DreamBook - Final Project Report

Raksha Ramesh Aria Xingni Shi

1. Title and Summary

This project introduces an intelligent diary analysis and visualization platform. Users can upload handwritten diary entries, audio recordings, or text directly, which are digitized through Google Cloud Vision and speech-to-text APIs. The extracted text is then summarized and analyzed with the Gemini API to identify key entities, themes, and emotions. A fine-tuned Stable Diffusion model (using DreamBooth and LoRA) generates personalized visuals that reflect the diary's content. A Gradio interface allows users to upload entries and enjoy AI-generated illustrations.

2. Goals and Target Audience

Goals

The primary goal of this project is to create an intelligent diary platform that seamlessly transforms handwritten/ spoken entries into personalized visual stories. The key objectives include:

- **Accurate Handwriting Recognition:** To extract handwritten text from images in various languages using the Google Cloud Vision API, aiming for high accuracy in character recognition.
- **Efficient Summarization:** To provide meaningful summaries of the diary entries using the Gemini API, focusing on both content and emotional context.
- **Visual Storytelling:** To generate personalized illustrations based on diary content using Stable Diffusion, enhancing the user's diary experience through engaging visuals.
- **User-Friendly Interface:** To create an intuitive interface where users can easily upload images/audio files, view summarized content, and view the AI-generated visual output.
- **Enhanced Emotional and Entity Visualization**: To help users reflect on their personal experiences through accurate emotion and entity recognition, providing deeper insights into their diaries.

Target Audience

This platform is designed for individuals who want to maintain digital diaries but seek more engaging, personalized experiences beyond just text. Specific target groups include:

- **Journal Enthusiasts:** People who keep daily or occasional diaries and want to add a creative element to their journaling through AI-generated content.
- **Mental Health and Wellness Users:** Individuals interested in reflecting on their emotional state through diary entries and visuals, offering a therapeutic experience.
- **Creative Writers and Visual Artists:** Writers and artists who want to convert their written ideas into visual stories, exploring the intersection of text and image generation.

By aligning the project's goals with the needs of these diverse user groups, we aim to provide a meaningful, interactive, and innovative journaling experience.

3. Pipeline and Baseline Expectations

1. **Handwriting Recognition**: *Google Cloud Vision API*: Extract handwritten text from images in multiple languages and convert to machine-readable text.
2. **Natural Language Processing**: *Gemini API*: Summarize text, identify named entities, main themes, and emotions.
3. **Image Analysis**: *Gemini 1.0 Pro Vision Model*: Analyze the diary writer's appearance from images.
4. **Image Generation**: *Stable Diffusion Model*: Generate personalized images based on diary content and writer's description.
5. **User Interface**: *Gradio*: Image Upload, Audio Input, Text Display, Generated Images Display

4. Plan to Improve the Baseline

- **Additional Input Sources**: **Audio Input**: Extend functionality to process audio inputs and convert them into text for further use in the system.
- **Enhancing Text Recognition Quality**
  Challenge: Poor quality images and spelling errors.
  Pre-processing: Apply contrast enhancement and noise reduction to improve image clarity before text recognition.
  Post-processing: Use a spell-checker to correct human errors after recognition.
- **Natural Language Processing** (Gemini API): Enhance text summarization by incorporating additional models like GPT-4 to generate dialogues and improve the depth of narrative storytelling. Perform extensive prompt engineering so that the main concepts of the text is identified and converted into prompts of the required format by the diffusion model.
- **Image Analysis and Generation**(Stable Diffusion + DreamBooth + LoRA):
  1. **DreamBooth**

     Purpose

     Customizes generative models to recognize and generate specific concepts, such as a unique character.

     Inputs

     - A small set of images of the target concept.
     - A textual prompt with a unique identifier.

Process

(a) Converts the prompt into text embeddings that capture semantic meanings.
(b) Adds varying levels of noise to the input images.
(c) Trains the model to iteratively denoise heavily noised images using the text embeddings, enhancing the model's ability to recreate the specific concept accurately.

Outcome

Improves the model's understanding and representation of the targeted concept through comprehensive retraining.

2. **Low-Rank Adaptation (LoRA)**

Purpose

Efficiently fine-tunes Stable Diffusion models, such as stable-diffusion-v1.4, to incorporate new concepts.

Approach

- Introduces and trains only a few new layers within the cross-attention layers that link image data with textual prompts.
- Avoids adjusting the entire model's gradients, maintaining the core structure.

