Deep Multimodal Neural Architecture Search

Zhou Yu, Yuhao Cui, Jun Yu*
Key Laboratory of Complex Systems Modeling and Simulation, School of Computer Science and Technology, Hangzhou Dianzi University, China
{yuz,cuiyh,yujun}@hdu.edu.cn

Meng Wang
School of Computer Science and Information Engineering, Hefei University of Technology, China
eric.mengwang@gmail.com

Dacheng Tao
UBTECH Sydney AI Centre, School of Computer Science, FEIT, The University of Sydney, Australia
dacheng.tao@sydney.edu.au

Qi Tian
Huawei Cloud & AI
China
tian.qi1@huawei.com

ABSTRACT

Designing effective neural networks is fundamentally important in deep multimodal learning. Most existing works focus on a single task and design neural architectures manually, which are highly task-specific and hard to generalize to different tasks. In this paper, we devise a generalized deep multimodal neural architecture search (MMnas) framework for various multimodal learning tasks. Given multimodal input, we first define a set of primitive operations, and then construct a deep encoder-decoder based unified backbone, where each encoder or decoder block corresponds to an operation searched from a predefined operation pool. On top of the unified backbone, we attach task-specific heads to tackle different multimodal learning tasks. By using a gradient-based NAS algorithm, the optimal architectures for different tasks are learned efficiently. Extensive ablation studies, comprehensive analysis, and comparative experimental results show that the obtained MMnasNet significantly outperforms existing state-of-the-art approaches across three multimodal learning tasks (over five datasets), including visual question answering, image-text matching, and visual grounding.

CCS CONCEPTS

• **Computing methodologies** → **Multi-task learning**; **Neural networks**.

KEYWORDS

multimodal learning; neural networks; neural architecture search

ACM Reference Format:
Zhou Yu, Yuhao Cui, Jun Yu, Meng Wang, Dacheng Tao, and Qi Tian. 2020. Deep Multimodal Neural Architecture Search. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3394171.3413977

*Jun Yu is the corresponding author.

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413977

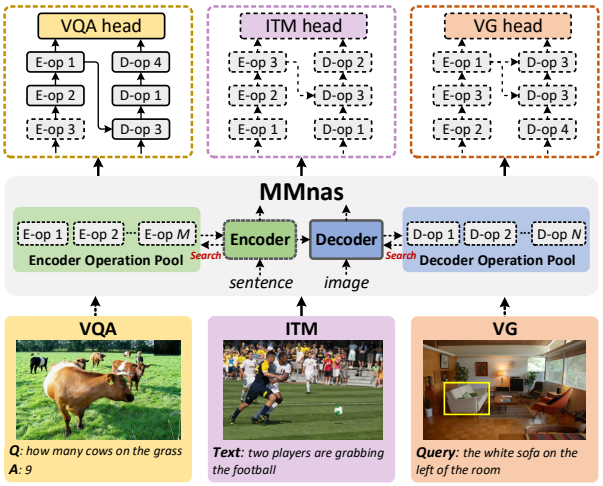


Figure 1: Schematic of the proposed generalized MMnas framework, which searches for the optimal architectures for the VQA, image-text matching, and visual grounding tasks.

1 INTRODUCTION

The developments in deep neural networks enable the machine to deal with complicated multimodal learning tasks that require a fine-grained understanding of both vision and language clues, *e.g.*, visual captioning [1, 51], visual grounding [43, 53], image-text matching [26, 36], and visual question answering (VQA) [14, 57]. Existing approaches have pushed state-of-the-art performance on respective tasks, however, their architectures are usually dedicated to one specific task, preventing them from being generalized to other tasks. This phenomenon raises a question: *Is it possible to design a generalized framework that can simultaneously adapt to various multimodal learning tasks?*

One promising answer to this question is the *multimodal-BERT* framework [9, 30, 34, 46], which is inspired by the de facto BERT model [11] in the natural language processing (NLP) community. Similar to the transfer learning paradigm [47, 61], BERT adopts a two-stage learning paradigm that first pre-trains a universal backbone via self-supervised learning, and then fine-tune the model for the specific task via supervised learning. Analogously,

the multimodal-BERT family pre-trains the Transformer-based backbone to obtain generalizable representations from a large-scale corpus consisting of paired multimodal data (*e.g.*, images and their associated captions). Thereafter, the generalized multimodal backbone is fine-tuned to downstream tasks such as VQA and visual grounding. Despite that the multimodal-BERT approaches deliver promising results on the benchmarks of various multimodal learning tasks, their computational costs are usually very high (*e.g.*, ~10M training samples [46] or ~300M model size [9, 34]), which severely limits their applicability.

In this paper, we tackle the generalized multimodal learning problem from another perspective. Rather than pre-training one generalized model for various tasks, we design a generalized framework instead, which can adaptively learn the optimal architecture for various tasks. To do this, we introduce neural architecture search (NAS) [64] into multimodal learning and propose a deep multimodal neural architecture search (MMnas) framework (see Figure 1). Inspired by the modularized MCAN model [56], we first define a set of primitive operations as the basic unit to be searched. Taking image and sentence features as inputs, we design a unified encoder-decoder backbone by respectively feeding features into the encoder and decoder. The encoder (or the decoder) consists of multiple encoder (or decoder) blocks cascaded in depth, where each block corresponds to an operation searched from the encoder operation pool. On top of the unified backbone, task-specific heads are respectively designed for each task (*e.g.*, VQA, visual grounding). By attaching the unified backbone with each head (*i.e.*, task), we use a gradient-based one-shot NAS algorithm to search the optimal architecture to the respective task. Compared to the *hand-crafted* architecture of MCAN, the *automatically searched* architecture of MMnas can better fit the characteristics of each task and hence lead to better performance. It is worth noting that the proposed MMnas framework is not conflict with the multimodal-BERT approaches. We can also apply the pre-training strategy on the searched architecture to further enhance its performance.

To summarize, the main contributions of this study is three-fold:

(1) We put forward a new generalized multimodal learning paradigm that uses the neural architecture search (NAS) algorithm to search for the optimal architecture for different tasks. Compared with the multimodal-BERT approaches that use large-scale data to pre-train a generalized model, our paradigm can better capture the characteristics of each task and be more parametric efficient.
(2) We devise a novel MMnas framework, which consists of a unified encoder-decoder backbone and task-specific heads to deal with different task, including visual question answering, image-text matching, and visual grounding.
(3) We conduct extensive experiments on five commonly used benchmark datasets. The optimal MMnasNet delivers new state-of-the-art performance, highlighting the effectiveness and generalizability of the proposed MMnas framework.

2 RELATED WORK

We briefly review previous studies on typical multimodal learning tasks and neural architecture search.

Multimodal Learning Tasks: Multimodal learning aims to build models that can understand and associate information from multiple modalities. From early research on audio-visual speech recognition [12, 60] to the recent explosion of interest in vision-and-language tasks [2, 8, 54], multimodal learning is a multi-disciplinary field of significant importance and potential. At present, multimodal learning with deep neural networks is the de facto paradigm for modern multimodal learning tasks, such as visual question answering (VQA) [2][26][56], image-text matching [24, 29], and visual grounding [55][53]. In the following, we briefly describe three typical multimodal learning tasks and a few representative approaches accordingly.

