Can Diffusion Models Be Used for Multilingual Symbol Generation

Anonymous ACL submission

Abstract

We investigate the use of denoising diffusion probabilistic models (DDPMs) with U-Net backbones for multilingual character generation across diverse writing systems. Our study spans Devanagari, English, Arabic, and Mayan scripts, exploring both script-specific and unified multi-script generation approaches. We systematically evaluate the effects of resolution scaling, training dataset size, and attention mechanisms on generation quality. Additionally, we incorporate conditional generation with T5 text embeddings to guide a unified model across scripts. Experiments show that diffusion models can faithfully learn individual script characters when trained in isolation. However, a unified model spanning multiple scripts faces significant challenges: cross-script generalization is sensitive to training data diversity, the number of training epochs, and visual similarity among scripts. We find that increasing image resolution yields only marginal quality gains unless accompanied by more training data. While T5-based conditioning improves control across scripts, overlapping character features across scripts still cause confusion. Our findings highlight both the potential of diffusion models for multilingual symbol generation and the practical challenges of achieving robust unified generation across diverse scripts.

1 Introduction

Multilingual or logographic character generation holds immense value in the context of language preservation, OCR systems, and universal character synthesis. Scripts like Devanagari, Arabic, and Mayan differ vastly in structure and style, requiring generative models to generalize across diverse visual representations. In this work, we explore whether Diffusion models that are known for their success in natural image generation (Ho et al., 2020; Nichol and Dhariwal, 2021) can learn to generate high-quality character images across multiple scripts.

2 Related Work

Several works have explored the use of GANs and VAEs for character generation (Azadi et al., 2018; Graves, 2013), especially for handwriting synthesis and font style transfer. Diffusion models have recently shown promise in high-fidelity image synthesis, but their application to symbol generation across diverse scripts remains underexplored. Prior work on diversity metrics and proxy-comparison (Sajjadi et al., 2018; Heusel et al., 2017) frameworks in generative models forms the foundation for evaluating cross-script generation.

3 Problem Statement

We define the following research questions:

- RQ1: Can diffusion models generate high-quality logographic and alphabetic characters?
- RQ2: How do data size and image resolution affect output quality?
- RQ3: Can these models serve as a proxy for cross-script comparison tasks?

These questions aim to understand the capability and limits of diffusion models in learning symbolic structure and style.

4 Methodology

We adopt a U-Net-based architecture using HuggingFace's *Diffusers* library (HuggingFace, 2023) under the DDPM framework. The model operates on grayscale character images ($1 \times 128 \times 128$ or $1 \times 256 \times 256$) with symmetric downsampling and upsampling blocks, skip connections, and six resolution stages ($128 \rightarrow \ldots \rightarrow 4$), with feature channels scaling from 128 to 512. Each block includes

one ResNet layer (`layers_per_block` $= 1$), and timestep embeddings condition the model throughout. To assess attention's role, we tested variants with no attention, bottleneck attention (1/4/8 heads at 4×4), and higher-resolution attention (16×16), keeping other hyperparameters fixed. Training was done from scratch in PyTorch (Paszke et al., 2019) using Gaussian noise and a configurable scheduler loop. While attention was hypothesized to help capture global stroke dependencies, results showed that a reduced-channel convolutional U-Net sufficed for high-quality generation, particularly under limited data conditions.

5 Dataset Preparation

5.1 Devanagari

Figure 1: Sample grid of Devanagari characters dataset

We constructed synthetic datasets for Devanagari, English, Arabic and Mayan scripts to train and evaluate our diffusion models. All images are grayscale (128×128 unless specified), with centered black glyphs on white backgrounds and no additional augmentations unless noted.

We rendered over 80 Devanagari characters—including vowels, consonants, and ligatures—across 305 Unicode-compliant fonts (Singh, 2025), yielding approximately 24,000 images. To study resolution effects, we created a 256×256 high-resolution subset for selected characters across many fonts. This version preserved finer stroke details but required reduced batch sizes or epochs due to memory constraints. The full 128px dataset has been released publicly.

