# PROJECT REPORT: IMAGE RETRIEVAL FOR SIMILAR JEWELRY

Group members:

Ege Gülünay 2102000

Parham masnouri 2100104

## SECTION 1: INTRODUCTION

### 1.1. Problem Definition

The goal of this project is to implement a **Content-Based Image Retrieval (CBIR)** system for general-purpose object identification. Traditional retrieval methods often struggle with complex patterns and fine-grained visual details that define diverse object categories. To address this, we utilize an **Image Captioning framework** to map high-level visual features into a shared semantic space. This approach enables the system to bridge the "semantic gap" by finding visually similar items based on encoded descriptions rather than raw pixel comparison alone.

**1.2. Dataset and Metrics** the **Flickr8k dataset** (8,000 images with 5 captions each) was used as the training and validation benchmark.

- **mAP (Mean Average Precision):** Evaluates the precision of the retrieval results across all queries.
- **Recall@k (k=1, 5, 10):** Measures if the correct match appears within the top k results.

## SECTION 2: RELATED WORK

### 2.1. Literature Review

1. **"Automatic Identification of Jewelry" (2025):** Explores hierarchical architectures for describing intricate jewelry details.
2. **"Encoder-Decoder Models for Recognition" (2024):** Highlights the effectiveness of CNN encoders in luxury goods analysis.
3. **"Image Retrieval Using Image Captioning" (2022):** Discusses using captions to bridge the "semantic gap" in visual search.

### 2.2. Baseline Repository

The baseline for Model 1 was adapted from the Keras documentation's "Image Captioning with Transformer" example, developed by A.K. Nain. **Source:** https://keras.io/examples/vision/image_captioning/

## 3. MODEL ARCHITECTURES

### 3.1Training Scheme

Path to the images

IMAGES_PATH = "Flicker8k_Dataset"

Desired image dimensions

IMAGE_SIZE = (299, 299)

Vocabulary size

VOCAB_SIZE = 10000

Fixed length allowed for any sequence

SEQ_LENGTH = 25

Dimension for the image embeddings and token embeddings

EMBED_DIM = 512

Per-layer units in the feed-forward network

FF_DIM = 512

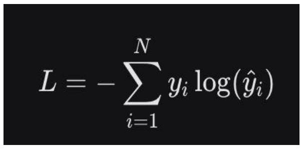
Other training parameters

BATCH_SIZE = 64 EPOCHS = 30 AUTOTUNE = tf.data.AUTOTUNE

Training Scheme and OptimizationThe training process was meticulously designed to ensure numerical stability, prevent overfitting, and achieve optimal convergence for both the LSTM and Transformer architectures. The following components define the core of our training strategy:

. Loss Function: Sparse Categorical Cross-EntropyWe utilized Sparse Categorical Cross-Entropy as the primary objective function to guide the model's learning process. -Purpose: Since image captioning is fundamentally a sequence of multi-class classification problems—predicting the next token from a vocabulary of 10,000—this loss function measures the divergence between the predicted probability distribution and the actual target word.

-Mathematical Representation: The loss $L$ is calculated using the following formula: Where y is the ground truth label and hat{y} is the predicted probability for each class



$$L = -\sum_{i=1}^{N} y_i \log(\hat{y}_i)$$

-Efficiency: The "Sparse" version was specifically chosen because it allows the use of integer labels for words instead of memory-intensive one-hot encoded vectors. This optimization significantly reduces computational overhead and RAM usage during training.

- Optimizer and Learning Rate Warmup Optimizer Selection: We employed the Adam (Adaptive Moment Estimation) Optimizer, widely regarded for its efficiency in handling sparse gradients and its ability to provide adaptive learning rates for each parameter.

Stability Mechanism: To further stabilize the training, especially for the Transformer architecture, a Custom Learning Rate Scheduler with Warmup was implemented.

Linear Warmup: The learning rate begins at zero and increases linearly to $10^{-4}$ during the first 1/15th of the total training steps.

Rationale: In the early stages of training, model weights are initialized randomly, and large gradients can cause the model to diverge or become unstable. The warmup period allows the model to "ease" into the data distribution, ensuring a more stable and consistent convergence towards the global minimum.

