Open Source LLM Experiment for BBS Puppet Show

Kazi Islam

Version 1.0

Prepared for Prof David Smith

Date [10-19-2025]

1. Abstract **3**
2. Objective **3**
3. Open Source LLMs **3**
3.1. Reason Behind Picking These LLMs 4
4. Scope **4**
5. Requirements and Constraints **5**
6 Dataset Plan **5**
6.1 Class list 5
6.2 Data augmentation (training time only) 5
6.3 Testing via single-image upload 5
7. Evaluation Plan **6**
7.1 Identification task & score (Top-K) 6
7.2 Latency score (speed per image) 6
7.3 Overall scoring 6
8. Model Comparison **6**
9. Sources (selected) **7**
10. Version History **7**

1. Abstract

This document outlines a structured framework to build a simple, open-source model for the Balanced Blended Space (BBS) puppet show that learns our characters and objects from pictures with short descriptions. After training, the model should recognize who or what appears in a new photo that is similar to the images in our dataset and label it for the show.

Our current focus is to collect and review open-source models that can be trained on image-and-text data. We will compile a shortlist based on clear criteria: permissive license, active community, documented training steps, ability to fine-tune on our images and captions, and support for evaluating recognition on new scenes. With the chosen models, we will organize our dataset, run small training tests, and measure simple outcomes: naming accuracy and whether the correct answer appears in the first few guesses. The result will be a practical, reusable recipe and model list that BBS (and similar projects) can adopt to recognize characters and props for future performances.

2. Objective

This study is aimed to build a simple, open-source LLM for the Balanced Blended Space (BBS) puppet show. We will train the LLM with a curated dataset of pictures of characters and objects along with their short descriptions. After training, it will be able to recognize what object or character appears in a new photo which is similar or related to the characters or objects that the model is trained on for the Balanced Blended Space Puppet show.

3. Open Source LLMs

We are selecting a small set of open, well-supported vision-language models we can fine-tune on our BBS image-and-caption dataset. Each model will learn our characters and props from labeled photos and short descriptions, then recognize similar new photos taken during rehearsals and shows. Our plan: organize clean images, write consistent captions, fine-tune with lightweight methods, and check results on fresh scenes with different backgrounds and lighting. We'll start with one model, compare speed to learn and reliability on look-alike puppets, and iterate by adding diverse views or clearer captions if needed. The goal is a practical, repeatable setup that we can share, reuse, and improve over time. Based on our goals we have picked five models to be the best fit for exploration for our BBS puppet show. These are-

1. Qwen2-VL-7B (Instruct- “Instruction-tuned.” The model was fine-tuned to follow plain, chat-style directions. You can ask it questions in simple language and it will try to follow your intent.) Hugging Face

2. LLaVA-OneVision-1.5 (4B/8B- Model size options: ~4 Billion (smaller, faster, easier to run; a bit less accurate) vs ~8 Billion (bigger, slower, needs more memory; usually more accurate) parameters.) GitHub+1
3. Idefics-2-8B Hugging Face+1
4. Phi-3.5-Vision (Instruct) Hugging Face+1
5. InternVL 2 (InternVL2-8B- A specific checkpoint name inside the InternVL2 family—the 8B-parameter version.Hugging Face+2InternVL+2)

3.1. Reason Behind Picking These LLMs

Below are the reasons why we picked these open source LLMs:

Qwen2-VL-7B

Great all-rounder for pictures + text. Understands prompts well and tends to give solid, on-topic answers. Qwen2-VL-7B-Instruct

LLaVA-OneVision-1.5 (4B or 8B)

Lots of tutorials and examples online, so it's easy to customize for our puppets/props. 4B is faster; 8B is more accurate. LLaVA-OneVision-1.5

Idefics-2-8B

Especially good when images include text (signs, labels on props). Helpful if some props have printed names or numbers. Idenfics-2

Phi-3.5-Vision

Smaller but surprisingly capable. Good choice if you want quick experiments and short training cycles. Phi-3.5-Vision

InternVL2-8B

Modern, well-maintained model that handles varied image resolutions nicely. Solid baseline for recognition + short captions. Hugging Face+2InternVL+2)