Benefits

- Enables the model to recognize new words as unique concepts without full retraining.
- Offers a scalable and resource-efficient solution compared to comprehensive methods like DreamBooth.

3. **Comparison**

- **DreamBooth**: Provides thorough customization by retraining the entire model, which can be resource-intensive.
- **LoRA**: Delivers a more efficient fine-tuning method by targeting specific layers, preserving the model's fundamental architecture and reducing computational demands.

Both DreamBooth and LoRA enhance generative models' ability to generate customized content, with LoRA offering a streamlined and scalable alternative to the more exhaustive retraining required by DreamBooth.

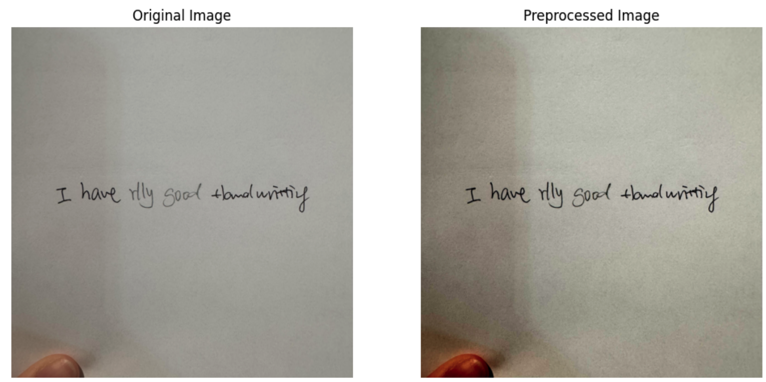
5. Preliminary Experiments and Results

5.1. Audio input

The Google Speech-to-Text API is utilized to convert an audio file into text. An M4A file is first converted to WAV format to ensure compatibility with the API.
Test case:
Audio content: it's quite ironic because I'm a scientist but I'm very addicted to psychic and you know like magic stuff
Transcribed Text: it's quite ironic because I'm a scientist but I'm very addicted to sidekick and you know like magic stuff The ***Word Error Rate (WER)*** is then calculated to compare the ground truth text with the transcribed output. In this case, a WER of **0.053** was observed, indicating the transcription process had a low error rate.

5.2. Input image enhancement

Contrast Enhancement: The image contrast was improved using the **Pillow** library to make the text more distinct.
Noise Reduction: **OpenCV** was applied for noise reduction, in particular Gaussian blur technique was tested to smooth the image, which gave a really great result.
Test Case: The handwriting consisted of both ink and pencil, where there was a noticeable difference in the darkness of the text. After processing, the legibility of the lighter pencil markings was significantly improved, and the distinction between the text and the background became much clearer.



5.3. Google Cloud Vision API

5.3.1 Test Case 1 (English)


```
Text detected:
Milk thistle makes you strong
```

5.4. Gemini API for Text Summarization

Initial Prompt

Input:

> Date: April 25, 2024
>
> Today was one of those days where everything seemed to go sideways from the moment I woke up. I overslept and missed my morning meeting, which set a stressful tone for the rest of the day. At lunch, I had a heart-to-heart conversation with Sarah about our upcoming project, and her insights were incredibly helpful. In the afternoon, I managed to catch up on some overdue reports, which gave me a sense of accomplishment. Despite the rocky start, ending the day with a relaxing walk in the park helped me regain my peace.

Output:

1. **Summary**: The diary entry describes a day filled with challenges and eventual success, starting with a stressful morning, followed by a productive meeting and a sense of accomplishment in the afternoon. The day ends with a peaceful walk in the park.
2. **Main Character**: The diary writer, whose perspective is presented throughout the entry.
3. **Theme**: Resilience and overcoming challenges.
4. **Emotion**: Initially stress and frustration, shifting to peace and contentment.

5. **Storyline (for a comic)**:
    1. **Panel 1**: Writer wakes up late, rushing, missing the meeting.
    2. **Panel 2**: Overwhelmed at desk with paperwork.
    3. **Panel 3**: Lunch with Sarah, productive conversation.
    4. **Panel 4**: Working on reports, feeling accomplished.
    5. **Panel 5**: Walking in the park, enjoying sunset.
    6. **Panel 6**: Reflecting in bed, feeling calm.

This prompt ended up being a little too vague and generic. The fine-tuned diffusion model requires first, only the prompt, and secondly, in the following format "A picture of CharacterName..."

