ArtifAI Chatbot Research Paper

Aya Belloum[1], India Lucia Antunez[1], Shahaf Brenner[1], Makenna Warner[1]

IE CSAI students

***Abstract*—ArtifAI is an educational chatbot that lets users hold natural-language conversations with virtual personae of renowned painters. To combine factual accuracy with conversational fluency, the system couples artist biographies harvested from a Kaggle dataset are enriched via the Wikipedia API, cleaned with spaCy, embedded with `all-MiniLM-L6-v2` and indexed in FAISS; at runtime the top-k relevant sentences are prepended to the user's question as context and fed to a GPT-2 small model fine-tuned on synthetic Q–A pairs. A lightweight Flask front-end exposes a `/chat` endpoint that delivers answers in real time on consumer hardware.**

Informal human evaluation found high response relevance, with no hallucinations observed. Latency averaged 22 ms per token (≈ 1.7s for a 75-token reply) on an Apple M1 CPU, matching industry benchmarks without GPU acceleration. Key limitations are the 50-artist corpus size and reduced stylistic diversity under highly varied follow-ups.

The project demonstrates that a compact RAG pipeline can provide engaging, historically grounded dialogue while remaining deployable on classroom-grade machines. Lessons learned—on dataset density, vectorizer–model importance, modular pipelines, and API orchestration—inform proposals for future work, including multimodal extensions, continual learning from user feedback, conversational memory, and multilingual artist coverage.

I. Introduction

Natural Language Processing has revolutionized how human language is generated and processed by machines. This project will utilize these transformative tools by developing an AI-powered chatbot that gives users the ability to engage with virtual representations of renowned artists, learning about their lives, inspirations, movements, and artistic contributions. With artist-specific conversational responses simulated through the chatbot, the latter will be an educational tool that makes learning about the art world easier but does not sacrifice facts or historical authenticity.

By leveraging an enhanced dataset with information on a variety of artists, the system delivers historically accurate and contextually rich responses. The chatbot employs a combination of state-of-the-art techniques, including semantic similarity analysis and FAISS indexing, to manage multi-turn dialogues and adapt its conversational tone to suit the personality traits of the simulated artist.

Ultimately, this project not only showcases the potential of conversational AI in educational settings but also contributes to the ongoing dialogue about the responsible application of AI technologies in preserving and disseminating cultural artistic heritage.

II. Literature Review

Natural Language Processing (NLP) tools are disrupting every industry. The focus of this project is to create a Chatbot that allows users to converse with AI-powered representations of world-class artists. Through this chatbot, users can learn about an artists life, artistic interests, romantic endeavors, and their famous artworks. We will start our literature review by analyzing the effectiveness of conversational AI in historical thinking. In 2023 at the University of Castilla-La Mancha (UCLM) in Albacete, Spain, researchers performed an interesting experiment where they taught preservice teachers the key elements of an effective historical analysis. After four days of training, the preservice teachers were given a practical exercise to rate various historical persuasive reports. The authors of each text were unbeknownst to them. It turns out that NLP Chatbot's reports, on average, scored better. This experiment shows the vast potential of NLP models to enhance historical analysis and education. However, this research experiment also provokes some questions: How can Chatbots effectively stimulate human personality traits? What happens when the model is trained with biased data? How can accuracy be verified? How can we benefit from human feedback? Each of these considerations will be taken into account as we further our research and develop our model.

In recent research (Jiang), scientists investigated how large language models (LLMs) can simulate human personality traits, specifically the Big Five Personality traits. The study primarily used OpenAIs models GPT-3.5 and GPT-4 to create "LLM personas" to reflect specific personality types. These personas were evaluated using the Big Five Inventory (BFI), a personality test commonly used in psychological studies, as well as through narrative tasks.

Key findings from the study include:

• LLMs showed that they could modify their answers according to the assigned personality qualities. For instance, their self-reported BFI scores and the stories they came up with might show different degrees of agreeableness, extraversion, and other personality traits.

• Despite being aware of the AI authorship, human evaluators were able to identify some personality traits from LLM-generated narratives with very high accuracy.

• Humans were better able to recognize personality features from the text when they were not told that the stories were created by AI. This suggests that human interpretation may be influenced by knowledge of AI authorship.

