Major Project report of

Emotion Aware AI Writing Assistant With Dynamic Visual Storytelling Support

In partial fulfillment of the requirements for the degree of

Bachelor of Technology

In

Computer Science Engineering

(Artificial intelligence & Machine learning)

Submitted By

Vilasagaram Pavani (22B81A6628)

Manchana Rajeshwari (22B81A6630)

Pocham Sushmitha (22B81A6652)

CVR COLLEGE OF ENGINEERING
MANGALPALLI, R.R. DIST.
INSTITUTION IN PURSUIT OF EXCELLENCE

Department of Artificial intelligence

& Machine learning

(Emerging Technologies)

CVR COLLEGE OF ENGINEERING

(UGC Autonomous Institution)

Affiliated to JNTU Hyderabad

Vastunagar, Mangalpalli (V), Ibrahimpatnam (M),

Ranga Reddy (Dist.), Hyderabad – 501510, Telangana State

CERTIFICATE

This is to certify that the seminar topic entitled **Emotion Aware AI Writing Assistant With Dynamic Visual Storytelling Support** being submitted by **'Vilasagaram Pavani','Manchana Rajeshwari','Pocham Sushmitha'** student of **IV B.Tech I Sem in Computer Science Engineering (Artificial intelligence & Machine learning),** CVR College of Engineering, Hyderabad, Telangana State, for partial fulfillment of the requirement for the award of Bachelor of Technology in **Computer Science Engineering (Artificial intelligence & Machine learning)** discipline.

ABSTRACT

This project aims to develop an Emotion-Aware AI Writing Assistant with Dynamic Visual Storytelling Support, an innovative AI-driven system that enhances the creative writing process by deeply understanding and responding to the emotional undertones within a user’s narrative.Unlike conventional writing tools that primarily focus on grammar, spelling, or basic stylistic suggestions, this assistant goes further by leveraging transformer-based NLP models such as BERT, RoBERTa, and GPT to analyze the emotional tone, sentiment, and intent behind the text. It provides real-time, context-sensitive suggestions that help users refine dialogue, plot development, and writing style in alignment with the desired emotional impact—whether it's joy, fear, suspense, or sorrow. The assistant enables writers to consciously shape the emotional pacing of their narratives, making their stories more authentic, immersive, and impactful.

Beyond textual support, the system integrates a dynamic visual storytelling module powered by text-to-image transformer models such as DALL·E and Stable Diffusion, which generates adaptive illustrations, mood boards, and character visuals that evolve with the emotional flow of the story. An interactive interface displays a visual emotional arc or timeline, helping users track emotional progression and make intentional adjustments to maintain consistency or experiment with narrative moods. By combining advanced natural language understanding with emotional visualization, this project aims to provide a multimodal storytelling environment ideal for writers, screenwriters, educators, students, and therapists. The result is a powerful, emotion-aware writing companion that fosters creativity, enhances emotional expression, and transforms storytelling into a more engaging, personalized, and expressive process.

INTRODUCTION

In recent years, the integration of artificial intelligence into creative domains has unlocked new possibilities for human expression, storytelling, and emotional engagement. One such innovation is the Emotion-Aware AI Writing Assistant with Dynamic Visual Storytelling, a system that bridges natural language understanding, emotional intelligence, and multimodal generation to create a unique, interactive storytelling experience.

Traditional writing assistants focus primarily on grammar correction, auto-completion, or idea suggestion — often ignoring the emotional tone behind user input. However, emotions play a vital role in how stories are written, understood, and felt. This project aims to address that gap by introducing an AI-powered system that not only detects the emotion behind a user's sentence but also uses it as a basis to generate both a contextually relevant story and a realistic visual representation of that story.

The system works in three main stages: emotion detection, story generation, and image creation. The input is a single user-provided sentence, from which the system identifies the dominant emotion using a pre-trained emotion classification model. Based on this emotion, the assistant generates a short story continuation using a lightweight language model, ensuring that the generated text maintains both narrative coherence and emotional consistency. To enhance the storytelling experience further, the system also generates a visual image using diffusion-based models, effectively translating emotional and narrative cues into artwork

By blending emotion AI, natural language generation, and visual storytelling, this project showcases how AI can enhance creativity, emotional literacy, and interactive storytelling in powerful and engaging ways.

1.1 Problem Statement

Despite the growing availability of AI-powered writing tools, most existing systems lack the ability to understand and respond to the emotional context of user input. They focus on grammatical correctness or idea generation without considering how emotions influence storytelling, tone, and creativity. Moreover, current tools often separate text and visual creativity, limiting the immersive potential of storytelling. Writers and users seeking emotionally relevant, vivid content must rely on multiple tools — some of which are costly or require API keys.

