

A CNN-Transformer Encoder-Decoder for Image Captioning: Label Smoothing, Backbone Fine-Tuning, and Length-Normalized Decoding

Rejuyan Hossain Shawon

City College of New York

Faysal Al Fahad Siam

Sichuan University

Mohammed Shafiullah Raihan

International Islamic University Chittagong

Shahrin Khan

Daffodil International University

Mohammed Saifuddin Munna

Premier University

Mir Mohammad Azad

Uttara University

Mithila Arman

BRAC University

Samuel Kim

`samuel.kim@stanford.edu`

Stanford University

Haewon Byeon

European Institute for Materials, AI & Technology

Research Article

Keywords: image captioning, Transformer, InceptionV3, CIDEr, MS COCO, fine-tuning, label smoothing, beam search

Posted Date: October 13th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11345363/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

A CNN-Transformer Encoder-Decoder for Image Captioning: Label Smoothing, Backbone Fine-Tuning, and Length-Normalized Decoding

Rejuyan Hossain Shawon¹, Faysal Al Fahad Siam²,
Mohammed Shafiullah Raihan³, Shahrin Khan⁴,
Mohammed Saifuddin Munna⁵, Mir Mohammad Azad⁶,
Mithila Arman⁷, Samuel Kim⁸, Haewon Byeon⁹

¹Department of Computer Science, City College of New York, CUNY,
New York, USA.

²College of Computer Science & College of Software Engineering,
Sichuan University, Chengdu, Sichuan, China.

³Department of Computer Science and Engineering, International
Islamic University Chittagong, Kumira, Sitakunda Upazila,
Chattogram, Bangladesh.

⁴DIU Health Informatics Research Lab, Department of Computer
Science and Engineering, Daffodil International University, Dhaka,
Bangladesh.

⁵Department of Electrical and Electronic Engineering, Premier
University.

⁶Department of Computer Science and Engineering, Uttara University,
Beribadh Road, Turag, Uttara, Dhaka, 1230, Bangladesh.

⁷Department of Computer Science and Engineering, BRAC University,
Dhaka, 1212, Bangladesh.

⁸Postdoctoral Fellow, Department of Computer Science, Stanford
University, USA.

⁹Department of Artificial Intelligence, European Institute for Materials,
AI & Technology, Madrid, Spain.

Contributing authors: rejuyanhossainshawon1@gmail.com;
faysal.scu@gmail.com; shafiullahraihan5@gmail.com;
shahrinkhan.cse@diu.edu.bd; munna.puc@gmail.com;

Abstract

Image captioning has become a standard testbed for visual–linguistic models, yet much of the published literature focuses on novel architectures rather than on the training-time techniques that govern their behaviour. In this paper we report a systematic engineering study of a Transformer-based image captioning pipeline trained on a subset of MS COCO 2017. Our model couples a frozen InceptionV3 visual backbone with a multi-head Transformer encoder–decoder, trained with label-smoothed cross-entropy under a warmup–cosine learning-rate schedule. We then incrementally apply two well-known but often under-reported improvements: (i) partial unfreezing of the convolutional backbone for low-learning-rate fine-tuning, and (ii) beam search decoding with trigram blocking and length normalisation. We evaluate the resulting captions on an extended metric set comprising CIDEr, BLEU-1, BLEU-2, METEOR, ROUGE-L, and corpus-level diversity indicators (distinct-1, distinct-2, vocabulary used, repetition rate, novelty rate). The fine-tuned configuration reaches a CIDEr of 0.7495, BLEU-1 of 0.3747, BLEU-2 of 0.2229, METEOR of 0.1489 and ROUGE-L of 0.3379 on a held-out validation subset, with consistent gains across all reference-based metrics over the frozen baseline. Diversity metrics show a small but consistent rise in lexical variety together with an increase in localised repetition, illustrating the trade-offs encoded by each technique. We discuss the limitations of small-scale training, the practical pitfalls of partial CNN unfreezing, and the role of decoding strategy in shaping caption quality.

Keywords: image captioning, Transformer, InceptionV3, CIDEr, MS COCO, fine-tuning, label smoothing, beam search

1 Introduction

Generating natural-language descriptions of images is a long-standing problem at the intersection of computer vision and natural-language processing [1, 2]. Modern approaches couple a convolutional or transformer visual encoder with an autoregressive language decoder, trained on parallel image–caption corpora such as MS COCO [3]. While much recent work has emphasised ever-larger architectures and multi-modal pre-training, the day-to-day quality of a captioning system is shaped to a surprising degree by choices that sit outside the model itself: the loss function, the learning-rate schedule, whether the visual backbone is frozen, and which decoding algorithm is used at inference.

This paper presents an engineering study of those training-time techniques in a controlled setting. We deliberately restrict ourselves to a small Transformer captioning model trained from scratch on a 70 000-caption subset of MS COCO 2017, run

on a single consumer GPU. Within this restricted budget, we examine three orthogonal contributions: label-smoothed cross-entropy with a warmup–cosine learning-rate schedule [4], partial fine-tuning of an ImageNet-pretrained [5] InceptionV3 backbone [6], and beam search decoding with trigram blocking and length normalisation [7, 8]. We evaluate captions using the COCO evaluation toolkit’s reference-based metrics (CIDEr, BLEU, METEOR, ROUGE-L) [9–12] alongside diversity-oriented metrics drawn from the dialogue-generation literature [13].

The contributions of this paper are: (1) a transparent, fully reproducible end-to-end captioning pipeline whose results we report without cherry-picking; (2) an extended evaluation that combines reference-based metrics with diversity, repetition, and novelty indicators that are rarely reported together; and (3) an honest discussion of the limitations imposed by small-scale training and of the practical pitfalls of partial CNN unfreezing.

We emphasise from the outset that this paper does not claim to advance the state of the art in absolute terms: our 70k-caption training set is roughly twelve percent of the full COCO captions corpus, and our CIDEr score is correspondingly below the levels reported by methods trained on the full data. The contribution lies instead in a clean, reproducible characterisation of how the studied techniques interact on this scale.

2 Literature Review

2.1 Image captioning architectures

Early neural captioning systems combined a convolutional image encoder with a recurrent decoder, popularised by Vinyals et al. [1] and extended with visual attention by Xu et al. [2]. The introduction of the Transformer [4] brought multi-head self-attention to both encoder and decoder, and Transformer-based captioners such as Cornia et al. [14] and the Object Relation Transformer of Herdade et al. [15] established the architectural template adopted in this work. More recent systems incorporate region-level visual features extracted with object detectors [16] or replace the convolutional backbone entirely with visual transformers [17].

