

Report Title: Technical Documentation

ArtConnect AI – AI-Powered Artist Promotion Assistant (POC)

Project: Final Project – ITAI 4373

Prepared for: Joe Fleishman (Client)

Team Name: Creative Intelligence Co. (CIC)

1. Introduction

ArtConnect AI is an AI-Powered Artist Promotion Assistant designed to streamline and optimize the social media engagement process for artist Joe Fleishman. The primary purpose of the system is to automatically identify high-value engagement opportunities from social media conversations (e.g., comments, mentions) and assist the artist in responding effectively.

Assistance for Joe Fleishman:

- **Saves Time:** Automates the laborious process of sifting through thousands of social media interactions.
- **Prioritizes Engagement:** Ranks potential interactions by an "Opportunity Score," ensuring the artist focuses on the most promising leads (e.g., potential buyers, collaborators, or positive media).
- **Maintains Brand Voice:** Generates draft replies that are consistent with the artist's established tone and style.

Proof-of-Concept (POC) Demonstration:

The POC demonstrates the core functionality of the system, focusing on:

- Data ingestion and preprocessing (simulated social data).
- A rule-based Opportunity Scoring Engine.
- A template-based Reply Generator.
- A "Human-in-the-Loop" workflow via a simple dashboard interface.

2. Architecture Overview

The ArtConnect AI system utilizes a modular, layered architecture to ensure separation of concerns, maintainability, and scalability for future enhancements.

Architecture Diagram (Text-based):

[Social Data (Simulated)]

|

v

[1. Data Input Layer]

|

v

[2. Preprocessing Layer]

| (Cleaned, Scored Data)

v

[3. AI Logic Layer (Opportunity Scoring)]

| (Opportunity Score, Rank)

v

[4. Interface Layer (Streamlit Dashboard)] <--> [5. Analytics & Logging Layer]

| (User Action: Approve/Edit/Reject)

v

[External Platform (Simulated Posting)]

Component Explanation:

- **Data Input Layer:** Responsible for ingesting raw data from simulated social media sources (CSV/JSON files), applying initial schema validation, and structuring the data for the next layer.
- **Preprocessing Layer:** Performs cleaning, feature extraction, and transformation on the raw text data, including text normalization, sentiment analysis, and keyword identification.
- **AI Logic Layer:** The core intelligence. Contains the rule-based Opportunity Scoring Engine which calculates a numerical score for each interaction based on predefined factors and weights.
- **Interface Layer (Dashboard):** The user-facing component built with Streamlit. It displays ranked opportunities, suggests draft replies, and facilitates the human-in-the-loop approval workflow.
- **Analytics & Logging Layer:** Records all system actions (data loads, scores), human decisions (approvals, edits), and tracks key performance indicators (KPIs) related to the system's effectiveness and the artist's engagement.

3. Technology Stack

The following technologies were selected for the POC based on their utility, ease of development, and suitability for rapid prototyping.

Technology	Rationale for Selection
------------	-------------------------

Python	Chosen as the primary programming language for its robust ecosystem of data science and AI libraries, and its simplicity for scripting and integration.
Pandas	Essential for data manipulation, cleaning, and processing the structured social media data efficiently (e.g., loading CSVs, running feature engineering).
JSON/CSV	Used to simulate external data feeds (Instagram, Twitter), providing a simple, accessible, and common data format for the POC's input layer.
Streamlit	Selected for rapid development of the interactive web dashboard (Interface Layer), requiring minimal front-end coding expertise.
Rule-based Scoring	Used instead of complex Machine Learning (ML) for the POC to provide transparent, explainable, and easily tunable scoring logic.
Template-based Reply Generation	A simple, effective approach for the POC to ensure replies adhere to a consistent brand voice and tone without needing a sophisticated generative model.

4. Data Input Module

The Data Input Module is responsible for mimicking the ingestion of social media activity.