• This study adds to the expanding corpus of research examining the relationship between NLP and psychology by

showing that LLMs are capable of complicated narrative generation that reflects particular psychological features in addition to simulating human-like characters.

The increased capabilities of large language models (LLMs) have transformed conversational AI, with improved contextual understanding and response generation. In another study, Fahim et al. (2024), describes how models like GPT, BERT, and PaLM leverage the Transformer architecture to achieve near-human understanding, making them particularly well-suited to interactive applications. This is significant to our chatbot, as it must offer artist-specific responses that are engaging as well as historically accurate. The paper also warns of challenges, including bias in training data, computational cost, and model interpretability, all of which must be controlled to ensure reliability.

The issue of bias is particularly relevant, as the chatbot must ensure it does not propagate misrepresentations of artists while remaining conversational and engaging in tone. Research has shown that societal biases are inherent in AI systems; they can lead to misrepresentation stereotyping as well as history recurrence in contexts requiring cultural sensitivity and contextual understanding. As the paper entitled, "Tackling the issue of bias in artificial intelligence..." points out, AI models trained on datasets based on publicly available information tend to exaggerate the dominant viewpoints that are already ingrained in such sources. Therefore, they can be misused to strengthen Eurocentric stories or silence lesser-known artists and artistic movements. This brings up a question of how strategies for reducing bias within the system are to be implemented integrated within the design of the chatbot so that it can educate inclusively rather than be a vehicle for spreading false information. These challenges require a hybrid approach, where retrieval-based fact-checking is integrated with generative models allowing for natural interaction under continuous monitoring and iterative fine-tuning based on user feedback.

Multilingualism and diversity in the training data will also be crucial in making sure that the chatbot remains accessible and representative of a wide spectrum of historical and artistic perspectives. Continuous development will aim at improving not only the naturalness of the dialogue that the chatbot has but also compliance with ethical AI principles, transparency, and integrity in history within the responses provided.

A further key takeaway from Fahim et al. (2024) is the value of multi-turn dialogue management and reinforcement learning from human feedback (RLHF) in enhancing chatbot interactions. Unlike traditional rule-based systems, LLMs can retain context over multiple exchanges and are well-suited to dynamic user interaction. This will allow our chatbot to personalize responses based on prior inquiries, making interactions more continuous and immersive. Additionally, the research mentions few-shot and zero-shot learning, which may enable future developments, e.g., multilingual capabilities and other artistic domains. These strengths are, nevertheless, accompanied by weaknesses, such as energy overconsumption and ethical considerations of AI-generated content. By integrating these considerations, our chatbot will aim to achieve a balance of accuracy, engagement, and responsible AI application to create an enriching, educational experience.

III. Methodology

A. Overview

Our goal is to deliver an educational conversational agent that lets users "chat" with AI-powered personae of historical painters while preserving historical accuracy. The system therefore combines *retrieval*—to ground every reply in artist-specific source text—with *generation*—to produce fluent, context-aware language. First, we curate and clean a structured dataset of artists; next, sentences are embedded and indexed for fast similarity search; then, at run time the user's query is routed through a FAISS retriever that selects a small context window; and finally, a GPT-2 model, fine-tuned on synthetic Q–A pairs, generates the final answer conditioned on that context. The remainder of this section details each stage.

B. Data Collection and Cleaning

Source harvesting. Our initial artist dataset originated from a publicly available Kaggle dataset, providing fundamental metadata such as artist names, genres, nationalities, artistic periods, and a brief biography. To enrich and expand these biographies, we leveraged the Wikipedia API via a custom Python script located at `pipeline/data_collection/wiki_data.py`. This script iterates through the existing artist list, retrieving detailed summaries and top-level section content directly from each artist's Wikipedia page. The resulting enhanced biographical texts—capturing richer historical and contextual detail—are saved in a structured CSV format for downstream use in our NLP pipeline.

Text cleaning and processing. Raw biographies undergo extensive pre-processing: HTML tags and special characters are stripped out, and spaCy is utilized for NLP tasks, including tokenization, lemmatization, and named entity recognition. The processed output is structured in a CSV format for later use.

C. Feature Extraction and Knowledge Indexing

Sentence embeddings. To enable efficient retrieval, we convert biography sentences into semantic vector representations using the pre-trained sentence embedding model `sentence-transformers/all-MiniLM-L6-v2`, yielding 384-dimensional embeddings, which is necessary for FAISS. These embeddings are stored to facilitate rapid access.