There is a need for a free, unified, emotion-aware storytelling system that can:

- Detect the user's emotional intent,
- Generate meaningful story continuations, and
- Produce visual content aligned with the emotional and narrative tone.

1.2 Problem Objectives

- To detect emotional context from a user-input sentence using a pre-trained, open-source emotion classification model.
- To generate a story continuation that is contextually and emotionally relevant to the input sentence using an open-source language model.
- To create a realistic image based on the detected emotion and generated story using a text-to-image generation model (e.g., Stable Diffusion).
- To integrate all components into a user-friendly interface using Streamlit, allowing real-time interaction and multimodal output.
- To ensure the entire system is cost-free, using only open-source models and tools that do not require API keys or paid access.
- To support creative writing, education, and therapeutic use, by enabling users to explore emotions through storytelling and visual expression.

LITERATURE SURVEY

2.1 Existing Work

1.Rule-Based Writing Assistants:

Early AI writing tools such as Grammarly or Pro Writing Aid focus on grammar correction, style improvement, and vocabulary suggestions. They use predefined linguistic rules and simple statistical methods but lack semantic or emotional understanding.

2. Basic Text Generation Models:

Models like GPT-2 and GPT-Neo have been used in creative writing tools to generate continuations of user-provided text. These models produce syntactically correct stories but do not adapt well to emotional context without fine-tuning.

3. Emotion Detection in NLP:

Emotion classification from text has been studied using machine learning models trained on datasets like Go Emotions, Emo Bank, and Twitter datasets. Models such as CardiffNLP/twitter-roberta-base-emotion provide state-of-the-art performance on short-text emotion detection.

4. Multimodal AI Projects:

Some advanced platforms attempt to merge text and visual AI, such as OpenAI's DALL·E and Mid Journey. These generate images from textual prompts but do not include emotional context detection as part of the pipeline.

5. Story Visualization Tools:

Projects like Image-to-Story (Visual Storytelling) and Story-to-Image Mapping are being explored in academic research, aiming to connect narrative content with matching visuals using deep learning methods.

2.2 Limitations of Existing Work

1. Lack of Emotion Awareness in Text Generation:

Most writing tools do not consider the emotional tone of the input. This leads to emotionally mismatched or generic story outputs that fail to resonate deeply with users.

2. Separation of Text and Image Generation:

Current models like ChatGPT (for writing) and DALL·E (for images) are not inherently connected. There's no emotional bridge between generated text and images unless the user manually provides descriptive prompts.

3. Limited Accessibility and Cost Barriers:

Many high-performing models require API keys, subscriptions, or paid services, making them inaccessible to students or small developers. They also depend heavily on cloud-based APIs.

4. No Unified Platform for Emotion + Text + Image:

Existing platforms rarely combine all three components — emotion detection, emotion-based storytelling, and image generation — into a single, free, and deployable application.

5. Over-reliance on Predefined Prompts:

Some AI art tools require detailed prompt engineering skills. Users unfamiliar with prompt crafting struggle to generate meaningful and coherent visual outputs.

SOFTWARE & HARDWARE SPECIFICATIONS

3.1 Software Requirements

Programming Language

- **Python 3.8+**

Used for emotion detection, story generation, image generation, and backend integration.

IDEs / Development Platforms

- Google Colab (Preferred for free GPU access)
- Jupyter Notebook
- Visual Studio Code (VS Code)
- PyCharm (optional)

Libraries & Tools

- **NLP & Emotion Detection** : transformers, torch, Cardiff NLP models
- **Text Generation** : transformers, auto-gpt, llama-cpp, sentence piece
- **Image Generation :** diffusers, stable-diffusion, PIL, safe tensors
- **Frontend UI** : Streamlit
- **Public Hosting** : pyngrok (for public URL in Colab)
- **Utilities** : pandas, numpy, requests, io, base64

Model Sources

- **HuggingFace Transformers Hub :** For emotion detection and open-source LLMs like Mistral, Phi-2.
- **HuggingFace Diffusers :** For Stable Diffusion image generation.

3.2 Hardware Requirements

- **System Specifications:** Minimum 8GB RAM, 256GB SSD
- **Operating System:** Windows 10/11, Linux (Ubuntu), or macOS
- **Processor:** Minimum Intel i5 (10th Gen)
- **GPU:** NVIDIA GTX 1050 or higher (for small-scale image generation)

PROPOSED SYSTEM DESIGN

4.1 Proposed Methods

The proposed system is designed to take a user’s input sentence, detect its emotional tone, generate an emotion-aware story continuation, and create a realistic visual that reflects the same emotion and story context. The solution uses a modular pipeline to maintain flexibility and scalability.

