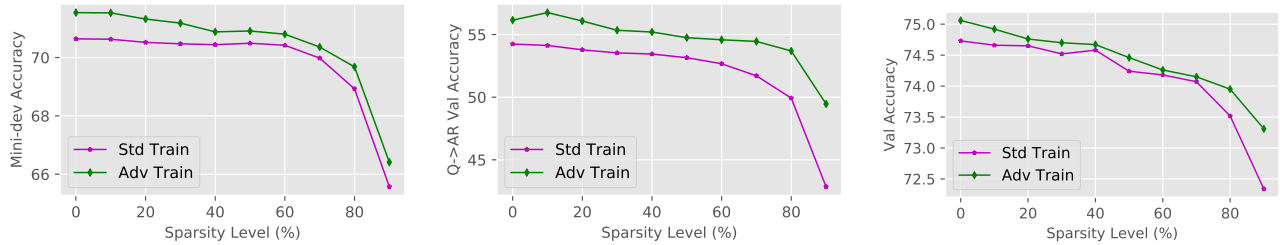
Q6: Do different VLP models behave differently? So far, we have focused on UNITER. Below, we experiment with

[2]This number changes to 99%/97% if VCR is not counted in.



(a) VQA (b) GQA (c) NLVR^2

Figure 4: The lottery ticket results of LXMERT on VQA, GQA, and NLVR^2.



(a) VQA (b) VCR (c) RefCOCO+

Figure 5: Performance of subnetworks that are found by adversarial training on the tasks of VQA, VCR and RefCOCO+.

Dataset	VQA mini-dev†	GQA test-dev	NLVR^2 dev
Sparsity	70%	70%	70%
LXMERT (paper)	69.90	59.80	74.95
LXMERT (reimp.)	69.95±0.03	59.91±0.07	74.90±0.26
×99%	69.25	59.31	74.15
Lottery Tickets	69.29±0.10	59.40±0.17	74.03±0.71
Random Pruning	65.22±0.05	47.88±0.55	51.38±0.45

Table 3: The LTH results of LXMERT on VQA, GQA, and NLVR^2. (†) The same mini-dev set as used in LXMERT.

Dataset	VQA (mini-dev†)	NLVR^2 (dev)
Sparsity	30%	30%
ViLT (reimp.)	70.88±0.05	75.82±0.20
×99%	70.17	75.06
Lottery Tickets	70.51±0.11	75.22±0.41
Random Pruning	65.16±0.05	56.14±0.40

Table 4: The lottery ticket results of ViLT on VQA and NLVR^2. (†) The same mini-dev set as used in ViLT.

LXMERT and ViLT to provide a more complete picture of VL lottery tickets. Results are summarized in Table 3, 4, and Figure 4. For LXMERT, similar observations can be found. Since both UNITER and LXMERT use the same visual features from object detection, but only differ in the use of one-/two-stream architecture, we conclude that the LTH observations are not sensitive to this one-/two-stream design. On the other hand, ViLT can only achieve a low sparsity ratio (30%) if we want to keep impaired performance. This is partially due to that ViLT directly takes image patches as input, all the modeling power needs to be absorbed in a single unified transformer, therefore less can be pruned, while for UNITER and LXMERT, the extracted image features are kept intact.

Q7: Can VLP models play lottery tickets adversarially? Lottery tickets are typically found via standard cross-entropy training. Here, we study whether adversarial training can be used to find winning tickets as well. Results are shown in Figure 5. Interestingly, on the 3 tasks considered, the ticket performance via adversarial training at 80% and 70% sparsity matches (or almost matches) the performance via standard finetuning at 70% and 60% sparsity, respectively. This suggests that adversarial training has the effect of making the sparse winning tickets 10% sparser in order to match the performance of a standard trained one.

Conclusion and Discussion

In this paper, we have presented a comprehensive study of the lottery ticket hypothesis (LTH) for vision and language. Below, we discuss some limitations of the current study. (i) *Efficiency*: We mainly focused on the scientific study of LTH. For future work, we plan to investigate the real speedup results on a hardware platform that is friendly to unstructured pruning, such as XNNPACK (Elsen et al. 2020). (ii) *Object Detection*: For UNITER/LXMERT, we studied the LTH for multimodal fusion, while keeping the object detection module untouched. In terms of end-to-end VLP, we focused on ViLT. For future work, we plan to study the LTH of object detection and other end-to-end VLP models.

References

Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*.

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Lawrence Zitnick, C.; and Parikh, D. 2015. Vqa: Visual question answering. In *ICCV*.

Cao, J.; Gan, Z.; Cheng, Y.; Yu, L.; Chen, Y.-C.; and Liu, J. 2020. Behind the Scene: Revealing the Secrets of Pre-trained Vision-and-Language Models. In *ECCV*.

Chen, T.; Cheng, Y.; Gan, Z.; Liu, J.; and Wang, Z. 2021a. Ultra-Data-Efficient GAN Training: Drawing A Lottery Ticket First, Then Training It Toughly. *arXiv preprint arXiv:2103.00397*.