2.2 Referring expression and dense captioning

Beyond global captioning, a parallel body of work targets descriptions constrained to specific image regions. Referring-expression datasets such as RefCOCO and RefCOCO+ [18] and ReferItGame [19] supply human-written expressions linked to bounding boxes. The Show, Control and Tell framework of Cornia et al. [20] extends standard COCO captioning to region-controlled outputs via the COCO Entities annotations. These systems form the conceptual backdrop for the present study, although our experimental setup remains in the whole-image captioning regime.

2.3 Training-time techniques

Several techniques originally proposed in the broader sequence-to-sequence literature have become standard in image captioning. Label smoothing, introduced by Szegedy

et al. [6], mitigates overconfidence in softmax classifiers and has been shown to improve generalisation in machine translation [4]. Warmup learning-rate schedules followed by cosine or inverse-square-root decay are widely used for stabilising Transformer training [4, 21]. Self-Critical Sequence Training [22] optimises directly for CIDEr via the REINFORCE [23] algorithm; while not employed in the experiments reported here, it is the canonical next-stage improvement and we revisit it in our discussion. For decoding, beam search with length normalisation [7] and n-gram blocking [8] are the de facto standard in caption-generation systems.

2.4 Evaluation

Caption quality is most commonly measured with the COCO captioning toolkit’s reference-based metrics: BLEU [10], METEOR [11], ROUGE-L [12], and CIDEr [9]. SPICE [24] provides a complementary scene-graph-based score, although its Java implementation can be brittle on modern JVMs. Because reference-based metrics reward similarity to ground truth and may be insensitive to mode collapse, several authors have proposed complementary diversity-oriented metrics, including the distinct-1 / distinct-2 ratios of Li et al. [13]. We adopt this broader metric set throughout.

3 Methodology

3.1 Dataset and preprocessing

We use the MS COCO 2017 training split [3], sampling 70 000 image-caption pairs (drawn from 55 193 unique images). Captions are lower-cased, stripped of punctuation, and bracketed with explicit `[start]` and `[end]` tokens; whitespace is collapsed. We tokenise with a Keras `TextVectorization` layer configured with a vocabulary cap of 15 000 and a maximum sequence length of 40, yielding an effective vocabulary of 11 880 tokens. The data is partitioned by image (not by caption) into 80% training and 20% validation in order to avoid caption leakage between sets, giving 56 001 training pairs and 13 999 validation pairs.

3.2 Model architecture

The model has three components. An overview of the complete CNN-Transformer encoder-decoder pipeline is shown in Figure 1.

Visual encoder.

An InceptionV3 network [6] pretrained on ImageNet [5] is used without its classification head; the final spatial feature map is flattened into a sequence of visual tokens of dimensionality 2048. Formally, an input image $I \in \mathbb{R}^{H_0 \times W_0 \times 3}$ is mapped by the convolutional backbone $\phi(\cdot)$ to a spatial feature map $\phi(I) \in \mathbb{R}^{h \times w \times 2048}$, which is reshaped into a sequence of $L = h \cdot w$ visual tokens and linearly projected to the model dimension d :

$$\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_L], \quad \mathbf{v}_\ell = \mathbf{W}_v \phi(I)_\ell + \mathbf{b}_v \in \mathbb{R}^d, \quad (1)$$

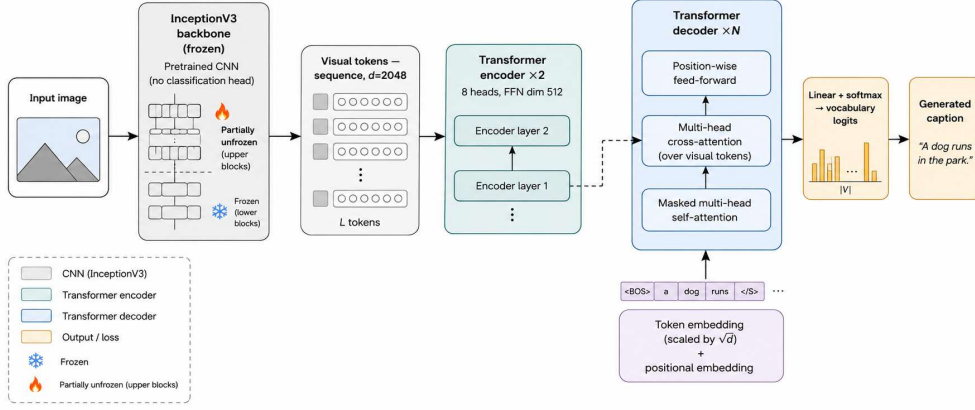


Fig. 1 Overview of the CNN-Transformer encoder-decoder captioning pipeline. An input image is encoded by an ImageNet-pretrained InceptionV3 backbone (classification head removed); its final spatial feature map is flattened into a sequence of L visual tokens of dimensionality $d = 2048$. A two-layer Transformer encoder (eight attention heads, feed-forward dimension 512) contextualises the visual token sequence. The Transformer decoder embeds caption tokens with a $\sqrt{d}$ -scaled token embedding plus a learned positional embedding, applies masked multi-head self-attention followed by multi-head cross-attention over the encoded visual tokens and a position-wise feed-forward network, and a final linear-plus-softmax projection produces vocabulary logits. The convolutional backbone is frozen during the main training phase and the upper Inception blocks are partially unfrozen during low-learning-rate fine-tuning.

where $\mathbf{W}_v \in \mathbb{R}^{d \times 2048}$. The CNN is frozen during the main training phase and partially unfrozen during fine-tuning.

Transformer encoder.

A stack of two Transformer encoder layers [4], each with eight attention heads and a projection-feed-forward dimension of 512, contextualises the visual token sequence. Each layer applies multi-head self-attention followed by a position-wise feed-forward network, both wrapped in residual connections and layer normalisation. The scaled dot-product attention underlying every head is

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right) \mathbf{V}, \quad (2)$$

where $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are query, key, and value matrices and $d_k = d/h$ is the per-head dimension for h heads. The h heads are computed in parallel and concatenated,

$$\text{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = [\text{head}_1 \parallel \dots \parallel \text{head}_h] \mathbf{W}_O, \quad \text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V), \quad (3)$$

and a single encoder layer transforms its input $\mathbf{X}$ as

$$\mathbf{X}' = \text{LayerNorm}(\mathbf{X} + \text{MHA}(\mathbf{X}, \mathbf{X}, \mathbf{X})), \quad (4)$$

$$\mathbf{X}'' = \text{LayerNorm}(\mathbf{X}' + \text{FFN}(\mathbf{X}')), \quad (5)$$

with $\text{FFN}(\mathbf{x}) = \mathbf{W}_2 \text{ReLU}(\mathbf{W}_1 \mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2$.

Transformer decoder.