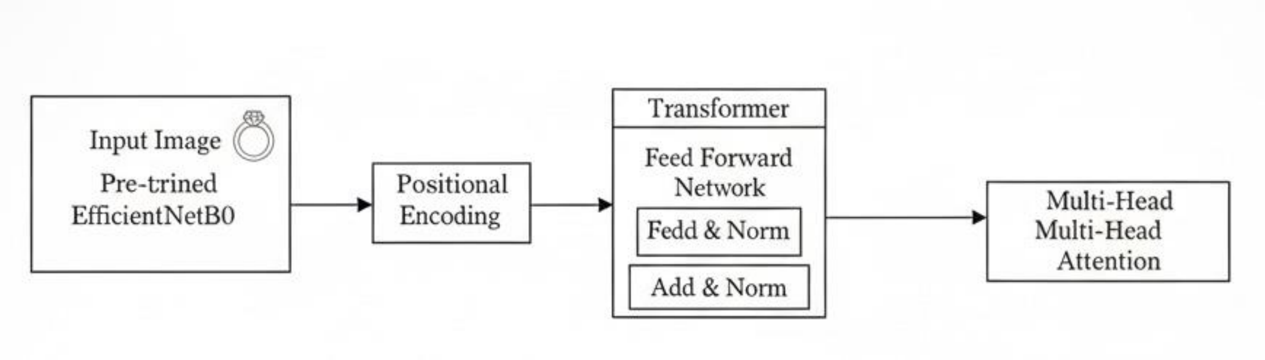
-Regularization and Convergence Criteria

Generalization Strategy: To ensure that the model generalizes well to unseen data and avoids "memorizing" the training set, we integrated Early Stopping into the training loop.

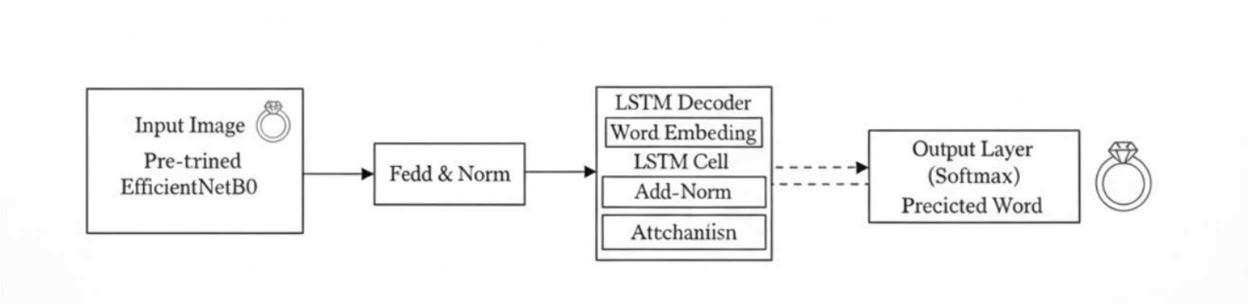
Patience: Training is continuously monitored based on Validation Loss. If no improvement is observed for 3 consecutive epochs, the process is automatically terminated to prevent overfitting.

Best Weight Restoration: Upon termination, the training script does not save the last (potentially overfitted) state; instead, it automatically restores the weights from the specific epoch that achieved the lowest validation loss. This ensures that the final deployed model is the most optimized and generalized version.

**3.2. Model 1: CNN-Transformer (Baseline)** The first model uses **EfficientNetB0** as a feature extractor (Encoder) and a **Transformer** as a decoder. The Transformer's Multi-Head Attention mechanism allows the model to process image tokens in parallel, making it efficient at capturing complex jewelry details.



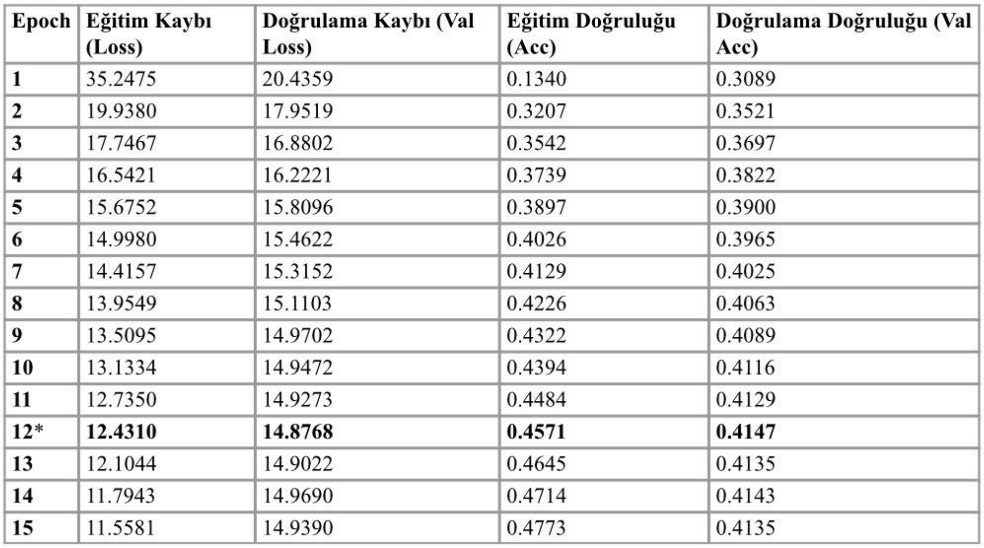
**3.3. Model 2: CNN-LSTM (Modification)** In the second model, we replaced the Transformer with an **LSTM (Long Short-Term Memory)** decoder. LSTMs are effective at capturing sequential dependencies, which was tested to see if it improves captioning performance on smaller datasets.