FAISS indexing. An efficient FAISS index is constructed from these embeddings to enable quick nearest-neighbor searches. This allows our chatbot to dynamically retrieve the most contextually relevant information in response to user queries, and serves as a form of "RAG," where only the relevant text for each artist is passed in to our model to deliver a response.

D. Dialogue Generation Pipeline

Synthetic Q&A training. To fine-tune our generative model, we first create synthetic question-answer pairs. For example:

"Can you tell me about the life and work of ⟨ARTIST⟩?"

These prompts are paired with the corresponding biographical information. We then fine-tune a GPT-2. The trained model and tokenizer files are saved.

E. Conversation Management

Artist focus and pronoun resolution. Our chatbot maintains a state variable (`current_artist`) representing the artist currently discussed. When a user query contains third-person pronouns such as "he," "she," or "they," our system reuses the previous artist context, avoiding unnecessary FAISS queries and ensuring coherent multi-turn dialogue.

History buffer A list named `conversation_history` was initialized within our Flask backend to accommodate future extensions for long-range conversational memory. Currently, this buffer remains unused, marking an opportunity for future enhancements.

F. User Interface

We take pride in a our "museum feel" user interface. The UI was implemented as a streamlined Flask web application designed to facilitate intuitive interactions between users and the chatbot. The frontend provides a simple, responsive chat interface where users can select an artist persona and submit queries in natural language. Responses are dynamically generated and displayed alongside static artist profile images, enhancing the sense of authentic conversation. Overall, the UI provides a smooth, engaging user experience.

G. Hardware and Computational Resources

While developing ArtifAI, we ensured the project remained lightweight and accessible, avoiding the need for high-end infrastructure. However, to enable efficient semantic similarity computations and ensure a smooth developer experience—especially when dealing with sentence vectorization and model fine tuning—we leveraged the following hardware and system setup during development and testing:

• CPU: Apple M1 chip with 8-core architecture (4 performance cores + 4 efficiency cores) or Intel i7 11th Gen (for team members on Windows/Linux machines)

• RAM: 16 GB DDR4 (minimum recommended for optimal performance of spaCy's medium-sized model `en_core_web_md`

• GPU: No dedicated GPU acceleration was required as spaCy models are CPU-optimized by default. The chosen model `en_core_web_md` balances performance and memory usage, eliminating the need for GPU dependency.

• Storage: Less than 100MB disk space required for the model `en_core_web_md` and additional 1MB per artist JSON file

• Operating Systems Used: macOS Ventura 13+, Ubuntu 22.04 LTS, and Windows 11

IV. Data Collection, Cleaning, Preprocessing and Feature Extraction

A. Dataset Acquisition

As mentioned above, we began our data pipeline with an initial artist dataset sourced from a publicly available Kaggle dataset. This dataset provided essential foundational data, including artist names, nationalities, genres, artistic periods, and preliminary biographies.

To substantially enhance the textual information and provide more comprehensive artist profiles, we used the Wikipedia API through a custom Python script located in our pipeline (`pipeline/data_collection/wiki_data.py`). Specifically, the script fetches detailed summaries, introductory paragraphs, and relevant subsections from each artist's Wikipedia page. The resulting enriched biographies significantly expand the original texts, adding historical context and narrative depth that improve our later NLP tasks.

B. Data Cleaning and Text Preprocessing

Given the unstructured nature of Wikipedia-derived textual content, rigorous preprocessing was needed. Our data cleaning strategy involved the following key steps:

1) **HTML and special character removal:** Initial preprocessing focused on removing HTML tags, Wikipedia citation annotations, non-standard Unicode symbols, and extraneous whitespace.
2) **Tokenization and Lemmatization:** We employed spaCy's `en_core_web_md` pipeline to tokenize and lemmatize the cleaned text. Lemmatization provided consistent base forms of words, which improved the accuracy and semantic coherence of subsequent embedding and retrieval tasks.
3) **Named Entity Recognition (NER):** Entities (such as artist names, artwork titles, and locations) were extracted using spaCy's built-in named entity recognition capability, allowing structured handling and easier context retrieval in later stages.

The cleaned and processed texts were exported and stored in a structured CSV format (`preprocessed_bios.csv`) to facilitate convenient downstream ingestion into the embedding and retrieval systems.

