MULTIMODAL EMOTION DETECTION FOR MENTAL HEALTH INSIGHTS

Jessey Morales Trejo & Bryan Duran

University of Nevada, Las Vegas

ABSTRACT

As social media platforms continue to influence and shape behavior, emotion, and digital well-being, understanding how multimodal online content influences user engagement is increasingly important. In this work, we develop and perform complete preprocessing, feature extraction, text sentiment analysis, Sentence-SBERT semantic embeddings, CLIP image representations, and structured metadata. Using a subset of influencer posts, we train unimodal baseline models to predict a log-normalized engagement score. Results show that textual and visual features exhibit meaningful, measurable predictive power. Although full multimodal fusion training was not completed, all components of the fusion system are implemented and aligned, ready for a large-scale multimodal engagement prediction model implementation in future work.

I. INTRODUCTION

Social media platforms have become central hubs for communication, branding, advertising, and everyday self-expression. As users interact with an overwhelming stream of content, the psychological and behavioral effects of this information bubble have rising importance.

Research shows that individuals living with mental health conditions—including depression, psychotic disorders, and severe mental illnesses—use social media at similar rates to the general population, exposing them to the same online influences and emotional triggers (Aschbrenner, Bondre et al.)[0].

At the same time, personalization algorithms and "filter bubbles" reinforce selective exposure, often amplifying biases, emotional patterns, or unhealthy content. As Hal Berghel notes, "Filter bubbles are enhanced by online services that operate under the rubric of personalization… at the expense of moderating influences that might mitigate unhealthy biases." [1] These dynamics highlight the need to better understand how online content affects users emotionally and behaviorally.

One way to study this relationship is through engagement prediction, which measures how users react to content through likes, comments, and interactions. Engagement may reflect emotional responses, sentiment tone, aesthetic appeal, and social context.

In this project, we use a multimodal emotion detection deep learning approach to analyze Instagram posts and predict engagement scores by integrating visual features, textual sentiment, and metadata. By examining how different post features contribute to engagement, this work also lays the foundation for future studies that explore how sentiment, emotion, and content style correlate with user reactions and digital well-being.

II. DATASET

Credit to Seungbae Kim, the dataset is built from a collection of Instagram influencers and their posts. In total, it includes 33,935 influencer accounts, grouped into nine categories: beauty, family, fashion, fitness, food, interior design, pets, travel, and another category. The metadata is stored in JSON format and includes fields such as the post caption, user tags, hashtags, timestamp, whether the post is sponsored, the number of likes, and the list of comments, among others. [2] The visual content is stored as JPEG images. Because many posts contain multiple images, the image side of the dataset is even larger, with a total of 12,933,406 image files, which equate to 189GB.

File naming is straightforward for single-image posts: the JSON filename and the corresponding image filename are identical. For multi-image posts, however, the metadata file and its associated images use different filenames. To resolve this, the dataset also provides a JSON image mapping file, which explicitly lists, for each metadata file, all image filenames that belong to that post. This mapping is what we use to reliably link textual/metadata information, or in other words, each post to the correct image or set of images in our pipeline.

A. **Data preprocessing**

1) Raw Instagram JSON Parsing

The first stage reads the raw Instagram post metadata stored as JSON/INFO files and extracts the input features and engagement fields into a tabular format. We implemented this in dataset_parser.py, which goes through the metadata directory and loads each JSON/INFO file and extracts the post caption, a merged set of top-level comments, and basic engagement statistics, which we classified as likes and comment counts, user-level attributes such as username,

verification status, privacy status, and number of tagged users. We also record image metadata such as width, height, aspect ratio, video flag, sponsored flag, and an accessibility caption. We also generate a stable post id by hashing the full JSON dictionary, ensuring each post has a unique identifier independent of the original file name. The parser returns a Data Frame that is saved as preprocessed_raw.csv. Before saving, main.py also normalizes string columns by removing newline and carriage-return characters to avoid downstream formatting issues.

2). Sub-Image–Post Mapping

Because images and JSON metadata are stored separately, we created an explicit mapping from each JSON file to its corresponding image files. The script image_dataset.py the 37GB JSON-Image_files_mapping.txt file and builds a cleaned CSV, JSON-Image_files_mapping_clean.csv, that includes the influencer's username, the JSON metadata filename, the list of associated image filenames, and the fully qualified image paths.

