Abode Behaviour Simulation: A Dual-Task Framework for Engagement Prediction and Marketing Tweet Generation

GDSC task
Vedavarshith, Shreya, Aditya rai, Rishabh
Adobe Challenge Track

***Abstract*—This report presents a comprehensive solution to the Adobe Behaviour Simulation Challenge, addressing both *Task 1: Behavior Simulation* (predicting tweet engagement via likes) and *Task 2: Content Simulation* (generating realistic marketing tweets). We propose a 3-stage hybrid model for Task 1 and a 6-stage VLM+LLM pipeline for Task 2. Both pipelines are designed for robustness across unseen brands and time periods, evaluated using RMSE, BLEU, ROUGE, and CIDEr on a 300K-sample dataset.**

***Index Terms*—Social Media Analytics, Multimodal AI, Engagement Prediction, Generative AI, Vision-Language Models, LoRA Fine-tuning**

I. Introduction

A. About Adobe Digital Experience Business

Adobe Experience Cloud gives digital businesses everything they need to design and deliver great customer experiences. Adobe solutions integrate best-in-class Adobe products to help its customers tackle pressing business challenges.

1) **Experience-led growth**: Adobe Experience Cloud gives our business clients the insights and tools to create individual and engaging experiences that their customers are looking for — the ones that drive growth for our customers' B2B or B2C business.
2) **Content supply chain**: Adobe's customers can build a scalable and efficient content supply chain with the help of Adobe GenStudio. This powerful solution combines best of content ideation, creation, production, and activation with the powers of generative AI.
3) **Personalization at scale**: Adobe Experience Cloud lets our clients reach every customer with engaging, personalized experiences — right when they need it. Our technology gives them real-time unified data and insights, connected customer journeys, and AI-assisted workflows for one-to-one moments and planned campaigns.
4) **B2B marketing**: Experience Cloud lets our customers deliver account-based experiences that differentiate their brand. Advanced features such as real-time data and account profiles and AI-driven workflows make every customer interaction relevant and let them deliver a seamless digital journey that's directly integrated with the sales process.

B. Background

Timely and targeted content generation is essential to yield desired marketing results for any business. This is carried through higher user engagement which in turn drives sales and broader customer outreach. Scoring high on user engagement KPIs boosts brand credibility, reach and product recall. This also helps marketers strengthen on algorithmic favorability which for a social media company decides on the post that gets propelled based on the engagement it drives.

II. Problem Description

The process of communication is defined by marketing researchers as below: A receiver, upon receiving a message from a sender over a channel, interacts with the message, thereby generating effects (user behavior). Any message is created to serve an end goal. For a marketer, the eventual goal is to get the desired effect (user behavior) i.e. such as likes, comments, shares and purchases, etc.

In this challenge, you will try to solve the problem of behavior simulation (Task-1) and content simulation (Task-2), thereby helping marketers to estimate user engagement on their social media content as well as create content that elicits the desired key performance indicators (KPI) from the audience.

(You can read these papers [1, 2] to understand both the tasks a bit better)

A. Dataset

Brands use Twitter to post marketing content about their products to serve several purposes, including ongoing product campaigns, sales, offers, discounts, brand building, community engagement, etc. User engagement on Twitter is quantified by metrics like user likes, retweets, comments, mentions, follows, clicks on embedded media and links. For this challenge, we have sampled tweets posted in the last five years from Twitter enterprise accounts. Each sample contains tweet ID, company name, username, timestamp, tweet text, media links and user likes.

III. Problem Statement

A. Task 1: Behavior Simulation

- Given the content of a tweet (text, company, username, media URLs, timestamp), the task is to predict its user engagement, measured by likes.
- **Sample IO**

date	content	username	media	inferred company
2019-07-05 11:32:00	Sandy, muddy terrain exploration...	Toyota_Fortuner	[Photo(...)]	toyota

TABLE I
INPUT EXAMPLE FOR TASK 1

Output: `likes: 10`

- Size of train data: 300K samples
- The results will be evaluated under 2 regimes (10K samples each):
  - Predicting likes for tweets of unseen brands (test dataset) but seen time period (training dataset: Jan-2018 to Dec-2020)
  - Predicting likes for tweets of unseen time period (test dataset) but seen brands (training dataset)

