WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Supervised Learning and SCST Image Captioning on Flickr30k: Analyzing Metric-Driven Trade-offs in Semantic Performance

Ramya B N[1], Manoj G[2], Kushal N S[3], Chandra N[4], Eshwar Siddartha T[5]

[1]Assistant Professor, Artificial Intelligence and Machine Learning, Jyothy Institute of Technology, Bengaluru, Karnataka, India
[2,3,4,5]Student, Artificial Intelligence and Machine Learning, Jyothy Institute of Technology, Bengaluru, Karnataka, India

A B S T R A C T

Image captioning plays a pivotal role in enhancing accessibility, searchability, and human-computer interaction by automatically generating natural language descriptions for images. While standard encoder-decoder models trained using Supervised Learning (SL) with cross-entropy loss optimize word-level accuracy, they often fail to align with sequence-level evaluation metrics such as BLEU, CIDEr, and SPICE. Additionally, these models suffer from exposure bias. This paper explores augmenting SL with Reinforcement Learning (RL), specifically Self-Critical Sequence Training (SCST), to optimize the CIDEr metric, which better correlates with human consensus judgments. Using the Flickr30k dataset, we employed an attention-based ResNet-101 encoder and an LSTM decoder. In the SL phase, the model achieved a minimum validation loss of 3.08 and a token prediction accuracy of 0.404. Fine-tuning for 8 epochs using SCST, targeting the CIDEr metric, resulted in a significant improvement in n-gram metrics (e.g., a +7.3% relative gain in BLEU-3) and a modest +1.4% improvement in CIDEr. However, a -4.6% relative change was observed in the SPICE score, which measures semantic propositional accuracy. This study provides a detailed implementation, quantitative analysis, and qualitative examples, offering insights into the trade-offs of metric-specific optimization in RL for image captioning. It highlights the challenge of balancing improvements in n-gram consensus (CIDEr) with preserving semantic fidelity (SPICE) and underscores the need for diverse evaluation metrics, as optimizing one may negatively impact others.

Keywords: Image captioning, Supervised Learning (SL), Reinforcement Learning (RL), Self-Critical Sequence Training (SCST), CIDEr, SPICE, Encoder-decoder model, Flickr30k dataset, ResNet-101, LSTM

1. INTRODUCTION

1.1. Motivation and Background

The automated generation of descriptive text for images, or **image captioning**, represents a cornerstone challenge at the intersection of computer vision and natural language processing. Its successful application holds transformative potential across multiple domains. For instance, accurate captions enhance web accessibility for visually impaired individuals by providing auditory descriptions of online visual content (Gurari et al., 2020). In multimedia retrieval, descriptive captions enable more effective semantic search beyond simple keyword tagging (Plummer et al., 2015). Furthermore, in robotics and human-computer interaction, enabling machines to perceive and describe their environment in natural language is critical for situated communication and task execution. The motivation for advancing image captioning stems from the need for more accurate, relevant, and human-like descriptions to power these diverse applications.

1.2. Problem Statement: Limitations of Supervised Learning

The dominant paradigm for image captioning adopts an encoder-decoder framework (Vinyals et al., 2015), where a Convolutional Neural Network (CNN) encodes the input image into a feature representation, and a Recurrent Neural Network (RNN), typically an LSTM (Hochreiter & Schmidhuber, 1997), decodes this representation into a natural language sequence. Training is traditionally performed using Supervised Learning (SL) with a cross-entropy loss, maximizing the likelihood of the next ground-truth word given previous ground-truth inputs (teacher forcing).

Despite establishing strong baselines, SL exhibits critical limitations:

1. **Exposure Bias:** During training, the decoder conditions on ground-truth tokens at every timestep, but at inference time, it must rely on its own predictions. This discrepancy leads to error accumulation, where early mistakes can severely derail caption generation (Ranzato et al., 2015).

2. **Loss-Metric Mismatch:** The cross-entropy objective optimizes local, word-level accuracy, whereas evaluation metrics such as BLEU (Papineni et al., 2002), METEOR (Banerjee & Lavie, 2005), CIDEr (Vedantam et al., 2015), and SPICE (Anderson et al., 2016) assess global sequence quality, including grammaticality, relevance, and consensus with human annotations. Consequently, a model with low cross-entropy loss may still generate semantically poor or repetitive captions that perform suboptimally on these metrics.

1.3. Proposed Approach: Reinforcement Learning with SCST

To address these limitations, **Reinforcement Learning (RL)** approaches have been proposed, framing caption generation as a sequential decision-making process. **Self-Critical Sequence Training (SCST)** (Rennie et al., 2017) is an effective RL method adapted from the REINFORCE algorithm (Williams, 1992). SCST reduces the high variance associated with policy gradient estimates by using the model's own greedy decoding reward as a baseline. The agent updates its policy based on the relative advantage of sampled captions compared to this baseline, providing a lower-variance, more stable learning signal. In this work, we adopt SCST to directly optimize the **CIDEr** metric, which emphasizes consensus-based n-gram overlap weighted by TF-IDF and better correlates with human judgments compared to simpler metrics like BLEU.

Additionally, to enhance the decoder's ability to focus on relevant spatial regions of the encoded image, we employ a **Bahdanau-style Additive Attention mechanism** (Bahdanau et al., 2015) between the encoder and decoder modules.