5.2 English

For English, we used a large-scale dataset containing stylized grayscale images rendered from over 85,000 unique fonts collected online. Each character (e.g., 'A', 'b', '5') is organized into separate ZIP archives, each containing thousands of 128×128 images with consistent formatting.

To evaluate performance on a minimal subset, we trained diffusion models specifically on the lowercase letter "g," sampling 100 images from the dataset. . We found that 100 images and 100 epochs provided a strong baseline, with additional training steps tested to explore convergence behavior.

5.3 Mayan

Two datasets were created using Mayan glyphs (MayaGlyphs.org, 2025). The first contained 100 augmented images of a single glyph ("mam," meaning grandfather), with variations including rotation, noise, and positional shifts. The second included 105 images across five different glyphs, with each class augmented to 20 samples via transformations such as additive noise, line thickness variation, and off-center alignment to simulate naturalistic variation.

Figure 2: Five mayan glyphs used for training, with 'mam' (center) employed for single-glyph training.

5.4 Arabic

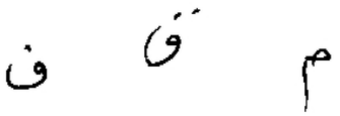

Figure 3: Sample images from our Arabic dataset for the letters *faa* (ف), *qaaf* (ق), and *meem* (م).

We constructed a dataset focusing on three Arabic letters: *faa* (ف, U+0641), *qaaf* (ق, U+0642), and *meem* (م, U+0645), using a mix of handwritten and synthetic glyphs. For each letter, we collected 90 handwritten samples from the HMBD-v1 dataset (Balaha et al., 2021) and generated 10 synthetic images using four open-source TTFs: Noto Naskh Arabic, Amiri, Lateef, and Harmattan.

All images were grayscale, center-cropped, and resized to 128×128 pixels, resulting in 100 images per character (300 total). The full dataset is publicly available at (Shrivastava, 2025)

Combined Datasets

To evaluate unified conditional diffusion models across scripts, we designed two combined datasets. All images were grayscale 128×128 px. Character and script labels were encoded via T5 embeddings (see Section 6).

Dataset C1: Uniform Synthetic Glyphs (Three Scripts)

This experiment used 1500 synthetic glyphs (100 per character) from five characters in each of three scripts. English (500): A, B, W, D, G., Devanagari (500)**:** क (*ka*), ल (*la*), ॐ (*om*), ग (*ga*), म (*ma*), and Arabic (500)**:** *faa* (ف), *qaaf* (ق), *ghayn* (غ), *laam* (ل), *meem* (م). All glyphs were generated using a single Noto-family font per script to ensure typographic uniformity.

Dataset C2: Diverse Source Glyphs (Four Scripts)

C2 aggregated data from earlier experiments, totaling 1000 images across four scripts. English (300)**:** A, B, G – 100 images each from varied fonts (see English dataset section). Devanagari (300): क (*ka*), ल (*la*), ग (*ga*) – 100 images each from diverse fonts (see Section 5). Arabic (300)**:** *faa*, *qaaf*, *meem* – 90 handwritten + 10 augmented synthetic images per letter (see Section 5.4). Mayan (100)**:** The glyph "mam," with 100 augmented samples. C2 introduces greater intra-script diversity and mixed real/synthetic content, in contrast to the uniform synthetic nature of C1.

6 Experimental Setup

We evaluated an unconditional Denoising Diffusion Probabilistic Model (DDPM) (Ho et al., 2020) with a U-Net backbone on Devanagari character generation. The model was trained to recover clean images from noise without using class labels or prompts—learning each character purely from its visual structure.

Training followed the standard DDPM procedure with $T = 1000$ diffusion steps and a linear noise schedule. Models were trained for up to 300 epochs, with early stopping based on convergence.