Link to Training Dataset Link to the Test Dataset

B. Task 2: Content Simulation

- Given the tweet metadata (company, username, media URL, timestamp), generate the tweet text.
- **Sample IO**

date	likes	username	media	inferred company
2019-07-05 11:32:00	10	Toyota_Fortuner	[Photo(...)]	toyota

TABLE II
INPUT EXAMPLE FOR TASK 2

Output: `content: Sandy, muddy terrain exploration? Yes { we've got you covered. Visit ...`

- Size of train data: 300K samples
- The results will be evaluated under 2 regimes (10K samples each):
  - Predicting tweet text for tweets of unseen brands (test dataset) but seen time period (training dataset: Jan-2018 to Dec-2020)
  - Predicting tweet text for tweets of unseen time period (test dataset) but seen brands (training dataset)

Link to Training Dataset Link to the Test Dataset

IV. Task 1: Behavior Simulation

A. Project Summary: A 2-Stage Hybrid Model for Tweet Engagement Prediction

This document summarizes the iterative modeling process for Task 1: Behavior Simulation, which aims to predict the number of likes (a regression task) for a given tweet.

The core challenge was the nature of the data:

1) **Skewed Distribution**: Most tweets get few likes (0-100), while a tiny fraction go "viral" (10,000+).
2) **The "Exploding Error" Problem**: The evaluation metric, RMSE (Root Mean Square Error), is brutally punished by large errors. This means a single bad guess on a "viral" tweet can ruin the entire model's score.
3) **Complex Data**: The model must understand nuanced text, brand voice, and timing to be successful.

B. Attempt 1: Single-Stage LightGBM + TF-IDF (The Baseline)

- **Model**: A single `LightGBM` Regressor.
- **Target**: `log1p(likes)`. We predicted the logarithm of likes to compress the skewed 0-1M range into a more normal 0-14 range.
- **Key Features**:
  - **Manual Features**: `hour`, `dayofweek`, `text_len`, `word_count`.
  - **Text Features**: TF-IDF (a "word counter").
- **Result**: A very high Real RMSE (3600).
- **Insight / Lesson Learned**: This model failed for two reasons:
  1) **"Exploding Error"**: The model was not very accurate. A small error on the log scale (e.g., guessing `log(likes)=6` when the answer was `log(likes)=2`) would "explode" when converted back, resulting in a prediction of 402 likes for a 10-like tweet.
  2) **"Dumb Features"**: TF-IDF is "dumb." It can count words but cannot understand the meaning or context of a tweet.

C. Attempt 2: LightGBM + BERT Embeddings (The "Smarter" Failure)

- **Model**: `LightGBM` Regressor (same as before).
- **Key Features**:
  - We replaced the "dumb" TF-IDF features with 384 "smart" features from a BERT Transformer.
  - We added "Brand Power" features (`company_avg_likes`, `sentiment`).
- **Result**: Still a very high Real RMSE (2800–3000) and a high Log RMSE (2.15) when tested properly with GroupKFold.
- **Insight / Lesson Learned**: Even with a "genius" set of features, the model was still forced to do two jobs at once (understand text and predict a precise number). The `log1p` "exploding error" problem was still poisoning the Real RMSE score.

D. Attempt 3: The Classifier Approach

We changed the problem entirely. Instead of predicting an exact number, we tried to predict a category.

- **Model**: A Transformer Classifier (`DistilBERT`, later upgraded to `RoBERTa`).
- **Task**: 4-Class Classification (predicting if a tweet would be "Low", "Medium", "High", or "Viral").
- **Result**: A 80% accuracy (F1-score: 0.78).

- **Insight / Lesson Learned**: This was a massive success. It proved:
  1) A Transformer can "sense" the difference between engagement levels.
  2) It was excellent at finding "Low" and "Medium" tweets (F1-score: 0.90 and 0.71).
  3) It was "shy" at predicting "Viral" tweets (low recall). This was because the "brain" was too small and the data was imbalanced.