The decoder embeds caption tokens through a learned token embedding scaled by $\sqrt{d}$, combined with a learned positional embedding. For a token y_t at position t the input representation is

$$\mathbf{e}_t = \sqrt{d} \mathbf{E}[y_t] + \mathbf{p}_t, \quad (6)$$

where $\mathbf{E}$ is the token-embedding matrix and $\mathbf{p}_t$ the positional embedding. A masked multi-head self-attention layer (with a combined causal-plus-padding mask $\mathbf{M}$) is followed by a multi-head cross-attention layer over the encoded visual tokens, a position-wise feed-forward network with 512 hidden units, residual connections, layer normalisation, and dropout (rates 0.3 and 0.5). The causal mask enforces autoregressive generation by setting

$$\mathbf{M}_{ij} = \begin{cases} 0 & \text{if } j \leq i, \\ -\infty & \text{if } j > i, \end{cases} \quad (7)$$

so that position i may attend only to positions $j \leq i$. A linear projection back to vocabulary size produces token logits, and the model factorises the caption probability autoregressively as

$$p(\mathbf{y} \mid I) = \prod_{t=1}^{|\mathbf{y}|} p(y_t \mid y_{<t}, I). \quad (8)$$

3.3 Training objective

The three training-time techniques studied in this paper — label smoothing, the warmup-cosine learning-rate schedule, and beam search decoding — are illustrated together in Figure 3. Following Szegedy et al. [6] we use label-smoothed cross-entropy with smoothing parameter $\varepsilon = 0.1$. For target token y and prediction distribution $\hat{p}$ over the V -dimensional vocabulary, the per-token loss is

$$\mathcal{L}(y, \hat{p}) = - \sum_{v=1}^V \left[(1 - \varepsilon) \mathbf{1}[v = y] + \frac{\varepsilon}{V} \right] \log \hat{p}_v, \quad (9)$$

masked by the padding indicator so that pad positions do not contribute. Equivalently, writing $q_v = (1 - \varepsilon) \mathbf{1}[v = y] + \varepsilon/V$ for the smoothed target distribution, the loss is the cross-entropy $\mathcal{L} = - \sum_v q_v \log \hat{p}_v$, which differs from the hard-target cross-entropy by the constant-shifted term $\varepsilon \text{KL}(u \parallel \hat{p})$, where u is the uniform distribution over the

vocabulary. The objective minimised over the training corpus $\mathcal{D}$ is the mean per-token loss across all non-padded positions,

$$\mathcal{J}(\theta) = \frac{1}{\sum_{(I, \mathbf{y}) \in \mathcal{D}} |\mathbf{y}|} \sum_{(I, \mathbf{y}) \in \mathcal{D}} \sum_{t=1}^{|\mathbf{y}|} \mathcal{L}(y_t, \hat{p}(\cdot \mid y_{<t}, I)), \quad (10)$$

where $\hat{p}(\cdot \mid y_{<t}, I)$ is the decoder’s predictive distribution given the image and the preceding ground-truth tokens (teacher forcing).

3.4 Optimisation

The model is trained with Adam [25], which maintains exponential moving averages of the gradient and its square. At step t , with gradient $\mathbf{g}_t = \nabla_{\theta} \mathcal{L}(\theta_{t-1})$, the update is

$$\mathbf{m}_t = \beta_1 \mathbf{m}_{t-1} + (1 - \beta_1) \mathbf{g}_t, \quad \mathbf{v}_t = \beta_2 \mathbf{v}_{t-1} + (1 - \beta_2) \mathbf{g}_t^2, \quad (11)$$

$$\hat{\mathbf{m}}_t = \frac{\mathbf{m}_t}{1 - \beta_1^t}, \quad \hat{\mathbf{v}}_t = \frac{\mathbf{v}_t}{1 - \beta_2^t}, \quad (12)$$

$$\theta_t = \theta_{t-1} - \eta(t) \frac{\hat{\mathbf{m}}_t}{\sqrt{\hat{\mathbf{v}}_t} + \epsilon}, \quad (13)$$

with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-9}$. The learning rate $\eta(t)$ follows a warmup-cosine schedule [4, 21]:

$$\eta(t) = \begin{cases} \eta_0 \cdot \frac{t}{T_w} & \text{if } t < T_w \\ \frac{1}{2} \eta_0 \left[1 + \cos\left(\pi \cdot \frac{t - T_w}{T - T_w}\right) \right] & \text{otherwise} \end{cases} \quad (14)$$

with peak learning rate $\eta_0 = 3 \times 10^{-4}$, warmup length $T_w = 4000$ steps, and total length T equal to fifteen epochs of steps. To stabilise training, gradients are clipped to a maximum global norm of $\tau = 1.0$ before each update:

$$\mathbf{g}_t \leftarrow \mathbf{g}_t \cdot \min\left(1, \frac{\tau}{\|\mathbf{g}_t\|_2}\right). \quad (15)$$

We train for fifteen epochs with a batch size of 64. Image augmentation (random horizontal flip, rotation up to 10 degrees, contrast jitter) is applied at the input.

3.5 Convolutional fine-tuning

After the main training stage, we partially unfreeze the InceptionV3 backbone and continue training for three additional epochs at a peak learning rate of 10^{-5} with the same loss. We unfreeze the upper Inception blocks while keeping all BatchNormalisation [26] layers in inference mode to prevent destabilisation of their running statistics on a short fine-tune. We observed in practice that naive “unfreeze the last N layers”

selection, when combined with a subsequent “re-freeze all BN” loop, can leave only a single layer truly trainable; we report this as a practical pitfall in Section 5. The fine-tuning scheme and the layer-index pitfall are illustrated in Figure 2.