After a lot of prompt engineering efforts, the finalize prompt was this:

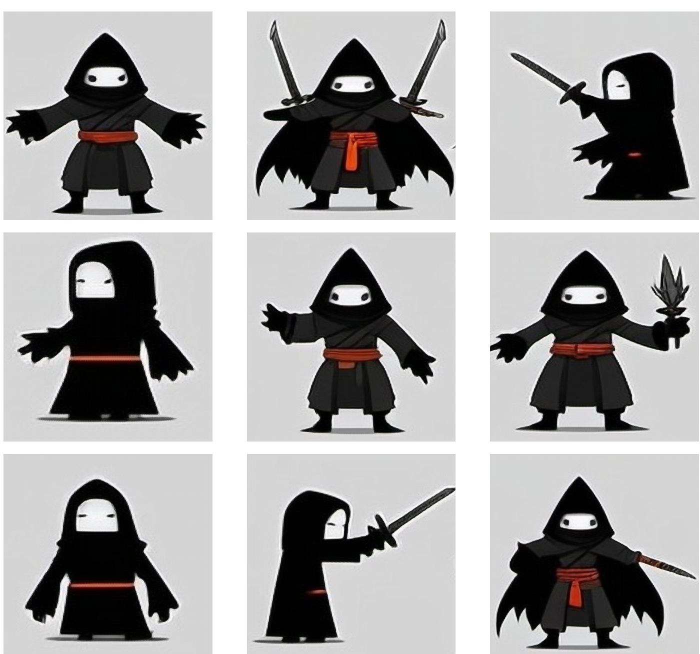
Final Prompt:

> prompt = Please analyze the following diary entry and understand the events that take place in the day and give me A sequence of 9 events or narrative that can be used to create a comic that represents the day depicted in the diary entry. Make sure the 9 events are in seperate lines. Make sure there is no numbering of the events. All the events should be of the following format: "A picture of FinalDude" followed by the action he's doing.
>
> Diary Entry: {DiaryEntry}

5.5. Finetuning Stable Diffution using DreamBooth + LoRA

We tested a lot of different characters, with different number of epochs for fine tuning. The following gave the best results:

Input Data for Fine-Tuning the *stable-diffusion-xl-base-1.0* model :



Hyper-parameters used for Fine-tuning:

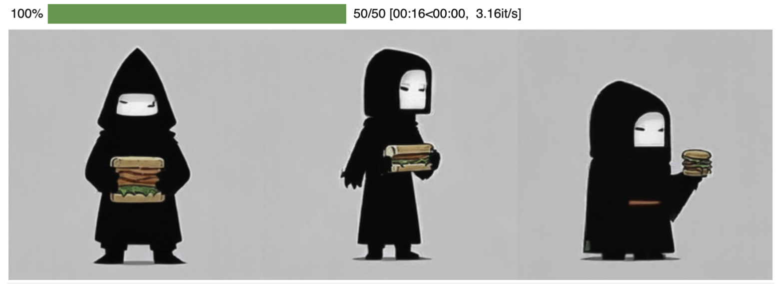
1. **pretrainedModelName**: stabilityai/stable-diffusion-xl-base-1.0
2. **pretrainedVaeModelName**: madebyollin/sdxl-vae-fp16-fix
3. **mixed_precision**: fp16
4. **instance_prompt**: FinalDude
5. **resolution**: 512
6. **train_batch_size**: 1
7. **gradient_accumulation_steps**: 2
8. **gradient_checkpointing**: Enabled
9. **learning_rate**: 1×10^{-4}
10. **lr_scheduler**: constant
11. **enable_xformers_memory_efficient_attention**: Enabled
12. **use_8bit_adam**: Enabled
13. **max_train_steps**: 4000
14. **checkpointing_steps**: 100
15. **seed**: 0

Outputs:

```
prompt = "A picture of FinalDude eating a sandwich"

image = pipe(prompt=prompt, num_inference_steps=50, num_images_per_prompt = 3)
image_grid(image.images, 1, 3)
```

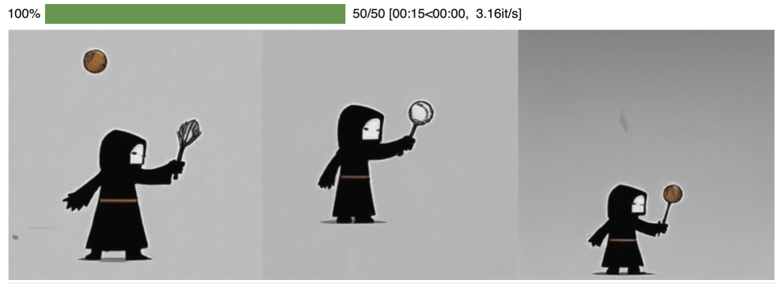
100% 50/50 [00:16<00:00, 3.16it/s]



```
prompt = "A picture of FinalDude playing catch with his friend"

image = pipe(prompt=prompt, num_inference_steps=50, num_images_per_prompt = 3)
image_grid(image.images, 1, 3)
```