E. The Final, Winning Pipeline: A 2-Stage "Hybrid" Model

This pipeline combines all our lessons learned. We use a "division of labor," hiring two specialists to do one job each.

Stage 1: The "Expert Critic" (Your RoBERTa Classifier)

- **Model**: `my_best_ROBERTa_model3` (A powerful `RoBERTa-base` Transformer).
- **Job**: To act as the "smart" text-understanding part of the pipeline. It reads the complex tweet and metadata.
- **Output**: An "expert opinion" in the form of four probabilities: `[prob_low, prob_medium, prob_high, prob_viral]`.

Stage 2: The "Master Accountant" (A LightGBM Regressor)

- **Model**: `LightGBM` Regressor (the model from our last notebook).
- **Job**: To take the "expert opinion" from Stage 1 and predict the final number.
- **Features**: It was trained on a "master spreadsheet" that combined:
  1) The simple manual features (`hour`, `dayofweek`, `sentiment`).
  2) The four powerful probability features from the Stage 1 classifier.
- **Result**: This 2-stage model was our most accurate by far, achieving a Log RMSE of `0.7347` (a 16.5% improvement over our previous best).

F. The Final Fix: The "Hybrid-Logic" Inference

This is the final, novel step that solves the "exploding error" problem and gives you the best possible Real RMSE.

When predicting on a new tweet (from the test set):

1) **Run Stage 1 (Classifier)**: Get the 4 probabilities.
2) **Ask One Question**: Is `prob_low > 90%`?
   - **IF YES**: The model is 99% sure this is a "boring" tweet. We STOP. We override the regressor and manually predict a safe, low number (e.g., 50 likes). This makes it impossible to have an "exploding error".
   - **IF NO**: The tweet has potential. We proceed to Stage 2.
3) **Run Stage 2 (Regressor)**: Feed the probabilities and manual features into the `LightGBM` model.
4) **Get Final Answer**: The model gives its highly accurate `log(likes)` prediction (e.g., 9.5), which we convert back to the real number (e.g., 13,350 likes).

This pipeline is the perfect solution: it is fast (runs in minutes), novel (uses a 2-stage hybrid), and highly accurate (uses the Transformer's "brain" and fixes the "exploding error").

G. SOTA Approach: Adding Vision (The Multi-Modal Pipeline)

Our analysis showed that our `ROBERTa` (text-only) classifier still struggled to differentiate the "High" and "Viral" classes perfectly. The logical next step is to incorporate the image data, which is a massive predictor of virality.

However, running an image model on every tweet (including 20,000 test samples) is computationally expensive and slow, which would hurt our "Efficiency of solution" score.

We propose a novel, SOTA-level 3-Stage "Conditional" Pipeline that combines efficiency and power:

1. Stage 1: Fast Triage (The RoBERTa Classifier)

- **Model**: Our existing `my_best_ROBERTa_model3`.
- **Job**: Run on all tweets. It's fast (text-only).
- **Output**: The 4 probabilities.

2. Stage 2: Conditional Image Analysis (The "Vision Expert")

- **Model**: A multi-modal model like `Gemini` or `ViT`.
- **Job**: This model is only activated if the Stage 1 prediction for "High" or "Viral" is above a certain confidence threshold (e.g., `prob_high + prob_viral > 0.3`).
- **Output**: A text caption of the image (e.g., "A high-quality photo of a new product") or a numerical "image embedding."

3. Stage 3: The Final Regressor (The LightGBM Accountant)

- **Model**: Our existing `LightGBM` Regressor.
- **Job**: Predict the final `log(likes)`.
- **Features**: It would be re-trained on an even richer dataset:
  1) Manual Features (`hour`, `sentiment`, ...).
  2) Stage 1 Probabilities (`prob_low`, ...).
  3) **New Vision Feature**: The image caption or embedding (or a "0" if Stage 2 was skipped).

This hybrid approach is the best of all worlds. It is highly efficient by skipping expensive image processing on 80% of the "Low" tweets, while still using the powerful vision signal on the "High" and "Viral" tweets where it matters most. This would be our final step to achieving the lowest possible RMSE.