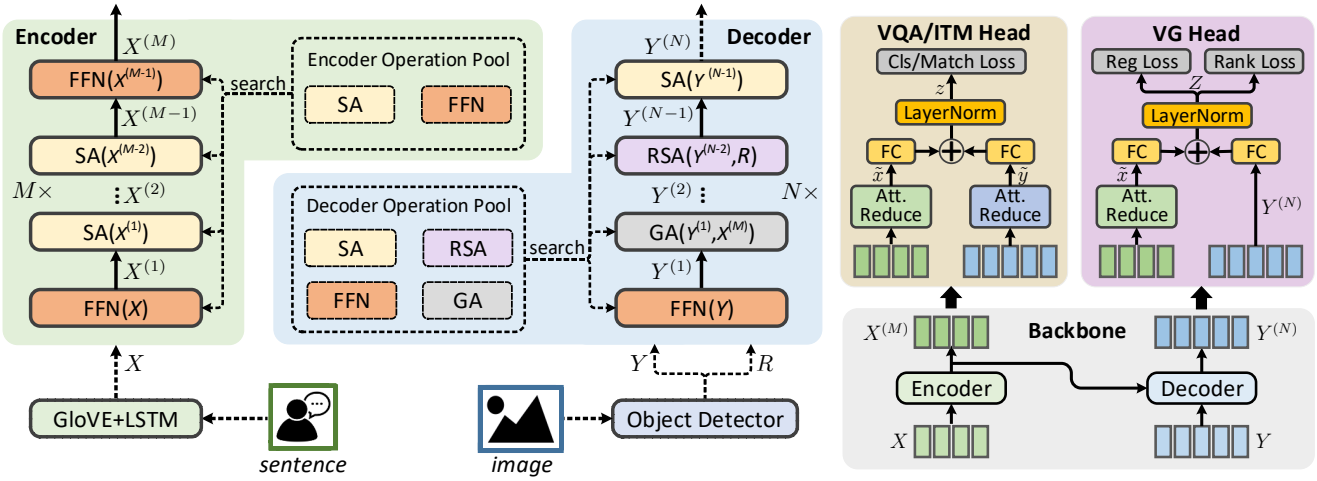
The VQA task aims to answer a question in natural language with respect to a given image, which requires a fine-grained and simultaneous understanding of both image and question. Antol *et al.* present a large-scale VQA benchmark with human annotations and some baseline methods [2]. Fukui *et al.* [14], Kim *et al.* [27], Ben *et al.* [4], and Yu *et al.* [57] devise different approximated bilinear pooling models to effectively fuse multimodal features with second-order interactions and then integrate them with attention-based neural networks. Most recently, deep co-attention models are proposed to integrate multimodal fusion and attention learning and deliver new state-of-the-art performance on the benchmark datasets [15, 26, 37]. Yu *et al.* introduce the idea of modularization into the deep co-attention model to characterize the fine-grained multimodal interactions by modularized attention units [56].

Image-text matching aims to learn two respective mapping functions for the image modality and the text modality, which are then projected into a common semantic space for distance measurement. Karpathy *et al.* propose a deep fragment embedding approach to learn the fine-grained similarity between the visual object in the image and textual word in the caption by maximizing their dot-product similarity under a multi-instance learning framework [24]. Lee *et al.* propose a stacked cross attention network to exploit the correspondences between textual words and image regions in discovering full latent alignments [29].

Visual grounding (*a.k.a*, referring expression comprehension) aims to localize an object in an image referred to by a textual query. Rohrbach *et al.* propose a GroundeR model to localize the referred object by reconstructing the sentence using attention mechanism [43]. Yu *et al.* introduce a modular attention network that simultaneously models the language-based attention and visual-based attention to capture rich contextual information for accurate localization [53].

The tasks above have the same input modalities (*i.e.*, image and text), however, their solutions are diverse and task-specific, thus preventing them from being generalized to other tasks. Inspired by the success of BERT model [11] in the NLP community, multimodal-BERT approaches are proposed to learn generalized multimodal representation in a self-supervised manner [9, 30, 34, 46]. Although promising results have been delivered, these methods usually suffer from tremendous computational costs which limits their usability in practical scenarios.

Neural Architecture Search: Neural architecture search (NAS), *a.k.a.* AutoML, has drawn an increasing interest in the last couple of years, and has been successfully applied to various deep learning tasks, such as image recognition [65], object detection [16], and



(a) Unified Encoder-Decoder Backbone

(b) Task-specific Heads

Figure 2: The flowchart of the MMnas framework, which consists of (a) unified encoder-decoder backbone and (b) task-specific heads on top the backbone for visual question answer (VQA), image-text matching (ITM), and visual grounding (VG).

language modeling [45]. Early NAS methods use the reinforcement learning to search neural architectures, which are computationally exhaustive [64, 65]. Recent works accelerate the searching process by using weight-sharing [40] or hypernetwork [6]. Although these methods bring acceleration by orders of magnitude, they usually require a meta-controller (*e.g.*, a hypernetwork or an RNN) which still burdens computational speed. Recently, one-shot NAS methods have been proposed to eliminate the meta-controller by modeling the NAS problem as a single training process of an over-parameterized supernet that comprises all candidate paths [5, 7, 32, 52].

The most closely related study to our work is the MFAS approach [39], which also incorporates NAS to search the optimal architecture for multimodal tasks. However, MFAS focuses on a simpler problem to search for the multimodal fusion model given two input features, which cannot be directly used to address the multimodal learning tasks in this paper.

3 THE MMNAS FRAMEWORK

In this section, we introduce a generalized multimodal learning framework MMnas via neural architecture search, which can be flexibly adapted to a wide range of multimodal learning tasks involving image-sentence inputs. As shown in Figure 2, MMnas contains a unified encoder-decoder backbone and task-specific heads. Taking an image and its associated sentence (*e.g.*, a question, a caption or a query) as inputs, the unified encoder-decoder backbone learns the multimodal interactions with a deep modularized network consisting of stacked encoder and decoder blocks, where each block is searched within a set of predefined primitive operations. On top of the unified backbone, we design task-specific heads to deal with the VQA, image-text matching (ITM), and visual grounding (VG) tasks, respectively. Before presenting the MMnas framework, we first introduce its basic building blocks, the primitive operations.

3.1 Primitive Operations

In the following, we present four types of primitive operations, termed as the *self-attention (SA)*, *guided-attention (GA)*, *feed-forward network (FFN)*, and *relation self-attention (RSA)* operations. First, we introduce a generalized formulation of the *scaled dot-product attention* proposed in [49], which is the core of our primitive operations below.