Simulated Data Feeds:

- **Instagram CSV Simulation:** Represents structured, high-volume data, likely including fields like comment text, commenter ID, post ID, and basic follower counts.
- **Twitter JSON Simulation:** Represents semi-structured data, capturing the complexity of API responses, including fields like tweet text, user profile data, timestamp, and metadata (e.g., retweet count).

Schema (Fields, Example Rows):

A unified internal schema is used after ingestion.

- **Key Fields:** `interaction_id`, `platform`, `timestamp`, `user_handle`, `user_followers`, `text_content`, `sentiment_score` (added in preprocessing), `opportunity_score`.
- **Example Row (Simplified):**
 - `interaction_id`: TWT-5928
 - `platform`: Twitter
 - `timestamp`: 2025-10-25 14:30:00
 - `user_handle`: @ArtLoverMeg
 - `user_followers`: 1200
 - `text_content`: "The technique in this piece is stunning! I'd love to commission something for my gallery."

Data Validation Steps:

- **Completeness Check:** Ensuring critical fields (`interaction_id`, `text_content`, `user_handle`) are present (non-null).
- **Format Validation:** Checking that `timestamp` is a valid datetime format and `user_followers` is a non-negative integer.
- **Source Integrity:** Validating the input file structure (e.g., checking if the JSON can be parsed without error). Invalid records are flagged and logged but not passed to the Preprocessing Module.

5. Preprocessing Module

The Preprocessing Module cleans and transforms the raw text data into structured features suitable for the Opportunity Scoring Engine.

Steps:

- **Text Cleaning:**
 - Removal of special characters, emojis, and URLs.
 - Conversion of all text to lowercase.
 - Tokenization and removal of common English stop words (e.g., 'the', 'a', 'is') to focus on core content.
- **Keyword Extraction:**
 - Identification of pre-defined keywords related to potential opportunities (e.g., "commission," "buy," "gallery," "interview," "collaboration").
 - Counting the frequency of these high-value keywords within the text content.
- **Sentiment Scoring:**
 - Application of a basic lexicon- or rule-based model (e.g., VADER) to assign a numerical sentiment score (e.g., -1.0 to +1.0) to the `text_content`. Only highly positive sentiment is considered a scoring factor.
- **Normalization:**
 - Scaling of numerical features like `user_followers` to a common range (e.g., 0 to 1) to prevent features with larger absolute values from disproportionately influencing the final score.

6. Opportunity Scoring Engine

The core of the AI Logic Layer, the Opportunity Scoring Engine calculates a value from 0 to 100 indicating the quality and urgency of an interaction.

Exact Scoring Factors:

The score is a weighted sum of the following factors:

1. **Keyword Presence (K):** Binary or count-based factor based on high-value keywords (e.g., "commission," "buy").
2. **Sentiment Strength (S):** The magnitude of positive sentiment in the comment.
3. **User Influence (U):** A normalized measure of the user's follower count or profile history.
4. **Recency/Urgency (R):** A factor that decays over time, prioritizing recent interactions.

Weights:

Each factor is assigned a weight based on the artist's priorities:

- Weight for Keyword Presence (\$W_K\$): High (e.g., 0.5)
- Weight for Sentiment Strength (\$W_S\$): Medium (e.g., 0.3)
- Weight for User Influence (\$W_U\$): Low (e.g., 0.15)
- Weight for Recency (\$W_R\$): Low (e.g., 0.05)
- Note: $\sum W = 1.0$

Logic for Generating a 0–100 Opportunity Score:

The raw score is calculated using the formula:

$$\text{Raw Score} = (K \cdot W_K) + (S \cdot W_S) + (U \cdot W_U) + (R \cdot W_R)$$

The final **Opportunity Score** is then scaled and normalized to the 0–100 range:

$$\text{Opportunity Score} = \min(100, \text{Raw Score} \times 100)$$

A key logic rule is a score of 0 is immediately assigned if the sentiment is highly negative or if the text is flagged by safety rules.