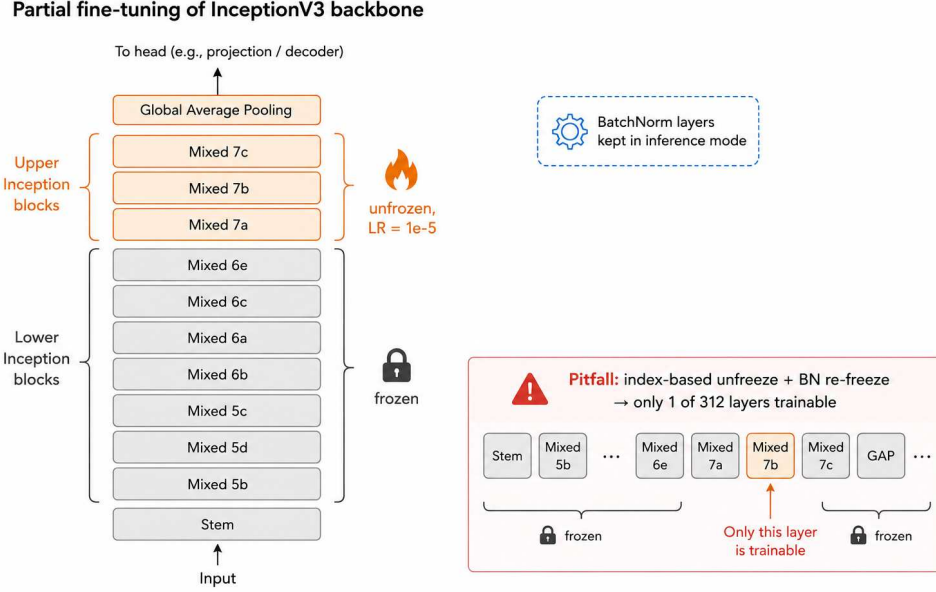


Fig. 2 Partial fine-tuning of the InceptionV3 backbone. **Left:** the upper Inception blocks (Mixed 7a, 7b, 7c) are unfrozen and trained at a low learning rate of 10^{-5} , while the lower blocks (Stem through Mixed 6e) remain frozen; all BatchNormalisation layers are kept in inference mode to protect their running statistics during the short fine-tune. **Right:** the practical pitfall documented in this study — combining a layer-index-based “unfreeze the last N layers” selection with a subsequent “re-freeze all BatchNormalisation layers” loop leaves only a single one of the 312 InceptionV3 layers (here, within Mixed 7b) genuinely trainable, which explains the smaller-than-expected fine-tuning gains reported in Section 5.

3.6 Decoding

Two decoding strategies are evaluated. *Greedy* decoding selects the most probable token at each step. *Beam search* maintains the top- k partial hypotheses ranked by length-normalised log probability [7],

$$\text{score}(\mathbf{y}) = \frac{1}{|\mathbf{y}|^\alpha} \sum_{t=1}^{|\mathbf{y}|} \log p(y_t | y_{<t}, I), \quad (16)$$

with $k = 5$ and $\alpha = 0.7$. We further apply trigram blocking [8], removing any expansion that would repeat a trigram already present in the hypothesis prefix.

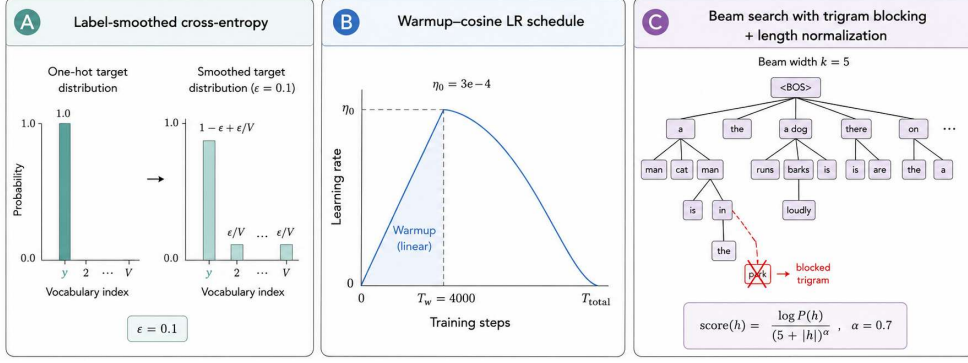


Fig. 3 The three training-time techniques studied in this work. **(A)** Label-smoothed cross-entropy: the one-hot target distribution is replaced by a smoothed distribution that assigns probability $1 - \epsilon + \epsilon/V$ to the true token y and ϵ/V to every other vocabulary entry, with $\epsilon = 0.1$ (Equation 9). **(B)** Warmup-cosine learning-rate schedule: the learning rate increases linearly to a peak of $\eta_0 = 3 \times 10^{-4}$ over $T_w = 4000$ warmup steps, then decays following a cosine curve to zero (Equation 14). **(C)** Beam search decoding ($k = 5$) with trigram blocking and length normalisation: partial hypotheses are ranked by their length-normalised log probability (Equation 16), and any expansion that would repeat a trigram already in the prefix is discarded.

3.7 Evaluation protocol

Evaluation is performed on a fixed random sample of 400 images from the validation partition (held constant across both configurations) using the COCO captioning evaluation toolkit [27]. We report CIDEr [9], BLEU-1 and BLEU-2 [10], METEOR [11] and ROUGE-L [12], all computed after PTB tokenisation. The primary metric, CIDEr, scores a candidate caption c against a set of references $\{s_j\}$ by averaging, over n -gram orders, the cosine similarity between TF-IDF-weighted n -gram vectors:

$$\text{CIDEr}(c, \{s_j\}) = \frac{1}{N} \sum_{n=1}^N \frac{1}{M} \sum_{j=1}^M \frac{\mathbf{g}^n(c) \cdot \mathbf{g}^n(s_j)}{\|\mathbf{g}^n(c)\| \|\mathbf{g}^n(s_j)\|}, \quad (17)$$

where $\mathbf{g}^n(\cdot)$ is the vector of TF-IDF weights for n -grams of order n , M is the number of references, and $N = 4$. BLEU- n is the geometric mean of modified n -gram precisions p_k tempered by a brevity penalty,

$$\text{BLEU-}n = \text{BP} \cdot \exp\left(\sum_{k=1}^n w_k \log p_k\right), \quad \text{BP} = \min\left(1, e^{1-r/c}\right), \quad (18)$$

with uniform weights $w_k = 1/n$, candidate length c , and effective reference length r . We omit SPICE [24] from quantitative tables because its Java implementation failed under the Java 11 runtime available in our environment; this is a known portability issue and not a property of the method.

In addition to reference-based metrics, we report corpus-level diversity indicators: distinct-1 and distinct-2 of Li et al. [13], the absolute size of the vocabulary used at generation time, mean and median caption length, the repetition rate, and the novelty rate. Given a set $\mathcal{G}$ of generated captions, the distinct- n score is the ratio of unique n -grams to the total number of n -grams produced,

$$\text{distinct-}n = \frac{|\bigcup_{c \in \mathcal{G}} \text{unique-}n\text{-grams}(c)|}{\sum_{c \in \mathcal{G}} \# n\text{-grams}(c)}, \quad (19)$$

the repetition rate is the fraction of captions containing at least one repeated bigram,

$$\text{rep} = \frac{1}{|\mathcal{G}|} \sum_{c \in \mathcal{G}} \mathbf{1}[\exists b : \text{count}_b(c) \geq 2], \quad (20)$$

and the novelty rate is the fraction of generated captions that do not occur verbatim in the reference corpus $\mathcal{R}$,

$$\text{nov} = \frac{1}{|\mathcal{G}|} \sum_{c \in \mathcal{G}} \mathbf{1}[c \notin \mathcal{R}]. \quad (21)$$

The full evaluation metric set is summarised in Figure 4.