Denote m queries and n key-value pairs as $Q \in \mathbb{R}^{m\times d}$, $K \in \mathbb{R}^{n\times d}$, $V \in \mathbb{R}^{n\times d}$ respectively, where d is the common dimensionality. The original scaled dot-product attention function in [49] obtains the output features $F \in \mathbb{R}^{m\times d}$ by weighted summation over all values V with respect to the attention learned from the scaled dot-product of Q and K:

$$F = \mathrm{A}(Q, K, V) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d}})V \tag{1}$$

Inspired by [21], we introduce the apriori relationship $R \in \mathbb{R}^{m\times n}$ between Q and K into Eq.(1) to obtain a more generalized formula:

$$F = \mathrm{A}(Q, K, V, R) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d}} + R)V \tag{2}$$

Without loss of generality, the commonly used multi-head mechanism [49] can also be incorporated with the generalized scaled dot-product attention function, which consists of h paralleled heads (*i.e.*, independent attention functions) to further improve the representation capacity of the attended features:

$$F = \mathrm{MHA}(Q, K, V, R) = [\mathrm{head}_1, \mathrm{head}_2, ..., \mathrm{head}_h]W^o \tag{3}$$

where each $\mathrm{head}_j = \mathrm{A}(QW_j^Q, KW_j^K, KW_j^V, R)$ refers to an independent scaled dot-product attention function. $W_j^Q, W_j^K, W_j^V \in \mathbb{R}^{d\times d_h}$ are the projection matrices for the j-th head, and $W^o \in \mathbb{R}^{h*d_h\times d}$. d_h is the dimensionality of the output features from each head and is usually set to $d_h = d/h$.

SA(X): Taking a group of input features $X \in \mathbb{R}^{m\times d_x}$ of dimension d_x, the output features $Z \in \mathbb{R}^{m\times d}$ of the SA operation are obtained by feeding the inputs through Eq.(3) as follows:

$$Z = \mathrm{SA}(X) = \mathrm{MHA}(X, X, X, \mathbf{0}) \tag{4}$$

where each $z_i \in Z$ encodes the intra-modal interactions between x_i and all features within X. $\mathbf{0}$ is an all-zero matrix indicating that no relation prior is provided.

GA(X, Y): Taking two group of features $X \in \mathbb{R}^{m\times d_x}$ and $Y \in \mathbb{R}^{n\times d_y}$ of dimension d_x and d_y respectively, the GA operation transforms them into $Z \in \mathbb{R}^{m\times d}$ as follows:

$$Z = \mathrm{GA}(X, Y) = \mathrm{MHA}(X, Y, Y, \mathbf{0}) \tag{5}$$

where each $z_i \in Z$ encodes the inter-modal interactions between x_i and all features within Y.

FFN(X): This operation is a two-layer MLP network with ReLU activation and dropout in between. Taking one group of input features $X \in \mathbb{R}^{m\times d_x}$, the transformed output features $Z \in \mathbb{R}^{m\times d}$ of the FFN operation are obtained as follows:

$$Z = \mathrm{FFN}(X) = \mathrm{FC}_d \circ \mathrm{Drop}_{0.1} \circ \mathrm{ReLU} \circ \mathrm{FC}_{4d}(X) \tag{6}$$

where $\mathrm{FC}_d(\cdot)$ is a fully-connected layer of output dimension d and $\mathrm{Drop}_p(\cdot)$ is a dropout layer with dropout rate p. The symbol $\circ$ denotes a composition of two layers.

RSA(X, R): This operation takes a group of features $X \in \mathbb{R}^{m\times d_x}$ along with their relation features $R \in \mathbb{R}^{m\times m\times d_r}$ as inputs, where d_r is the dimensionality of the relation features. The output features $Z \in \mathbb{R}^{m\times d}$ of the RSA operation are obtained as follows:

$$\begin{aligned} \mathrm{MLP}(R) &= \mathrm{ReLU} \circ \mathrm{FC}_1 \circ \mathrm{ReLU} \circ \mathrm{FC}_{d_h}(R) \\ Z = \mathrm{RSA}(X, R) &= \mathrm{MHA}(X, X, X, \log(\mathrm{MLP}(R) + \epsilon)) \end{aligned} \tag{7}$$

where $\mathrm{MLP}(R)$ denotes a two-layer MLP network with transformations applied on the last axis of R. $\epsilon = 1e^{-6}$ is a small constant to avoid the underflow problem.

For each of the primitive operation above, shortcut connection [20] and layer normalization [3] are respectively applied to it. The reason why we use these four operations is two-fold: 1) they are effective and complementary at modeling different kinds of interactions for multimodal learning; and 2) we expect MMnas to be a neat baseline thus only involve the essential operations. Without losing of generality, more effective operations can be included to enlarge the operation pool seamlessly in the future.

3.2 Unified Encoder-Decoder Backbone

Inspired by [56], we construct a unified encoder-decoder as the backbone to model the deep interactions between the bimodal inputs consisting of an image and its associated sentence. In the following, we describe each component of the backbone in detail.

Sentence and Image Representations: The input sentence is first tokenized and then trimmed (or zero-padded) into a sequence of m words. Each word is represented as a 300-D vector using the pre-trained GloVe word embeddings [38]. The word embeddings are fed into a one-layer LSTM network with d_x hidden units, resulting in the final sentence features $X \in \mathbb{R}^{m\times d_x}$.

Following the strategy in [1], the input image is represented as a set of objects extracted from a pre-trained object detection model (*e.g.*, Faster R-CNN). For each image, the object detector predicts n objects with each object being represented as a group of visual features and relation features, respectively. The visual features $Y \in \mathbb{R}^{n\times d_y}$ are obtained by pooling the convolutional features from the detected objects. The relation features $R \in \mathbb{R}^{n\times n\times 4}$ are calculated by the relative spatial relationships of object pairs[1].

Sentence Encoder and Image Decoder: Taking the word-level sentence features X as inputs, the sentence encoder learns the intra-modal interactions of sentence words by passing X through M encoder blocks $\{b^{(1)}_{\mathrm{enc}}, b^{(2)}_{\mathrm{enc}}, ..., b^{(M)}_{\mathrm{enc}}\}$ recursively:

$$X^{(i)} = b^{(i)}_{\mathrm{enc}}(X^{(i-1)}) \tag{8}$$

where $i \in \{1, 2, ..., M\}$ and $X^{(0)} = X$. Each $b^{(i)}_{\mathrm{enc}}(\cdot)$ corresponds to an operation searched from an encoder operation pool with independent operation weights. Similar to [56], the encoder operation pool consists of two candidate operations: SA and FFN.

Analogous to the sentence encoder, we design an image decoder consisting of N decoder blocks $\{b^{(1)}_{\mathrm{dec}}, b^{(2)}_{\mathrm{dec}}, ..., b^{(N)}_{\mathrm{dec}}\}$. Slightly different from that of the encoder, the decoder operation pool contains four operations: SA, RSA, GA, and FFN. Taking the visual features Y and relation features R from the image, along with the output features $X^{(M)}$ from the sentence encoder as inputs, the image decoder models the intra- and inter-modal interactions of the multimodal inputs in a recursive manner:

$$Y^{(i)} = b^{(i)}_{\mathrm{enc}}(Y^{(i-1)}, R, X^{(M)}) \tag{9}$$

where $i \in \{1, 2, ..., N\}$ and $Y^{(0)} = Y$. Each $b^{(i)}_{\mathrm{dec}}(\cdot)$ takes at least one input (*i.e.*, $Y^{(i-1)}$) and may have an additional input (*i.e.*, R or $X^{(M)}$) if specific operation is searched (*i.e.*, RSA or GA).