1.Emotion Detection

- Load a pre-trained emotion classification model from Hugging Face (Cardiff NLP/twitter-roberta-base-emotion).
- Preprocess the input sentence (tokenization, padding).
- Predict the dominant emotion (e.g., joy, sadness, fear, anger, love, etc.).

2. Prompt Engineering for Story Generation

- Use the detected emotion and original input to create a prompt for the language model.
- Example: *“Continue this sad story: ‘She stood alone in the rain…’”*
- Ensure prompts are emotionally aligned and grammatically structured for better results.

3. Story Generation

- Load a small, open-source language model such as: Mistral-7B, Phi-2, or quantized versions via llama-cpp.
- Feed the prompt to the model.
- Generate a short, coherent story continuation that reflects both the emotion and context.

4. Image Generation

- Construct a descriptive image prompt from:
    - The original sentence
    - The generated story
    - The detected emotion
- Use **Stable Diffusion (via diffusers)** to generate a realistic image.
- Display the image using PIL within Streamlit.

5. Web Interface with Streamlit

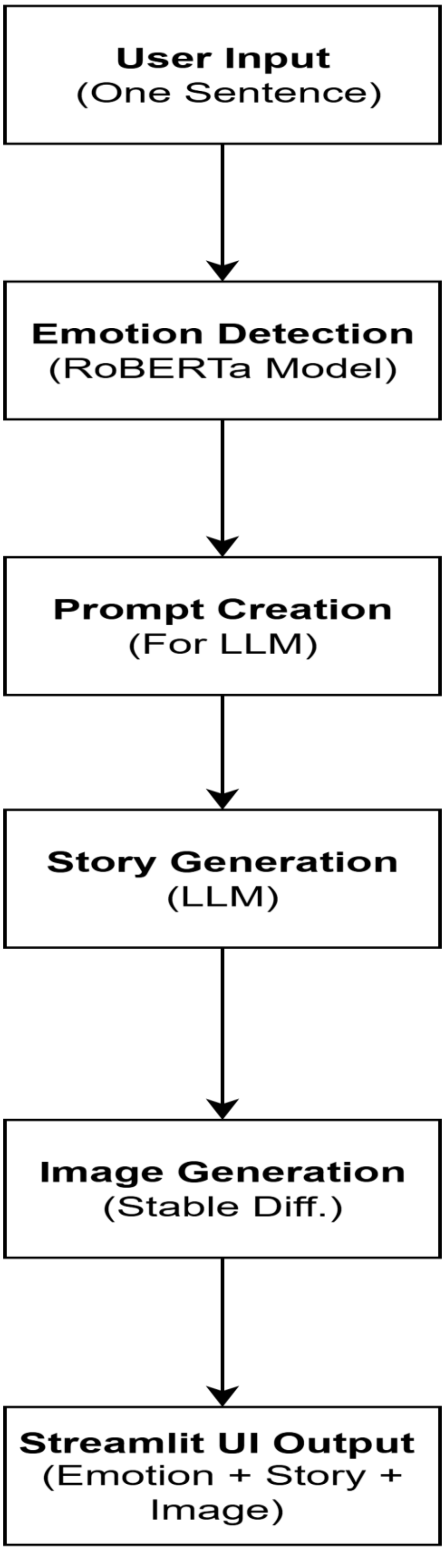
- Build a fully interactive Streamlit app with:
  - Text input box
  - Display of detected emotion
  - Generated story output
  - Rendered image display
- Use pyngrok to deploy in Google Colab with a public URL.

6. Execution Environment

- Use Google Colab to ensure the tool runs even on low-end machines.
- Install necessary dependencies and provide interface through the browser.
- All operations handled via Python with no external APIs required.

4.2 Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) represents the logical flow of data in the Emotion-Aware AI Writing Assistant with Dynamic Visual Storytelling system. It highlights how a user's input sentence is processed through multiple AI modules, including emotion detection, story generation, and image generation, before producing the final multimodal output.


User Input
(One Sentence)
Emotion Detection
(RoBERTa Model)
Prompt Creation
(For LLM)
Story Generation
(LLM)
Image Generation
(Stable Diff.)
Streamlit UI Output
(Emotion + Story + Image)

Components of the Proposed System

1.User Interface

The system begins with a web-based user interface developed using Streamlit. Here, the user enters a single sentence, which serves as the input prompt for the system. The interface provides input fields and dynamically displays the detected emotion, generated story continuation, and corresponding image.