Pseudocode for the Algorithm:

FUNCTION Calculate_Opportunity_Score(interaction_data):

 // 1. Feature Extraction (from Preprocessing Layer)

 K = Extract_Keyword_Factor(interaction_data.text_content) // 0-1.0

 S = Extract_Sentiment_Score(interaction_data.text_content) // 0-1.0 (Positive scale)

 U = Normalize_User_Followers(interaction_data.user_followers) // 0-1.0

 R = Calculate_Recency_Factor(interaction_data.timestamp) // 0-1.0

 // 2. Define Weights

 W_K = 0.50

 W_S = 0.30

 W_U = 0.15

 W_R = 0.05

 // 3. Apply Safety Check (e.g., profanity filter, spam check)

 IF Is_Unsafe_Content(interaction_data.text_content):

 RETURN 0

// 4. Calculate Raw Weighted Score

$\text{Raw_Score} = (K * W_K) + (S * W_S) + (U * W_U) + (R * W_R)$

// 5. Final Scaling to 0-100

$\text{Opportunity_Score} = \text{MIN}(100, \text{Raw_Score} * 100)$

RETURN Opportunity_Score

7. Reply Generator Module

The Reply Generator Module creates context-appropriate, brand-compliant draft responses for the artist to review.

Brand-Voice Template System:

- A library of pre-written templates (JSON/YAML format) is maintained, categorized by interaction type (e.g., "Commission Inquiry," "Collaboration Request," "Simple Praise," "Technical Question").
- Templates include placeholders (e.g., [USER_HANDLE], [ARTWORK_TITLE]) which are dynamically filled with data from the interaction record.

Tone Logic:

- The system selects a template based on a combination of the *highest-weighted keyword* and the *sentiment score*.
- For high-score opportunities (e.g., Commission), the tone is professional, appreciative, and includes a clear call to action (e.g., "Please email me at...").
- For simple praise, the tone is grateful and personal (e.g., "Thank you, [USER_HANDLE]!").
- Templates are engineered to align with the artist's known professional yet approachable "brand voice."

Safety Rules:

A dedicated set of rules prevents the generation of inappropriate or risky replies:

- **PII/Privacy:** Never generate a reply that solicits or reveals personal contact information other than the artist's public email/website.
- **Financial/Pricing:** Never include exact pricing or financial negotiations in the draft reply; instead, direct the user to a private channel.
- **Hate Speech/Harassment:** Templates are scrubbed to ensure they are universally

positive and non-inflammatory. The system will default to a neutral, "Thank you for your comment" template if the incoming message's safety rating is ambiguous.

Examples of Safe Replies:

- **For "Commission Inquiry" Keyword:** "Thank you so much for your interest, [USER_HANDLE]! I'd be happy to discuss a commission. Could you please send me an email at [ARTIST_EMAIL] with the details of your idea?"
- **For "Simple Praise":** "I really appreciate your kind words, [USER_HANDLE]! It means a lot to me."

8. Approval Workflow (Human-in-the-loop)

The Human-in-the-loop (HITL) workflow is a critical ethical and compliance safeguard, ensuring the artist retains final control over all public communication. This process occurs in the Interface Layer.

Workflow Actions:

- **Approve:** The artist clicks 'Approve,' and the final reply text (drafted or edited) is immediately logged and passed to a simulated posting module. This is the fastest path for high-confidence, pre-approved responses.
- **Edit:** The artist modifies the generated draft reply. The *original draft* and the *final edited version* are both logged. This provides valuable feedback data to refine future templates.
- **Reject:** The artist dismisses the opportunity, deciding not to reply (or to reply manually outside the system). This action is logged as a "Rejection," providing a signal that the Opportunity Score or Reply Generator may need tuning.