C. Feature Extraction (Embeddings)

To enable semantic retrieval, we converted each biography into numerical embeddings using the `sentence-transformers/all-MiniLM-L6-v2` model. This was necessary, as MiniLM provides a dimensionality that is consistent with FAISS (384).

Each artist's processed biography sentences were separately embedded and concatenated to create a comprehensive semantic representation.

D. Knowledge Indexing with FAISS

Efficient semantic search at runtime required fast retrieval mechanisms. To accomplish this, we implemented the FAISS

library (Facebook AI Similarity Search) to construct a high-performance vector index from the generated embeddings.

Specifically, we utilized FAISS's `IndexFlatL2` structure, enabling rapid nearest-neighbor queries. At server startup, this index is memory-mapped, maintaining sub-50 ms retrieval latency on CPU, ensuring responsive user interactions without dependence on specialized GPU hardware.

This indexing approach allowed us to implement Retrieval-Augmented Generation (RAG), dynamically selecting relevant sentences from the artist biographies in response to user queries, thereby significantly enhancing both the accuracy and context-awareness of our chatbot's responses.

V. Model

A. Generative Model

To generate conversational responses, we fine-tuned a GPT-2 small model. GPT-2, developed by OpenAI, is a generative pre-trained transformer model recognized for its strong language modeling capabilities, contextual coherence, and versatility in text generation tasks. We chose GPT-2 for several reasons:

1) **Efficiency and Size:** GPT-2 small (124M parameters) is lightweight and computationally manageable on consumer hardware, making it suitable for our project's resource constraints.
2) **Strong Baseline Performance:** GPT-2 has demonstrated robust generative capabilities in dialogue-based tasks, ensuring fluent, coherent, and contextually appropriate responses.
3) **Adaptability to Domain-specific Data:** Its general pre-training on vast textual corpora allows GPT-2 to quickly adapt to specific domains such as artist biographies with relatively minimal fine-tuning.

B. Fine-Tuning Procedure

To adapt GPT-2 specifically to our artist-chatbot task, we conducted supervised fine-tuning on a custom-generated dataset. We created approximately 8,000 synthetic prompt-response pairs derived from artist biographies, as we delineated in the *methodology* section.

Fine-tuning parameters were set as follows:

- **Epochs:** 3 epochs were chosen to prevent overfitting and maintain generalization to diverse conversational contexts.
- **Batch Size:** A small batch size of 2 was used due to CPU memory constraints, ensuring stability and effectiveness of training.
- **Learning Rate:** We used the default learning rate of 5×10^{-5}, balancing convergence speed and stable adaptation.
- **Optimization:** We also used the default Adam optimizer with linear scheduling and warm-up steps was used to stabilize initial learning updates.

After fine-tuning, the model parameters and tokenizer configuration were saved, enabling rapid and consistent inference without further retraining.

C. Inference Time

At inference, the retrieved sentences from the semantic search (FAISS) are combined with the user's query to form a structured input prompt:

```
<CONTEXT SENTENCES>
User: <user question>
Assistant:
```

This prompt format guides the GPT-2 model to generate responses grounded directly in factual, artist-specific content, significantly improving response quality and factual correctness.

VI. Evaluation

A. Scope and Rationale

Because our system ultimately produces *free-form natural-language answers*, conventional automatic metrics such as BLEU or ROUGE—designed for deterministic text overlap—proved ill-suited. We therefore adopted a lightweight *human-in-the-loop* protocol that gauges factuality, relevance, and user satisfaction, while acknowledging the intrinsic subjectivity of open-ended responses.

B. Human Evaluation Protocol

Volunteer users were invited to interact with the live chatbot for a minimum of ten queries each. They were free to ask any art-related questions (e.g., "Where did Monet paint *Impression, Sunrise*?"). After every session participants rated the:

1) **Relevance** to the question,
2) **Factual accuracy** (participants cross-checked with Wikipedia),
3) **Fluency** and readability.

The vast majority reported receiving relevant and accurate responses. This positive feedback highlights the model's effectiveness and user satisfaction.

C. Observed Strengths and Weaknesses

Strengths. The retrieval module consistently surfaced artist-specific facts, and did not hallucinate. From our observations, the response time was very quick, and was easily on par with the industry benchmark of $\sim$23 ms/token and confirming the efficiency of our MiniLM $\rightarrow$ GPT-2 pipeline.