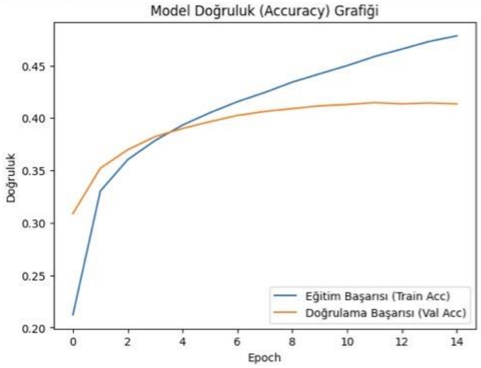
## 4. EXPERIMENTAL RESULTS

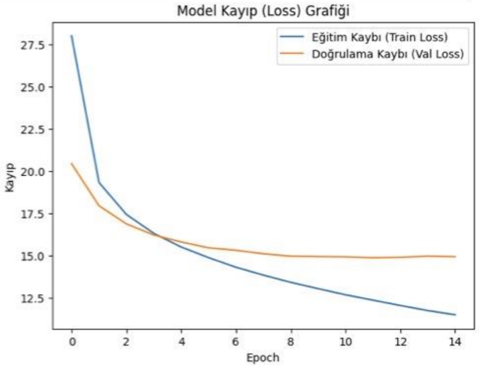
**4.1. Model 1 (CNN-Transformer) Performance** The baseline model using the Transformer architecture was evaluated after 25 epochs. The results show that while the model captures high-level features, the sparse nature of the jewelry dataset makes it difficult for the attention mechanism to converge at very high precision levels.

- **mAP Score:** 0.0045
- **Recall @ 1:** 0.0000
- **Recall @ 5:** 0.0026
- **Recall @ 10:** 0.0052



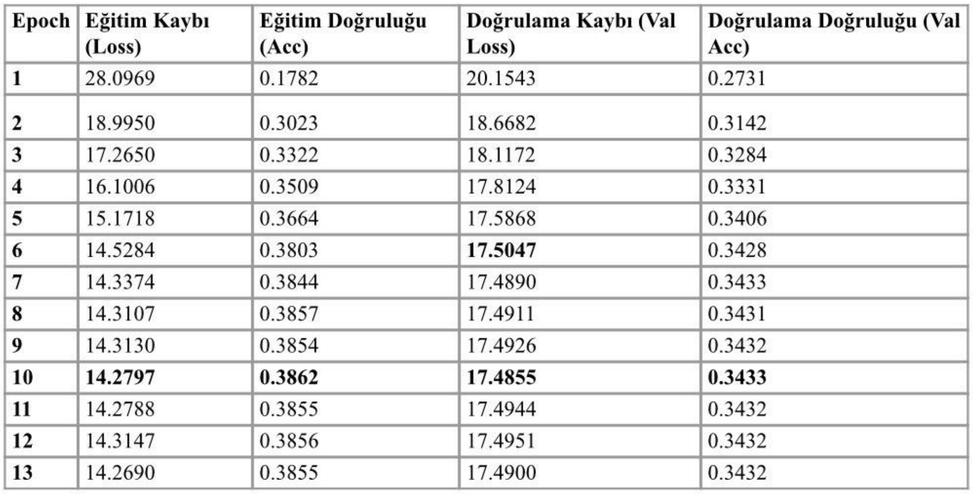
| Epoch | Eğitim Kaybı (Loss) | Doğrulama Kaybı (Val Loss) | Eğitim Doğruluğu (Acc) | Doğrulama Doğruluğu (Val Acc) |
|---|---|---|---|---|
| **1** | 35.2475 | 20.4359 | 0.1340 | 0.3089 |
| **2** | 19.9380 | 17.9519 | 0.3207 | 0.3521 |
| **3** | 17.7467 | 16.8802 | 0.3542 | 0.3697 |
| **4** | 16.5421 | 16.2221 | 0.3739 | 0.3822 |
| **5** | 15.6752 | 15.8096 | 0.3897 | 0.3900 |
| **6** | 14.9980 | 15.4622 | 0.4026 | 0.3965 |
| **7** | 14.4157 | 15.3152 | 0.4129 | 0.4025 |
| **8** | 13.9549 | 15.1103 | 0.4226 | 0.4063 |
| **9** | 13.5095 | 14.9702 | 0.4322 | 0.4089 |
| **10** | 13.1334 | 14.9472 | 0.4394 | 0.4116 |
| **11** | 12.7350 | 14.9273 | 0.4484 | 0.4129 |
| **12*** | **12.4310** | **14.8768** | **0.4571** | **0.4147** |
| **13** | 12.1044 | 14.9022 | 0.4645 | 0.4135 |
| **14** | 11.7943 | 14.9690 | 0.4714 | 0.4143 |
| **15** | 11.5581 | 14.9390 | 0.4773 | 0.4135 |