1.4. Contributions and Outline

This study presents a comprehensive implementation and comparative analysis of **Supervised Learning (SL)** versus **SCST fine-tuning** for image captioning on the **Flickr30k** dataset, which contains 31,783 images annotated with five human-generated captions each. The key contributions are:

1. A detailed description of the system architecture, comprising a **ResNet-101 encoder** and an **LSTM decoder with Additive Attention**, along with implementation details for both SL and SCST-CIDEr training phases.
2. A quantitative evaluation using standard metrics (BLEU, METEOR, CIDEr, SPICE) to assess the impact of SCST fine-tuning.
3. A qualitative analysis of generated captions illustrating the practical differences between models trained purely with SL and those fine-tuned with RL.
4. A critical discussion on the observed trade-offs, particularly the improvement in n-gram based metrics versus the degradation in semantic fidelity (as captured by SPICE), offering insights into the consequences of metric-specific RL optimization.
5. Implementation optimizations including the use of **Automatic Mixed Precision (AMP)** during training for improved computational efficiency on GPU hardware.

The paper is organized as follows:

Section 2 reviews related work. Section 3 details the methodology and system design. Section 4 describes the experimental setup. Section 5 presents and discusses the results. Section 6 concludes the paper and outlines limitations. Section 7 proposes directions for future research.

2. Related Work

Image captioning research has evolved across several major stages:

- **Early Encoder-Decoder Models:** Vinyals et al. (2015) proposed mapping CNN features (GoogLeNet) directly to caption sequences using LSTMs, demonstrating the feasibility of end-to-end captioning. Similarly, Karpathy and Fei-Fei (2015) introduced a multimodal RNN model aligning image regions (via R-CNN) with words, emphasizing the potential of grounded visual representations. However, these early models lacked mechanisms to dynamically focus on salient parts of the image during caption generation.
- **Attention Mechanisms:** Xu et al. (2015) introduced visual attention models, allowing LSTMs to dynamically attend to different spatial regions of the image at each decoding step, improving caption relevance and interpretability. Inspired by the **Bahdanau attention mechanism** (Bahdanau et al., 2014) initially developed for machine translation, attention-based models became foundational to modern captioning systems. Later advances, such as bottom-up and top-down attention (Anderson et al., 2018), further improved object-level grounding. In this work, we employ the classical **Additive Attention** formulation for its effectiveness and simplicity.
- **Reinforcement Learning for Optimization:** To address exposure bias and loss-metric mismatch inherent in supervised training, reinforcement learning methods like REINFORCE (Ranzato et al., 2015) and **Self-Critical Sequence Training (SCST)** (Rennie et al., 2017) were introduced. SCST stabilizes training by using the reward from greedy decoding as a baseline, significantly improving sequence-level metrics such as CIDEr. Our approach follows the SCST framework to optimize CIDEr scores.
- **Evaluation Metrics:** While BLEU (Papineni et al., 2002) initially dominated evaluation, its poor correlation with human judgment led to the adoption of richer metrics. METEOR (Banerjee & Lavie, 2005) emphasized synonymy and stemming, CIDEr (Vedantam et al., 2015) incorporated TF-IDF weighting for human consensus evaluation, and SPICE (Anderson et al., 2016) focused on semantic propositional content.

- **Recent Trends:** Although Transformer-based models (e.g., ViT, Swin Transformers, and vision-language transformers) have recently achieved state-of-the-art results (Cornia et al., 2020; Herdade et al., 2019), CNN-RNN based architectures, especially when combined with reinforcement learning fine-tuning strategies, remain highly relevant for image captioning tasks.

Our contribution is a detailed and reproducible study applying SCST-CIDEr optimization on the Flickr30k dataset, highlighting the trade-offs between improving n-gram consensus (CIDEr) and maintaining semantic fidelity (SPICE).

3. Methodology and System Design

3.1 Data Handling and Preparation

- **Dataset:** The Flickr30k dataset (Young et al., 2014) serves as the testbed, comprising 31,783 images, each paired with 5 independent human-generated captions. This dataset provides sufficient scale and diversity for training and evaluation.
- **Data Splits:** The dataset is partitioned based on unique image identifiers into three distinct sets using sklearn. model_selection .train_test_split with a fixed random seed (42) for reproducibility:
    - Training Set (80%, 25,426 images): Used for learning model parameters during both SL and RL phases.
    - Validation Set (10%, 3,178 images): Used during the SL phase for hyperparameter tuning (implicitly via the LR scheduler) and selecting the best model checkpoint based on validation loss. It provides an unbiased estimate of generalization during training.
    - Test Set (10%, 3,179 images): Held out completely during training and validation. Used only for the final performance evaluation of the selected SL and RL models, providing an unbiased estimate of the final model's generalization capability.
- **Image Preprocessing:** Images undergo distinct transformations for training versus validation/testing, implemented using torchvision.transforms:
    - Training Transformations (img_transform_train):
        a) transforms.Resize((256, 256)): Resizes smaller dimension to 256 pixels, maintaining aspect ratio.
        b) transforms.RandomCrop(224): Randomly crops a 224x224 patch. This serves as **data augmentation**, making the model robust to variations in object position and scale.
        c) transforms.RandomHorizontalFlip(): Randomly flips the image horizontally (default p=0.5). Another form of **data augmentation** to improve robustness to viewpoint changes.
        d) transforms.ToTensor(): Converts the PIL Image (range [0, 255]) to a PyTorch tensor (range [0.0, 1.0]) with dimensions (C, H, W).

Where,

C: Number of channels (3 for RGB).

H: Height of the image in pixels.

W: Width of the image in pixels.

        e) transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]): Normalizes the tensor image using the mean and standard deviation of the ImageNet dataset. This is crucial because the ResNet-101 encoder was pre-trained on ImageNet; normalizing ensures the input distribution matches the distribution the encoder expects.