2. Emotion Detection Module

Once the input sentence is submitted, it is passed through a pre-trained emotion classification model (CardiffNLP/twitter-roberta-base-emotion) loaded via Hugging Face's transformers library. The model processes the sentence and outputs the dominant emotion, such as *joy*, *anger*, *sadness*, *fear*, or *love*.

3. Prompt Engineering and Story Generation

The detected emotion and original user input are used to formulate a tailored prompt. This prompt is passed to a language generation model such as Phi-2, Mistral-7B, or any compatible open-source LLM. The model generates a story continuation that aligns both contextually and emotionally with the input.

4. Image Generation Module

A descriptive prompt combining the story theme and detected emotion is sent to a Stable Diffusion model (via the diffusers library). This model generates a realistic AI image that visually represents the emotional tone and content of the story.

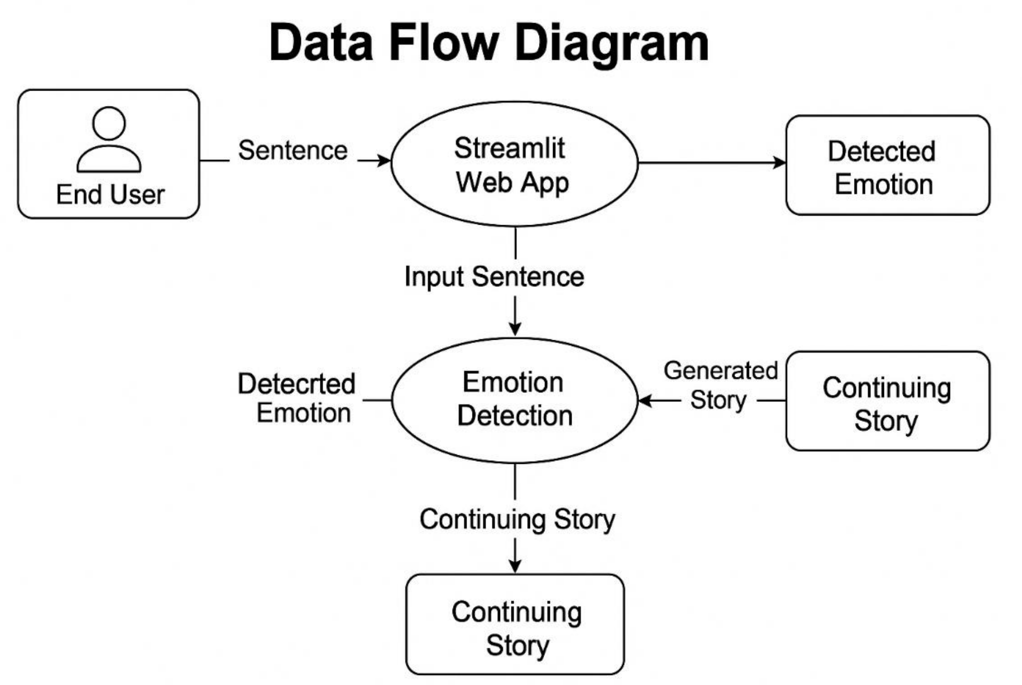
5. Output Rendering

The final output, including:

- The detected emotion,
- The generated story, and
- The corresponding AI image, is rendered and displayed in real-time on the Streamlit interface. This creates a multimodal storytelling experience for the user — seamlessly blending natural language understanding, creative writing, and visual art.

4.3 System Architecture

The system architecture diagram provides a high-level overview of the components involved in the Emotion-Aware AI Writing Assistant with Dynamic Visual Storytelling system. It outlines how various modules interact to process user input, detect emotion, generate emotionally aligned story continuations, and create realistic visuals — all in real time.



Components:

1.End User

The user interacts with the system through a simple, intuitive web interface built using Streamlit. They input a sentence or a short prompt, which is used as the emotional and narrative seed for the story and image generation process.

2. Streamlit Web App

This serves as the frontend of the application. It captures user input, displays output (detected emotion, story, and image), and handles user interaction.
It also manages communication with backend modules including emotion detection, story generation, and image creation.

3. Emotion Detection Module

Once the user submits the sentence, it is passed to a pre-trained transformer-based emotion classification model (CardiffNLP/twitter-roberta-base-emotion) loaded via Hugging Face. This module identifies the dominant emotion (e.g., joy, sadness, anger, fear, love) expressed in the input text. The result is used as emotional context for downstream story and image generation.