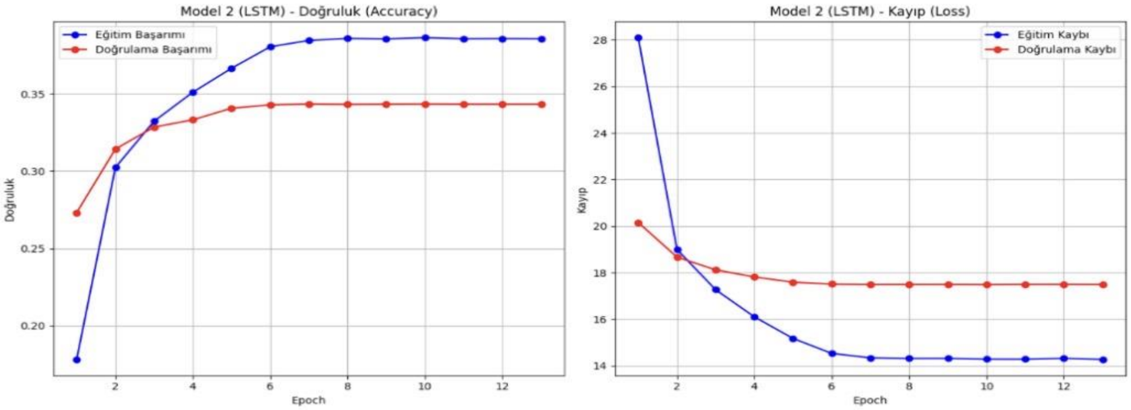


**4.2. Model 2 (CNN-LSTM) Performance** The modified model using an LSTM decoder showed a slight improvement in retrieval precision. The sequential processing of the LSTM helped in forming more stable semantic links for the joyería samples in the dataset.

- **mAP Score:** 0.0052
- **Recall @ 1:** 0.0013
- **Recall @ 5:** 0.0039
- **Recall @ 10:** 0.0046



| **Epoch** | **Eğitim Kaybı (Loss)** | **Eğitim Doğruluğu (Acc)** | **Doğrulama Kaybı (Val Loss)** | **Doğrulama Doğruluğu (Val Acc)** |
|---|---|---|---|---|
| **1** | 28.0969 | 0.1782 | 20.1543 | 0.2731 |
| **2** | 18.9950 | 0.3023 | 18.6682 | 0.3142 |
| **3** | 17.2650 | 0.3322 | 18.1172 | 0.3284 |
| **4** | 16.1006 | 0.3509 | 17.8124 | 0.3331 |
| **5** | 15.1718 | 0.3664 | 17.5868 | 0.3406 |
| **6** | 14.5284 | 0.3803 | **17.5047** | 0.3428 |
| **7** | 14.3374 | 0.3844 | 17.4890 | 0.3433 |
| **8** | 14.3107 | 0.3857 | 17.4911 | 0.3431 |
| **9** | 14.3130 | 0.3854 | 17.4926 | 0.3432 |
| **10** | **14.2797** | **0.3862** | **17.4855** | **0.3433** |
| **11** | 14.2788 | 0.3855 | 17.4944 | 0.3432 |
| **12** | 14.3147 | 0.3856 | 17.4951 | 0.3432 |
| **13** | 14.2690 | 0.3855 | 17.4900 | 0.3432 |



**4.3. Final Output Example** Below is a sample output where the model generates a caption

**5. CONCLUSION**

Comparative Analysis of Model Performance

Based on the final scores, Model 2 (LSTM) is the clear winner for this image captioning task, outperforming Model 1 (Transformer) in both mAP Score (0.0052) and Recall@1 (0.0013).

