International Journal of Computer Engineering in Research Trends
ISSN: 2349-7084/ https://doi.org/10.22362/ijcert/2024/v11/i1/v11i101
Volume 11, Issue 1, January 2024

Research Paper

Stylistic Image Captioning with Adversarial Learning: A Novel Approach

Sushma Jaiswal[1*], Harikumar Pallthadka[2], Rajesh P. Chinchewadi[3], Tarun Jaiswal[4]

[1]*Guru Ghasidas Central University, Bilaspur (C.G.) and Post-Doctoral Research Fellow, Manipur International University, Imphal, Manipur, jaiswal1302@gmail.com, https://orcid.org/0000-0002-6253-7327*
[2]*Manipur International University, Imphal, Manipur, vc@miu.edu.in, https://orcid.org/0000-0002-0705-9035.*
[3]*Manipur International University, Imphal, Manipur, rajesh.cto@miu.edu.in.*
[4] *National Institute of Technology, Raipur, tjaiswal_1207@yahoo.com, https://orcid.org/0000-0003-3963-4548*

**Sushma Jaiswal: jaiswal1302@gmail.com*

Received: 16/10/2023, Revised: 07 /11/2023, Accepted: 19/12/2023 Published: 03/01/2024

Abstract: - *this paper present "Attention-GAN," a new image captioning model that synergistically integrates attention mechanisms and Generative Adversarial Networks (GANs) to revolutionize image caption production. Attention-GAN has two main parts. First, an attention-based caption generator that strongly correlates visual regions with caption segments. This attention mechanism helps the model highlight important visual aspects and provide meaningful, contextual captions. Second, an adversarial training process adds aesthetic diversity to the caption generator. Adversarial training produces more subtle and different stylized descriptions, resulting in captions that express the image's content and aesthetic and stylistic variances. More interesting and varied image captions result from our dual-component technique, which blends attention-based modelling precision with adversarial learning inventiveness. Attention-GAN generates contextually relevant and artistically appealing captions in extensive benchmark dataset trials. Quantitative and qualitative analyses show that the model is capable of creating captions that match image content and have varied stylistic subtleties. Attention-GAN is a promising image captioning technology that can bridge the gap between factual description and creative expression for a variety of computer vision and natural language processing applications.*

Keywords- *CNN, LSTM, Image Caption, BLSTM, Attention-GAN*

---

1. Introduction

Image captioning is a fascinating interdisciplinary challenge at the interface of computer vision and natural language processing that automatically generates appropriate image captions. Advanced techniques have been used to create captions that describe visual content and have artistic and stylistic variances. We offer "Attention-GAN," a revolutionary image captioning model that uses attention processes and Generative Adversarial Networks (GANs) to revolutionize picture caption production. Attention-GAN's dual-component architecture enhances caption production. The first component uses an attention-based caption generator to help the model match image regions to caption segments. The algorithm generates captions with fine-grained information and contextual relevance by allocating different amounts of attention to different sections of the image, mimicking human perception. Second, a well-designed adversarial training mechanism complements the attention-based generator. Adversarial training, a GAN trademark, adds stylistic diversity to captions. This allows the caption generator to provide descriptions with a wide range of creative styles and artistic emotions. The antagonistic interaction improves the model's inventiveness and style, making captions more interesting and appealing. We examine Attention-GAN's architecture, operation, and experimental validation in this study. We demonstrate how this novel approach improves factual correctness and relevancy of generated captions while adding stylistic variety. Attention-GAN takes image captioning to a new level by combining attention mechanisms and adversarial learning to create appealing and informative textual narratives from visual data.

The image captioning model "Attention-GAN" has these primary contributions:

1. The model can dynamically focus on different image regions with an attention mechanism, producing more contextually accurate captions. When describing the image, this helps understand its features.
2. The algorithm generates captions with varied stylistic elements via adversarial training. This

ensures correct captions with a variety of aesthetic and stylistic variances, enriching the descriptions.

3. Attention-GAN synergizes attention-based caption generation and adversarial training. This fusion creates contextually meaningful and stylistically different captions, making captioning more versatile and interesting.
4. Adversarial training and attention improve caption realism and expressiveness. Thus, the image captioning model becomes more interesting and informative as descriptions better transmit visual details.
5. By integrating attention, the model can better correlate visual attributes with textual portions. This multimodal comprehension boosts captions by correlating descriptions with image highlights.

2. Literature Survey

The authors [1] used policy gradient approaches and function approximation to advance reinforcement learning. The research uses function approximation to improve reinforcement learning efficiency and efficacy for policy optimization for complicated decision-making tasks. The work advances reinforcement learning by revealing how to optimize policies with function approximation, a fundamental component of real-world applications. The authors [2] proposed a method to fix sequence generation's training-inference mismatch. Sequence prediction models improve by gradually switching from ground truth tokens to predicted tokens during training, making the model more robust in generating sequences. The unique framework of GANs incorporates two neural networks, a generator and a discriminator, competing. The discriminator distinguishes between real and created data, while the generator creates data that looks like real data. Adversarial training generates high-quality data, which advanced generative modelling and became a cornerstone of machine learning and artificial intelligence [3].