Training Hyperparameters

For single-character tasks (e.g., "la"), we used 305 training images and trained for 300 epochs (90K steps). For larger datasets (e.g., multi-character or 256px images), 100 epochs sufficed due to the greater data volume. We used a batch size of 32 for 128px images, and 16 for 256px to fit memory constraints. All models were trained using Adam (Kingma and Ba, 2014) with learning rate 1×10^{-4} and default β values ($\beta_1 = 0.9$, $\beta_2 = 0.999$), minimizing MSE loss between predicted and target noise.

Sampling was performed using the standard DDPM procedure over 1000 steps.

Mayan We conducted two primary experiments using the diffusion model.

Experiment 1 We trained the model using 100 images for 1 alphabet for two different durations, 100 epochs, a short training run to establish a baseline. 800 epochs, a long training run to observe convergence behavior.

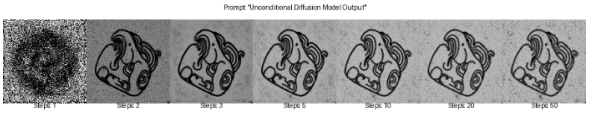

Figure 4: Generated glyph outputs for the Mayan character "mam" (meaning "grandfather") after 800 epochs of training.

Experiment 2 We used the 105 augmented images representing five distinct alphabets to train the same model for 800 epochs. The aim was to evaluate whether data diversity through augmentation could compensate for the limited size of the dataset.

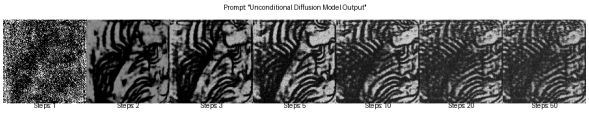

Figure 5: Generated glyph outputs for 5 Mayan characters after 800 epochs of training.

Arabic Script Experiments

To evaluate diffusion model performance on Arabic, we conducted two focused experiments using the letter *faa* (ف). Both used the U-Net DDPM setup and hyperparameters outlined earlier.

AR1: Learning Progression. A model was trained for 800 epochs on 100 images (90 handwritten samples from HMBD-v1 (Balaha et al., 2021) and 10 synthetic glyphs). The synthetic images, kept unaugmented, served as canonical references. Outputs were evaluated at epochs 100, 200, 300, 500, and 800.

AR2: Effect of Minor Dataset Increase. To assess sensitivity to training set size, we trained a second model for 200 epochs using 110 images (adding 10 handwritten samples to AR1's dataset). Outputs were compared at epochs 100, 150, and 200 using MSE and KL divergence metrics.

Experimental Setup for Combined Conditional Models

We also trained conditional diffusion models on combined multi-script datasets using the 'UNet2DConditionModel' from Hugging Face Diffusers (HuggingFace, 2023; von Platen et al., 2022). These models were conditioned on script and character information via text embeddings.

Model Architecture

The model operated on 128×128 grayscale images ('in_channels=1', 'out_channels=1') with four downsampling and upsampling stages ('block_out_channels = (128, 256, 512, 512)', 'layers_per_block = 2'). Textual conditioning was enabled via cross-attention layers ('CrossAttnDownBlock2D', 'CrossAttnUpBlock2D'), with embeddings passed using 'addition_embed_type="text"' and 'cross_attention_dim=512' (matching the T5-small encoder).

Text Conditioning with T5

For conditional generation, we encoded character labels (e.g., "A", क, ف, "MAM") using the 'google-t5/t5-small' model (Raffel et al., 2020). These embeddings guided the U-Net via cross-attention during denoising.

Training: Experiment C1 (Uniform Glyphs). The model was trained on the C1 dataset (1500 synthetic glyphs across three scripts and five characters per script; see Section 5.4) for 200 epochs. We used DDPM training with a DDPMScheduler, AdamW optimizer, and a cosine LR schedule with warmup.