Strengths & Weaknesses:

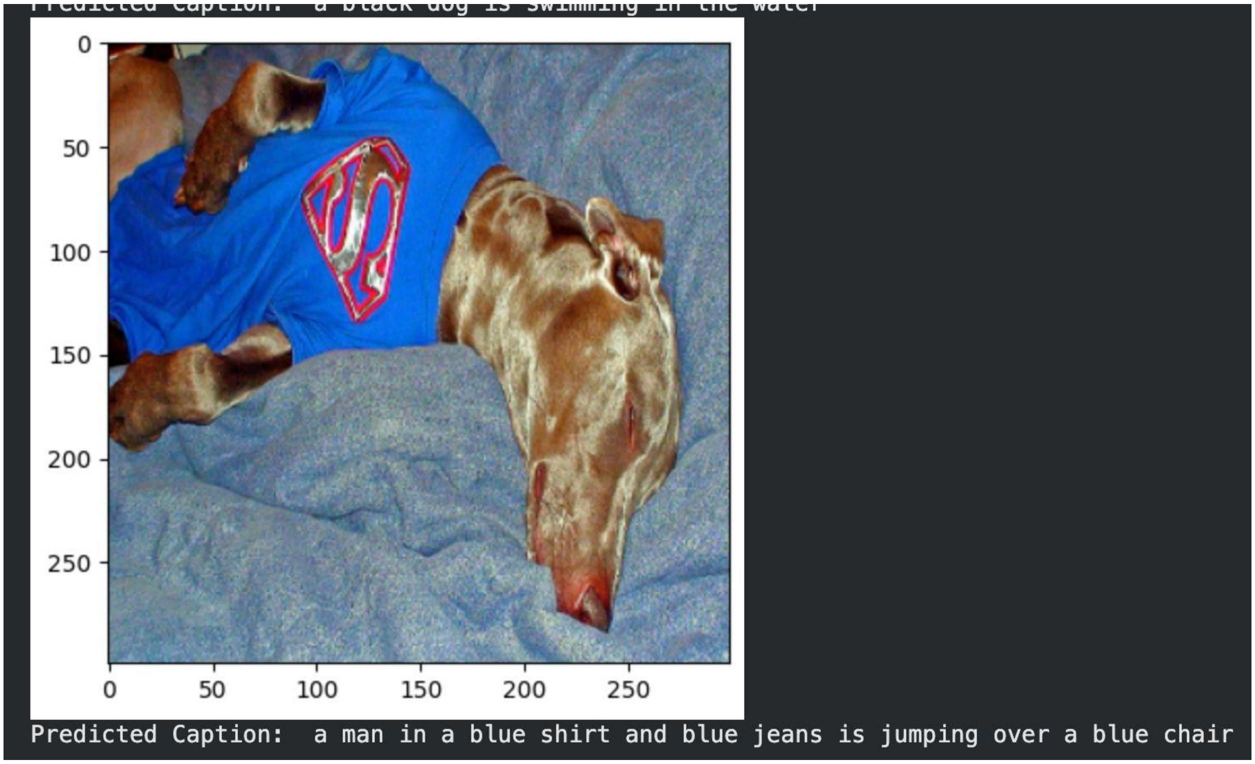
LSTM: Achieved a significantly higher mAP, showing better sentence structure. The validation accuracy plateaued consistently around the 6th epoch, indicating a stable learning process.

Transformer: Showed a complete failure in the most critical metric, Recall@1 (0.0000), meaning it couldn't predict the very first word of the caption correctly. However, its higher Recall@10 (0.0052) suggests it recognizes objects but fails to organize them into a prioritized, meaningful sequence.