100% 50/50 [00:15<00:00, 3.16it/s]



```
prompt = "A picture of FinalDude going for a walk in the forest"

image = pipe(prompt=prompt, num_inference_steps=50, num_images_per_prompt = 3)
image_grid(image.images, 1, 3)
```

100% 50/50 [00:15<00:00, 3.15it/s]

```python
prompt = "A picture of FinalDude wearing a unicorn onesie"

image = pipe(prompt=prompt, num_inference_steps=50, num_images_per_prompt = 3)
image_grid(image.images, 1, 3)
```

100% 50/50 [00:15<00:00, 3.15it/s]

6. User Interface

Gradio is used for application setup. The interface is designed to allow seamless interaction with 3 distinct flows: **Audio to Comic**, **Image to Comic**, **Text to Comic**

6.1. Audio to Comic

- **Input:**
  - Audio file upload (supported format: .m4a).
  - A number input to specify the number of inference steps (range: 1 to 100).
- **Output:**
  - A textbox displaying the transcribed text from the audio.
  - The generated 3x3 comic

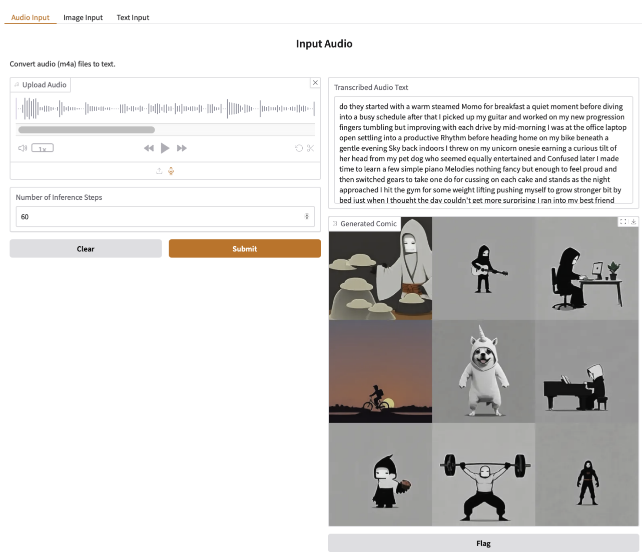


Figure 1: Application's Interface (Audio).

6.2. Image to Comic

- **Input:**
  - Image file upload.
  - A number input to specify the number of inference steps (range: 1 to 100).
- **Output:**
  - Preprocessed Image.
  - A textbox displaying the transcribed text from the image.
  - The generated 3x3 comic.

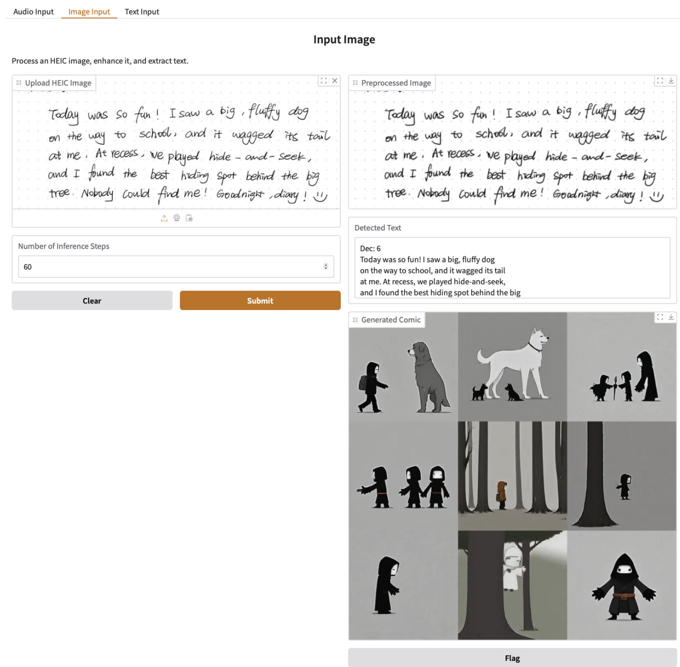


Figure 2: Application's Interface (Image).