Chen, T.; Frankle, J.; Chang, S.; Liu, S.; Zhang, Y.; Carbin, M.; and Wang, Z. 2020a. The Lottery Tickets Hypothesis for Supervised and Self-supervised Pre-training in Computer Vision Models. *arXiv preprint arXiv:2012.06908*.

Chen, T.; Frankle, J.; Chang, S.; Liu, S.; Zhang, Y.; Wang, Z.; and Carbin, M. 2020b. The lottery ticket hypothesis for pre-trained bert networks. In *NeurIPS*.

Chen, T.; Sui, Y.; Chen, X.; Zhang, A.; and Wang, Z. 2021b. A Unified Lottery Ticket Hypothesis for Graph Neural Networks. *arXiv preprint arXiv:2102.06790*.

Chen, T.; Zhang, Z.; Liu, S.; Chang, S.; and Wang, Z. 2021c. Long Live the Lottery: The Existence of Winning Tickets in Lifelong Learning. In *ICLR*.

Chen, X.; Cheng, Y.; Wang, S.; Gan, Z.; Wang, Z.; and Liu, J. 2020c. EarlyBERT: Efficient BERT Training via Early-bird Lottery Tickets. *arXiv preprint arXiv:2101.00063*.

Chen, X.; Zhang, Z.; Sui, Y.; and Chen, T. 2021d. GANs Can Play Lottery Tickets Too. In *ICLR*.

Chen, Y.-C.; Li, L.; Yu, L.; El Kholy, A.; Ahmed, F.; Gan, Z.; Cheng, Y.; and Liu, J. 2020d. Uniter: Universal image-text representation learning. In *ECCV*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Dou, Z.-Y.; Xu, Y.; Gan, Z.; Wang, J.; Wang, S.; Wang, L.; Zhu, C.; Liu, Z.; Zeng, M.; et al. 2021. An Empirical Study of Training End-to-End Vision-and-Language Transformers. *arXiv preprint arXiv:2111.02387*.

Elsen, E.; Dukhan, M.; Gale, T.; and Simonyan, K. 2020. Fast sparse convnets. In *CVPR*.

Evci, U.; Pedregosa, F.; Gomez, A.; and Elsen, E. 2019. The difficulty of training sparse neural networks. *arXiv preprint arXiv:1906.10732*.

Fang, Z.; Wang, J.; Hu, X.; Wang, L.; Yang, Y.; and Liu, Z. 2021. Compressing Visual-linguistic Model via Knowledge Distillation. In *ICCV*.

Frankle, J.; and Carbin, M. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *ICLR*.

Frankle, J.; Dziugaite, G. K.; Roy, D.; and Carbin, M. 2020. Linear mode connectivity and the lottery ticket hypothesis. In *ICML*.

Frankle, J.; Schwab, D. J.; and Morcos, A. S. 2020. The Early Phase of Neural Network Training. In *ICLR*.

Gale, T.; Elsen, E.; and Hooker, S. 2019. The state of sparsity in deep neural networks. *arXiv preprint arXiv:1902.09574*.

Gan, Z.; Chen, Y.-C.; Li, L.; Zhu, C.; Cheng, Y.; and Liu, J. 2020. Large-scale adversarial training for vision-and-language representation learning. In *NeurIPS*.

Girish, S.; Maiya, S. R.; Gupta, K.; Chen, H.; Davis, L.; and Shrivastava, A. 2020. The Lottery Ticket Hypothesis for Object Recognition. *arXiv preprint arXiv:2012.04643*.

Goyal, Y.; Khot, T.; Summers-Stay, D.; Batra, D.; and Parikh, D. 2017. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *CVPR*.

Han, S.; Mao, H.; and Dally, W. J. 2015. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*.

Huang, Z.; Zeng, Z.; Huang, Y.; Liu, B.; Fu, D.; and Fu, J. 2021. Seeing Out of tHe bOx: End-to-End Pre-training for Vision-Language Representation Learning. In *CVPR*.

Huang, Z.; Zeng, Z.; Liu, B.; Fu, D.; and Fu, J. 2020. Pixel-BERT: Aligning Image Pixels with Text by Deep Multi-Modal Transformers. *arXiv preprint arXiv:2004.00849*.

Hudson, D. A.; and Manning, C. D. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*.

Kalibhat, N. M.; Balaji, Y.; and Feizi, S. 2020. Winning Lottery Tickets in Deep Generative Models. *arXiv preprint arXiv:2010.02350*.

Kim, W.; Son, B.; and Kim, I. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*.

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*.

Lee, K.-H.; Chen, X.; Hua, G.; Hu, H.; and He, X. 2018. Stacked cross attention for image-text matching. In *ECCV*.

Li, G.; Duan, N.; Fang, Y.; Jiang, D.; and Zhou, M. 2019a. Unicoder-VL: A Universal Encoder for Vision and Language by Cross-modal Pre-training. *arXiv preprint arXiv:1908.06066*.

Li, J.; Selvaraju, R. R.; Gotmare, A. D.; Joty, S.; Xiong, C.; and Hoi, S. 2021. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*.

