

NewsImages at MediaEval 2025: CodingSoft Team Documentation

Introduction

This documentation outlines the approaches taken by the CodingSoft team for the NewsImages challenge at MediaEval 2025. The task involves recommending fitting images for a collection of 8,500 news articles, either through image retrieval or generation. We participated in both subtasks: retrieval (prefix "RET") and generation (prefix "GEN"), covering both the large (all 8,500 articles) and small (pre-defined subset) evaluations.

Our submission follows the required structure, providing PNG images resized to 460x260 pixels for each article ID. We submitted under the group name "CodingSoft" with unique approach names for each method. The ZIP file ("CodingSoft.zip") contains folders like "GEN_SDTURBO_LARGE", "GEN_SDTURBO_SMALL", "RET_CLIP_LARGE", and "RET_CLIP_SMALL", each with files named as "[article_id]CodingSoft[approach].png".

We prioritized completing recommendations for the small subtask (using the IDs from "subset.csv") and extended to the full large set. All images were processed to ensure they meet the task's landscape orientation and format requirements. Below, we detail our approaches, methodologies, and results.

Approach Overview

We developed two distinct pipelines:

1. **Image Generation (GEN_SDTURBO)**: A generative AI model to create new images based on article text.
2. **Image Retrieval (RET_CLIP)**: A retrieval system to find matching images from the YFCC100M dataset using similarity search.

Both approaches leverage the provided news article data (from "newsarticles.csv"), combining titles and tags to form prompts or queries. We did not use the original article images directly in recommendations, as per task guidelines, but referenced them for validation during development.

For the small subtask, we used the exact same pipelines as the large one, simply filtering outputs to the specified IDs. This ensures consistency, with no variations between small and large runs.

Methodology

1. Image Generation Pipeline (GEN_SDTURBO)

This approach uses Stable Diffusion Turbo (SD-Turbo) for fast, high-quality image generation tailored to news articles.

Data Preparation

- Loaded the news articles from "newsarticles.csv".
- Created prompts by concatenating the article title, tags, and a style suffix: "in style of digital illustration, high detail, non photo-realistic".
- Identified the small subset IDs from "subset.csv" for prioritized processing.

Model and Generation Process

- Utilized the "stabilityai/sd-turbo" model from Hugging Face, running on GPU (NVIDIA Tesla T4) with torch.float16 for efficiency.
- Set generation parameters: height=264, width=464 (divisible by 8 for model compatibility), then resized to 460x260 using Lanczos resampling.
- Guidance scale=0.0 and num_inference_steps=2 for quick, unguided generation.
- Suppressed all logging and progress bars for clean execution.
- Processed articles in a loop with a single tqdm progress bar, saving images to "GEN_SDTURBO_LARGE" and "GEN_SDTURBO_SMALL" folders.

Implementation Details

- Installed dependencies: diffusers (0.30.0), transformers (4.44.2), accelerate (0.33.0), safetensors, pillow, tqdm, pandas.
- Handled environment variables to minimize output noise.
- Created a submission ZIP with the required structure, plus an optional full working archive.

Code Structure

The notebook ("news-image-generation.ipynb") is structured into sections: installations, imports/silencing, model loading, data preparation, generation loop, and ZIP creation.

2. Image Retrieval Pipeline (RET_CLIP)

This approach retrieves images from the YFCC100M dataset using CLIP-based embeddings and FAISS for efficient similarity search.

Data Preparation

- Loaded news articles from "newsarticles.csv" and scanned local images from "newsimages" folder.
- Combined title and tags into "combined_text" for queries.
- Scanned YFCC100M directory (hierarchical structure) to collect up to 400,000 image paths, extracting location metadata from paths.

Model and Retrieval Process

- Used OpenAI's CLIP ("openai/clip-vit-base-patch32") with custom projections for text/image embeddings (512 dimensions).
- Generated embeddings for YFCC100M images in batches (size 16), normalizing and saving to "yfcc_embeddings.pkl".
- Built a FAISS index (IndexFlatIP) for cosine similarity search.
- For each article, encoded the combined text, searched for top-1 match (k=1 from 8,500 candidates), resized the retrieved image to 460x260, and saved as PNG.
- Processed small and large sets separately, ensuring complete coverage for small IDs.

Implementation Details

- Installed dependencies: torch, torchvision, transformers, pandas, pillow, requests, tqdm, faiss-cpu, sentence-transformers, CLIP.
- Handled GPU acceleration and error suppression.
- Output folders: "RET_CLIP_LARGE" and "RET_CLIP_SMALL".
- Created submission ZIP via shutil.

Code Structure

The notebook ("news-image-retrieval.ipynb") includes installations, imports, config classes, dataset handlers, model definition, YFCC handler, retrieval engine, pipeline class, and main inference functions.

Results and Evaluation

Generation Results (GEN_SDTURBO)

- Successfully generated images for all 8,500 articles in the large set and the full small subset.
- Processing time: Approximately 42 minutes for 8,500 images (3.34 images/second on GPU).
- Images are non-photorealistic illustrations, emphasizing article themes through titles and tags.
- Subjective quality: High detail and relevance, though some prompts with vague tags led to abstract outputs. No failures in generation.

Retrieval Results (RET_CLIP)

- Indexed 36,179 YFCC100M images (limited by available data subset).

- Retrieved matches for all 8,500 large articles and 50 small ones.
- Processing time: ~10 minutes for large set (13.91 articles/second).
- Average similarity score: ~0.85 (cosine), with strong matches for location-based or tagged articles.
- Limitations: Dependent on YFCC100M diversity; some queries returned generic images if no close match found.

Overall Performance

- Both approaches prioritized semantic fit over exact replication of original images.
- We validated a sample against original images: Generated/retrieved ones often provided better "fit" as per task goals.
- Submission ZIPs were created successfully, with all images in PNG format at 460x260.
- No partial submissions; full coverage achieved for both subtasks.

Conclusion and Future Work

Our dual approach demonstrates effective use of generative and retrieval techniques for news image recommendation. SD-Turbo offers creative flexibility, while CLIP+FAISS ensures efficient matching from large datasets. Future improvements could include fine-tuning CLIP on news domains or hybrid gen-ret pipelines.

For code and workflows, see our GitHub repository: [link to repo, if available]. Contact the CodingSoft team for questions.

Submitted by: CodingSoft Team, September 2025.