6.3. Text to Comic

Text Input:

- **Input:**
  - Text box to input text directly to the interface.
  - A number input to specify the number of inference steps (range: 1 to 100).
- **Output:**
  - The generated 3x3 comic

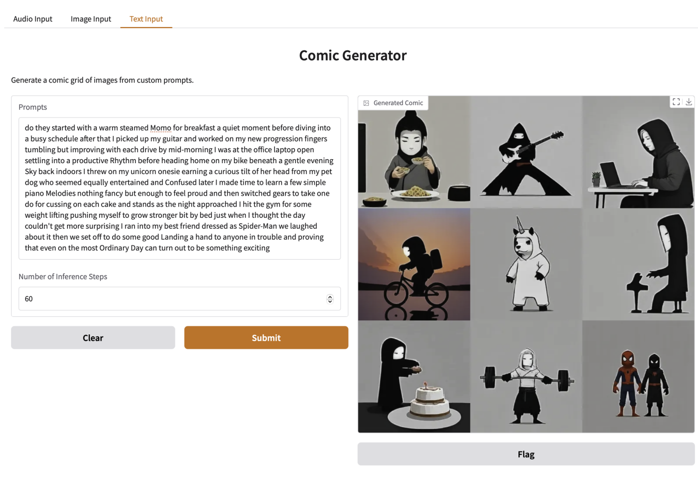


Figure 3: Application's Interface (Text).

7. Project timeline, Milestones, Work Distribution

Project timeline

1. **By Midterm submission (Nov 7th):** Experiment and implement the enhancements proposed and finalize the final

pipeline. Have an MVP up and running.
2. **By Preliminary Final Report Submission (Nov 26th):** Have a basic end-to-end description of the pipeline, each component fully integrated with the next.
3. **By Final Report (Dec 10th):** Any minor modifications and adjustments based on some user feedback and polishing up the project.

Milestones

1. Each component of the pipeline works independently.
2. Integration of multiple components of the project.
3. Modifications and improvements to individual components.
4. User interface and end-to-end testing.
5. Final results and demo.

Work Distribution

1. **Handwriting Recognition**: Aria
2. **Audio to Text**: Aria
3. **Natural Language Processing**: Raksha
4. **Image Generation**: Raksha
5. **User Interface**: Aria & Raksha

8. Conclusion

This project introduces a platform that transforms multi-languages handwritten diary and audio entries into comics using Google Cloud Vision, Gemini APIs, Pillows, OpenCV, Stable Diffusion and, DreamBooth.

9. Future Work

- **Expanded Multimodal Integration:** Incorporate richer media types, such as short animated sequences or background music, to enhance the user's diary experience.
- **Enhanced Language and Handwriting Support:** Support a wider range of languages, dialects, and complex handwritten scripts while improving OCR for cursive writing and low-quality scans.
- **Adaptive Personalization:** Enable the model to learn and evolve with the user's stylistic preferences, ensuring visuals and narratives become increasingly tailored over time.
- **Context-Aware Visual Generation:** Integrate external contextual data (e.g., user's location, season, or sensor inputs) to produce visuals that reflect not just the diary content but also the writer's environment.
- **Improved Privacy and Security:** Implement stronger encryption, on-device processing, and enhanced permission controls to ensure user data remains secure and confidential.
- **Scalability and Performance Optimization:** Optimize inference times, load balancing, and caching to support larger user bases and ensure smoother performance.

10. Learnings

- **Complexities of API Integration:** Integrating multiple APIs (Google Cloud Vision, Speech-to-Text, Gemini, Stable Diffusion) requires handling varied data formats, authentication, and error management.
- **Significance of Preprocessing and Prompt Engineering:** Effective image preprocessing and well-crafted prompts greatly influenced OCR accuracy and the coherence of generated comics.
- **Balancing Performance and Quality:** Finding the right trade-off between computational efficiency (LoRA) and high-fidelity results (DreamBooth) was a key insight.
- **User-Centric Design:** Ensuring a user-friendly interface and incorporating user feedback early improved the overall utility and appeal of the platform.
- **Cross-Disciplinary Collaboration:** Specialization in different components (OCR, NLP, Image Generation) benefited the project, but frequent communication and integration testing were essential.
- **Value of Continuous Experimentation:** Iterative refinement, prompt tweaking, and parameter tuning proved crucial for overcoming unexpected challenges and improving output quality.

11. Link to Drive

This Link to Drive has the code and Demo video for this project: `https://drive.google.com/drive/folders/1H7ZsgMwg2jOFaDkrm2rViOKkIZO2cqV3?usp=sharing`