Li, L. H.; Yatskar, M.; Yin, D.; Hsieh, C.-J.; and Chang, K.-W. 2019b. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Li, X.; Yin, X.; Li, C.; Hu, X.; Zhang, P.; Zhang, L.; Wang, L.; Hu, H.; Dong, L.; Wei, F.; et al. 2020. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. *arXiv preprint arXiv:2004.06165*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Liu, Z.; Sun, M.; Zhou, T.; Huang, G.; and Darrell, T. 2019. Rethinking the Value of Network Pruning. In *ICLR*.

Lu, J.; Batra, D.; Parikh, D.; and Lee, S. 2019a. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*.

Lu, J.; Goswami, V.; Rohrbach, M.; Parikh, D.; and Lee, S. 2019b. 12-in-1: Multi-Task Vision and Language Representation Learning. *arXiv preprint arXiv:1912.02315*.

Ma, H.; Chen, T.; Hu, T.-K.; You, C.; Xie, X.; and Wang, Z. 2021. Good Students Play Big Lottery Better. *arXiv preprint arXiv:2101.03255*.

Malach, E.; Yehudai, G.; Shalev-Schwartz, S.; and Shamir, O. 2020. Proving the lottery ticket hypothesis: Pruning is all you need. In *ICML*.

Ordonez, V.; Kulkarni, G.; and Berg, T. 2011. Im2text: Describing images using 1 million captioned photographs. In *NeurIPS*.

Pensia, A.; Rajput, S.; Nagle, A.; Vishwakarma, H.; and Papailiopoulos, D. 2020. Optimal lottery tickets via subsetsum: Logarithmic over-parameterization is sufficient. *arXiv preprint arXiv:2006.07990*.

Prasanna, S.; Rogers, A.; and Rumshisky, A. 2020. When bert plays the lottery, all tickets are winning. *arXiv preprint arXiv:2005.00561*.

Renda, A.; Frankle, J.; and Carbin, M. 2020. Comparing rewinding and fine-tuning in neural network pruning. In *ICLR*.

Savarese, P.; Silva, H.; and Maire, M. 2020. Winning the Lottery with Continuous Sparsification. In *NeurIPS*.

Sharma, P.; Ding, N.; Goodman, S.; and Soricut, R. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*.

Su, W.; Zhu, X.; Cao, Y.; Li, B.; Lu, L.; Wei, F.; and Dai, J. 2019. VL-BERT: Pre-training of Generic Visual-Linguistic Representations. *arXiv preprint arXiv:1908.08530*.

Suhr, A.; Zhou, S.; Zhang, A.; Zhang, I.; Bai, H.; and Artzi, Y. 2018. A corpus for reasoning about natural language grounded in photographs. *arXiv preprint arXiv:1811.00491*.

Tan, H.; and Bansal, M. 2019. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *EMNLP*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Wang, C.; Zhang, G.; and Grosse, R. 2020. Picking Winning Tickets Before Training by Preserving Gradient Flow. In *ICLR*.

Wang, J.; Hu, X.; Zhang, P.; Li, X.; Wang, L.; Zhang, L.; Gao, J.; and Liu, Z. 2020a. MiniVLM: A Smaller and Faster Vision-Language Model. *arXiv preprint arXiv:2012.06946*.

Wang, W.; Wei, F.; Dong, L.; Bao, H.; Yang, N.; and Zhou, M. 2020b. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *arXiv preprint arXiv:2002.10957*.

Xie, N.; Lai, F.; Doran, D.; and Kadav, A. 2019. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*.

Xue, H.; Huang, Y.; Liu, B.; Peng, H.; Fu, J.; Li, H.; and Luo, J. 2021. Probing Inter-modality: Visual Parsing with Self-Attention for Vision-Language Pre-training. In *NeurIPS*.

You, H.; Li, C.; Xu, P.; Fu, Y.; Wang, Y.; Chen, X.; Baraniuk, R. G.; Wang, Z.; and Lin, Y. 2020. Drawing Early-Bird Tickets: Toward More Efficient Training of Deep Networks. In *ICLR*.

Yu, H.; Edunov, S.; Tian, Y.; and Morcos, A. S. 2020. Playing the lottery with rewards and multiple languages: lottery tickets in rl and nlp. In *ICLR*.

Yu, L.; Poirson, P.; Yang, S.; Berg, A. C.; and Berg, T. L. 2016. Modeling context in referring expressions. In *ECCV*.

Zellers, R.; Bisk, Y.; Farhadi, A.; and Choi, Y. 2019. From recognition to cognition: Visual commonsense reasoning. In *CVPR*.

Zhang, P.; Li, X.; Hu, X.; Yang, J.; Zhang, L.; Wang, L.; Choi, Y.; and Gao, J. 2021. VinVL: Making Visual Representations Matter in Vision-Language Models. *arXiv preprint arXiv:2101.00529*.

Zhou, L.; Palangi, H.; Zhang, L.; Hu, H.; Corso, J. J.; and Gao, J. 2019. Unified vision-language pre-training for image captioning and vqa. *arXiv preprint arXiv:1909.11059*.