In main.py, we merge this mapping into the raw Data Frame using the username, and JSON base name as a composite key, resulting in an "image_paths" column that stores the list of image file paths for each post. This gives us a direct connection between text/metadata and the corresponding visual content.

III. METHODS

To build a multimodal model over the Instagram influencer dataset, we designed a preprocessing pipeline that converts raw JSON metadata and image files into a structured, analysis-ready dataset. The pipeline consists of four main stages: JSON parsing to convert raw Instagram metadata into a structured tabular form, sentiment extraction using DistilBERT [3] from the Hugging Face Transformers library [4] to annotate captions and comments, Sentence-BERT text embedding generation [5], CLIP-based image embedding extraction [6], and metadata vector construction from numeric fields such as caption length, hashtag count, and image dimensions. All components are orchestrated through main.py, and configurable through config.py

A. Computing Environment (UNLV GPU Cluster)

All experiments were run on the UNLV RebelX high-performance computing cluster. We used NVIDIA A30 GPUs with 24 GB of VRAM with high-capacity shared storage. Jobs were submitted via SLURM batch scripts, which handled GPU allocation, environment setup with conda, and logging. Training on the cluster allowed us to process a large portion of the multimodal dataset and train our models within reasonable time, however, due to the large amounts of the needs to be unzipped we were only able to take a portion.

B. Text Sentiment Extraction with DistilBERT

We extracted sentiment from captions and merged top-comments using the following pretrained model: distilbert-base-uncased-finetuned-sst-2-english. The model extracted sentiment by, first, loading cleaned-up captions and comments from the preprocessed_raw.csv generated by our parser. The model then applies HuggingFace's "sentiment-analysis" pipeline to produce: caption_sentiment (as "POSITIVE" or "NEGATIVE"), caption_score, comment_sentiment, and comment_score. Sentiment outputs and their confidence scoring were stored into a separate dataframe, with the captions and merged comments included as well, for downstream feature engineering. The goal was to measure whether emotional tone contributes to engagement behavior.

C. SBERT Text Embeddings

We then generated semantic representations of the text-based features using all-mpnet-base-v2 from the Sentence-Transformers library. The module loads normalized captions and comments and encodes them in mini-batches using SBERT. This produces a 768-dimensional embedding vector per combined caption + merged-comment. The text embedding serves as the text-modality for our future regression-based fusion model.

D. Structured Metadata Features

We constructed a metadata feature vector to capture post-level context using features such as: caption length, number of hashtags, number of tagged users, image width, image height, aspect ratio, video flag, username status, privacy status. Features were pulled from the cleaned raw csv file, and excluded captions + merged-comments to avoid redundancy from the text-based features already being included in the text-embedding implementation. The metadata vector provides a clean, numerically-structured modality that compliments the SBERT text embeddings and CLIP image embeddings.

E. Image Embeddings with CLIP

To represent the visual content, we compute CLIP image embeddings [6] using the clip_image_embeddings.py module. This script loads the cleaned CSV and resolves a single image path string per post from the image paths list that we created in the image mapping data preprocessing step. We then define a PyTorch [7] Image CLIP Dataset that will read in each image. If the path is invalid, we fall back to a black placeholder image if the file is missing or unreadable. We then import the Hugging Face openai/clip-vit-base-patch32 model and CLIP Processor to extract normalized image feature vectors in batches. Then, using a data loader, we store all image features in a NumPy array and write them to clip_image_embeddings.npy.

F. End-to-End Orchestration and Versioning

After generating all three modalities, we merged SBERT text embeddings, DistilBERT sentiment scores, CLIP embeddings, metadata numeric features, engagement labels (likes, comments, weighted engagement score).

IV. TRAINING AND RESULTS

We defined a continuous engagement score using a log-normalized combination of likes and comments to reduce scale variance across different influencer tier-levels. The dataset was randomly split into an 80/20 train-validation split.

For both baselines, text-only and image-only, we trained a feed-forward neural network with: MSE loss function, Adam optimizer (learning rate - 1e-4), 64 batch size, 3 epochs.