V. Task 2: Content Simulation

A. Project Goal and Methodology

The project aims to create realistic, effective marketing content by implementing a structured six-stage machine learning process.

1) Multi-Stage Pipeline: The content generation process is divided into the following stages:

1) **Data Cleaning**: Initial preprocessing and standardization of input data.
2) **Image Captioning**: Using a VLM to generate rich, objective descriptions of images.

3) **Training Prompt Preparation**: Engineering custom prompts for the fine-tuning process.
4) **LLM Fine-tuning**: Training the content generation model.
5) **Inference**: Generating the simulated tweet content.
6) **Evaluation**: Measuring the quality and similarity of the generated content.

2) Previous Approach / Alternate Experiment: Before settling on the current two-stage approach (VLM for caption, LLM for tweet), we also attempted a pipeline where the VLM was used to process the Image along with other content (such as username, likes, date, etc.) to directly generate the final tweet, without an intermediate, explicit captioning step. This alternative was explored to see if the VLM could inherently fuse all multimodal data for content generation, but the refined two-step method was ultimately chosen for better control and result quality.

B. Model Architecture and Technology Stack

The system utilizes a dual-model approach, combining specialized models for vision and language tasks.

TABLE III
MODEL ARCHITECTURE OVERVIEW

Component	**Base Model**	**Role**
Vision-Language Model (VLM)	Qwen2-VL-2B-Instruct	Generates detailed image captions. Supports batch processing.
Large Language Model (LLM)	LLaMA 3.2-3B-Instruct	Generates the final marketing tweet content. Fine-tuned using LoRA with 4-bit quantizat

1) Generation Context: The fine-tuned LLaMA model is designed for context-aware generation, considering the following factors for realism:

- Brand identity
- Media content/context (via the generated caption)
- Temporal information
- User engagement metrics (e.g., likes, usernames)

C. Key Project Features and Components

The repository is structured with specific files dedicated to each major functionality:

TABLE IV
REPOSITORY STRUCTURE AND FUNCTIONALITY

File/Directory	**Functionality**
`data_cleaning.py`	Cleans and standardizes input data (dates, likes, URLs).
`image_captioning.ipynb`	Integrates Qwen2-VL for detailed descriptions.
`prompt.py`	Creates custom prompt templates for marketing context.
`fine_tune_llama.ipynb`	Implements LoRA fine-tuning and quantization.
`metric.py`	Scripts for calculating performance scores.
`test_pipeline.ipynb`	Final notebook for executing the full generation process.

D. Evaluation Strategy

The quality of the generated outputs (tweets and captions) is assessed using standard Natural Language Generation (NLG) metrics:

- **BLEU Scores (BLEU-1 through BLEU-4)**: Measures the n-gram overlap similarity to reference texts.
- **ROUGE Metrics (ROUGE-1, ROUGE-2, ROUGE-L)**: Measures the overlap of n-grams, word sequences, and longest common subsequences.
- **CIDEr Score**: Measures the consensus of a generated caption with human-written references, primarily for VLM output evaluation.

VI. EVALUATION GUIDELINES

Submission Deadline - 15th December Testing Dataset will be shared on 14th December

Evaluation metric(s) – 50% weightage For Task 1: RMSE (Root Mean Squared Error) between predicted and ground-truth likes. For Task 2: BLEU 1-4, ROUGE, CIDEr. Refer this [1] for the above metrics.

Approach – 35% weightage 1. Efficiency of solution 2. Novelty of approach

Presentation – 15% weightage

VII. SUBMISSION GUIDELINES

1. Link to GitHub repository (access made public on the day of final submission) 2. Submit a report detailing your approach, results, or any inference/assumption you feel is important. The maximum length can be 3 pages in ACL LaTeX format (excluding appendix and references).

Dataset shared is open source. The IP to final solution will belong with Adobe.

VIII. CONCLUSION

We deliver a production-ready dual-task system:

- **Task 1**: 16.5% RMSE reduction via conditional vision and hybrid modeling.
- **Task 2**: High-fidelity tweet generation with decoupled VLM+LLM pipeline.

The modular design ensures efficiency, scalability, and interpretability.