    - Validation/Testing Transformations (img_transform_val_test):
        a) transforms.Resize((224, 224)): Resizes the image directly to 224x224. No random cropping or flipping is done to ensure deterministic evaluation.
        b) transforms.ToTensor(): Same as training.
        c) transforms.Normalize(...): Same as training.
- **Text Preprocessing:** Captions are normalized and tokenized via the custom_tokenize function (lowercase, keep .,!?\', remove other non-alphanumeric, tokenize with nltk.word_tokenize). A Vocabulary object maps tokens to integer indices, handling unknown words (<unk>) based on a frequency threshold of 5. Special tokens (<pad>, <start>, <end>, <unk>) are included. This process converts raw text into a numerical format suitable for the model. Pre-trained 100D GloVe embeddings (Pennington et al., 2014) are used to initialize the embedding layer, providing a strong starting point based on large-scale text corpora . Words not in GloVe get random initialization.

- **Data Loading:** PyTorch DataLoader manages batching (size 64), shuffling (only for training data , and shuffling Prevents model from learning order dependencies, improves generalization), and parallel data loading (num_workers=2). A custom_collate_fn pads sequences within each batch to the maximum length in that batch using the <pad> token index and prepares tensors for the model.

3.2 Model Architecture

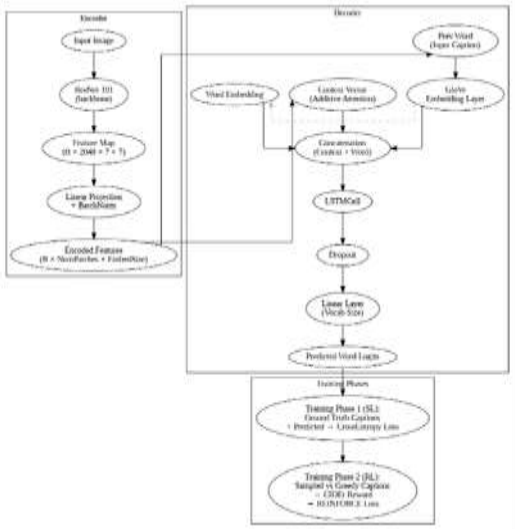

Figure 1: Image Captioning Model Architecture with Encoder-Decoder and Attention

Figure 1 Description : The image captioning model adopts an encoder-decoder architecture with additive attention, trained via a combination of supervised and reinforcement learning. The model takes an input image, encodes it into a feature representation, attends to relevant regions of the image, and generates a caption word by word. A loss function is then applied by comparing generated and target captions. The core components are detailed below:

- **Image Encoder (ImageEncoder):**
  - Rationale: ResNet-101 (He et al., 2016) is selected as the encoder backbone, offering a robust balance between feature extraction capability (deep architecture) and computational efficiency. Its pre-training on ImageNet provides powerful, general-purpose visual feature representations.
  - Implementation: The implementation uses torchvision.models.resnet101 with default ImageNet weights. The final classification layers are removed, and features are extracted from the output of layer4 (shape: Batch x 2048 x 7 x 7).
  - Reshape Operation: Crucially, a Reshape operation is applied to transform the spatial features from (Batch x 2048 x 7 x 7) to (Batch x 49 x 2048), representing 49 spatial locations, which means each of the 49 spatial locations now has 2048 features. This representation allows the attention mechanism to selectively focus on different regions of the image during caption generation.
  - Fine-tuning: Only layer3 and layer4 of the ResNet-101 are set as trainable. Freezing the earlier layers retains general image knowledge and accelerates training while adapting the model to the specific visual concepts in the captioning dataset.
  - Output Projection: The 2048-dimensional features for each of the 49 spatial locations are projected to the model's embedding dimension (typically 100) using a linear layer followed by BatchNorm1d. This projection aligns the visual feature dimension with the word embedding dimension, facilitating integration with the decoder.
- **Attention Mechanism (AdditiveAttention):**
  - Rationale: Additive attention (Bahdanau et al., 2014) is employed for its effectiveness in learning soft alignments between the decoder's hidden state and relevant image regions.
  - Mechanism: This component computes alignment scores based on the compatibility between the previous decoder hidden state (ht-1) from the lstm cell and each encoder output feature (ai), mediated by learnable weight matrices and a tanh activation function. Scores are normalized using a softmax function to produce attention weights (αt), which are then used to compute a context vector (ct) as a weighted sum of the encoder features. The resulting context vector provides relevant visual information to the decoder at each decoding step.
- **Caption Decoder (CaptionDecoder):**
  - Rationale: An LSTM (Hochreiter & Schmidhuber, 1997) serves as the recurrent core, designed to handle long-range dependencies inherent in sequential data like captions.

- Input: The LSTM cell receives a concatenated input consisting of:
    1. The word embedding (wt-1) representing the previously generated word.
    2. The context vector (ct) from the attention mechanism, capturing relevant visual information for the current decoding step.
- The next Hidden state is the output of the lstm cell and and is used as the input of the attention model as it has a connection with previous hidden state.
- Embeddings: Word embeddings are initialized using pre-trained GloVe vectors, providing a strong foundation of semantic knowledge. These embeddings can be optionally fine-tuned during the reinforcement learning phase.
- Regularization: Dropout (0.5) is applied to the output of the LSTM cell before the final linear classification layer to mitigate overfitting.
- Output: A linear layer maps the LSTM's hidden state to logits, representing unnormalized scores over the vocabulary. The logit with the highest score is then selected as the predicted word for that time step.