Protection of Ethics and Compliance:

- **Ethical Oversight:** The HITL process explicitly prevents the system from autonomously posting to social media, mitigating the risk of algorithmic errors, tone misinterpretation, or accidental non-compliance (Section 12).
- **Brand Consistency:** Ensures every communication maintains the artist's authentic voice and professional standards, safeguarding their brand reputation.
- **Legal/Financial Compliance:** The artist, as the subject matter expert, can verify that replies involving business matters (commissions, sales) adhere to all applicable laws and personal business policies before public dissemination.

9. Dashboard (UI/UX)

The Interface Layer is a user-friendly Streamlit dashboard designed for Joe Fleishman to efficiently review and manage opportunities.

Key Components:

- **Opportunity Table (The Inbox):**
 - Displays all ingested interactions, sorted descending by the **Opportunity Score**.
 - Key columns include: **Opportunity Score**, **Platform**, **User Handle**, **Time Since Post**, and a snippet of the **Text Content**.
 - Allows filtering by platform and score range.
- **Reply Panel (The Detail View):**
 - Activated by selecting a row in the Opportunity Table.
 - Shows the full interaction text, the calculated scoring factors (e.g., Keywords detected, Sentiment), and the system-generated draft reply.
 - Contains the **Approve**, **Edit** (with a text box), and **Reject** buttons for the HITL workflow.
- **Analytics Page:**
 - A separate tab or view displaying the Key Performance Indicators (KPIs) and data visualizations from the Analytics Module (Section 10).
 - Provides metrics on system performance and artist engagement over time.
- **Navigation Layout:**
 - A simple, three-section layout: **Opportunity Review** (main workspace), **Analytics**, and **Configuration** (for updating scoring weights and reply templates).

10. Analytics Module

The Analytics Module processes logged data to provide insights into the system's performance and the artist's interaction strategy.

KPIs Tracked:

- **AI Suggestion Approval Rate (ASAR):** The percentage of generated draft replies that were ultimately *Approved* by the artist.
- **Time-to-Reply (TTR):** The average time taken between data ingestion and the artist's action (Approve/Reject).
- **Opportunity Recall:** The percentage of interactions containing a high-value keyword that were correctly assigned an Opportunity Score above a certain threshold (e.g., 75).
- **Platform Breakdown:** Volume of interactions processed and replies sent per social media platform.

How the System Calculates Them:

- KPIs are calculated by querying the logged data in the Analytics & Logging Layer.
- $ASAR = (Total\ Approved / Total\ Opportunities\ Reviewed) \times 100\%$
- TTR is calculated as the mean difference between the **log_timestamp** (System

Action) and the `ingestion_timestamp` (Data Input).

Sample Analytics Outputs (Text-based):

- **System Efficacy Rate:** 85%
- **Top 3 Keywords Detected:** Commission (45), Gallery (12), Collaborate (8)
- **Rejection Analysis:** 4% of high-score opportunities were rejected; primary reason: Ambiguous Intent.
- **TTR (Average):** 4.2 hours

11. Data Flow Summary

The following steps explain the full end-to-end workflow of a single social media interaction through the ArtConnect AI system:

1. **Load Data:** The Data Input Layer consumes a batch of simulated social media data (CSV/JSON), validates the schema, and assigns a unique `interaction_id` and `ingestion_timestamp`.
2. **Preprocess:** The Preprocessing Layer cleans the text, extracts features like keywords, calculates the sentiment score, and normalizes all numerical features (e.g., followers).
3. **Score:** The AI Logic Layer's Opportunity Scoring Engine applies the pre-defined weights and factors to calculate the final 0–100 **Opportunity Score**.
4. **Generate Reply:** The Reply Generator Module uses the primary keyword and sentiment to select an appropriate brand-voice template and fills in placeholders to create a draft reply.
5. **Human Review:** The Interface Layer (Dashboard) displays the interaction, ranked by its Opportunity Score, along with the system-generated draft reply, for the artist's review.
6. **Log Actions:** The Analytics & Logging Layer records the artist's decision (Approve, Edit, or Reject), the final message content, and the `log_timestamp`.
7. **Update Analytics:** The logged data is immediately processed by the Analytics Module to refresh the KPI metrics (e.g., System Efficacy Rate, TTR) displayed on the Analytics Page.