Osindero and Mirza developed Conditional Generative Adversarial Nets [4], Based on GANs, cGANs provide additional information to the generator and discriminator for conditional data production. This allows data collection under certain conditions, making data generation more controlled and focused. cGANs have been used in image-to-image translation and data synthesis, expanding traditional GANs' conditional data generating capabilities.

Sequence generation using generative adversarial networks (GANs) and policy gradient approaches is presented in SeqGAN [5]. SeqGAN uses reinforcement learning to solve discrete data generation problems like natural language. It uses a generator network to generate sequences and a discriminator network to reward reinforcement learning. The model generates cohesive and realistic sequences thanks to this novel combination, boosting language modelling and text synthesis.

Silver et al. [6] described AlphaGo, an AI system that mastered Go, in their landmark work. Deep neural networks and Monte Carlo Tree Search are used. Deep neural networks assessed board situations and guided the search, while MCTS assisted with decision-making and move selection. Artificial intelligence's AlphaGo system proved that powerful machine learning and computational search methodologies might win complex strategic games.

Hochreiter and Schmidhuber developed the highly important Long Short-Term Memory (LSTM) network for recurrent neural networks [7]. LSTMs are specialised RNNs that simulate sequential data long-term dependencies by solving the vanishing gradient problem. Key innovation: memory cells and gates manage information flow to capture critical information over long sequences. LSTMs are vital in natural language processing, time series analysis, and other fields that need sequential data processing.

Conditional Generative Adversarial Networks improve image captioning, according to Chen et al. A unique technique to picture caption refinement uses cGANs for more targeted and realistic generation [8]. This method improves image captioning by conditioning picture production on more information, resulting in better quality and coherence captions that match image content and context.

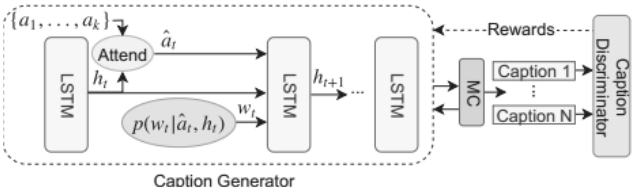

Figure. 1: The architecture of the Attend-GAN model.

3. The proposed Attention-GAN Model

In the attention GAN proposed model, attention is applied to fine grained image regions denoted by $a = \{a_i, \ldots, a_k\}$, where $a_i \in \mathfrak{D}$. Each region a_i belongs to space $\mathfrak{D}$ with D dimensions, and the total number of regions is K.

The objective is to generate an image caption $x = \{x_1, \ldots, x_T\}$, where $x_i \in \mathfrak{Y}$, from a vocabulary $\mathfrak{Y}$ of size N. The length of the generated caption is denoted by T.

In this context, the attention mechanism dynamically adjusts its focus across diverse fine-grained image regions during different time steps. This dynamic attention allocation allows the model to emphasize specific regions of the image while generating each word in the caption. Consequently, the attention-driven caption generation process yields a descriptive and contextually relevant narrative that accurately portrays the contents of the image.

The caption generator utilizing reinforcement learning and Monte Carlo (MC) search with LSTM involves a model represented by G_θ, parameterized by θ. This LSTM-based generator takes an image I and sequentially generates a caption x. The objective is to maximize expected reward $E[R]$, where R is a reward function based on caption quality. MC search is employed to sample sequences by iteratively generating words using the LSTM. The LSTM's ability to model sequences, combined with reinforcement learning and MC search, leads to the

generation of diverse, contextually relevant, and high-quality image captions.

Incorporating Long Short-Term Memory (LSTM) units for sequence generation in the caption generator with reinforcement learning and Monte Carlo (MC) search involves several components.

3.1 LSTM based Caption Generator

The caption generator is based on an LSTM neural network, which generates captions sequentially. Let G_θ denote the LSTM-based caption generator parameterized by θ. The generator takes an image I as input and generates a sequence of words $x = \{x_1, \ldots, x_T\}$, as output, where x_i is the word at time step t. The LSTM generates the next word based on the previous words and the image features.

The LSTM generates the next word based on the previous words and the image features.

$$x_t \sim G_\theta(x_{t-1}, h_{t-1}, c_{t-1}, I) \quad (1)$$

Here, h_{t-1} and c_{t-1} are the LSTM hidden state and cell state from the previous time step.

3.2 Reinforcement Learning Objective

In reinforcement learning, the objective is to maximize the expected reward E[R] where R is a function of the generated caption.

$$J(\theta) = E[R(x)] = \sum_x P(x;\theta) \cdot R(x) \quad (2)$$

The goal is to maximize this expected reward by adjusting the LSTM's parameters θ.