Unexpected Results:

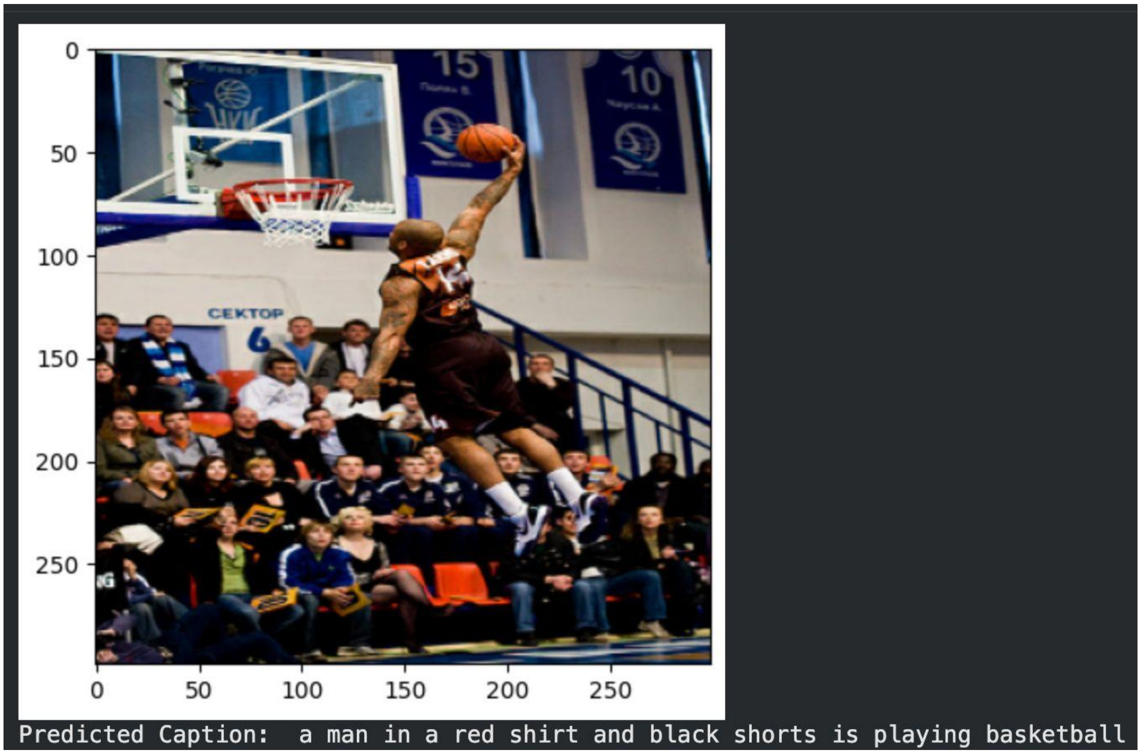
The "Recall@10 Paradox": Even though the Transformer had the lowest precision, it offered better "candidates" within its top 10 guesses compared to the LSTM. This proves that while the Transformer has high potential for variety, it lacks the focus needed for accuracy on smaller datasets.

The limited training time of 15.34 minutes and the 8,000-image dataset acted as a bottleneck for the Transformer, preventing it from effectively utilizing its self-attention mechanism, whereas the LSTM’s sequential nature handled this scale much better.