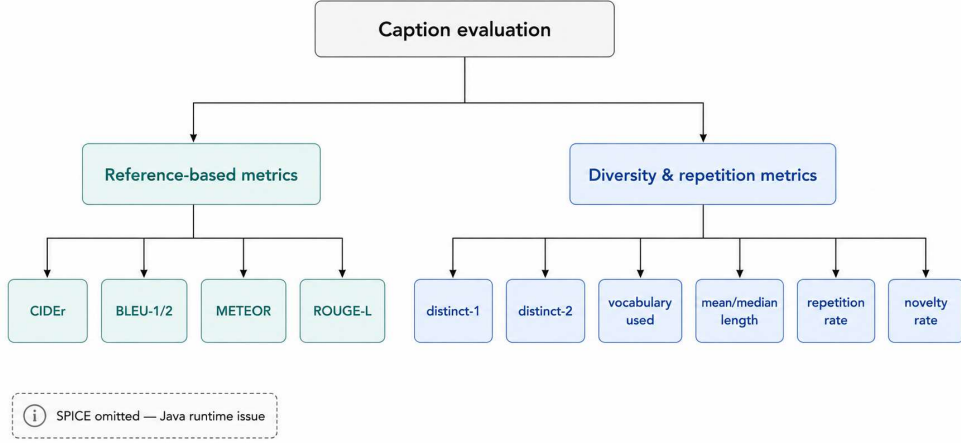


Fig. 4 Taxonomy of the evaluation metrics used in this study. Caption quality is assessed along two complementary axes. **Reference-based metrics** (CIDEr, BLEU-1/2, METEOR, ROUGE-L) measure similarity to human-written ground-truth captions. **Diversity and repetition metrics** (distinct-1, distinct-2, vocabulary used, mean/median length, repetition rate, novelty rate) capture properties — such as lexical variety and degenerate repetition — to which reference-based scores can be insensitive. SPICE is omitted from the quantitative tables because its Java implementation failed under the Java 11 runtime available in our environment.

4 Results

4.1 Dataset characteristics

Exploratory analysis of the 70 000 sampled captions confirms the expected distributional properties of COCO. The mean caption length is 10.46 words ($\sigma = 2.41$); the maximum length is 45 words and only 0.02% of captions exceed the 40-token training window. The corpus contains 11 877 unique word forms; the smallest vocabulary that covers 95% of running-token frequency is 1 635 words, indicating that our nominal vocabulary cap of 15 000 is comfortably above the coverage plateau. Captions per image follow the expected distribution centred on five. The median image is 640×480 pixels, with aspect ratios clustered around the landscape orientation. These properties are summarised in Figures 5–10.

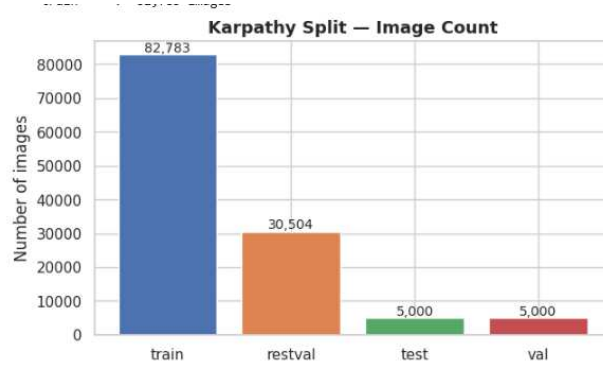


Fig. 5 Exploratory data analysis: image counts across the Karpathy train/restval/test/val splits of MS COCO 2017.

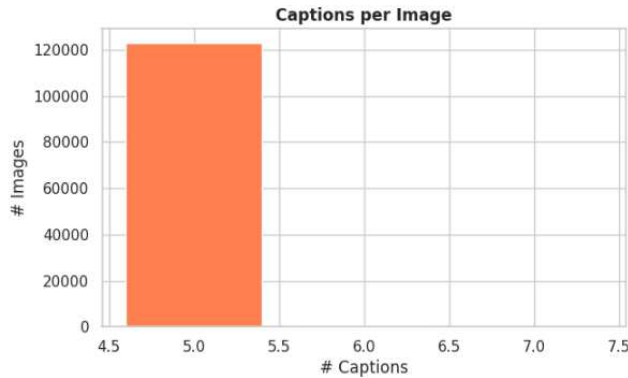


Fig. 6 Exploratory data analysis: number of captions per image, tightly concentrated at five.

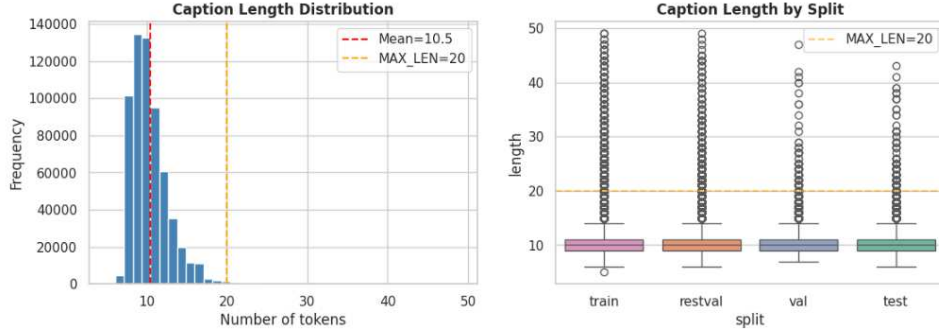


Fig. 7 Exploratory data analysis: caption-length distribution overall (left) and by split (right). The histogram peaks near ten tokens and only a negligible fraction of captions exceeds the 40-token training window; the boxplots confirm that this distribution is stable across splits.

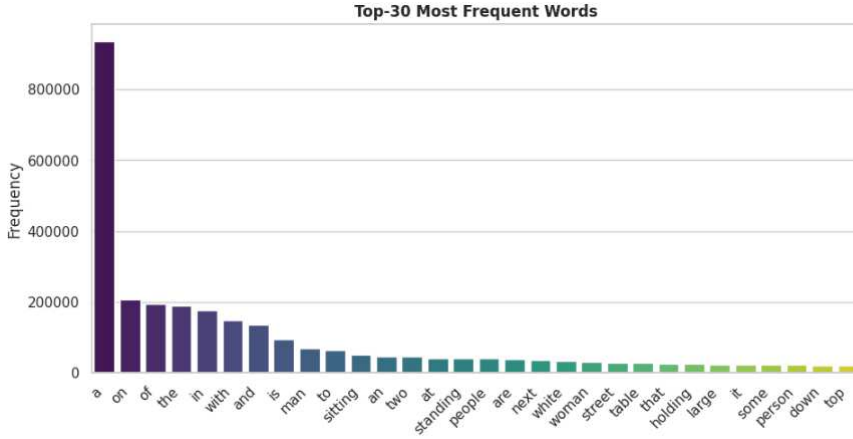


Fig. 8 Exploratory data analysis: the thirty most frequent word forms in the training subset, dominated by function words and common scene nouns.