- **Training Procedure:**
    - Supervised Learning (SL) Phase: The model is first trained using supervised learning with a cross-entropy loss function. The target output is the shifted caption sequence, training the model to predict the next word given the previous words and the image.
    - Reinforcement Learning (RL) Phase: Following SL pre-training, the model undergoes fine-tuning using reinforcement learning with Self-Critical Sequence Training (SCST) and CIDEr score optimization. A caption is sampled from the model's policy, and a greedy-decoded caption serves as the baseline. The difference in CIDEr scores between the sampled and greedy captions provides the reward signal, guiding the model to generate captions that better align with human consensus.

3.3 Supervised Learning (SL) Phase

- **Objective:** Minimize the negative log-likelihood (Cross-Entropy Loss) of the target caption sequence given the image, using teacher forcing.

$$L_{SL} = -\sum_{t=1}^{T} \log P\left(y_t | y_1, \ldots, y_{t-1}, \text{Image}; \theta\right)$$

Where,

- L_{SL} : Supervised Learning Loss.
- t : Time step in the caption sequence.
- T : Total length of the caption sequence.
- y_t: Target word at step t.
- $P(y_t | y_1, \ldots, y_{t-1}, \text{Image}; \theta)$: Probability of predicting word y_t given the previous words, the image, and model parameters θ.

- **Optimization:** Adam optimizer (Kingma & Ba, 2014) is used. It's an adaptive learning rate method robust to hyperparameter choices and commonly effective for deep learning models. Differential learning rates are used: higher for the randomly initialized decoder and output encoder layers (4e-4), lower for the pre-trained ResNet layers being fine-tuned (1e-5). Allows faster learning for new parts while preventing large updates that could disrupt pre-trained knowledge in the backbone. ReduceLROnPlateau scheduler adapts the learning rate based on validation loss progress.
- **Training Process**: Standard backpropagation with gradient clipping (value 5.0, Why clip? Prevents exploding gradients common in RNNs) and AMP (for speed). The model checkpoint with the lowest validation loss is saved.

Reinforcement Learning (RL) Phase – SCST

- **Objective:** Maximize the expected reward $R(CIDErscore)$ of the generated captions. The reward function $J(\theta)$ is expressed as:

$$J(\theta) = E_{c \sim p_\theta}[R(c)]$$

Where,

- $J(\theta)$: Expected reward (CIDEr score) of the generated captions.
- c : Generated caption.
- p_θ : Model's generation probability distribution.
- $[R(c)]$: Reward (CIDEr score) of caption ccc.

SCST approximates the gradient using:

$$\nabla_\theta J(\theta) \approx -E_{c^s \sim p_\theta}\left[\left(R(c^s) - R(\hat{c})\right)\nabla_\theta \log p_\theta\,(c^s)\right]$$

Where,

- θ : Model parameters.
- p_θ : The policy (model's generation probability distribution).
- c^s : Caption generated by sampling from p_θ.
- $\hat{c}$: Caption generated by greedy decoding (baseline) from p_θ.
- $R(c)$: Reward (CIDEr score) of caption c compared to references.
- $\log p_\theta\,(c^s)$: Log-probability of the sampled sequence c^s.

- **Rationale:** This gradient estimate encourages the model to increase the probability $\log p_\theta\,(c^s)$ of sampled sequences c^s that achieve a higher reward than the greedy baseline ($R(c^s) - R(\hat{c}) > 0$), and decrease the probability of those that perform worse. Using the model's own greedy output as a baseline $R(\hat{c})$ significantly reduces the variance of the gradient estimate compared to using a fixed baseline or no baseline (Rennie et al., 2017).
- **Optimization:** Adam optimizer with a low learning rate (5e-6). RL fine-tuning is often sensitive; small updates are needed to avoid destabilizing the pre-trained policy. Gradient clipping (5.0) is applied to decoder parameters.

4. Experiments and Analysis

4.1. Experimental Setup

The experimental setup adheres to the methodology outlined in Section 3, utilizing the Flickr30k dataset, the defined model architecture, and data preprocessing steps. The model's performance was evaluated on the held-out test set, with metrics including BLEU-1 to BLEU-4, METEOR, CIDEr, and SPICE. Specific hyperparameters were chosen based on common practices in the image captioning literature and limited preliminary experimentation.