3.3 Monte Carlo (MC) Search for Caption Sampling

During MC search, sequences of words are sampled using the LSTM-based generator. Starting with an initial word, subsequent words are generated based on the probability distribution output by the LSTM at each time step.

$$x_t \sim P(x_t | x_{t-1}, h_{t-1}, c_{t-1}, I) \quad (3)$$

The MC search involves sampling multiple sequences and evaluating their quality using a reward function R(x). Sequences are generated by iteratively sampling from the LSTM until an end token is generated or a maximum length is reached.

$$x^* = \arg\max_x \sum_{t=1}^{T} R(x_t) \quad (4)$$

Subject to $x_t \sim P(x_t | x_{t-1}, h_{t-1}, c_{t-1}, I)$

Where x_i are sampled sequences from the LSTM, and T is the length of the sequence. The reward function R(x) is based on caption quality metrics.

By combining the LSTM-based caption generator with reinforcement learning and Monte Carlo search, we can efficiently generate diverse and contextually relevant captions for images. The LSTM's sequential generation ability, reinforcement learning objective, and MC search's sampling mechanism collectively contribute to generating improved and coherent image captions.

3.4 Image Caption discriminator based on Wasserstein GAN (WGAN)

The discriminator in a Wasserstein Generative Adversarial Network (WGAN) is a key component designed to evaluate the quality of the generated data. Unlike traditional GANs, WGANs utilize a Wasserstein distance metric, also known as Earth Mover's distance, as the objective function. Here's a brief description of the discriminator in a WGAN:

3.4.1 Wasserstein Distance Metric

The WGAN utilizes the Wasserstein distance as the metric to measure the difference between the distribution of real data and the distribution of generated data. This metric provides a more meaningful and stable measure compared to traditional GANs that use binary cross-entropy or other measures.

There are some experiments trying to optimize the similar architecture based on Wasserstein GAN (WGAN) loss, such as the paper dualGAN [9]. Theoretically, with the use of EM distance, WGAN will stabilize the training process of GAN.

The EM distance is a measure of distance between two distributions:

$$W(P_r, P_g) = \inf_{x,y\sim\gamma} \mathbb{E}_{x,y}[\| x - y \|] \quad (5)$$

After some transformations, our goal becomes:

$$\max_{w\in W} \mathbb{E}_y[f_w(y)] - \mathbb{E}_z[f_w(g_\theta(z))] \quad (6)$$

Where $\{f_w(x)\}_{w\in W}$ are all K-Lipchitz for some K.

Objective of conditional WGAN

$$\min_{w\in W}\{-\mathbb{E}_y[f_w(x,y)] + \mathbb{E}_z[f_w(x, g_\theta(x,z))]\} \quad (7)$$

According to the original GAN, in order to apply WGAN, we must:

- Adjust the loss function so that it takes the EM distance into account.
- Take out the discriminator's sigmoid function.
- Apply the Lipchitz constraint by using a weight clipping method.
- Switch to RMSProp as the optimization function.

The weight clipping approach has been shown to have some surprising behavior by [10]. Gradient Penalty is a superior approach for enforcing the Lipchitz constraint.

$$L = \mathbb{E}_{\tilde{x}\sim P_g}[D(\tilde{x})] - \mathbb{E}_{x\sim P_P}[D(x)] + \lambda\mathbb{E}_{\hat{x}\sim P_{\hat{\omega}}}\mathbb{E}\left[\left(\|\Delta_{\hat{x}}D(\hat{x})\|_2\right)^2\right] \quad (9)$$

Where $P_{\hat{x}}$ sampling uniformly along straight lines between pairs of points sampled from the data distribution P_r and the generator distribution P_g.

Wasserstein GAN with gradient penalty term

The fact that the WGAN substantially restricts the critic's capacity to learn about weight reduction is one of its main complaints. In the original WGAN paper, the authors say quite clearly that "Weight clipping is a bad way to enforce a Lipschitz constraint." The performance of a WGAN as a critic is crucial because without accurate gradients, the Generator cannot gradually modify its weights to generate better samples. Therefore, one of the most recent WGAN adaptations is the Wasserstein GAN Gradient Penalty (WGAN-GP) architecture. The definition and compilation procedure of the Wasserstein GAN Generator are identical to those of the WGAN-GP Generator. The punishment gradient sentence can be incorporated by simply changing the critic.