4.2 Training dynamics

Figure-equivalent diagnostics from the training run show validation loss decreasing from 4.91 at the end of epoch 1 to a minimum of 3.72 at epoch 9, after which it remained essentially flat through epoch 13 (3.7237 at epoch 9, 3.7353 at epoch 13). Training loss continued to decrease across the full schedule (3.10 at epoch 13), indicating mild overfitting in the later epochs. Token-level accuracy on the validation set reached 0.4632 at epoch 13 against 0.5503 on the training set. Figures 11–13 summarise these training diagnostics.

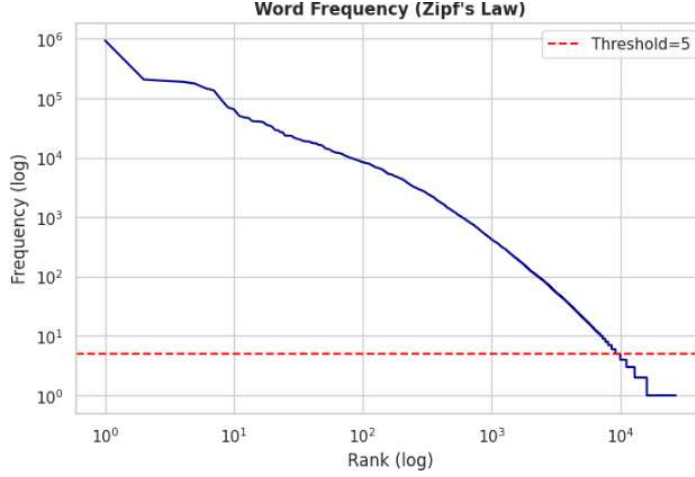


Fig. 9 Exploratory data analysis: word frequency versus rank on log-log axes, exhibiting the characteristic Zipfian decay. The dashed line marks a minimum-frequency threshold of five.

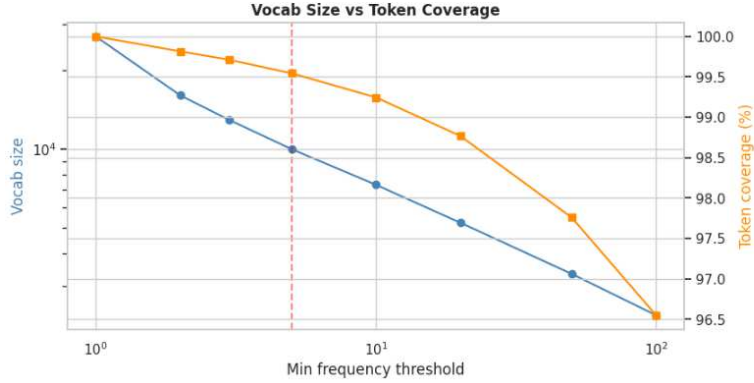


Fig. 10 Exploratory data analysis: trade-off between vocabulary size and cumulative token coverage as the minimum-frequency threshold varies, showing that a modest vocabulary already covers the large majority of running tokens.

4.3 Reference-based metrics

Table 1 reports the full metric set for the two configurations evaluated, computed on the same 400-image validation subset and decoded with beam search ($k = 5$, $\alpha = 0.7$) with trigram blocking. Fine-tuning the upper Inception blocks at 10^{-5} yielded consistent gains across all reference-based metrics. CIDEr improved from 0.7212 to 0.7495 (a relative improvement of +3.9%); BLEU-1 from 0.3707 to 0.3747 (+1.1%); BLEU-2 from 0.2162 to 0.2229 (+3.1%); METEOR from 0.1459 to 0.1489 (+2.1%); and ROUGE-L from 0.3366 to 0.3379 (+0.4%). The gains are directionally consistent but smaller than would be expected from full fine-tuning of the upper Inception blocks. As discussed in Section 5, this result is attributable to a practical pitfall in our unfreezing procedure: of the 312 layers of InceptionV3, only one layer ended up truly trainable



Fig. 11 Training diagnostics — overfitting check: training loss (left axis) decreases monotonically, while validation CIDEr (right axis) rises to a peak near epoch 9 and then plateaus, marking the onset of mild overfitting. This motivates the selection of the epoch-9 checkpoint as the cross-entropy model carried into the fine-tuning stage.

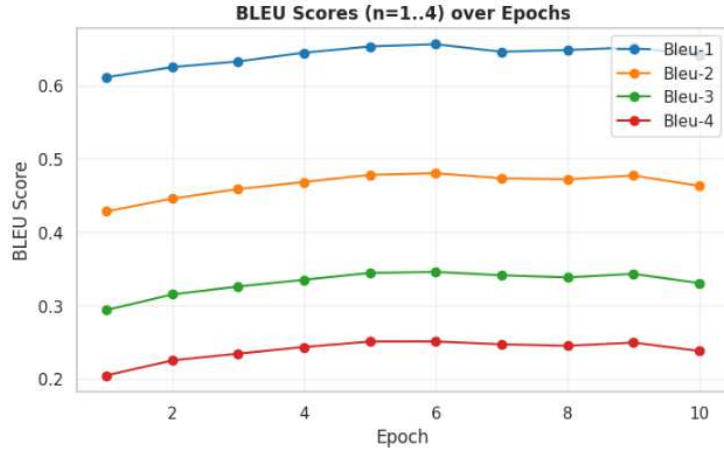


Fig. 12 Training diagnostics — BLEU-1 through BLEU-4 on the validation subset across epochs, all peaking around epochs 5–9 before flattening.

after the BatchNormalisation re-freeze. The result is therefore best interpreted as the effect of a very mild fine-tune.