Training: Experiment C2 (Diverse Glyphs). Using the same architecture, we trained on the C2 dataset (1000 glyphs from four scripts with real and synthetic samples; see Section 5.4) for 800 epochs. Checkpoints were saved every 25 epochs to monitor progression and cross-script learning.

Unless otherwise noted, hyperparameters (batch size, learning rate, optimizer) matched those in the unconditional model setup. For inference, we used the DPM Solver Multistep Scheduler.

7 Results and Analysis

7.1 Devanagari

To assess whether diffusion models can learn to generate high-fidelity Devanagari characters, we trained on 305 images of the character "la" (*la*) rendered in different fonts. The model consistently

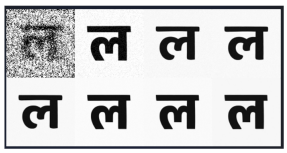

Figure 6: Progressive generation of the Devanagari character "ल" via diffusion.

produced accurate outputs capturing the core structure of "la," including the horizontal header line and relative proportions of its components. Generated samples appeared as plausible variations rather than replicas of any one font, indicating successful generalization across styles.

A common observation was the emergence of *averaged* glyphs—outputs exhibited intermediate stylistic features (e.g., line thickness) reflective of training data diversity. This regression-to-the-mean behavior is consistent with known properties of generative models and suggests that diffusion models interpolate between seen styles rather than inventing new ones.

Effect of Dataset Size

We evaluated how the number of training examples per character affects generation quality. With only 10 images of क (*ka*), the model captured the basic shape but showed inconsistencies—some outputs overfit to specific fonts, while others lacked detail. Increasing the dataset to 50 or 100 images significantly improved quality, with 100 samples yielding clean and consistent glyphs without excessive training. In contrast, longer training on very small datasets led to memorization rather than generalization. These results indicate that data diversity is more important than epoch count for stable learning, with diminishing returns beyond a few hundred fonts.

Effect of Image Resolution

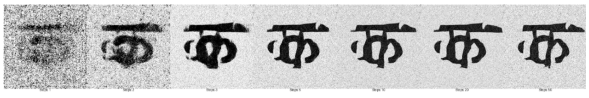

Figure 7: Denoising progression for the Devanagari character क at 128×128 resolution shows limited generalization and blurred outputs due to training on only 5 images.

We compared models trained on 128×128 and 256×256 Devanagari images to assess resolution impact. While 256px offers finer detail, it in-

creases model complexity and data needs. Without sufficient data, it yielded marginal or worse results, highlighting limited benefits in low-data settings.

Figure 8: Improved generation of क at 256×256 resolution with clearer strokes, enabled by training on a larger dataset.

Effect of Image Resolution

While 256px models captured finer details, quality gains over 128px were marginal—especially with limited data, where outputs were often fuzzier and less consistent. With 100–300 samples, 256px models improved visually but remained structurally similar to 128px. Upsampling 128px generations yielded comparable results, suggesting that higher resolution offers only incremental benefits without additional data or model capacity.

English

We used the same model architecture on different fonts of the letter “g” in the English language to experiment with dataset size and the number of epochs. Figure 9 shows a few of the fonts used in the training dataset:

The goal was to check what is the smallest dataset size and the smallest number of epochs required to train a model that can generate the letter “g” with high fidelity. First we attempted to train the model with a dataset size of 100 images (128x128 px) for 100 epochs. The results of this experiment are shown in Figure 10.

The model was not able to learn the shape of letter “g”. Since 100 images did not yield a good result, reducing the dataset size or number of epochs would not have improved the results further. In subsequent experiments, the dataset size was kept the same at 100 images, and epochs were increased to

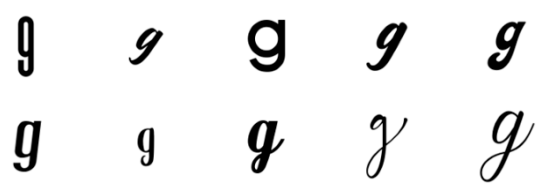

Figure 9: Samples from the ”g” dataset used in training the diffusion model. Total 100 images of different fonts were used.