Gradient penalty

It is 1 -Lipschitz if gradients of the norm at most 1 occur everywhere for a differential function f. Therefore, for f, the gradient norm of interpolated points between generated and real data should equal 1. Rather than using weight clipping to punish the model, the gradient penalty is applied if the gradient norm diverges from its intended norm value of 1.

$$L = \underbrace{\mathbb{E}_{\hat{x}\sim\mathbb{P}_g}[D(\bar{x})] - \mathbb{E}_{x\sim\mathbb{P}_r}[D(x)]}_{\text{Original critic loss}} + \underbrace{\lambda\,\mathbb{E}_{\hat{x}\sim\mathbb{P}_{\hat{x}}}\left[\left(\|\nabla_{\hat{x}}D(\hat{x})\|_2 - 1\right)^2\right]}_{\text{Our gradient penalty}}$$

$\bar{x}$ is sampled from $\bar{x}$ and x is sampled t-uniformly [0,1] . Lambda (λ) is usually set to ten. Batch normalisation is avoided by the critic. Batch normalisation is used to create correlations between samples that are part of the same batch. By penalising the discriminator's gradient norm with regard to its inputs instead of the current weight clipping, it can improve classification accuracy.

Implementation of Wasserstein with gradient penalty

Algorithm 1 WGAN with gradient penalty.

We use default values of $\boldsymbol{\lambda = 10, n_{\text{critic}} = 5, \alpha = 0.0001, \beta_1 = 0, \beta_2 = 0.9}$.

Require: The gradient penalty coefficient $\boldsymbol{\lambda}$, the number of critic iterations per generator iteration $\boldsymbol{n_{\text{critic}}}$, the batch size $\boldsymbol{m}$, Adam hyperparameters $\boldsymbol{\alpha, \beta_1, \beta_2}$.

Require: initial critic parameters $\boldsymbol{w_0}$, initial generator parameters $\boldsymbol{\theta_0}$.

while $\boldsymbol{\theta}$ has not converged do

for $\boldsymbol{t = 1, \ldots, n_{\text{critic}}}$ do

for $\boldsymbol{i = 1, \ldots, m}$ do

Sample real data $\boldsymbol{x \sim \mathbb{P}_r}$, latent variable $\boldsymbol{z \sim p(z)}$, a random number $\boldsymbol{\epsilon \sim U[0,1]}$.

$$\tilde{\boldsymbol{x}} \leftarrow \boldsymbol{G_\theta(z)}$$

$$\hat{\boldsymbol{x}} \leftarrow \boldsymbol{\epsilon x + (1-\epsilon)\tilde{x}}$$

$$\boldsymbol{L^{(i)} \leftarrow D_w(\tilde{x}) - D_w(x) + \lambda\left(\|\nabla_{\hat{x}} D_w(\hat{x})\|_2 - 1\right)^2}$$

end for

$$\boldsymbol{w \leftarrow \text{Adam}\left(\nabla_w \frac{1}{m}\sum_{i=1}^{m} L^{(i)}, w, \alpha, \beta_1, \beta_2\right)}$$

end for

Sample a batch of latent variables $\boldsymbol{\{z^{(i)}\}_{i=1}^{m} \sim p(z)}$.

$$\boldsymbol{\theta \leftarrow \text{Adam}\left(\nabla_\theta \frac{1}{m}\sum_{i=1}^{m} - D_w(G_\theta(z)), \theta, \alpha, \beta_1, \beta_2\right)}$$

end while

Table 1. Implementation of WGAN-GP [11]

In a Wasserstein Generative Adversarial Network (WGAN) for image captioning, the discriminator aims to approximate the Wasserstein distance between the distribution of real captions and the distribution of generated captions. Let's define the mathematical components involved:

3.5 Discriminator Objective

The discriminator, denoted as D, is trained to estimate the Wasserstein distance W between the real caption distribution P_r and the generated caption distribution P_g using a Wasserstein loss function Wasserstein $L_{\text{Wasserstein}}$: $L_{\text{Wasserstein}}(D) = \sup_{\|f\|_L \le 1} E_{z\sim P_r}[f(x)] - E_{z\sim P_g}[f(G(z))]$ (10)

Here, f is a 1-Lipschitz continuous function (defined by the Lipschitz constant $\|f\|_L \le 1, G(Z)$ is the generated caption from noise z, x is a real caption, and z is a noise vector.

3.6 Training the Discriminator

The training process for the discriminator involves maximizing the difference between the scores assigned to real and generated samples. This difference is the approximation of the Wasserstein distance. The Wasserstein distance encourages the generator to produce data that is closer to the real data distribution, promoting stable training and better convergence.

The discriminator in a WGAN plays a crucial role in encouraging the generator to produce high-quality data that is closer to the real data distribution. The use of the Wasserstein distance metric transforms the traditional GAN training into a more stable and efficient process, addressing challenges such as mode collapse and vanishing gradients.

During training, the discriminator D aims to maximize the Wasserstein loss by adjusting its parameters θ_D:

$$\theta_D \leftarrow \theta_D + \alpha \cdot \nabla_{\theta_D} L_{\text{Wasserstein}}(D) \quad (11)$$

Where α is the learning rate.