3.3 Task-specific Heads

The output sentence features $X^{(M)} = [x^{(M)}_1, x^{(M)}_2, ..., x^{(M)}_m]$ and image features $Y^{(N)} = [y^{(N)}_1, y^{(N)}_2, ..., y^{(N)}_n]$ from the unified encoder-decoder backbone contain rich information about the attentive interactions between the sentence words and image objects. On top of the backbone, we attach task-specific heads to address the visual question answering (VQA), image-text matching (ITM), and visual grounding (VG) tasks, respectively.

VQA Head: Similar to most existing works [2, 26, 57], we resolve the VQA problem by predicting the best-matched answer to the question from a large answer vocabulary. Inspired by the multi-modal fusion model in [56], we use two independent attentional reduction models for $X^{(M)}$ and $Y^{(N)}$ to obtain their reduced features $\tilde{x}$ and $\tilde{y}$, respectively:

$$\begin{aligned} \alpha &= \mathrm{softmax}(\mathrm{MLP}(X^{(M)})), \quad & \beta &= \mathrm{softmax}(\mathrm{MLP}(Y^{(N)})) \\ \tilde{x} &= \sum_{i=1}^{m} \alpha_i x^{(M)}_i, & \tilde{y} &= \sum_{i=1}^{n} \beta_i y^{(N)}_i \end{aligned} \tag{10}$$

where $\alpha \in \mathbb{R}^m, \beta \in \mathbb{R}^n$ are the attention weights to be learnt. $\mathrm{MLP}(X) = \mathrm{FC}_1 \circ \mathrm{ReLU} \circ \mathrm{FC}_d(X)$ corresponds to a two-layer MLP

[1]Denote the location of the i-th object as $\{x_i, y_i, h_i, w_i\}$, where x_i, y_i refer to the center of the object, and w_i, h_i refer to the width and height of the object, respectively. Following the strategy in [21], the 4-D relation feature between the m-th object and the n-th object is defined as $[\log(|x_m - x_n|/w_m), \log(|y_m - y_n|/h_m), \log(w_n/w_m), \log(h_n/h_m)]$.

network. After that, the reduced features are fused together as follows:

$$z = \text{LayerNorm}(W_x^T \tilde{x} + W_y^T \tilde{y}) \tag{11}$$

where $W_x, W_y \in \mathbb{R}^{d \times d_z}$ are two projection matrices to embed the input features into a d_z-dimensional common space. LayerNorm is appended on the fused feature to stabilize training [3].

The fused feature z is then projected into a vector $p \in \mathbb{R}^k$ and then fed into a k-way classification loss, where k denotes the size of the answer vocabulary. For the dataset that provides multiple answers to each question, we formulate it as a multi-label classification problem and use binary cross-entropy (BCE) loss to train the model. For the dataset that only has one answer to each question, we regard it as a single-label classification problem and use the softmax cross-entropy loss instead.

ITM Head: Image-text matching aims to learn a matching score to measure the cross-modal similarity between the image-text pair. Since the outputs of the ITM and VQA tasks are similar, we therefore reuse part of the model in the VQA head. On top of the fused feature z from Eq.(11), the matching score $s \in (0, 1)$ is obtained as follows:

$$s = \sigma(W_z^T z) \tag{12}$$

where $W_z \in \mathbb{R}^{d_z}$ and $\sigma(\cdot)$ denotes the sigmoid function. Denote the predicted matching score of an input image-text pair as $s(\mathcal{I}, \mathcal{T})$, where $(\mathcal{I}, \mathcal{T})$ represents a positive sample with correspondence. We use BCE loss with hard negatives mining for $(\mathcal{I}, \mathcal{T})$ as our loss function to train the matching model:

$$\begin{aligned} \mathcal{L}_{\text{match}} = &\log(s(\mathcal{I}, \mathcal{T})) + \log(1 - s(\mathcal{I}, \mathcal{T}')) \\ &+ \log(s(\mathcal{I}, \mathcal{T})) + \log(1 - s(\mathcal{I}', \mathcal{T})) \end{aligned} \tag{13}$$

where $\mathcal{T}'$ and $\mathcal{I}'$ denote the hard negative text and image samples for $(\mathcal{I}, \mathcal{T})$ mined from the whole training set per training epoch.

VG Head: We address the visual grounding task by predicting a ranking score and a refined bounding box for each visual object in the image referred to by the query. To do this, we first feed the word-level query features $X^{(M)}$ into the attentional reduction model in Eq.(10) to obtain the reduced feature vector $\tilde{x}$. After that, $\tilde{x}$ is broadcasted and integrated with the object-level image features $Y^{(N)}$ as follows:

$$Z = \text{LayerNorm}(W_x^T \tilde{x} + W_y^T Y^{(N)}) \tag{14}$$

where $Z \in \mathbb{R}^{n \times d_z}$ correspond to the fused features of n objects in the image. Each object feature $z \in Z$ is then linearly projected into a ranking score $s \in \mathbb{R}$ and a 4-D bounding box offset $b \in \mathbb{R}^4$, respectively. Similar to [59], we design a multi-task loss function consisting of a ranking loss $\mathcal{L}_{\text{rank}}$ and a regression loss $\mathcal{L}_{\text{reg}}$:

$$\mathcal{L} = \mathcal{L}_{\text{rank}} + \lambda \mathcal{L}_{\text{reg}} \tag{15}$$

where λ is a hyper-parameter to balance the two terms. The $\mathcal{L}_{\text{rank}}$ term penalizes the KL-divergence between the predicted scores $S = [s_1, s_2, ..., s_n] \in \mathbb{R}^n$ and the ground-truth scores $S^* \in \mathbb{R}^n$ for n objects, where S^* are obtained by calculating the IoU scores of all objects with respect to the unique ground-truth bounding box. Softmax normalizations are respectively applied to S and S^* to form a score distribution. The $\mathcal{L}_{\text{reg}}$ term penalizes the smoothed L_1 distance [17] between the predicted offset b and the ground-truth offset b^* for the objects with their IoU scores S^* larger than a threshold σ. The offset $b^* \in \mathbb{R}^4$ is obtained by calculating the translations between the bounding box of the input object and the bounding box of ground-truth object [17].

Algorithm 1: Search Algorithm for MMnasNet.