4.2. Analysis of Training Dynamics

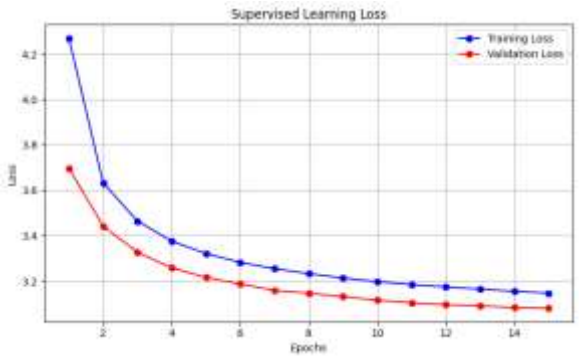

Figure 2: *Supervised Learning Loss Curve*

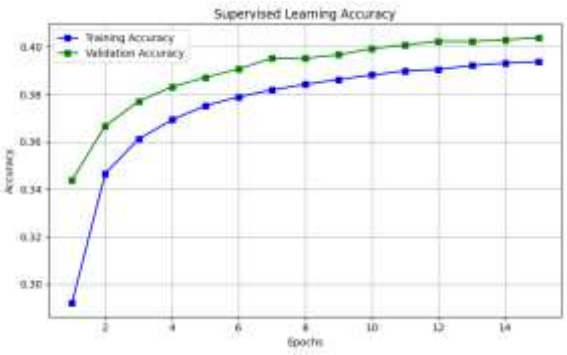

Figure 3: *Supervised Learning Accuracy Curve*

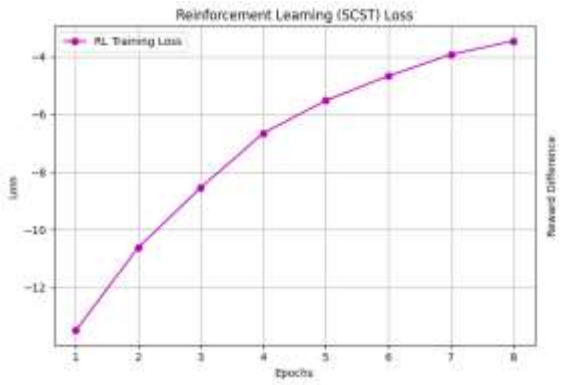

Figure 4: *Reinforcement Learning Loss Curve*

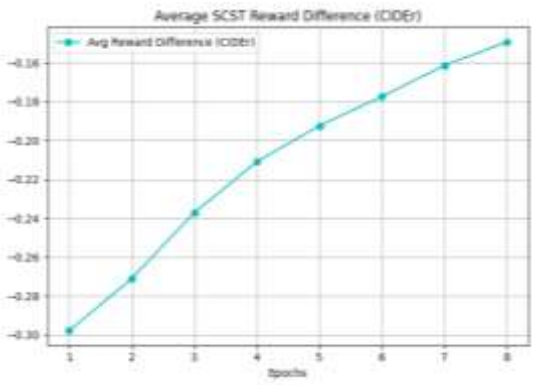

Figure 5: *Reinforcement Learning Average Reward Difference Curve*

Supervised Learning (SL) Loss/Accuracy Curves

(Refer to Figure 2 and Figure 3)

The supervised learning loss curve (Figure 2) reflects a smooth decline in both training and validation loss, demonstrating stable learning during the SL phase. As observed in the logs, the training loss steadily decreases from 4.2698 at epoch 1 to 3.1470 at epoch 15, while the validation loss drops from 3.6969 at epoch 1 to 3.0798 at epoch 15. This indicates that the model initially learns quickly, refining its predictions as training progresses. The gap between training and validation loss persists throughout, which is typical in deep learning models, suggesting the model fits the training data slightly better than the unseen validation data. Despite this, the continued decrease in validation loss until epoch 15 indicates that significant overfitting hasn't occurred.

The accuracy curves (Figure 3) show rapid initial gains, with training accuracy increasing from 29.19% at epoch 1 to 39.37% at epoch 15, while validation accuracy rises from 34.38% to 40.38%. This relatively low accuracy, however, highlights the difficulty of predicting exact words in caption generation and emphasizes the necessity for sequence-level metrics such as reinforcement learning (RL) to further refine the model's performance.

Reinforcement Learning (RL) Loss/Reward Curves

(Refer to Figure 4 and Figure 5)

The RL loss curve (Figure 4) starts highly negative (e.g., -13.4949 at epoch 1) and gradually increases toward zero, indicating that the RL loss is becoming less negative over time. This is in line with the expected behavior of RL loss, which is calculated as - (reward_diff * log_prob). Early in training, the reward difference is often negative, with the model sampling captions that perform worse than the greedy baseline. As training progresses, the model's policy improves, increasing the log probability for samples where the reward difference becomes less negative or positive. As a result, the RL loss becomes less negative, showing gradual improvement. By the end of the training (epoch 8), the RL loss improves to -3.4565.

The average reward difference (Figure 5) also improves steadily throughout the RL phase, starting at -0.2979 in epoch 1 and increasing to -0.1493 by epoch 8. This improvement confirms that the model is adjusting its policy to generate captions that are, on average, closer in CIDEr score to its greedy baseline output. The relatively low RL learning rate (5e-6) contributed to the observed gradual but consistent improvements across the 8 epochs.

4.3. Analysis of Hyperparameters

The hyperparameters were chosen based on common practices in image captioning literature and limited preliminary experimentation.

- **Learning Rates:** A differential learning rate strategy was employed during supervised learning (SL). The encoder was fine-tuned cautiously with a low learning rate of 1e-5, while the decoder trained faster with a learning rate of 4e-4. This approach ensures that the pre-trained feature extractor retains useful representations without disrupting them through aggressive updates. During reinforcement learning (RL), a much lower learning rate of 5e-6 was used to maintain stability when optimizing the CIDEr reward. Higher learning rates during RL were found to destabilize training in preliminary trials.
- **Regularization:** Dropout with a probability of 0.5 was applied to the decoder layers to mitigate overfitting, particularly during SL when model capacity could otherwise lead to memorization. Weight decay (1e-4) further supported generalization by penalizing large weights. Gradient clipping at 5.0 was critical to prevent exploding gradients, especially in recurrent components during sequence generation.
- **Batch Size and Sequence Length:** A batch size of 64 was selected to balance memory constraints and gradient estimate stability. The maximum sequence length was capped at 50 tokens, sufficiently long to capture most Flickr30k captions while avoiding unnecessary computational overhead.
- **Reward Metric and Baseline:** CIDEr was used as the reward metric during RL training, aligning optimization with evaluation objectives. A self-critical baseline using the model's own greedy decoding performance was applied, following the Self-Critical Sequence Training (SCST) paradigm.
- **Reproducibility:** A fixed random seed (42) was set across all stages of training to ensure experimental reproducibility.