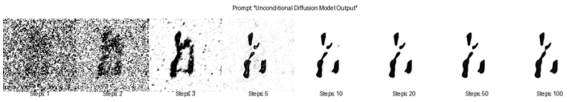

Figure 10: Results of the diffusion model trained with 100 images for 100 epochs.

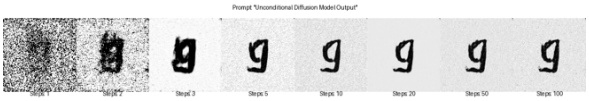

Figure 11: Results of the diffusion model trained with 100 images for 200 epochs.

200. The results of this experiments is shown in Figures 11.

Training for 200 epochs showed notable improvement over 100, enabling the model to capture the basic structure of the English character “g,” though edge roughness persisted. Extending training to 300 and 600 epochs (Figures 12 and 13) further improved output quality, with 600 epochs yielding high-fidelity, well-defined glyphs. These results demonstrate that, even with a fixed dataset size, longer training enhances generation quality. The English dataset’s font diversity also aided generalization, unlike the more limited Devanagari dataset, which may require additional data or epochs to achieve similar fidelity—highlighting the role of dataset diversity in learning efficiency and output quality.

Mayan

In Experiment 1, training for 800 epochs on 100 original glyphs yielded clear outputs resembling authentic symbols. In Experiment 2, despite high number of epochs for training, outputs remained noisy. This suggests that model needs more runs and data diversity (5 characters) increased the cost of training.

Arabic

Experiments with Arabic script, primarily on *faa* (ف), revealed distinct learning dynamics and highlighted the script’s sensitivity to training duration

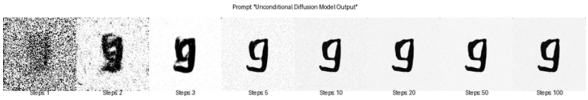

Figure 12: Results of the diffusion model trained with 100 images for 300 epochs.

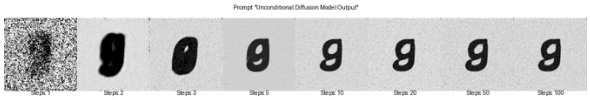

Figure 13: Results of the diffusion model trained with 100 images for 600 epochs

and dataset composition.

Qualitative Learning Progression (Experiment AR1). Generation of *faa* (ف) from the 100-image dataset (90 HMBD-v1 handwritten, 10 canonical synthetic) showed slow but steady refinement over 800 epochs (Figure 17). Initially, at 100 epochs, outputs were predominantly noise with no discernible character structure. The main circular body and tail began forming correctly at 300 epochs. Continued training to 500 epochs improved stroke coherence. However, consistent generation of a well-formed *faa* (ف), including its vital dot, was only achieved after approximately 800 epochs. This extended training underscores the learning challenge for this character, particularly the late emergence of the nuqta, suggesting that fine, semantically critical details require extensive training or more targeted data.

Effect of Minor Dataset Size Increase (Experiment AR2). Experiment AR2 examined the impact of increasing the *faa* (ف) dataset to 110 images (100 HMBD-v1, 10 synthetic), with the model trained for 200 epochs and compared against AR1 at early stages (100, 150, and 200 epochs). Qualitatively, the 110-image dataset produced visibly clearer and better-formed glyphs at these checkpoints than the 100-image set (Figure 14). For instance, at 200 epochs, AR2 glyphs showed improved definition and reduced noise. This was corroborated by Mean Squared Error (MSE) loss and Kullback-Leibler (KL) divergence metrics, which indicated a more favorable training trajectory for the larger dataset. This aligns with findings for Devanagari and RQ2, reinforcing that even minor data increases can enhance learning efficiency and output quality.