4. Prompt Engineering and Story Generation Module

Using the original user input and the detected emotion, a custom prompt is created for story continuation.This prompt is passed to an open-source language model such as Mistral-7B, Phi-2, or LLaMA.The model generates a short, coherent, emotionally consistent story continuation, designed to reflect the input sentiment and narrative theme.

5. Image Generation Module (Stable Diffusion)

From the user input, detected emotion, and generated story, a descriptive image prompt is created .This prompt is sent to the Stable Diffusion model (accessed via the diffusers library) to produce a realistic image that visually represents the emotion and context of the story. This enables dynamic visual storytelling directly tied to user input.

6. Output Rendering

The final output — including the:

- Detected emotion
- Generated story continuation, and
- AI-generated image — is rendered in the Streamlit web interface. Users can instantly view the full storytelling experience, which combines natural language understanding, emotional intelligence, and visual creativity.

4.4 Technology Description

The Emotion-Aware AI Writing Assistant integrates various modern technologies to process natural language input, detect emotions, generate emotionally relevant stories, and produce matching visuals. Below is a breakdown of the core technologies used in this system:

Python

Python is the primary programming language used to develop the entire pipeline—from text preprocessing to emotion detection, story generation, image generation, and frontend deployment. Its flexibility and vast ecosystem make it ideal for AI-based applications.

Transformers (Hugging Face)

The transformers library from Hugging Face is used to load and run pre-trained NLP models, especially for:

- **Emotion classification** using models like CardiffNLP/twitter-roberta-base-emotion

- **Language generation** using open-source LLMs such as Mistral, Phi-2, or similar

These models enable accurate understanding and continuation of text based on emotional cues.

Pandas & NumPy

Used during text preprocessing and data manipulation to:

- Format input sentences
- Handle tokenized text data
- Clean and structure the data flow through various components

Streamlit

Streamlit is the frontend framework used to build the interactive web interface. It allows users to:

- Input a sentence
- View detected emotion
- Read the AI-generated story
- See the corresponding generated image
  All outputs are displayed in real time in a seamless, user-friendly manner.

Stable Diffusion (via Diffusers)

Stable Diffusion, accessed via the diffusers library, is used for realistic image generation. Based on the user's emotional input and story context, a descriptive prompt is sent to the model, which generates a visual scene matching the tone and emotion.

Model Serialization (Joblib or Torch Save)

Although model training is done offline, the models can be saved using tools like joblib or torch.save() for reuse during deployment, reducing startup time and resource usage in the app.

Google Colab & Pyngrok

The entire system is designed to run in Google Colab, making it accessible without local installations. Pyngrok is used to expose the local Streamlit app to the web, allowing public access via a secure URL - even from a notebook environment.

Jupyter Notebook / VS Code / PyCharm

These Integrated Development Environments (IDEs) were used during development for:

- Writing and testing individual modules
- Debugging emotion detection, story generation, and image generation pipelines
- Integrating the full system into one cohesive application

Future Scope

1.Multilingual Emotion Detection and Story Generation

Extend support to multiple languages to enable users across different regions to interact in their native languages, increasing accessibility and emotional relevance.

2.Voice Input and Speech Output Integration

Add voice-based interaction, enabling users to speak their inputs and receive audio narration of generated stories using text-to-speech (TTS) systems. This makes the tool more inclusive and interactive.

3.Emotion-Driven Character Personalization

Introduce consistent AI-generated characters and avatars that reflect the detected emotions visually and narratively, improving engagement in storytelling.

4.User Profile-Based Personalization

Customize story tone, emotion sensitivity, and visual style based on user preferences or past interactions to create a more personalized storytelling experience.

5.Enhanced Visual Generation with Animation

Move beyond static images by integrating AI video generation or animation tools to bring story scenes to life.

6.Interactive Storytelling with Branching Narratives

Allow users to choose between multiple emotional paths or story continuations (like a choose-your-own-adventure), fostering creativity and interaction.

7.Mental Health and Therapeutic Applications

Utilize the system for emotional expression and self-reflection, potentially collaborating with psychologists to develop therapy-friendly tools for emotional awareness and journaling.

8.Integration with Educational Platforms

Incorporate the assistant into e-learning systems to support emotionally rich story-based learning in language education or creative writing classes.

9.Improved Emotion Detection Accuracy

Use multimodal emotion recognition (text, voice tone, facial expression) for more accurate emotional analysis in real-time user interactions.

10.Offline or Low-Resource Model Support

Optimize the system to work offline or with limited internet and hardware resources, making it usable in rural and low-income settings.