The goal of this discriminator in the WGAN context is to approximate the Wasserstein distance between real and generated captions. By training the discriminator to maximize the Wasserstein loss, the generator is

encouraged to produce captions that are increasingly similar to those from the real caption distribution, leading to higher-quality and more accurate generated captions.

By implementing the discriminator as a CNN and training it to maximize the Wasserstein loss, the model is encouraged to generate captions that closely resemble real captions, enhancing the quality and relevance of the generated captions.

3.7 Algorithm on Attention GAN

In Algorithm1, we pre-train our caption generator for a set number of epochs. Next, we generate example captions using the best generator model. Real captions are chosen from reality. Our caption discriminator is pre-trained with generated and real captions for a set number of epochs. Caption generator and discriminator are pre-trained on facts. We start adversarial training on a positive or negative sentiment dataset. We train the caption generator and discriminator forg- and d-steps. We improve the caption generator and discriminator with this mechanism. The caption generator updates its parameters based on the rewards received from the discriminator.

IMAGE CAPTION GENERATOR:

INITIALIZATION:

Initialize LSTM-based caption generator G with attention mechanism and parameters θ.

REINFORCEMENT LEARNING TRAINING LOOP

FOR EACH

Sample a mini-batch of images I and corresponding ground-truth captions x_{gt}.

Generate captions for the images using the LSTM-based generator with attention: $x_{gen} \sim G_\theta(I)$.

Compute the reward $R(x_{gen})$ based on caption quality.

Update the parameters θ of the generator using policy gradients with Wasserstein loss:

$$\theta \leftarrow \theta + \alpha \nabla_\theta \left(E_{xgm}[D(x_{gen})] - R(x_{gen}) \right)$$

FOR END

MC SEARCH FOR CAPTION GENERATION:

FOR EACH IMAGE

Initialize an empty caption $x = [\,]$.

Repeat until the maximum caption length is reached or an end token is generated:

- Use the LSTM-based generator G_θ with attention to predict the next word given the current image and partial caption.
- Sample the next word using the predicted probability distribution.
- Update the partial caption with the sampled word.

Save the generated caption.

FOR END

IMAGE CAPTION DISCRIMINATOR:

INITIALIZATION:

Initialize CNN-based discriminator D with parameters ϕ.

WGAN TRAINING LOOP

For each training iteration:

- Sample a mini-batch of real captions x_{real} and generated captions $x_{gen.}$
- Compute the Wasserstein loss using the CNN-based discriminator:

$$L_{\text{Wasserstein}}(D) = \frac{1}{m} \sum_{i=1}^{m} D(x_{real}[i]) - D(x_{gen}[i])$$

- Update the parameters $\emptyset$ of the discriminator using the gradient of the Wasserstein loss:

$$\emptyset \leftarrow \emptyset - \beta \cdot \nabla_\emptyset L_{\text{Wasserstein}}(D) .$$

For End

ADVERSARIAL TRAINING:

Train the generator adversarially to improve caption quality:

- Generate captions for images using the generator.
- Compute the adversarial loss based on the CNN-based discriminator's evaluation of the generated captions.
- Update the generator parameters to minimize the adversarial loss.

LOSS FUNCTIONS:

GENERATOR LOSS

Incorporate adversarial loss to align the generated captions with the real caption distribution.

CNN-BASED DISCRIMINATOR LOSS

Utilize Wasserstein loss to guide the generator towards producing captions closer to the real caption distribution.

MODEL EVALUATION AND TERMINATION:

MODEL EVALUATION:

Periodically evaluate the model's performance on a validation set using appropriate evaluation metrics like BLEU score, CIDEr, etc.

TERMINATION

Stop training based on predefined criteria such as a certain number of epochs or achievement of satisfactory performance.

4. Experimental Setup

MSCOCO image-caption dataset for training models, T.-Y. Lin et al. [12] the Microsoft Common Objects in Context (COCO) dataset. The dataset is a significant contribution to computer vision, providing a large-scale collection of images with comprehensive annotations, including object segmentations, keypoints, and captions. With over 82,000 images and 413,000 captions in the training set, COCO has become a vital resource for training and evaluating models across various computer vision tasks, promoting advancements in object recognition, segmentation, and image captioning. The dataset has facilitated benchmarking and challenges, driving progress and innovation in the field.

SentiCap [13] is a system designed to generate image descriptions infused with sentiments. By incorporating emotions and sentiments into image captions, SentiCap aims to enhance the expressiveness and human-like nature of the generated descriptions. The paper explores the integration of sentiment analysis and natural language generation to create captions that reflect not only the visual content but also the emotional context of the images, ultimately contributing to more engaging and emotive image descriptions. SentiCap, collection contains favourable and negative attitudes. The positive section has

2,873 captions with 998 photos for training and 2019 captions with 673 images for testing. 2,468 captions with 997 photos are for training and 1,509 with 503 for testing in the negative area.