Figure 14: Qualitative comparison for *faa* (ف) generation at 200 epochs. Left: Trained on 100 images (Exp. AR1). Right: Trained on 110 images (Exp. AR2). The additional 10 handwritten samples improve clarity.

Results and Analysis of Combined Conditional Experiments

Our combined multi-script experiments, which utilized a conditional diffusion model architecture with T5 textual embeddings, aimed to assess the model's ability to generate characters from multiple scripts simultaneously under different data conditions.

Experiment C1: Uniform Synthetic Glyphs (Three Scripts). Training the conditional model on the C1 dataset (1500 uniform synthetic images across English, Devanagari, and Arabic; 5 characters each) for 200 epochs provided initial insights into cross-script learning with minimal intra-script font variance. By 200 epochs, the model successfully generated four of the five specified English characters (A, B, D, G) with good fidelity, indicating that English, was learned most readily (representative samples in Figure 15, top row). A significant observation was the tendency for both Devanagari and Arabic characters to converge towards the Devanagari letter ॐ (*om*).(Figure 15, middle and bottom rows).

This outcome suggests that even with textual conditioning, certain visually dominant or perhaps more easily representable characters in the shared latent space (like ॐ (*om*) in this dataset) can overpower the generation for other characters from different scripts, especially with relatively short training (200 epochs) and a dataset composed entirely of clean, uniform synthetic glyphs.

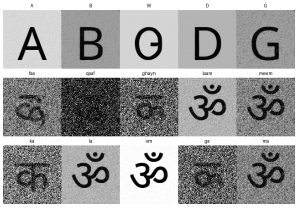

Figure 15: Sample outputs from Experiment C1 for prompted English (top), Arabic (middle), and Devanagari (bottom) characters.

Experiment C2: Diverse Source Glyphs (Four Scripts). Experiment C2 involved training the same conditional architecture for an extended 800 epochs on a more challenging dataset of 1000 images from diverse sources. The details are in Section 5.4). Key observations (referencing Figure 16) include:

- English**:** The three target characters (A, B, G) were consistently generated with high fi-

delity, reinforcing English as the most readily learned script by this model configuration.

- Devanagari (Hindi): Results were mixed. The character क (*ka*) was generated accurately. However, other targeted Devanagari characters, such as ल (*la*) and ग (*ga*), often failed to generalize correctly, frequently collapsing to forms resembling क (*ka*) or other simpler Devanagari structures, even after 800 epochs.
- Arabic**:** Showed notable improvement compared to the C1 outcomes and its own behavior at earlier epochs in C2. The characters *faa* and *qaaf* were generated with good structural accuracy and clear distinction from Devanagari forms by 800 epochs. However, the character *meem* (م) still exhibited a tendency to be misshapen or to converge towards forms with features reminiscent of the Devanagari क (*ka*).
- Mayan**:** The single 'mam' glyph demonstrated promising, albeit partial, generation. Outputs often captured key structural components or resembled "half a face" of the target glyph. This indicates the model was learning some complex features but struggled with complete reconstruction, likely due to the extreme low-resource nature (100 examples of one intricate glyph) and high visual complexity, even within an 800-epoch combined training run.

The C2 results suggest that while the conditional model can handle significant data diversity, challenges such as script-feature overlap (e.g., between certain Arabic and Devanagari characters), insufficient or less distinct representation for some characters within a script (e.g., some Devanagari characters beyond क), and extreme data scarcity for complex logographies (Mayan) remain significant hurdles. Textual conditioning clearly aids in directing generation, but the model's learned visual feature space can still exhibit confusions, especially when characters from different scripts share underlying visual primitives.

8 Discussion

Our findings from both individual script training and the combined multi-script conditional diffusion modeling help us answer our research questions RQ1, RQ2 and RQ3. We observe that diffusion models can generate high quality characters.