Input: A supernet parameterized by the architecture weights θ and the model weights W. Training set $\mathcal{D}_a$ and $\mathcal{D}_m$ are used to optimize θ and W, respectively. T_w and T_j denote the number of epochs for the warming-up and iterative optimization stages, respectively. u is a factor to balance the update frequencies of θ and W.
Output: The searched optimal architecture a^*

```
Random initialize θ and W;
# The warming-up stage;
for t = 1 to T_w do
    Random sample an architecture a ~ A;
    Random sample a mini-batch d_m ⊆ D_m;
    Update W by descending ∇_W L_train(N(a, W)) on d_m;
end
# The iterative optimization stage;
for t = 1 to T_j do
    for i = 1 to u do
        Random sample a mini-batch d_m ⊆ D_m;
        Sample an architecture a ~ A(θ) with respect to θ;
        Update W by descending ∇_W L_train(N(a, W)) on d_m;
    end
    Random sample a mini-batch d_a ⊆ D_a;
    Sample an architecture a ~ A(θ) with respect to θ;
    Update θ by descending ∇_θ L_train(N(a, W)) on d_a;
end
Return a* by picking the operation with the largest value in θ* for each block.
```

4 SEARCH ALGORITHM

To obtain the optimal MMnasNet architecture for each task on specific dataset, we introduce an efficient one-shot search algorithm that search the optimal architecture within an over-parameterized supernet with weight sharing.

Denote a supernet as $\mathcal{N}(\mathcal{A}(\theta), W)$ that encodes the whole search space $\mathcal{A}$ of MMnas, where W and θ correspond to the model weights and architecture weights of all the possible operations in the supernet, respectively[2]. The optimal architecture is obtained by minimizing the expectation with respect to θ and W jointly:

$$(\theta^*, W^*) = \underset{\theta, W}{\text{argmin}} \ \mathbb{E}_{a \sim \mathcal{A}(\theta)}[\mathcal{L}(\mathcal{N}(a, W))] \tag{16}$$

where for each of the three tasks above, $\mathcal{L}(\mathcal{N}(a, W))$ indicates using the task-specific loss function to optimize the weights of network architecture $\mathcal{N}(a, W)$, where a is a valid architecture sampled from the supernet with respect to θ. Based on the obtained optimal θ^*, the optimal architecture a^* is obtained by selecting the operation with the largest architecture weight in each block of the backbone.

Inspired by the strategy in [7], we adopt an iterative algorithm to optimize the architecture weights θ and the model weights W alternatively. We first separate the training set into two non-overlapping sets $\mathcal{D}_m$ and $\mathcal{D}_a$. When training the model weights W, we first freeze the architecture weights θ and stochastically sample exactly one operation for each block with respect to θ after softmax activation, which results in a valid architecture a. After

[2]Given a MMnas supernet consisting of M encoder blocks and N decoder blocks, the size of the search space is $2^M \times 4^N$ and the number of all the possible operations in the supernet is $2M+4N$, where 2 and 4 correspond to the sizes of the encoder and decoder operation pools, respectively.

that, we update the model weights W activated by a via standard gradient descent on $\mathcal{D}_m$. When training the architecture weights θ, we freeze the model weights W, sample a valid architecture a, and then update θ via gradient descent on $\mathcal{D}_a$.

As claimed in [10], the iterative optimization of W and θ inevitably introduces bias to certain architectures and leave the rest ones poorly optimized. To alleviate the problem, we introduce an additional *warming-up* stage before the iterative optimization. In the warming-up stage, we do not train the architecture weights θ and sample operations uniformly to train the model weights W. This ensures that the model weights W are well initialized thus leading to more impartial and robust architecture search.

The detailed search algorithm is illustrated in Algorithm 1.

5 EXPERIMENTS

We evaluate the searched MMnasNets on three multimodal learning tasks and perform comparative analysis to the state-of-the-art methods on five benchmark datasets thoroughly. Furthermore, we conduct comprehensive ablation experiments to explore the reasons why MMnas is effective.

5.1 Datasets

VQA-v2 is a commonly-used dataset for the VQA task [18]. It contains human annotated question-answer pairs for COCO images [31]. The dataset is split into three subsets: train (80k images with 444k questions); val (40k images with 214k questions); and test (80k images with 448k questions). The test subset is further split into test-dev and test-std sets that are evaluated online. The results consist of three per-type accuracies (Yes/No, Number, and Other) and an overall accuracy.

Flickr30K contains 31,000 images collected from Flickr website with five captions each. Following the partition strategy of [24, 29], we use 1,000 images for validation and 1,000 images for testing and the rest for training.

RefCOCO, RefCOCO+ and RefCOCOg are three datasets to evaluate visual grounding performance. All three datasets are collected from COCO images [31]. RefCOCO and RefCOCO+ are split into four subsets: train (120k queries), val (11k queries), testA (6k queries about people), and testB (5k queries about objects). RefCOCOg is split into three subsets: train (81k queries), val (5k queries), and test (10k queries)

The statistics and evaluation metrics of the datasets are summarized in Table 1.

5.2 Experimental Setup

Universal Setup: We use the following hyper-parameters for MMnasNet as the default settings unless otherwise stated. For each primitive operation, the latent dimensionality in the multi-head attention d is 512 and the number of heads h is 8. The dimensionality of the fused features d_z is set to $2d$. The number of encoder blocks M and decoder blocks N are respectively set to 12 and 18 to match the number of blocks in the 6-layer MCAN model[3] [56].

[3]A L-layer MCAN model corresponds to a special case of the MMnasNet model consisting of $2L$ encoder blocks (with repeated SA-FFN operations) and $3L$ decoder blocks (with repeated SA-GA-FFN operations).

Table 1: The detailed statistics and evaluation metrics of the tasks and datasets.

Task	Dataset	Image Source	#Img.	#Sent.	Metric
VQA	VQA-v2 [18]	COCO	204K	1.1M	Accuracy
ITM	Flickr30K [41]	Flickr	31K	155K	Recall@K
VG	RefCOCO [25]	COCO	20K	142K	Accuracy
	RefCOCO+ [25]		20K	142K	
	RefCOCOg [35]		26K	95K	

For each dataset, we use its train split to perform architecture search. The train set is further random split into two subsets D_m and D_a with $|D_m|/|D_a| = 4$. Each randomly initialized model is warmed-up for $T_w = 50$ epochs and then searched for another $T_j = 20$ epochs with the early stopping strategy. The frequency ratio u for updating the model and architecture weights is set to 5. With the searched optimal architecture, we train the MMnasNet model again from scratch to obtain the final model.

VQA Setup: For VQA-v2, we follow the setting in [56] that all questions are processed to a maximum length of $m = 14$ and the size of the answer vocabulary is set to 3129. The visual features and relation features are extracted from a pre-trained Faster R-CNN model on Visual Genome [1]. The number of extracted objects $n \in [10, 100]$ is determined by a confidence threshold.