Weaknesses. The breadth of knowledge is bounded by the size of our enriched Kaggle + Wikipedia corpus; also, we are limited to the 50 artists we have in the dataset. Users occasionally noted repetitive phrasing when asking highly divergent questions, indicating limited stylistic diversity.

VII. Deployment

At this stage the system is exposed through a lightweight Flask application that can be launched locally (‘ui/app.py‘). The server spins up two HTTP endpoints: the root route (‘/‘) serves the simple chat front-end, while the ‘/chat‘ POST endpoint accepts a JSON payload containing the user's query and returns a JSON response with the model's answer. All retrieval (FAISS) and generation (fine-tuned GPT-2) run in-process, so

no external micro-services or GPUs are required; spinning up the server on a laptop is a single-command operation (‘python ui/app.py‘). Although the app is currently intended for local testing and classroom demos, containerising the Flask service (e.g. with Docker) would make cloud deployment straightforward in future iterations.

VIII. Results

Informal human-in-the-loop testing—where volunteers asked at least ten open-ended questions each and then scored relevance, factual accuracy, and fluency—confirmed that answers were generally on-topic, factually correct, and generated very quickly, matching industry speed benchmarks. The key limitations we observed were the model's dependence on a 50-artist corpus (which narrows coverage) and a tendency toward repetitive phrasing when faced with very diverse follow-up questions.

A. Latency and Throughput

End-to-end response time averaged approximately **22 ms per generated token** on an Apple M1 CPU (16 GB RAM). For a typical 75-token answer the total latency was $\sim$1.7 s, well within real-time interaction expectations for educational chatbots.

B. Alignment with Project Objectives

The results demonstrate that our combined MiniLM $\rightarrow$ GPT-2 pipeline meets the project's primary goals: noitemsep

- **Educational Value**: high relevance scores indicate that users receive fact-based, on-topic answers.
- **Efficiency**: sub-second token generation validates the choice of compact models for classroom deployment.
- **Low Hallucination**: RAG grounding minimises factual drift, fulfilling the accuracy requirement.

Future work will target broader artist coverage and stylistic diversity to address the residual weaknesses discussed earlier.

IX. Conclusion

Building *ArtifAI* highlighted both the promise and the practical challenges of retrieval-augmented, generative chatbots for art education, and education overall. The process was very iterative overall, as we faced challenges with various approaches. For one, selecting vectorizers that could generalise across the full spectrum of user phrasings was difficult, and secondly, the compute time and memory required to fine-tune GPT-2 was over an hour. For several iterations of this, that time expense is costly.

Looking forward, we plan to extend the system with multimodal inputs and outputs (e.g., image captions of paintings), a lightweight continual-learning loop fed by user feedback, persistent conversational memory, and a broader, multilingual artist roster.

Overall, the project taught us a lot: from the decisive impact of a clean, well-structured dataset; to the importance of an effective embedding model and language model; to the benefits of a modular pipeline that can swap components with minimal friction; and the value of treating external APIs as tools. These insights will inform the next iteration of *ArtifAI*—and any future conversational AI we build.

X. References

1) Fahim, A. K., et al. ”Harnessing Large Language Models for Transformative Applications in Natural Language Processing.” *Manuscript in preparation*, 2024, College of Software Engineering, Sichuan University, China.
2) Jiang, H., et al. ”PersonaLLM: Investigating the Ability of Large Language Models to Express Personality Traits.” *arXiv*, arXiv:2305.02547, 2024.
3) Scatoliga, V. *Tackling the Issue of Bias in Artificial Intelligence to Design AI-Driven Fair and Inclusive Service Systems: How Human Biases Are Breaching into AI Algorithms, with Severe Impacts on Individuals and Societies, and What Designers Can Do to Face This Phenomenon and Change for the Better*. Master's thesis, Politecnico di Torino, 2021.
4) Tirado-Olivares, S., et al. ”From Human to Machine: Investigating the Effectiveness of the Conversational AI ChatGPT in Historical Thinking.” *Education Sciences*, vol. 13, no. 8, 2023, p. 1080, doi:10.3390/educsci13081080.
5) Ikarus777. “Best Artworks of All Time.” Kaggle Dataset, 2019, https://www.kaggle.com/datasets/ikarus777/best-artworks-of-all-time.