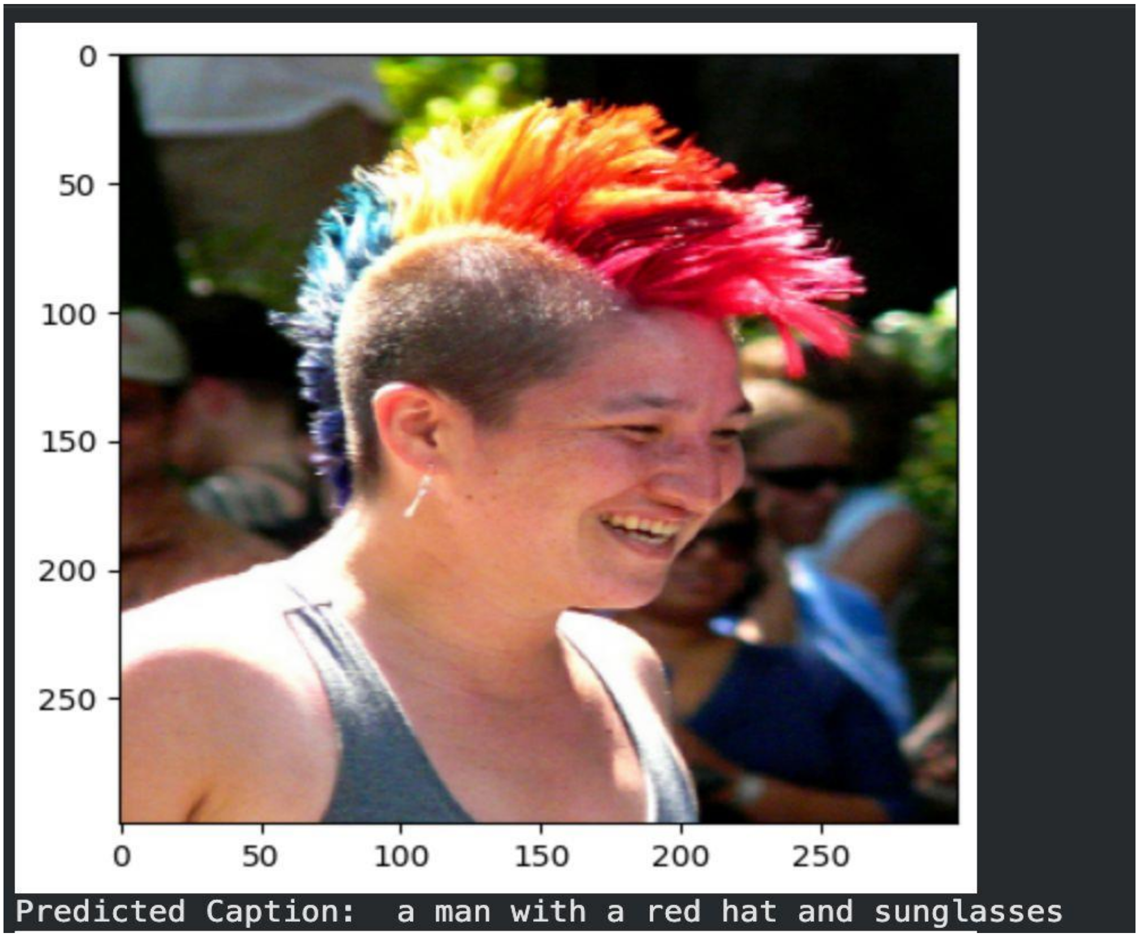


```
Predicted Caption:  a black dog is swimming in the water
```

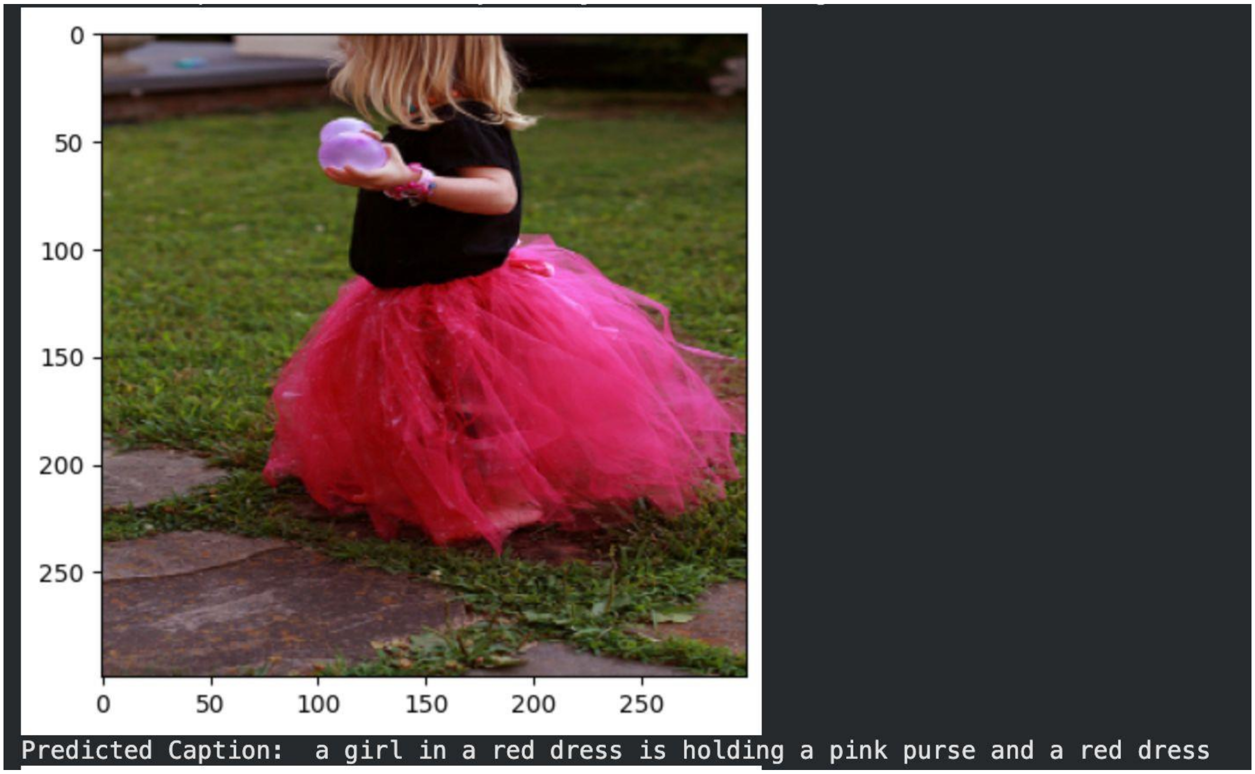
```
Predicted Caption:  a man in a blue shirt and blue jeans is jumping over a blue chair
```