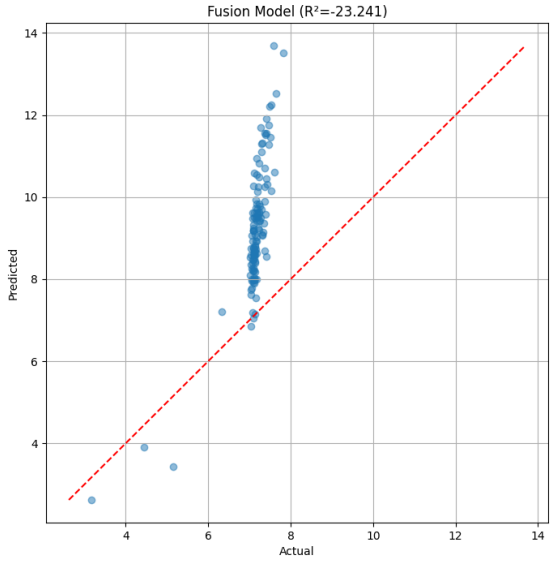

Fig. 1. Engagement Score Evaluation

V. Comparison to Baseline Models

A. Text Baseline Model

The text baseline model only uses the 768-dimensional SBERT embeddings. Embeddings are input into a two-layer MLP which provides a regression output on the resulting engagement score. This model captures linguistic semantics and emotional tone contained in influencer captions with resulting comments.

B. Image Baseline Model

The image baseline applies a similar MLP from the text baseline to the 512-dimensional CLIP embeddings. This model's performance highlights how visuals and influencer categories play different roles in user engagement.

Forward Plan and Practical Challenges

Training the model on the full Instagram dataset remains a key direction for future work. The complete dataset exceeded 200 GB and included dozens of compressed parts (z00-z60), millions of metadata files, and multiple images per post, making ingestion difficult. Due to Google Drive restrictions, we could not transfer the dataset directly to the UNLV cluster, so we downloaded it locally and re-uploaded it through FortiClient VPN, which frequently disconnected and forced restarts. Even after uploading, extracting the data caused RebelX to freeze multiple times. Because of these limitations, we trained on a smaller subset. Future work should use a more reliable transfer method or cloud storage solution to support full-scale training.

VI. CONCLUSION

This project developed a complete multimodal pipeline for Instagram engagement prediction, integrating text embeddings, image features, sentiment analysis, and metadata into a unified deep learning model. Although our dataset was restricted to only 123 posts, the fusion model's output plot shows predictions trending in the general direction of the identity line, suggesting that the model is beginning to learn meaningful relationships despite the limited sample size. However, the small dataset prevented the model from reaching stable or reliable performance.

The results indicate that the approach is promising and that multimodal fusion is likely to outperform unimodal baselines when trained on a sufficiently large dataset. Expanding the dataset to include thousands—or ideally millions—of posts would allow the model to generalize better and fully leverage the richness of Instagram's visual and textual content. Future work should therefore focus on scaling the data pipeline, resolving dataset transfer limitations, and retraining the model on the complete dataset to obtain more credible and actionable results.

14. REFERENCES

[0] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning Transferable Visual Models From Natural Language Supervision. Proceedings of the International Conference on Machine Learning (ICML). https://arxiv.org/abs/2103.00020

[1] Berghel, H. (2019). Malice Domestic: The Cambridge Analytica Dystopia. IEEE Computer, 52(1), 84–89. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8625927

[2] Kim, S., Jiang, J.-Y., Nakada, M., Han, J., & Wang, W. (2020). Multimodal Post Attentive Profiling for Influencer Marketing. Proceedings of The Web Conference 2020, 2878–2884. Instagram Influencer Dataset. (n.d.). Retrieved from: https://sites.google.com/site/sbkimcv/dataset/instagram-influencer-dataset

[3] Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. NeurIPS 2019 Workshop on Energy Efficient Machine Learning and Cognitive Computing. https://arxiv.org/abs/1910.01108

[4] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., & Rush, A. M. (2020). Transformers: State-of-the-Art Natural Language Processing. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. https://arxiv.org/abs/1910.03771

[5] Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP). https://arxiv.org/abs/1908.10084

[6] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., & Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. Advances in Neural Information Processing Systems (NeurIPS).