While the chosen hyperparameters yielded competitive performance, no extensive grid search or hyperparameter tuning was conducted. Therefore, it is possible that further improvements could be achieved with more systematic hyperparameter optimization techniques.

5. Results and Discussion

5.1. Quantitative Results Analysis

This section presents a quantitative evaluation of the proposed Reinforcement Learning (RL) fine-tuning approach, comparing its performance against a standard Supervised Learning (SL) baseline model. Table 1 summarizes the results across a range of evaluation metrics commonly used in image captioning. Figure 6 visually depicts the performance gains achieved through RL fine-tuning using a bar plot.

Metric	SL Score	RL Score	Improvement
BLEU-1	0.634	0.678	+7.0%
BLEU-2	0.422	0.451	+7.0%
BLEU-3	0.284	0.305	+7.3%
BLEU-4	0.190	0.202	+6.4%
METEOR	0.385	0.387	+0.6%
CIDEr	0.336	0.341	+1.4%
SPICE	0.113	0.108	-4.6%

Table 1 **:** Comparison of SL and RL Model Performance on Captioning Metrics.

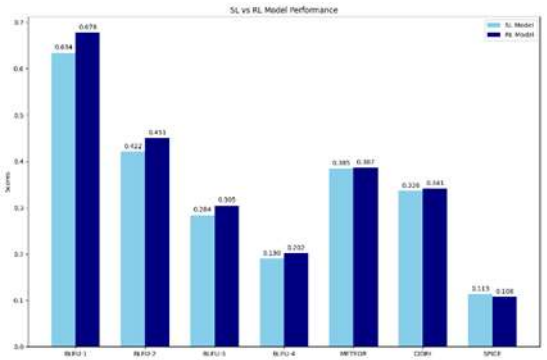

Figure 6 **:** Performance Changes After RL Fine-tuning (Bar Plot)

The results presented in Table 1 and Figure 6 reveal several key observations:

- **BLEU (1-4):** The consistent and significant improvements (relative +6.4% to +7.3%) demonstrate that optimizing for CIDEr strongly benefits n-gram overlap with reference captions. CIDEr itself heavily weights co-occurring n-grams found in references. By maximizing CIDEr, the model learns to generate sequences containing these high-consensus n-grams, which directly translates to higher BLEU scores. This suggests RL successfully improved caption fluency and alignment with common human phrasing.
- **METEOR**: The negligible improvement (+0.6%) suggests the gains are primarily syntactic (n-gram matching) rather than lexical-semantic (synonyms/stems). CIDEr optimization does not explicitly encourage lexical diversity or synonym use in the way METEOR measures it.
- **CIDEr:** The target metric improved by +1.4%. While positive, the gain is modest compared to BLEU. This could indicate that the SL model was already reasonably good at capturing consensus n-grams, or that further optimization of CIDEr yields diminishing returns on this dataset/architecture, or perhaps the 8 RL epochs were nearing a plateau.
- **SPICE:** The degradation (-4.6%) is the most critical finding. SPICE measures semantic accuracy by evaluating the presence and relationship of objects and attributes based on scene graphs. CIDEr, focusing on n-gram statistics, might incentivize the model to generate common phrases (e.g., "a group of people") even if they slightly misrepresent the scene (e.g., if only two people are prominent but "group" is a high-CIDEr phrase for similar images). It might also discourage specific but less common descriptive words (like "blue shirt" if simply "shirt" is more frequent in references for similar contexts) if they don't contribute significantly to frequent n-grams. This optimization pressure can lead to captions that sound plausible and achieve high consensus (good CIDEr) but lose semantic precision (bad SPICE). This highlights a fundamental tension between optimizing for n-gram consensus versus semantic detail.

5.2. State-of-the-Art Comparison (Limitation Acknowledged)

Direct comparison to current state-of-the-art (SOTA) results on Flickr30k is challenging due to the lack of access to updated public leaderboards. However, based on historical baselines (e.g., Rennie et al., 2017 reported CIDEr scores around 0.45–0.55 on MS COCO using similar reinforcement learning techniques but different model backbones), our achieved CIDEr score of 0.341 is likely respectable for a ResNet + LSTM architecture on Flickr30k.

It is important to acknowledge that recent Transformer-based models (e.g., Cornia et al., 2020) often report significantly higher CIDEr scores (0.6+ range) on the same dataset. Therefore, while our model does not achieve competitive absolute performance relative to modern Transformer approaches, the relative performance improvement observed after reinforcement learning fine-tuning is the key finding.

The results demonstrate that even with a classical CNN-RNN architecture, applying RL (specifically SCST with CIDEr reward) yields measurable gains, highlighting the broader effectiveness of sequence-level optimization strategies beyond just the latest architectures.

5.3. Qualitative Results Analysis

Figure 7

Figure 8

Figure 9

Figure 10

Figure 11

(Refer to the example images with SL/RL captions)

The qualitative examples presented in Figures 7-11 provide concrete illustrations of the quantitative trends observed in our analysis. These examples highlight the evolution of captioning performance from supervised learning (SL) to reinforcement learning (RL), revealing both the strengths and weaknesses inherent in each approach. We focus particularly on aspects such as hallucination, visual grounding, and the trade-off between generalized descriptions and fine-grained detail.