ITM Setup: For Flickr30K, the maximum length of texts (*i.e.*, captions) is set to $m = 50$. The visual features and relation features are extracted from a Faster R-CNN model pre-trained on Visual Genome with the number of objects $n = 36$ [1]. For each positive image-text pair $(\mathcal{I}, \mathcal{T})$ in the training set, we use the following hard sample mining strategy before each training epoch: we randomly sample 64 negative images per text and 64 negative texts per image from the whole training set to generate negative image-text pairs. Thereafter, we feed all these negative pairs to the current model checkpoint to predict their matching scores and regard the top-5 ranked negative samples as the hard negative samples according to their scores. Finally, we randomly pick one hard image sample $\mathcal{I}'$ and one hard text sample $\mathcal{T}'$ from the candidate hard negative samples, respectively.

VG Setup: We use the same settings for the three visual grounding datasets. For the textual queries, the maximum length is set to $m = 14$. For the images, we adopt two pre-trained object detectors to extract the visual features: 1) a Mask R-CNN model trained on COCO [19]; and 2) a Faster R-CNN model trained on Visual Genome [42]. During the training data preparation for the two detectors, we excluded all the images that exist in the training, validation and testing sets of RefCOCO, RefCOCO+, and RefCOCOg to avoid the data leakage problem. For both detectors above, we detect $n = 100$ objects for each image to extract the visual and relation features. The loss weight λ is set to 1.

5.3 Ablation Experiments

We run a number of ablations experiments on VQA-v2 to analyze the reason behind MMnasNet's effectiveness. Results shown in Table 2 are discussed in detail next.

Search Space: In Table 2a, we compare the MMnasNet models searched from different decoder operation pools. From the results, we can see that: 1) modeling the intra-modal attention among visual

Decoder operations	All	Y/N	Num	Other
{GA, FFN}	66.5	84.9	45.2	58.2
{SA, GA, FFN}	67.4	85.0	49.7	58.7
{RSA, GA, FFN}	67.6	84.9	51.3	58.8
{SA, RSA, GA, FFN}	**67.8**	**85.1**	**52.1**	**58.9**

(a) ***Search Space:*** **Per-type accuracies of MMnasNet with different decoder operation pools. All models use the same encoder operation pool of** {SA, FFN}.

M	N	MCAN (Size)	MMnasNet (Size)
4	6	66.1 (27M)	67.1 (28M)
8	12	66.9 (41M)	67.7 (44M)
12	18	67.2 (56M)	**67.8** (58M)
16	24	67.2 (68M)	67.7 (76M)

(b) ***Model Depth:*** **Overall accuracies and sizes of MCAN and MMnasNet with different number of encoder blocks M and decoder blocks N.**

Encoder	Decoder	Accuracy
R	R	66.9
S	R	67.1
R	S	67.6
S	S	**67.8**

(c) ***Random v.s. Searched:*** **Overall accuracies of MMnasNet with random (R) or searched (S) architecture for the encoder-decoder.**

Table 2: Ablation experiments for MMnasNet on VQA-v2. We train on the *train* split and report the results on the *val* split.

objects by SA or RSA is vital to object counting performance (*i.e.*, the *number* type answers), which is consistent with the results reported in [56]; 2) introducing the RSA operation which models the relative spatial relationships between paired objects can further facilitate the object counting performance; and 3) SA and RSA are complementary to each other, hence modeling them together leads to the best performance on all answer types.

Model Depth: In Table 2b, we compare MMnasNet to the reference MCAN model [56] under different model depths (*i.e.*, number of encoder blocks M and decoder blocks N). The results reveal that: 1) MMnasNet consistently outperforms MCAN, especially when the model depth is relatively shallow (*e.g.*, $M \leq 8$). This can be explained that the optimal architectures for different model depths are quite different; 2) with the same M and N, the model size of MMnasNet is slightly larger than MCAN. This is because MMnasNet tends to use more FFN operations, which introduces more parameters to increase the nonlinearity of the model; and 3) with the increase of model depth, both MCAN and MMnasNet saturate at M=12 and N=18, which reflects the bottleneck of the used deep encoder-decoder framework.

Random *vs.* Searched: To prove the necessity and superiority of the searched architectures over randomly generated ones, we conduct the experiments in Table 2c by alternatively using the searched or random architectures for the encoder and decoder, respectively. From the results, we can see that: 1) the searched architectures outperforms the random counterparts by up to 0.9 points; 2) the design of the decoder architecture is much more important than the encoder architecture; and 3) the all-random architecture also performs well compared to some recent works [15, 26]. This suggests the used primitive operations that constitute the architecture also play a key role in model performance.

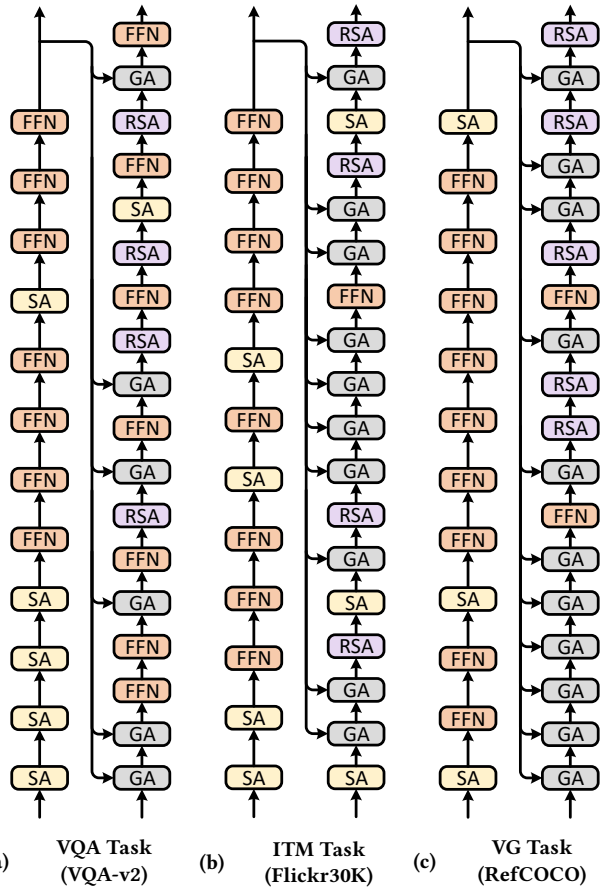


Figure 3: The optimal MMnasNet backbones searched for different tasks (over specific datasets).

5.4 Main Results

Taking the ablation studies into account, we compare the best-performing MMnasNet models (with M=12 and N=18) to the state-of-the-art approaches on five benchmark datasets. Figure 3 illustrates the optimal MMnasNet backbones searched for different tasks (over specific datasets). This verifies our hypothesis that the optimal architectures for different tasks may vary prominently. Note that we do not compare MMnasNet to the multimodal-BERT approaches (*e.g.*, LXMRET [46] or UNITER [9]), since they introduce additional training datasets for model pre-training thus may lead to unfair comparison.

In Table 3, we compare MMnasNets to the state-of-the-art methods on VQA-v2. The demonstrated results show that: 1) compared with existing state-of-the-art approaches, MMnasNet outperforms them by a clear margin on all answer types; and 2) MMnasNet significantly outperforms existing approaches on the object counting performance (*i.e.*, the number type), which owes to the effectiveness of the RSA operation we introduce.

Table 4 contains the image-text matching results on Flickr30K. Similar to most existing works [29, 50], we report the matching results in terms of Recall@K, where K denotes the top-K results

Table 5: Accuracies (with IoU>0.5) on RefCOCO, RefCOCO+ and RefCOCOg to compare with the state-of-the-art methods. All methods use *detected objects* to extract visual features. COCO [31] and Genome [28] denote two datasets for training the object detectors. SSD [33], FRCN [42] and MRCN [19] denote the used detectors with VGG-16 [44] or ResNet-101 [20] backbones.

Method	Object Detector			RefCOCO			RefCOCO+			RefCOCOg	
	Dataset	Model	Backbone	TestA	TestB	Val	TestA	TestB	Val	Test	Val
CMN [22]	COCO	FRCN	VGG-16	71.0	65.8	-	54.3	47.8	-	-	-
VC [62]	COCO	FRCN	VGG-16	73.3	67.4	-	58.4	53.2	-	-	-
Spe.+Lis.+Rein.+MMI [55]	COCO	SSD	VGG-16	73.7	65.0	69.5	60.7	48.8	55.7	59.6	60.2
Spe.+**Lis.**+Rein.+MMI [55]	COCO	SSD	VGG-16	73.1	64.9	69.0	60.0	49.6	54.9	59.2	59.3
MAttNet [53]	COCO	MRCN	ResNet-101	81.1	70.0	76.7	71.6	56.0	65.3	67.3	66.6
DDPN [59]	Genome	FRCN	ResNet-101	80.1	72.4	76.8	70.5	54.1	64.8	67.0	66.7
MMnasNet (ours)	COCO	MRCN	ResNet-101	82.5	78.4	81.5	70.9	62.3	69.8	72.7	73.1
MMnasNet (ours)	Genome	FRCN	ResNet-101	**87.4**	77.7	**84.2**	**81.0**	**65.2**	**74.7**	**75.7**	**74.7**

Table 3: Accuracies on the *test-dev* and *test-std* splits of VQA-v2. All methods use the same visual features [1] and are trained on the *train+val+vg* splits, where *vg* denotes the augmented dataset from Visual Genome.

Method	Test-Dev				Test-Std
	All	Y/N	Num	Other	All
Bottom-Up [48]	65.32	81.82	44.21	56.05	65.67
MFH+CoAtt [58]	68.76	84.27	49.56	59.89	-
BAN-8 [26]	69.52	85.31	50.93	60.26	-
BAN-8 (+G+C) [26]	70.04	85.42	54.04	60.52	70.35
DFAF-8 [15]	70.22	86.09	53.32	60.49	70.34
MCAN-6 [56]	70.63	86.82	53.26	60.72	70.90
MMnasNet (ours)	**71.24**	**87.27**	**55.68**	**61.05**	**71.46**

Table 4: Recall@{1, 5, 10} on Flickr30K to compare with the state-of-the-art methods.

Method	Text Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10
VSE++ [13]	52.9	80.5	87.2	39.6	70.1	79.5
DAN [36]	55.0	81.8	89.0	39.4	69.2	79.1
DPC [63]	55.6	81.9	89.5	39.1	69.2	80.9
SCO [23]	55.5	82.0	89.3	41.1	70.5	80.1
$SCAN_{t\rightarrow i}$ [29]	61.8	87.5	93.7	45.8	74.4	83.0
$SCAN_{i\rightarrow t}$ [29]	67.7	88.9	94.0	44.0	74.2	82.6
CAMP [50]	68.1	89.7	95.2	51.5	77.1	85.3
MMnasNet (ours)	**78.3**	**94.6**	**97.4**	**60.7**	**85.1**	**90.5**

retrieved from a database and ranges within $\{1, 5, 10\}$. The cross-modal matching results from two directions (*i.e.*, the text retrieval and image retrieval) are demonstrated in Table 4 to compare with the state-of-the-art approaches. From the results, we can see that MMnasNet significantly outperforms existing state-of-the-art methods in terms of all evaluation metrics.

In Table 5, we report the comparative results on RefCOCO, RefCOCO+, and RefCOCOg, respectively. We use the commonly used accuracy metric [53], where a prediction is considered to be correct if the predicted bounding box overlaps with the ground-truth of IoU >0.5. With the standard visual features (*i.e.*, the MRCN model pre-trained on COCO), MMnasNet reports a remarkable improvement over MAttNet on all the three datasets. Be equipped with the powerful visual features (*i.e.*, the FRCN model pre-trained on Visual Genome), MMnasNet obtains further improvement and delivers the new state-of-the-art performance across all datasets.

6 CONCLUSION

In this paper, we present a generalized deep multimodal neural architecture search (MMnas) framework for various multimodal learning tasks. Different from the existing approaches that design hand-crafted and task-specific architectures to address only a single task, MMnas can be generalized to automatically learn the optimal architectures of different tasks. To achieve this, we construct a unified encoder-decoder backbone with each encoder/decoder block corresponding to an operation searched from a candidate set of predefined operations. On top of the unified backbone, we attach task-specific heads to deal with different tasks. The optimal architecture for each task is learned by an efficient neural architecture search (NAS) algorithm to obtain task-specific MMnasNet. Extensive experiments are conducted on the VQA, visual grounding, and image-text matching tasks to show the generalizability and effectiveness of the proposed MMnas framework. Comprehensive results from five benchmark datasets validate the superiority of MMnasNet over existing state-of-the-art methods.

Different from existing multimodal-BERT approaches that use large-scale multimodal pre-training, we introduce an alternative way to address the generalized multimodal learning problem via a NAS framework. We hope our work may serve as a solid baseline to inspire future research on multimodal learning.

ACKNOWLEDGMENTS

This work was supported in part by the National Key R&D Program of China under Grant 2018AAA0100603, and in part by National Natural Science Foundation of China under Grant 61702143, Grant 61836002, Grant 61725203 and Grant 61732008.

REFERENCES

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018).

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *IEEE International Conference on Computer Vision (ICCV)*. 2425–2433.

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).

[4] Hedi Ben-Younes, Rémi Cadene, Matthieu Cord, and Nicolas Thome. 2017. Mutan: Multimodal tucker fusion for visual question answering. In *IEEE International Conference on Computer Vision (ICCV)*.

[5] Gabriel Bender, Pieter-Jan Kindermans, Barret Zoph, Vijay Vasudevan, and Quoc Le. 2018. Understanding and Simplifying One-Shot Architecture Search. In *International Conference on Machine Learning (ICML)*. 550–559.

[6] Andrew Brock, Theodore Lim, James Millar Ritchie, and Nicholas J Weston. 2018. SMASH: One-Shot Model Architecture Search through HyperNetworks. In *International Conference on Learning Representations (ICLR)*.

[7] Han Cai, Ligeng Zhu, and Song Han. 2018. Proxylessnas: Direct neural architecture search on target task and hardware. *arXiv preprint arXiv:1812.00332* (2018).

[8] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325* (2015).

[9] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Uniter: Learning universal image-text representations. *arXiv preprint arXiv:1909.11740* (2019).

[10] Xiangxiang Chu, Bo Zhang, Ruijun Xu, and Jixiang Li. 2019. Fairnas: Rethinking evaluation fairness of weight sharing neural architecture search. *arXiv preprint arXiv:1907.01845* (2019).

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the NAACL-HLT*. 4171–4186.

[12] Stéphane Dupont and Juergen Luettin. 2000. Audio-visual speech modeling for continuous speech recognition. *IEEE Transactions on Multimedia* 2, 3 (2000), 141–151.

[13] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2017. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612* (2017).

[14] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding. *Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2016).

[15] Peng Gao, Zhengkai Jiang, Haoxuan You, Pan Lu, Steven CH Hoi, Xiaogang Wang, and Hongsheng Li. 2019. Dynamic fusion with intra-and inter-modality attention flow for visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6639–6648.

[16] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. 2019. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 7036–7045.

[17] Ross Girshick. 2015. Fast r-cnn. In *IEEE International Conference on Computer Vision (ICCV)*. 1440–1448.

[18] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017).

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2961–2969.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[21] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. 2018. Relation networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3588–3597.

[22] Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. 2017. Modeling relationships in referential expressions with compositional modular networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1115–1124.

[23] Yan Huang, Qi Wu, Chunfeng Song, and Liang Wang. 2018. Learning semantic concepts and order for image and sentence matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6163–6171.

[24] Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3128–3137.

[25] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 787–798.

[26] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. 2018. Bilinear attention networks. In *Advances in Neural Information Processing Systems (NIPS)*. 1564–1574.

[27] Jin-Hwa Kim, Kyoung Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. 2017. Hadamard Product for Low-rank Bilinear Pooling. In *International Conference on Learning Representation (ICLR)*.

[28] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision* 123, 1 (2017), 32–73.

[29] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. 2018. Stacked cross attention for image-text matching. In *European Conference on Computer Vision (ECCV)*. 201–216.

[30] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557* (2019).

[31] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*. 740–755.

[32] Hanxiao Liu, Karen Simonyan, and Yiming Yang. 2018. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055* (2018).

[33] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. 2016. Ssd: Single shot multibox detector. In *European Conference on Computer Vision (ECCV)*. Springer, 21–37.

[34] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems (NIPS)*. 13–23.

[35] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *IEEE International Conference on Computer Vision (ICCV)*. 11–20.

[36] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. 2017. Dual attention networks for multimodal reasoning and matching. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 299–307.

[37] Duy-Kien Nguyen and Takayuki Okatani. 2018. Improved Fusion of Visual and Language Representations by Dense Symmetric Co-Attention for Visual Question Answering. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), 6087–6096.

[38] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation.. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Vol. 14. 1532–1543.

[39] Juan-Manuel Pérez-Rúa, Valentin Vielzeuf, Stéphane Pateux, Moez Baccouche, and Frédéric Jurie. 2019. Mfas: Multimodal fusion architecture search. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6966–6975.

[40] Hieu Pham, Melody Guan, Barret Zoph, Quoc Le, and Jeff Dean. 2018. Efficient Neural Architecture Search via Parameters Sharing. In *International Conference on Machine Learning (ICML)*. 4095–4104.

[41] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*. 2641–2649.

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems (NIPS)*. 91–99.

[43] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. 2016. Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision (ECCV)*. 817–834.

[44] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[45] David R So, Chen Liang, and Quoc V Le. 2019. The evolved transformer. *arXiv preprint arXiv:1901.11117* (2019).

[46] Hao Tan and Mohit Bansal. 2019. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 5103–5114.

[47] Jinhui Tang, Xiangbo Shu, Zechao Li, Guo-Jun Qi, and Jingdong Wang. 2016. Generalized deep transfer networks for knowledge propagation in heterogeneous domains. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 12, 4s (2016), 1–22.

[48] Damien Teney, Peter Anderson, Xiaodong He, and Anton Van Den Hengel. 2018. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR*. 4223–4232.

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*. 6000–6010.

[50] Zihao Wang, Xihui Liu, Hongsheng Li, Lu Sheng, Junjie Yan, Xiaogang Wang, and Jing Shao. 2019. CAMP: Cross-Modal Adaptive Message Passing for Text-Image

Retrieval. In *IEEE International Conference on Computer Vision (ICCV)*. 5764–5773.
[51] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C Courville, Ruslan Salakhutdinov, Richard S Zemel, and Yoshua Bengio. 2015. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention.. In *International Conference on Machine Learning (ICML)*, Vol. 14. 77–81.
[52] Yuhui Xu, Lingxi Xie, Xiaopeng Zhang, Xin Chen, Guo-Jun Qi, Qi Tian, and Hongkai Xiong. 2019. Pc-darts: Partial channel connections for memory-efficient differentiable architecture search. *arXiv preprint arXiv:1907.05737* (2019).
[53] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. 2018. Mattnet: Modular attention network for referring expression comprehension. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1307–1315.
[54] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. 2016. Modeling context in referring expressions. In *European Conference on Computer Vision (ECCV)*. Springer, 69–85.
[55] Licheng Yu, Hao Tan, Mohit Bansal, and Tamara L Berg. 2017. A joint speaker-listener-reinforcer model for referring expressions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 7282–7290.
[56] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. 2019. Deep Modular Co-Attention Networks for Visual Question Answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6281–6290.
[57] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. 2017. Multi-modal Factorized Bilinear Pooling with Co-Attention Learning for Visual Question Answering. *IEEE International Conference on Computer Vision (ICCV)* (2017), 1839–1848.
[58] Zhou Yu, Jun Yu, Chenchao Xiang, Jianping Fan, and Dacheng Tao. 2018. Beyond Bilinear: Generalized Multimodal Factorized High-Order Pooling for Visual Question Answering. *IEEE Transactions on Neural Networks and Learning Systems* 29, 12 (2018), 5947–5959.
[59] Zhou Yu, Jun Yu, Chenchao Xiang, Zhou Zhao, Qi Tian, and Dacheng Tao. 2018. Rethinking Diversified and Discriminative Proposal Generation for Visual Grounding. *International Joint Conference on Artificial Intelligence (IJCAI)* (2018).
[60] Ben P Yuhas, Moise H Goldstein, and Terrence J Sejnowski. 1989. Integration of acoustic and visual speech signals using neural networks. *IEEE Communications Magazine* 27, 11 (1989), 65–71.
[61] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. 2018. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3712–3722.
[62] Hanwang Zhang, Yulei Niu, and Shih-Fu Chang. 2018. Grounding Referring Expressions in Images by Variational Context. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[63] Zhedong Zheng, Liang Zheng, Michael Garrett, Yi Yang, and Yi-Dong Shen. 2017. Dual-path convolutional image-text embedding with instance loss. *arXiv preprint arXiv:1711.05535* (2017).
[64] Barret Zoph and Quoc V Le. 2016. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578* (2016).
[65] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. 2018. Learning transferable architectures for scalable image recognition. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*. 8697–8710.