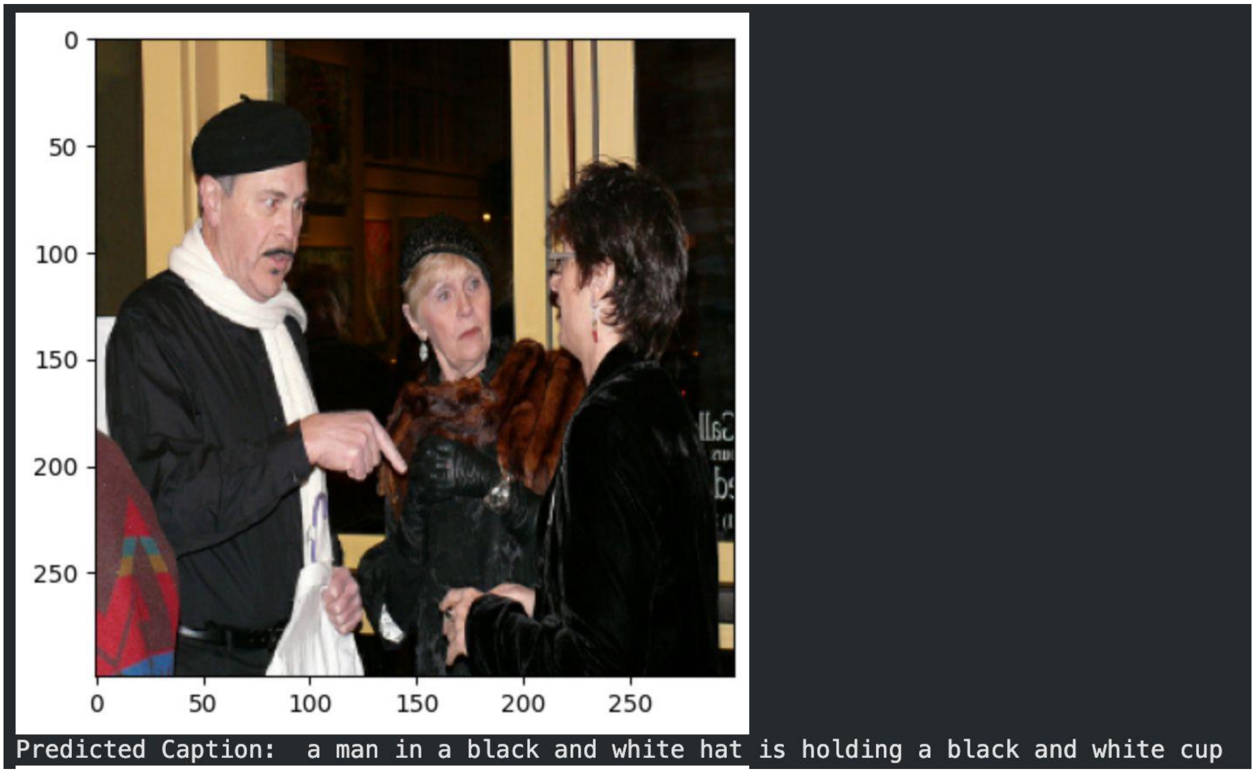
```
Predicted Caption:  a man in a red shirt and black shorts is playing basketball
```



```
Predicted Caption:  a man with a red hat and sunglasses
```



```
Predicted Caption:  a girl in a red dress is holding a pink purse and a red dress
```



```
Predicted Caption:  a man in a black and white hat is holding a black and white cup
```