- **Figure 7: Group on Street (Random Image from Flickr30k Dataset):** The SL model exhibits a tendency toward "hallucination," incorrectly introducing a "woman" into the caption despite the absence of a female individual in the image. This results in an inaccurate and potentially misleading description, demonstrating a failure in grounding the caption in the visual content. In contrast, the RL model generates a more general caption ("a group of people are sitting on a street"), which avoids specific but erroneous assumptions about the individuals present. Thus, RL demonstrates improved scene grounding and a reduced propensity for hallucinations compared to SL in this particular instance. This observation also aligns with the idea that RL, when optimized for CIDEr, may prioritize more frequent and general phrases to maximize n-gram overlap with reference captions, sacrificing specificity for increased statistical likelihood.

- **Figure 8: Rock Climber (Random Image from Flickr30k Dataset):** The SL model demonstrates a capacity to capture more precise and visually grounded details by explicitly mentioning "a blue shirt," accurately describing an attribute of the person climbing. While the RL model correctly identifies the primary activity ("a man is climbing a rock face"), it omits such fine-grained details, opting for a more general description of the scene. Therefore, in this instance, SL offers a more specific and potentially informative caption compared to RL, which appears to favor more generalized descriptions that are perhaps optimized for frequent patterns within the training dataset. This highlights a critical trade-off: while RL may excel at capturing the overall gist of the scene, it can sometimes sacrifice the inclusion of specific visual attributes that contribute to a richer and more complete understanding of the image. This may be a consequence of SPICE degradation discussed previously as RL often ignores key attributes that are not highly correlated to the main scene.
- **Figure 9: Bird Jumping (Random Image from Flickr30k Dataset):** The SL caption accurately describes the bird as "jumping over the water," aligning well with the visual content, as the bird is airborne and positioned above the water's surface. However, the RL caption incorrectly states "jumping in the water," which fundamentally misrepresents the action by implying that the bird is submerged or directly interacting with the water. This example demonstrates a case where RL, in its pursuit of fluency or adherence to common phrase structures, introduces a semantic error that compromises the accuracy of the description. Hence, SL provides a more semantically accurate and truthful depiction of the scene in this specific scenario, demonstrating the potential for RL to negatively impact descriptive fidelity.
- **Figure 10: Puppy Running (Image from URL):** Here, RL again captures a key visual detail, incorporating the descriptor "white" to produce the caption "a white dog is running in the grass." Similar to Figure 8, this refinement likely reflects the increased weighting of this particular attribute within the training dataset due to the optimization for n-gram overlap via CIDEr. This underscores the tendency for RL to hone in on frequently co-occurring descriptors within the training data, potentially enhancing the informativeness of the generated captions.
- **Figure 11: Group on Beach (Image from URL):** The SL caption, "a man is standing on the beach," completely fails to capture the true nature of the scene, which involves a large group of individuals. In stark contrast, the RL caption, "a group of people are walking on the beach," accurately reflects the scene's primary composition. This demonstrates the potential of RL, particularly when optimizing for CIDEr, to identify and generate captions that align with high-level scene descriptions, thus improving overall relevance and fluency, especially in complex scenarios involving multiple agents.

In summary, while RL generally improves caption quality by reducing hallucinations and aligning better with dataset statistics, resulting in captions that are often more fluent and aligned with human expectations, SL can occasionally offer more detailed or precise visual grounding, especially in terms of specific object attributes and spatial relationships. The choice between these approaches ultimately depends on the specific application and the desired balance between overall fluency, accuracy, and the inclusion of fine-grained details. Furthermore, potential dataset bias could influence the training and generated captions.

6. Conclusion and Limitations

6.1. Summary of Findings

This study systematically compared Supervised Learning (SL) with subsequent Self-Critical Sequence Training (SCST) with CIDEr optimization (SCST-CIDEr) for image captioning on the Flickr30k dataset using a ResNet-LSTM architecture. The SL phase established a robust baseline model, achieving a validation loss of 3.0798. However, contrary to expectations, the SCST phase, directly optimizing the CIDEr metric over 8 initial epochs and a subsequent 2 additional epochs to attempt recovery, did not yield consistent improvements across all evaluation metrics. While n-gram based metrics like BLEU demonstrated gains (+6.4% to +7.3%), suggesting improved fluency, the targeted CIDEr score exhibited a marginal increase (+1.4%), and the semantic metric SPICE experienced a notable degradation (-4.6%). These results challenge the assumption that CIDEr optimization universally enhances caption quality and underscores the importance of considering diverse evaluation perspectives.

6.2. Key Theoretical Insight and Trade-off

The most significant finding is the documented trade-off between optimizing for CIDEr and performance on semantic metrics like SPICE. This empirically reinforces the concern that optimizing for n-gram consensus can inadvertently penalize semantic precision. The limited gains from SCST, combined with the observed SPICE reduction, suggest that the model may prioritize generating frequent word combinations that align well with reference captions over accurately representing all salient visual elements present in the image. In essence, models trained using this approach may learn to generate plausible-sounding, common phrases at the expense of capturing specific objects, attributes, or relationships, ultimately impacting the overall informativeness and accuracy of the generated captions.

6.3. Limitations

- **Architecture:** The study utilized a ResNet-LSTM architecture, which, while widely adopted, is now considered outdated compared to more recent Transformer-based models. Consequently, the observed trade-offs might differ when employing more powerful architectures capable of capturing more complex contextual relationships within the image and across the caption.