12. Ethical Design Considerations

The system is built on a foundation of responsible AI practices, prioritizing the artist's control and user safety.

- **No Auto-Posting:** The system operates strictly as an **assistant**, requiring explicit human approval for every single public communication. This eliminates the risk of an algorithm causing reputational damage.
- **No Scraping:** The POC is based on *simulated* data. Future production versions will rely solely on approved, official API access with explicit user consent, adhering to platform terms of service.
- **Safe Tone Guidelines:** The template system and safety rules (Section 7) are

engineered to prevent replies that are aggressive, manipulative, or could be misconstrued as hate speech or harassment.

- **Transparency:** The dashboard explicitly shows the user *why* an interaction received a high score (i.e., the detected keywords and scoring factors), making the "AI" decision process transparent and auditable.

13. Limitations of the POC

The current Proof-of-Concept, while effective for demonstrating core logic, has several inherent limitations:

- **Simulated Data:** The system operates entirely on static, pre-formatted files, not real-time, live social media data streams.
- **No ML Scoring:** The core intelligence relies on a static, **rule-based** scoring engine, which cannot adapt or learn from new data patterns or complex language nuances.
- **No API Integration:** There is no connection to external social media APIs for true data ingestion or for automated posting of approved replies. Posting is simulated.
- **Limited Platforms:** The POC only simulates data from Instagram and Twitter, excluding platforms like Facebook, TikTok, or other art-specific communities.

14. Future Technical Improvements

To evolve ArtConnect AI from a POC to a deployable product, the following technical improvements are planned:

- **API Integration:** Implement robust, authenticated connections to social media platform APIs to enable real-time data ingestion and posting of approved replies.
- **ML-based Models:** Replace the rule-based scoring engine with a Machine Learning classification or regression model trained to predict "conversion" or "commission likelihood," providing greater accuracy.
- **Multi-platform Support:** Extend the Data Input Module and Reply Generator to process and categorize data from a wider range of platforms using a standardized, internal data schema.
- **Database & Cloud Deployment:** Migrate data storage from flat files (JSON/CSV) to a scalable cloud database (e.g., PostgreSQL, Firestore) and deploy the Streamlit application on a cloud platform (e.g., AWS, GCP) for 24/7 availability.

15. Appendix (Optional)Data Schema

Field Name	Data Type	Description	Example Value
<code>interaction_id</code>	String	Unique system ID for the	TWT-5928

		interaction.	
platform	String	Source social media platform.	Twitter
timestamp	Datetime	Time of the original interaction.	2025-10-25 14:30:00
user_handle	String	User's public identifier.	@ArtLoverMeg
user_followers	Integer	Follower count of the interacting user.	1200
text_content	String	The full text of the comment/mention.	"...I'd love to commission something..."
keywords_hit	List[String]	Keywords detected by Preprocessing	["commission", "love"]
sentiment_score	Float	Polarity score (-1.0 to 1.0).	0.92
opportunity_s	Integer	Final score (0-	94

core		100).	
------	--	-------	--

Pseudocode (Detailed)

Refer to the pseudocode provided in Section 6: Opportunity Scoring Engine. Table Structures

1. Opportunities Table (Database Mockup)

- `interaction_id` (PK)
- `platform`
- `timestamp`
- `opportunity_score`
- `status` (NEW, REVIEWED, APPROVED, REJECTED)
- `draft_reply`
- `final_reply`

2. Logs & Analytics Table (Database Mockup)

- `log_id` (PK)
- `interaction_id` (FK)
- `action_type` (INGEST, SCORE, APPROVE, REJECT, EDIT)
- `log_timestamp`
- `user_id` (Joe Fleishman)
- `data_change` (JSON object detailing original vs. edited reply, etc.)