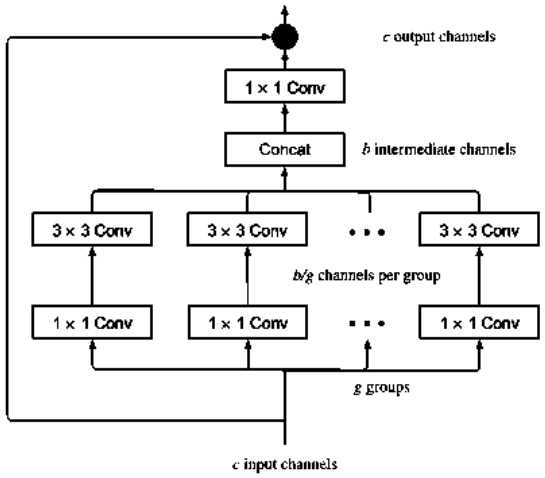

Figure. 2: The ResNeXt block

4.1 Evaluation Metrics

The proposed method assessed using BLEU [14], METEOR [15], ROUGE-L [16] and CIDEr [17].

BLEU (Bilingual Evaluation Understudy) [14] is a popular and widely used automatic evaluation metric in natural language processing, particularly in machine translation. It quantifies the similarity between a candidate translation and one or more reference translations. BLEU calculates the precision of words and phrases in the candidate translation that match the references. It's based on n-gram comparisons, typically unigrams (single words) or bigrams (pairs of adjacent words). BLEU is valuable for quick and automated evaluation, although it has limitations, especially in capturing nuances of meaning and fluency. Despite its shortcomings, it provides a convenient and standardized way to evaluate the quality of machine-translated text.

METEOR (Metric for Evaluation of Translation with Explicit ORdering) [15] is an automatic evaluation metric commonly used in the field of machine translation. It assesses the quality of a translation by comparing it to one or more reference translations. METEOR considers word-to-word matches, stemming, synonymy, and the ordering of words, offering a more comprehensive evaluation compared to metrics like BLEU. It calculates precision, recall, and a harmonic mean score, providing a balanced assessment of the translation quality while considering both content and word order. The metric is widely utilized to evaluate the effectiveness of machine translation systems.

ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation - Longest Common Subsequence) [16] is an automatic evaluation metric used primarily in natural language processing and machine translation to assess the quality of automatically generated text, such as summaries or translations. ROUGE-L measures the overlap in terms of the longest common subsequence (LCS) between the generated text and reference text. It emphasizes recall, assessing how much of the reference text is covered by the generated text. ROUGE-L is effective for evaluating text summarization and related tasks, providing a quantitative measure of the quality and informativeness of the generated content.

CIDEr (Consensus-based Image Description Evaluation) [17] is an automatic evaluation metric used to assess the quality of image captions generated by models in the field of computer vision. Unlike traditional metrics like BLEU, CIDEr takes into account not only lexical similarity but also semantic similarity by considering consensus among human-generated captions. It leverages a consensus-driven approach, rewarding descriptive and diverse captions. CIDEr calculates similarity based on n-grams, providing a more nuanced evaluation of image captions. It's widely recognized for its ability to align well with human judgments and has become a standard evaluation metric in image captioning research.

One metric used to evaluate the calibre of image captions produced by automated systems is called Spice (Semantic Propositional Image Caption Evaluation) [18]. By highlighting semantic propositions or the links between objects and their attributes in the image, it focuses on semantic content and diversity and assesses how well captions match ground truth captions. Spice provides a more comprehensive assessment of the performance of the picture captioning model than just lexical matching by evaluating n-gram overlap, precision, recall of semantic propositions, and caption variety. By highlighting the accurate representation of significant relationships and concepts in generated captions, this statistic is essential for fine-tuning and enhancing picture captioning systems.Most image captioning research focuses on the standard metrics mentioned above.

4.2 Implementation Details

We are also interested in the linguistic features of proposed image captions, particularly connected to sentiment. We use Stanford part-of-speech tagger software to extract adjectives from our models and identify those with strong sentiment values in the SentiCap dataset (see Table 1) . We quantify lexical selection variation by calculating Entropy of the distribution of these adjectives. We compare our models to Mathewset al.[19] baseline models: The CNN+RNN model is trained exclusively on the MSCOCO dataset. The system includes ANP-Replace, ANP-Scoring, RNN-Transfer, and SentiCap, which uses two LSTM modules to learn from factual and sentiment-bearing captions. We also compare SF-LSTM+Adapt, which uses an attention mechanism to weight factual and sentiment-based information. Table 3 shows the findings of all these models from their sources. We compare our models to baseline models. For five generative models, we test the quality of our implementations of existing models CNN+RNN (Baseline-1), ANP-Replace (Baseline-2), ANP-Scoring (Baseline-3), RNN-Transfer (Baseline-4) and SentiCap (Baseline-5). Which uses two LSTM modules to learn from factual and sentiment-bearing captions) as well as the quality of our image encoders, where we compare ResNeXt-152. We report performance on the COCO caption dataset [12]. We evaluate BLEU

[14], ROUGE-L [16], CIDEr [17], METEOR [15] and SPICE [18] and compare model's performances to state-of-the-art models.