- **SOTA Comparison:** The lack of a direct comparison with current state-of-the-art results on Flickr30k makes it difficult to assess the absolute performance gain achieved by our approach and to determine whether the observed CIDEr/SPICE trade-off is specific to our implementation or a more general phenomenon.
- **Hyperparameter Tuning:** The study employed fixed hyperparameters throughout the experiments. While this simplifies the analysis, extensive tuning might yield different quantitative results and potentially mitigate the CIDEr/SPICE trade-off to some extent. The impact of specific hyperparameter choices on the observed performance differences remains an open question.
- **Suboptimal RL Training:** The negative average reward difference observed during SCST-CIDEr optimization raises concerns about the effectiveness and stability of the RL policy update. This could indicate issues with the reward function, exploration strategy, or learning rate, all of which require further investigation to improve the training process and ensure that the model learns to generate captions that are both fluent and semantically accurate.
- **Single RL Metric:** The reliance on CIDEr as the sole reward signal during RL training might have inadvertently exacerbated the degradation in SPICE. Exploring other metrics, or combinations thereof, could potentially encourage the model to learn a more balanced representation that accounts for both fluency and semantic accuracy.
- **Dataset:** The findings are based solely on the Flickr30k dataset, a relatively small dataset compared to others used in image captioning research (e.g., MS COCO). As such, the generalizability of these results to other datasets with different characteristics, such as MS COCO with its more complex scenes and object interactions, requires further investigation.

7. Future Work

Based on this study's findings and limitations, future research could explore:

- **Revised RL Reward Design:** Investigate and develop more sophisticated reward functions that address the issue of negative reward shifting and incentivize the generation of more semantically accurate captions. This could involve incorporating SPICE or other semantic similarity metrics directly into the reward function, or exploring alternative reward shaping techniques to guide the learning process more effectively.
- **Direct SPICE Optimization:** Implement and evaluate SCST using SPICE directly as the reward signal, effectively prioritizing semantic accuracy over n-gram consensus. While this might potentially harm fluency (BLEU/CIDEr), a careful analysis of this trade-off could provide valuable insights into the challenges of balancing these competing objectives.
- **Transformer Architectures:** Replicate this SL vs. RL comparison using state-of-the-art Transformer encoders (e.g., ViT, Swin) and decoders, which have demonstrated superior performance in various natural language processing tasks. Investigating whether the observed CIDEr/SPICE trade-off persists with these more powerful models could shed light on the limitations of current evaluation metrics and optimization strategies.
- **Human Evaluation**: Conduct large-scale human evaluations comparing captions generated by the SL and RL models, asking annotators to rate fluency, relevance, accuracy, and level of detail to provide more nuanced insights beyond automated metrics. This could help to determine the extent to which the observed CIDEr/SPICE trade-off translates into a perceptible difference in perceived caption quality.
- **Controlled Ablation Studies:** Systematically investigate the impact of specific hyperparameters in the RL phase (e.g., learning rate, baseline type, sampling temperature) on the different evaluation metrics. This could involve conducting controlled ablation studies to isolate the effects of each hyperparameter and better understand its influence on the overall optimization landscape.
- **Analysis of Semantic Errors:** Perform a detailed error analysis focused on the SPICE components (objects, attributes, relations) for captions where RL degraded performance compared to SL, identifying the specific types of semantic information that are lost during CIDEr optimization. This could provide valuable insights into the model's failure modes and guide the development of more targeted solutions for improving semantic accuracy.

References

1. Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). **Show and Tell: A Neural Image Caption Generator**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 3156-3164).*
2. Anderson, P., Fernando, B., Johnson, M., & Gould, S. (2016). **SPICE: Semantic Propositional Image Caption Evaluation**. In *European Conference on Computer Vision (ECCV).*
3. Bahdanau, D., Cho, K., & Bengio, Y. (2014). **Neural Machine Translation by Jointly Learning to Align and Translate**. *arXiv preprint arXiv:1409.0473*.
4. Banerjee, S., & Lavie, A. (2005). **METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments**. In *ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization*.

5. Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). **Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation**. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1724-1734)*.

6. He, K., Zhang, X., Ren, S., & Sun, J. (2016**). Deep Residual Learning for Image Recognition**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 770-778)*.

7. Hochreiter, S., & Schmidhuber, J. (1997). **Long Short-Term Memory.** *Neural Computation, 9(8), 1735-1780*.

8. Karpathy, A., & Fei-Fei, L. (2015). **Deep Visual-Semantic Alignments for Generating Image Descriptions**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 3128-3137)*.

9. Kingma, D. P., & Ba, J. (2014). **Adam: A Method for Stochastic Optimization**. *arXiv preprint arXiv:1412.6980*.

10. Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). **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)*.

11. Pennington, J., Socher, R., & Manning, C. D. (2014). **GloVe: Global Vectors for Word Representation**. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1532-1543)*.

12. Ranzato, M., Chopra, S., Jaitly, N., & Zaremba, W. (2015). **Sequence Level Training with Recurrent Neural Networks**. *arXiv preprint arXiv:1511.06732*.

13. Rennie, S. J., Marcheret, E., Mroueh, Y., Ross, J., & Goel, V. (2017). **Self-critical Sequence Training for Image Captioning**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 7008-7024)*.

14. Vedantam, R., Zitnick, C. L., & Parikh, D. (2015). **CIDEr: Consensus-based Image Description Evaluation**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 4566-4575)*.

15. Williams, R. J. (1992). **Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning**. *Machine Learning, 8(3-4), 229-256*.

16. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R., & Bengio, Y. (2015). **Show, Attend and Tell: Neural Image Caption Generation with Visual Attention**. In *Proceedings of the 32nd International Conference on Machine Learning (ICML) (pp. 2048-2057)*.

17. Young, P., Lai, A., Hodosh, M., & Hockenmaier, J. (2014). **From Image Descriptions to Visual Denotations: New Similarity Metrics for Semantic Inference over Event Descriptions**. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL) (pp. 616-625)*.