4. Scope

We will build a small first version that teaches an image-and-text AI (a vision-language model) to name known BBS puppets and props from a single photo. We'll lightly fine-tune the model on our own labeled pictures with short captions so it learns our characters. The system will run offline on a normal laptop (using a GPU if available, or a CPU if not). For testing, we will give the model one photo at a time and check if it names the right character or prop. We'll repeat this across a fixed test set with different backgrounds, angles, and lighting. We will compare several models using two main metrics: accuracy (how often the name is correct—Top-1 and Top-3) and latency (time per image to make a prediction).

5. Requirements and Constraints

We'll keep the LLM small so it can run on a normal laptop. If a graphics card is available, we'll use its memory (VRAM); if not, it should still work on the regular processor (CPU), just slower. We'll process only a few images at a time (small batch size) so we don't run out of memory. During the BBS puppet show, the model should answer quickly, aim for under a second per image (low latency) and work offline with no internet. We'll watch costs by running locally first and only using the cloud if needed, with a clear monthly limit. For privacy and ethics, we'll use only images we have permission to use, avoid or blur people and sensitive signs, respect copyrights and trademarks (IP), and store all data securely. [InternVL2-8B-AWQ]

6 Dataset Plan

Build a small, clean image-text dataset so a vision-language model can learn the names of known BBS puppets and props and recognize them in new photos.

6.1 Class list

- Make a fixed list of images for a particular character/prop (10–40 images in a list).
- Give each list a short, unique name (e.g., `Blue_Dragon`, `Red_Fan`) to every list of characters.
- Add a one-line description per list (color, material, any unique feature).
- Formats of images: `.jpg/.jpeg/.png` (RGB), 480p–4K; max 10 MB each.
- File name example: `Blue_Dragon_2025-06-01_side_medium_bright.jpg`.
- Delete blurry/overexposed shots unless needed for "hard cases."

6.2 Data augmentation (training time only)

- Light crop/resize, horizontal flip, slight brightness/contrast.
- Avoid heavy color shifts that change identity cues.

6.3 Testing via single-image upload

- **Input:** We will upload one image in order for the model to make a set number of guesses about which list the image/ character belongs to or resembles with.

7. Evaluation Plan

7.1 Identification task & score (Top-K)

- Give the model one uploaded image. It returns an ordered list of guesses for the correct class/list.
- Default K = 5 guesses (we can also run K = 3 for a quicker test).
- Score per image (Top-5 mode):
  1st guess correct → **5/5**
  2nd → **4/5**
  3rd → **3/5**
  4th → **2/5**
  5th → **1/5**
  Not in Top-5 → **0/5**
  (If using Top-3: 1st=3/3, 2nd=2/3, 3rd=1/3, else 0/3.)

7.2 Latency score (speed per image)

Measure time for prediction:

- **< 1s → 3/3**
- **< 2s → 2/3**
- **< 3s → 1/3**
- **≥ 3s → 0/3**

7.3 Overall scoring

- Per image total = Identification (0–5) + Latency (0–3) = 0–8.

- Model score = average over all test images, also report median latency.

8. Model Comparison

We will make a chart consisting of the scores of all the models. We will order them based on accuracy and latency time. Then we will pick the top 2 models for our BBS puppet show.

9. Sources (selected)

- Qwen2-VL-7B-Instruct model card (Apache-2.0; dynamic resolution & benchmarks). Hugging Face
- LLaVA-OneVision-1.5 GitHub (fully open framework); LLaVA-OneVision-1.5 paper (recent results/benchmarks). GitHub+1
- Idefics2-8B model card & blog (OCR/document understanding; Apache-2.0). Hugging Face+1
- Phi-3.5-Vision-Instruct model card & model page (128K context; MIT). Hugging Face+1
- InternVL2-8B model card & blog; AWQ quantized release (deployment efficiency). Hugging Face+2InternVL+2

10. Version History

Version	Created on	Created by
1.0	10/05/2025	Kazi Islam
2.0	10/19/2025	Kazi Islam