ResNeXt-152 (see Figure 2) extends ResNet with grouped convolutions and a split-transform-merge technique for scalability and efficiency. ResNeXt-152 is a deep neural network design that excels at challenging computer vision applications. It's part of ResNeXt, a ResNet extension. The network has 152 layers, hence its name "152". Using split-transform-merge and grouped convolutions, ResNeXt-152 is unique. Multiple parallel information flow paths improve feature learning with grouped convolutions. To optimise network representation and scalability, "cardinality" describes the number of these paths inside a group. ResNeXt-152 excels at image classification, object detection, and segmentation with large datasets by balancing model complexity and performance. Its capacity to efficiently handle and learn from large volumes of data while retaining high speed makes it a useful tool in computer vision, achieving top benchmark results. Our dictionary includes 9,703 words from MSCOCO and SentiCap datasets for all models. A 300-dimensional vector contains each word.

In generator and discriminator the hidden state and memory cell size of our LSTM is set to 512. Optimize the caption generator with the Adam function [20] and set the learning rate to 0.0001. Utilize mini-batches are 64. Discriminator mini-batches are 80. We use Monte Carlo search five times. To converge, models are trained for 20 epochs. Table 3, models utilized the same SentiCap dataset training/test folds for comparability. Our entire model proposed model outperforms the state-of-the-art in all image captioning measures, both positive and negative, in the SentiCap dataset. We present average results to demonstrate our models' improvements over the current state-of-the-art model (See Table 1 & Figure 3). Compared to the prior model, proposed model demonstrated significant increases of 6.15, 6.45, 3.00, and 2.95 points utilizing BLEU-1, ROUGE-L, CIDEr, and BLEU-2 metrics. Other measures indicate minor but positive gains.

Since CNN+RNN is trained primarily on MSCOCO, it has few sentiment ANPs. SentiCap has more sentences containing sentiment words than the other baseline techniques, as expected after word-level regularization. SentiCap's higher sentiment word count compared to ANP-Replace and ANP-Scoring indicates that it actively promotes sentimental ANP usage in sentences.

Table 2. Algorithm on Attention GAN Image Caption

Section	Baseline Method	B-1	B-2	B-3	B-4	R	M	C	S
+Ve	Baseline-1	48.7	28.1	17.0	10.7	36.6	15.3	55.6	-
	Baseline-2	48.2	27.8	16.4	10.1	36.6	16.3	55.2	-
	Baseline-3	48.3	27.9	16.6	10.1	36.5	16.6	55.4	-
	Baseline-4	49.3	29.3	17.9	10.9	37.2	17.0	54.1	-
	Baseline-5	49.1	29.1	17.5	10.8	36.5	16.8	54.4	-
	Proposed Method	56.9	33.6	20.3	12.5	44.3	18.8	61.6	15.9
-Ve	Baseline-1	47.6	27.5	16.3	9.8	36.1	15.0	54.6	-
	Baseline-2	48.1	28.8	17.7	10.9	36.3	16.0	56.5	-
	Baseline-3	47.9	28.7	17.7	11.1	36.2	16.0	57.1	-
	Baseline-4	47.8	29.0	18.7	12.1	36.7	16.2	55.9	-
	Baseline-5	50.0	31.2	20.3	13.1	37.9	16.8	61.8	-
	Proposed Method	56.2	34.1	21.3	13.6	44.6	17.9	64.1	16.2