Figure 16: Sample outputs from Experiment C2 (diverse source glyphs, 800 epochs). Top row: English (A, B, G). Second row: Arabic (*faa*, *qaaf*, *meem*). Third row: Devanagari (*ka*, *la*, *ga*). Bottom right: Mayan ('mam'). Note the varied success, with good English, improved Arabic *faa*/*qaaf*, but challenges for some Devanagari characters and Arabic *meem*, and partial Mayan generation.

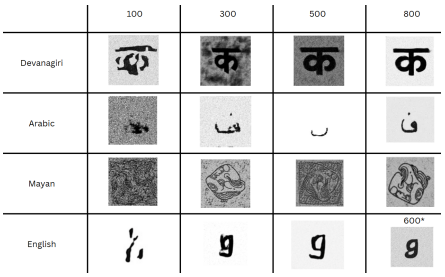

Figure 17: Character generation across scripts (Devanagari, Arabic, Mayan, English) at increasing training stages (100–800 epochs/samples). Outputs improve in clarity and accuracy over time, with simpler scripts converging faster and complex ones like Mayan requiring more training for structural fidelity.

Furthermore we also see observe the influence of data characteristics and training paradigms (RQ2), and demonstrate their potential as tools for comparative analysis of script learnability (RQ3). This research also highlights key challenges in achieving robust, universal multilingual generation, particularly when employing unified architectures. A visual summary of comparative generation quality is presented in Figure 17.

RQ1: Feasibility of High-Quality Symbol Generation. The U-Net based diffusion model architecture, when trained on individual scripts, generally demonstrated its capability to learn and generate high-quality representations of both alphabetic (English; elements of Devanagari, Arabic) and more complex logographic characters (Mayan 'mam' glyph; Devanagari conjuncts; complete Arabic letter forms). High-fidelity generation was observed for characters such as English 'g', Devanagari ला (*la*), Arabic *faa* (ف), and the Mayan 'mam' glyph, contingent upon sufficient training epochs (typically 500-800, varying by script) and appropriate datasets (see results in Section 7.1 and corresponding individual script result sections).

This confirms that a common diffusion architecture can, in principle, adapt to a wide range of symbol structures. The conditional models in combined experiments (C1, C2) further supported this by producing recognizable, script-specific characters when prompted, although generation quality and stability were notably dependent on the specific script and dataset composition within the combined training (Section 7.1).

RQ2: Impact of Data Characteristics and Training. Dataset properties and training duration significantly influenced generation quality and learning efficiency. *Data Size:* Increased data volumes correlated with improved results. This was evident in Devanagari experiments and for Arabic *faa* (ف) (Experiment AR2), where a modest addition of 10 handwritten samples enhanced glyph clarity and training metrics at earlier epochs. These observations show the benefit of larger datasets for refining character generation, particularly with variable handwritten or stylistically diverse script data. *Data Composition (Source and Diversity):* The nature of training data directly impacted character generation quality. For Devanagari, diverse synthetic fonts yielded good quality and an "average" stylistic output. Similar observations were also made when running experiment AR2. Increasing the dataset size by 10 images resulted in higher quality image generation *Training Duration:* Longer training consistently improved quality for individual scripts, up to practical limits. Scripts involving more complex characters or those with scarcer data, such as Arabic and Mayan, particularly benefited from extended training. The observed progressive refinement (e.g., from noise to detailed structure for Arabic *faa*) is characteristic of diffusion model learning. *Image Resolution:* Devanagari experiments (128px vs. 256px) suggested that higher resolution did not substantially improve perceived quality without a corresponding increase in data, implying that for simpler glyphs or in low-data scenarios, 128px resolution offers an effective balance of detail and learnability.

RQ3: Diffusion Models for Cross-Script Comparison and Challenges in Unified Architectures.