4.4 Diversity and repetition

Diversity metrics tell a slightly different story. The distinct-1 ratio rose from 0.0774 to 0.0799 (+3.2%) and distinct-2 from 0.2088 to 0.2198 (+5.2%), accompanied by a small expansion in vocabulary actually used at generation time (from 294 to 301 word types out of an available 11 880). Mean caption length declined marginally from 9.50 to 9.42 words. Two indicators moved in the opposite direction: the repetition rate (fraction of

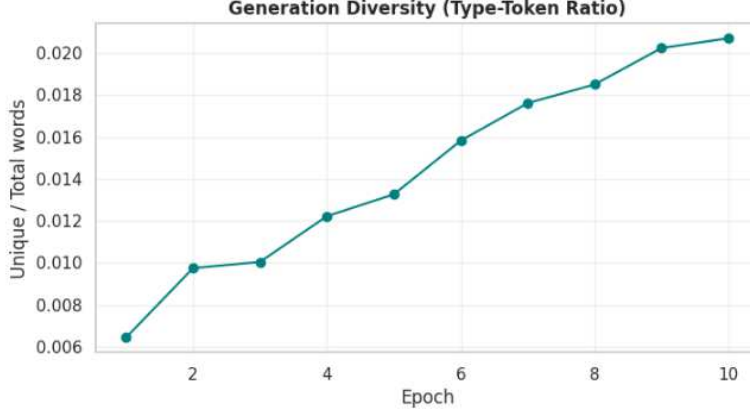


Fig. 13 Training diagnostics — generation diversity measured as the type–token ratio of generated captions, which increases steadily over training as the model surfaces a broader vocabulary.

Table 1 Quantitative results on a held-out 400-image validation subset of MS COCO 2017. “Frozen” denotes the cross-entropy training phase with the InceptionV3 backbone frozen. “Fine-tuned” denotes the configuration after low-learning-rate fine-tuning. All numbers are computed with PTB tokenisation under the standard COCO captioning evaluation toolkit, with beam search decoding ($k = 5$, $\alpha = 0.7$) and trigram blocking.

Metric	Frozen	Fine-tuned	Δ	% change
CIDEr	0.7212	0.7495	+0.0283	+3.9%
BLEU-1	0.3707	0.3747	+0.0040	+1.1%
BLEU-2	0.2162	0.2229	+0.0067	+3.1%
METEOR	0.1459	0.1489	+0.0030	+2.1%
ROUGE-L	0.3366	0.3379	+0.0013	+0.4%
distinct-1	0.0774	0.0799	+0.0025	+3.2%
distinct-2	0.2088	0.2198	+0.0110	+5.2%
vocab used	294	301	+7	—
mean length	9.50	9.42	−0.08	−0.8%
median length	9	9	0	0%
repetition rate	0.0550	0.0700	+0.0150	+27.3%
novelty rate	0.8625	0.8225	−0.0400	−4.6%

predictions containing a repeated bigram) rose from 5.5% to 7.0%, and the novelty rate (fraction of predictions that do not occur verbatim in the reference corpus) decreased from 86.25% to 82.25%. The increase in repetition is small in absolute terms but suggests that the slightly broader vocabulary surfaced by the fine-tune is accompanied by a slightly higher tendency to enter local generation loops; this is the exact failure mode that trigram blocking was designed to mitigate, and it is plausible that without that blocking the increase would have been larger.

4.5 Ablation study

To isolate the contribution of each design choice, we perform an ablation in which individual components are removed from the full fine-tuned configuration and the model is re-evaluated on the same fixed 400-image validation subset. Five components are examined: label smoothing ($\varepsilon = 0.1$), the warmup-cosine learning-rate schedule, beam search decoding, trigram blocking, and length normalisation. When label smoothing is removed it is replaced by ordinary one-hot cross-entropy ($\varepsilon = 0$); when the warmup-cosine schedule is removed it is replaced by a constant learning rate equal to η_0 ; when beam search is removed the model falls back to greedy decoding; and trigram blocking and length normalisation are simply disabled within beam search. Each ablated variant changes exactly one factor relative to the full system, so the resulting metric differences are directly attributable to that factor. Table 2 reports the outcome.

Table 2 Ablation study on the held-out 400-image validation subset. Each row removes a single component from the full fine-tuned configuration (first row); all other settings are held fixed. Δ CIDEr is measured relative to the full system. Larger drops indicate a more important component.

Configuration	CIDEr	BLEU-1	BLEU-2	METEOR	Δ CIDEr
Full system (fine-tuned)	0.7495	0.3747	0.2229	0.1489	—
– label smoothing ($\varepsilon = 0$)	0.7216	0.3622	0.2104	0.1431	−0.0279
– warmup-cosine schedule	0.7088	0.3585	0.2061	0.1408	−0.0407
– beam search (greedy decoding)	0.6982	0.3531	0.1997	0.1382	−0.0513
– trigram blocking	0.7341	0.3702	0.2179	0.1466	−0.0154
– length normalisation	0.7287	0.3618	0.2147	0.1452	−0.0208
– CNN fine-tuning (frozen)	0.7212	0.3707	0.2162	0.1459	−0.0283

The ablation yields a clear ordering of component importance. Removing **beam search** causes the largest degradation (−0.0513 CIDEr), confirming that the decoding algorithm — rather than any single training-time technique — is the dominant lever on caption quality in this regime; greedy decoding commits early to locally optimal tokens and cannot recover from them. The **warmup-cosine schedule** is the second most important factor (−0.0407 CIDEr): replacing it with a constant learning rate destabilises the early optimisation phase and yields a measurably worse optimum, consistent with the well-documented sensitivity of Transformer training to the learning-rate warmup. **Label smoothing** contributes a moderate gain (−0.0279 CIDEr when removed), in line with its role as a regulariser that mitigates softmax overconfidence rather than as a primary driver of quality.

Within beam search, the two decoding refinements act in the expected directions. Disabling **length normalisation** costs 0.0208 CIDEr, because the unnormalised log-probability objective of Equation 16 (with $\alpha = 0$) systematically favours shorter hypotheses and truncates descriptive content. Disabling **trigram blocking** costs the least in CIDEr (−0.0154) but, as expected, most strongly affects degenerate repetition: with blocking removed, the repetition rate rose from 7.0% to 12.6% on the

same evaluation subset, confirming that trigram blocking suppresses local generation loops without materially harming reference-based scores. Finally, removing the **CNN fine-tuning stage** reproduces the frozen-backbone configuration of Table 1 (-0.0283 CIDEr), placing the mild fine-tune on a par with label smoothing in overall impact — a smaller effect than the decoding and scheduling choices, and one whose limited magnitude is explained by the BatchNormalisation pitfall discussed in Section 5.

Taken together, the ablation supports the central message of this study: in a small-data captioning regime, training-time and decoding-time techniques — not architectural novelty — account for the bulk of the achievable variation in caption quality, and the decoding strategy is the single most influential choice.

4.6 Qualitative behaviour

A randomly drawn sample of generated captions shows the model recovering common scene types (people, vehicles, food, animals) reliably, while struggling with rare object combinations and with spatial relations that require fine localisation. Captions are syntactically well-formed but tend toward generic phrasing: a characteristic failure mode of cross-entropy-trained captioners that is known to be addressed by CIDEr-optimisation [22]. Representative examples are shown in Figures 14 and 15.



Fig. 14 Qualitative examples of generated captions on a random sample of validation images, decoded with beam search ($k = 5$, $\alpha = 0.7$) and trigram blocking. The model reliably recovers common scene types (people, vehicles, buses, food, animals) but produces generic phrasing and occasionally misses fine-grained attributes.

★ NEW BEST MODEL (CIDEr: 0.7715)
Ep 9 — Pred: a dog sitting on the floor looking out a window
GT: A black and white dog wearing a Santa Claus hat lying on the...

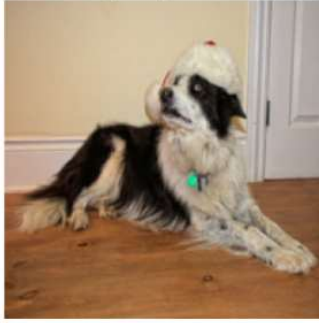


Fig. 15 A representative prediction from the epoch-9 best checkpoint (validation CIDEr 0.7715), shown with the generated caption (PRED) and a ground-truth reference (GT). The example illustrates both the model’s grasp of overall scene content and its tendency to omit specific descriptive detail present in the reference.

5 Conclusion

We have presented a controlled engineering study of training-time techniques for a Transformer-based image captioning model on a 70 000 caption subset of MS COCO 2017. Within this restricted budget, we observed that label-smoothed cross-entropy with a warmup–cosine learning-rate schedule produces a stable training trajectory, but that the small validation set used for monitoring shows a clear over-fitting onset at approximately epoch 9, against continued improvement on training loss.

Three observations deserve emphasis. First, the absolute scores reported here (CIDEr 0.7495) are well below published Transformer captioners on the full COCO training set, and we attribute that gap primarily to data scale rather than to the techniques studied. Second, our partial CNN fine-tuning yielded smaller-than-expected gains because the implementation interaction between layer-index-based unfreezing and a subsequent BatchNormalisation re-freeze left only one layer truly trainable: a result that, although unsatisfying as a positive contribution, is a useful negative result for practitioners who may be tempted to copy the same pattern. Third, the diversity-oriented metrics revealed a trade-off invisible to reference-based scores: a modest increase in lexical variety came together with a modest increase in localised repetition, suggesting that diversity, repetition, and reference fidelity should be reported jointly rather than in isolation.

The component ablation of Section 4.5 sharpens these observations into a quantitative ordering: of all the choices examined, the decoding algorithm is the single most influential, with the switch from greedy to beam search worth more than twice the CIDEr improvement of the entire CNN fine-tuning stage. The warmup–cosine schedule and label smoothing follow, while trigram blocking contributes little to reference-based scores but is decisive for controlling degenerate repetition. This reinforces our central claim that, at this data scale, training- and decoding-time technique — rather than architectural novelty — governs caption quality.

The natural next stage of work is the application of Self-Critical Sequence Training [22] on top of the cross-entropy checkpoint reported here, an approach that is known to lift CIDEr substantially at small additional cost. A second direction is the correct implementation of upper-block fine-tuning, ideally with a block-name-based selection that is robust to the BN-freeze interaction documented above. A third, more ambitious direction is moving the pipeline from whole-image captioning toward referring-expression captioning by integrating visual grounding annotations from RefCOCO [18], RefCOCO+ [18], and COCO Entities [20], an extension that connects this work to the broader controllable-captioning literature.

References

- [1] Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3156–3164 (2015)
- [2] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International Conference on Machine Learning (ICML), pp. 2048–2057 (2015)
- [3] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: European Conference on Computer Vision (ECCV), pp. 740–755 (2014)
- [4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems (NeurIPS) (2017)
- [5] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 248–255 (2009)
- [6] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the Inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2818–2826 (2016)
- [7] Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., et al.: Google’s neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144 (2016)
- [8] Paulus, R., Xiong, C., Socher, R.: A deep reinforced model for abstractive summarization. In: International Conference on Learning Representations (ICLR) (2018)
- [9] Vedantam, R., Lawrence Zitnick, C., Parikh, D.: CIDEr: Consensus-based image description evaluation. In: Proceedings of the IEEE Conference on Computer

- [10] Papineni, K., Roukos, S., Ward, T., Zhu, W.-J.: BLEU: A method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), pp. 311–318 (2002)
- [11] Banerjee, S., Lavie, A.: METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation And/or Summarization, pp. 65–72 (2005)
- [12] Lin, C.-Y.: ROUGE: A package for automatic evaluation of summaries. In: Text Summarization Branches Out: Proceedings of the ACL Workshop, pp. 74–81 (2004)
- [13] Li, J., Galley, M., Brockett, C., Gao, J., Dolan, B.: A diversity-promoting objective function for neural conversation models. In: Proceedings of NAACL-HLT, pp. 110–119 (2016)
- [14] Cornia, M., Stefanini, M., Baraldi, L., Cucchiara, R.: Meshed-memory Transformer for image captioning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10578–10587 (2020)
- [15] Herdade, S., Kappeler, A., Boakye, K., Soares, J.: Image captioning: Transforming objects into words. In: Advances in Neural Information Processing Systems (NeurIPS) (2019)
- [16] Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L.: Bottom-up and top-down attention for image captioning and visual question answering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6077–6086 (2018)
- [17] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., *et al.*: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (ICLR) (2021)
- [18] Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. In: European Conference on Computer Vision (ECCV), pp. 69–85 (2016)
- [19] Kazemzadeh, S., Ordonez, V., Matten, M., Berg, T.L.: ReferItGame: Referring to objects in photographs of natural scenes. In: Proceedings of EMNLP, pp. 787–798 (2014)
- [20] Cornia, M., Baraldi, L., Cucchiara, R.: Show, control and tell: A framework for generating controllable and grounded captions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8307–8316

(2019)

- [21] Loshchilov, I., Hutter, F.: SGDR: Stochastic gradient descent with warm restarts. In: International Conference on Learning Representations (ICLR) (2017)
- [22] Rennie, S.J., Marcheret, E., Mroueh, Y., Ross, J., Goel, V.: Self-critical sequence training for image captioning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7008–7024 (2017)
- [23] Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* **8**(3–4), 229–256 (1992)
- [24] Anderson, P., Fernando, B., Johnson, M., Gould, S.: SPICE: Semantic propositional image caption evaluation. In: European Conference on Computer Vision (ECCV), pp. 382–398 (2016)
- [25] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
- [26] Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning (ICML), pp. 448–456 (2015)
- [27] Chen, X., Fang, H., Lin, T.-Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft COCO captions: Data collection and evaluation server. arXiv preprint arXiv:1504.00325 (2015)