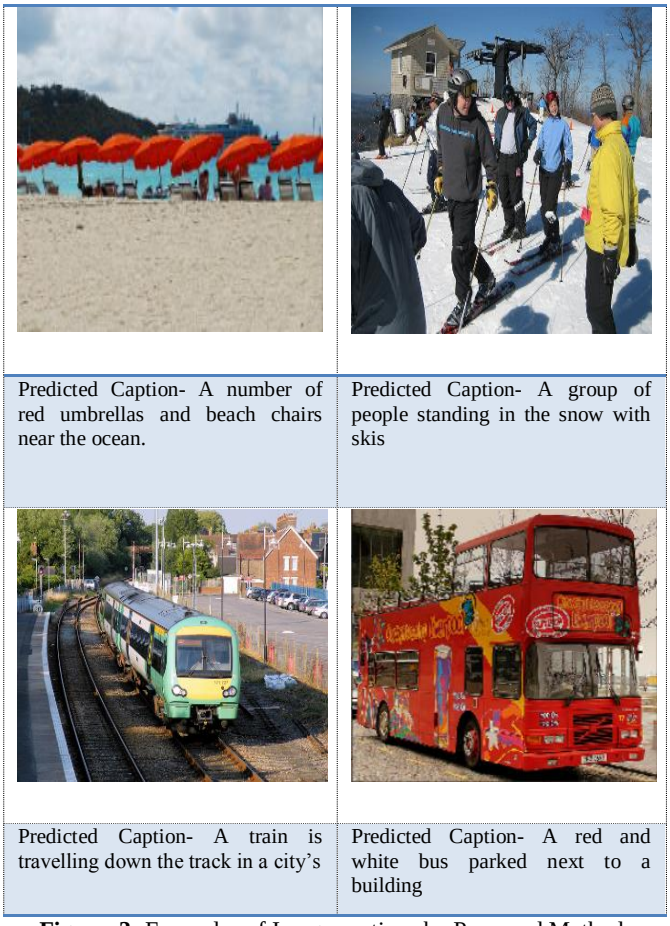

Figure. 3: Examples of Image captions by Proposed Method.

5 Conclusion

This innovative method of producing image captions is based on the combination of reinforcement learning and Monte Carlo (MC) search with a Long Short-Term Memory (LSTM) network. It is further enhanced by a Wasserstein GAN (WGAN)-based caption discriminator that uses Convolutional Neural Networks (CNN). When combined with MC search, the reinforcement learning mechanism improves the LSTM's ability to create diverse, interesting, and contextually accurate captions. This is further improved with the addition of the switching RNN architecture, which divides sentiment-focused word production from factual word generation to enable the creation of complex and emotionally charged captions. Regarding assessment, the generated captions are robustly evaluated by the WGAN and CNN-based discriminator, which guarantees that they correspond with human-like

descriptions and improves the quality of the created content as a whole. This integrated model demonstrates a promising path forward for image captioning, generating captions that eloquently combine emotional comprehension with accuracy to enhance the storyline of visual content.

References

[1]. R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in Advances in neural information processing systems, 2000, pp. 1057–1063.

[2]. S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in Advances in Neural Information Processing Systems, 2015, pp. 1171–1179

[3]. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in Advances in neural information processing systems, 2014, pp. 2672– 2680.

[4]. M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv preprint arXiv:1411.1784, 2014

[5]. L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient." in AAAI, 2017, pp. 2852–2858.

[6]. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot et al., "Mastering the game of go with deep neural networks and tree search," Nature, vol. 529, no. 7587, pp. 484–489, 2016

[7]. S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997

[8]. Chen Chen, Shuai Mu, Wanpeng Xiao, Zexiong Ye, Liesi Wu, and Qi Ju. 2019. Improving image captioning with conditional generative adversarial nets. In AAAI. 8142–8150.

[9]. Z. Yi et al. "DualGAN: Unsupervised Dual Learning for Image-to-Image Translation". In: ArXiv e-prints (Apr. 2017). arXiv: 1704.02510 [cs.CV].

[10]. I. Gulrajani et al. "Improved Training of Wasserstein GANs". In: ArXiv e-prints (Mar. 2017). arXiv: 1704.00028 [cs.LG].

[11]. Milne, Tristan and Adrian I. Nachman. "Wasserstein GANs with Gradient Penalty Compute Congested Transport." ArXiv abs/2109.00528 (2021): n. pag.

[12]. T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick.Microsoft COCO: Common Objects in Context.InEuropean Conference on Computer Vision, pp. 740–755

[13]. P. Mathews, L. Xie, and X. He.SentiCap: Generating Image Descriptions with Sentiments.InAAAI Conference on Artificial Intelligence, pp. 3574–3580 (2016).

[14]. K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu.BLEU: A Method for Automatic Evaluation of Machine Translation.InAnnual Meeting of the Association for Computational Linguistics, pp. 311–318 (Association for Computational Linguistics, 2002).

[15]. M. Denkowski and A. Lavie.Meteor Universal: Language Specific Translation Evaluation for any Target Language.InConference on Machine Translation, pp. 376–380 (2014).

[16]. C.-Y. Lin.Rouge: A Package for Automatic Evaluation of Summaries. Text Summarization Branches Out (2004).

[17]. R. Vedantam, C. Lawrence Zitnick, and D. Parikh.CIDEr: Consensus-Based Image Description Evaluation.InIEEE Conference on Computer Vision and Pattern Recognition, pp. 4566–4575 (IEEE, 2015).

[18]. Anderson, Peter, Basura Fernando, Mark Johnson and Stephen Gould. "SPICE: Semantic Propositional Image Caption Evaluation." ArXiv abs/1607.08822 (2016): n. pag.

[19]. Mathews, A., Lexing Xie and Xuming He. "SentiCap: Generating Image Descriptions with Sentiments." ArXiv abs/1510.01431 (2015): n. pag.

[20]. D. P. Kingma and J. Ba.Adam: A Method for Stochastic Optimization. arXiv Preprint arXiv:1412.6980 (2014).