The varied learning trajectories across scripts highlight diffusion models as effective tools for comparing script learnability. English was learned with the greatest ease, followed by Devanagari, aided by diverse synthetic fonts. Arabic posed challenges in rendering diacritics and handling handwritten variability, while Mayan glyphs—due to their logographic complexity and data scarcity—were the hardest to model. Conditional experiments (C1 and C2) revealed challenges in building a universal symbol generator, with T5-based conditioning showing promise but also exposing issues like script confusion and feature entanglement. Robust multilingual generation likely requires more balanced, script-specific datasets and refined conditioning or architectural strategies to promote feature disentanglement. Future work should explore targeted data augmentation and conditioning techniques to address these limitations and better understand symbol-specific learning difficulty.

In summary, diffusion models are effective generative tools for individual scripts. However, creating a single, unified multilingual model that performs optimally across diverse and potentially overlapping scripts introduces significant data and modeling challenges. Our study provides a foundation for future research aimed at developing more robust and versatile multilingual systems.

9 Conclusion

Diffusion models can generate multilingual characters effectively with sufficient data and training. Attention layers help capture global character structure. However, resolution scaling and data diversity remain limiting factors.

Limitations

This study has several limitations that constrain generalizability and cross-script performance. First, while Arabic results are presented, the script lacks the same level of Unicode diversity and font standardization as English. For English, we had access to a wide variety of fonts and Unicode-compliant data, whereas for other scripts such as Devanagari and Arabic, font diversity and character coverage were limited.

Mayan glyphs present an even greater challenge. Unlike modern scripts, they lack standardized digital resources and font libraries. Identifying a reliable dataset was difficult, and the resulting small training set restricted the model's ability to learn detailed logographic patterns.

Our experiments also did not include noisy or handwritten inputs, which limits applicability to real-world tasks such as OCR or degraded script reconstruction. Including such data would be crucial for improving robustness and generalization across

writing styles.

Finally, diffusion models are computationally demanding. Although we utilized a Colab A100 instance with 80 GB RAM, our initial training schedule of 100 epochs proved insufficient. Extending training to 800 epochs improved performance but limited the scope for broader experimentation. Building a more generalizable, multilingual diffusion model would require larger datasets and access to more powerful hardware.

Ethical Considerations

This work involves modeling scripts with cultural, religious, or historical significance, such as Devanagari and Mayan -requiring sensitivity to potential misuse or misrepresentation. The generative nature of the model raises concerns around forgery and counterfeit reproduction, especially for official or sacred texts. However, all data used in this study is synthetically generated, and no personally identifiable or private information was involved.

References

Samaneh Azadi, Matthew Fisher, Vladimir Kim, Zhaowen Wang, Eli Shechtman, and Trevor Darrell. 2018. Multi-content gan for few-shot font style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Hossam Magdy Balaha, Hesham Arafat Ali, Mohamed Saraya, and Mahmoud Badawy. 2021. HMBD-v1: Handwritten modern arabic benchmark, version 1. https://github.com/HossamBalaha/HMBD-v1.

Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a nash equilibrium. In *Advances in Neural Information Processing Systems*.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*.

HuggingFace. 2023. Diffusers library. https://huggingface.co/docs/diffusers.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

MayaGlyphs.org. 2025. Mayan characters dataset. https://mayaglyphs.org/. Accessed: May 2025.

Alexander Quinn Nichol and Prafulla Dhariwal. 2021. Improved denoising diffusion probabilistic models. *International Conference on Machine Learning*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, and 1 others. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. 2018. Assessing generative models via precision and recall. In *Advances in Neural Information Processing Systems*.

Aditya Shrivastava. 2025. Multiscript glyph images (v1). https://huggingface.co/datasets/adishri/multiscript_glyph_images_v1. Accessed: May 2025.

Mayank Pratap Singh. 2025. Devanagari characters image dataset. https://huggingface.co/datasets/Mayank022/Devanagari-Characters-Image.

Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. 2022. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers.