Predicting TikTok Virality with Late-Fusion Multimodal Learning

Stanford CS229 Project

Oscar Rodriguez
Stanford University
oscarrm@stanford.edu

Mahda Soltani
Stanford University
soltanim@stanford.edu

Ela Naz Sigin
Stanford University
elasigin@stanford.edu

Abstract

Social media platforms like TikTok have become one of the dominant channels through which information, entertainment, and political and cultural trends propagate at scale. A defining feature of these platforms is the extreme skewness in attention: while the vast majority of uploaded videos receive little to no exposure, a small fraction achieve *viral* reach, accumulating orders of magnitude more views within a short time window. This heavy-tailed distribution of engagement naturally raises the question of how to operationalize and predict "virality" in a principled way.

This investigation explores the use of Machine Learning techniques to pin down a formal framework for the task of predicting the likelihood of a TikTok video going viral using both intrinsic (e.g. video content) and extrinsic features (e.g. video author).

1 Introduction

Social media platforms such as TikTok have become central channels through which information, entertainment, and cultural trends spread at massive scale. A defining feature of these platforms is the extreme inequality in attention: while most uploaded videos receive minimal engagement, a small fraction rapidly accumulates millions of views. This highly skewed distribution raises an important question: can we predict which content is likely to go viral?

In this project, we develop a machine learning framework to estimate the probability that a TikTok video becomes viral by leveraging both intrinsic signals (features derived from the video content itself) and extrinsic signals (information about the creator and posting context). Using a multimodal architecture that integrates text, video, and tabular metadata, we model engagement dynamics and classify videos that fall into the top tail of the engagement distribution.

Our results provide insight into the extent to which measurable platform signals can explain virality, while highlighting the persistent role of stochastic and external factors in shaping online attention.

2 Methods

2.1 Data collection and processing

We started with a base dataset called TikTok-10M The Data Company (2024) from which we sourced a rich subset of examples that we complemented with a scraping step to download the videos and publisher accounts to extract additional signals (e.g. video bytes, aspect ratio) as well as account information (e.g. account statistics, publishing history). With all this raw data at hand, we produced a curated dataset Rodriguez (2025) for our task. Table 1 documents the dataset dimensions.

2.2 Task definition

The definition of virality (`is_viral`) we produced is the target dimension we optimize for. It is meant to capture outliers relative to any given account stack-ranked in absolute terms with the goal of producing a signal that's useful for any account size. We compute two scores for each video: **engagement** and **view velocity**.

Engagement

$$\text{engagement}_i = \frac{\text{share\_count} \times w_0 + \text{save\_count} \times w_1 + \text{comment\_count} \times w_2 + \text{like\_count} \times w_3}{\text{play\_count}} \tag{1}$$

Table 1: Dataset dimensions: intrinsic content signals and extrinsic author/metadata signals.

Dimension	Type	Description	Extraction
User / author signals			
user_id/username	ID/String	Unique identifier and handle of the creator	Scraped
user_verified	Boolean	Verification status of the author	Scraped
is_private	Boolean	Whether the account is set to private	Scraped
author_stats	Floats	Follower, hearts, video, and friend counts	Scraped; $\log(1+x)$
Video content & metadata			
description	String	User-generated caption and hashtags	Base dataset
temporal_features	Float	Hour of day and day of week	Sine/cosine encoding
duration	Integer	Total length of the video in seconds	Base dataset
challenges	String	Associated hashtag challenges or trends	Base dataset
video_bytes	Binary	Scraped video resized to 256×256 @ 1fps	`yt-dlp` + `moviepy`
detected_objects	String	Objects detected in video frames	DETR-ResNet-50
Audio & technical specs			
music_info	String	Title, album, and author of the audio	`yt-dlp` metadata
resolution	Integer	Video width and height in pixels	`yt-dlp`
aspect_ratio	Float32	Ratio of width to height	Calculated
vq_score	Float64	Visual quality score	Base dataset
Engagement stats & target labels			
engagement_score	Float64	Weighted sum of interactions per view	Engineered
view_velocity	Float64	Views accumulated per hour since creation	$\log(1+x)$ scaled
is_viral	Binary	Classification (P=0.95) of engagement	Quantile threshold

Engagement represents the volume of "interactions" a video had relative to its play count, making the score relative to the account size. As shown in Figure 1, accounts with a larger audience will most likely outperform their counterparts in absolute measurements. This signal instead captures a degree of engagement normalized to audience size: a small account with a few hundred followers that generates thousands of interactions would have a comparable engagement to an account with millions of followers that accumulated hundreds of thousands of interactions.

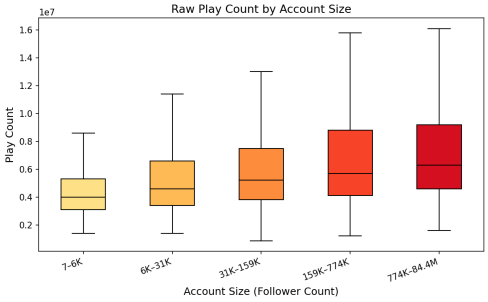

Figure 1: Raw play count by account size. Larger accounts naturally accumulate more views, motivating our normalized engagement signal.

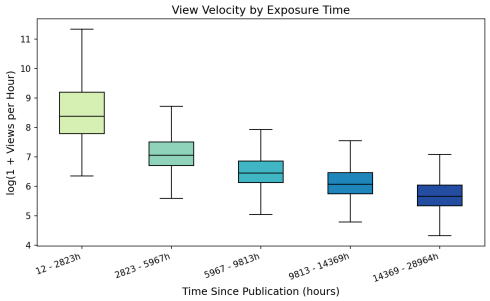

Figure 2: View velocity by exposure time. Videos accumulate views most rapidly in the first few hours, with velocity dropping sharply thereafter.

View velocity

View velocity normalizes play count by the total time a video has been live (Figure 2). A video with high view velocity accumulates interactions faster:

$$\text{view velocity}_i = \frac{\text{play\_count}}{\text{delta\_hours}} \tag{2}$$

Producing the signal

We combine both measurements into a signal S_i for each video and consider the top 5% as "viral":

$$S_i = \frac{\text{engagement}_i}{\text{engagement}_{max}} + \frac{\text{view velocity}_i}{\text{view velocity}_{max}} \tag{3}$$

$$y^{(i)}_{\text{is\_viral}} = \begin{cases} 1 & \text{if } S_i \geq F_S^{-1}(0.95) \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Our reasoning for this definition of virality is rooted in convenience and usefulness. There are too many dynamic forces that influence virality, like current social discourse, political events, or sheer luck, which makes it near impossible to capture all contributing factors. Instead, we condense a signal that is useful even for small accounts: rolling out a video that is a couple of folds more viral relative to their expected baseline is considered a win.

2.3 Model architecture

The architecture implements a mixture of processing techniques for different data modalities; all branches are concatenated and passed through a late fusion module to produce 2 regression heads (engagement and view velocity) and 1 classification head (virality). The outputs are stitched together into a final weighted loss. Figure 3 shows the full pipeline.

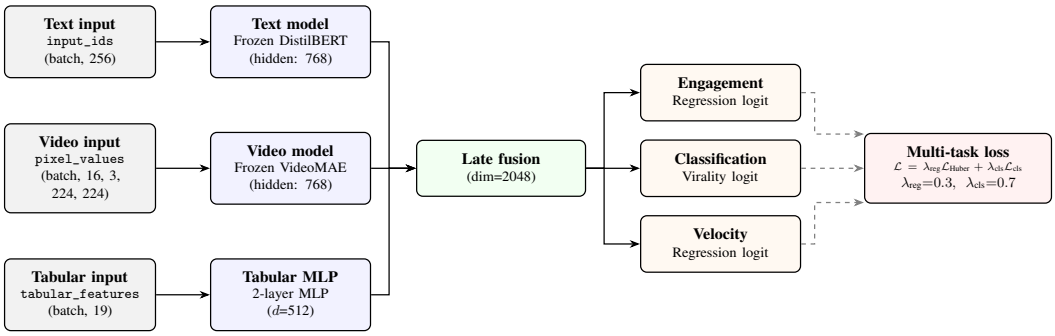

Figure 3: End-to-end architecture: features are extracted from multimodal inputs, fused into a shared latent space, and passed to specialized heads. The final objective is a weighted combination of Huber regression loss and classification loss.

Text input The text input is a concatenation of textual dimensions: "`[CLS] [TITLE] ... [MUSIC] ... [CHALLENGES] ... [OBJECTS] ... [LOC] ...`" fed into a pretrained, frozen DistilBERT Sanh et al. (2019) backbone to extract a 768-dimensional semantic representation from the [CLS] token.

Video input Videos were downsized to 224×224 pixels at 1 FPS. We sample 16 frames and feed them into a frozen VideoMAE Tong et al. (2022) model, then apply global average pooling across the output tokens to yield a 768-dimensional embedding capturing visual features and temporal dynamics.

Tabular input The tabular branch processes 19 heterogeneous features including author signals, temporal metadata, and video quality scores. We apply $\log(1+x)$ scaling to high-variance author statistics and sine/cosine cyclic encoding to temporal features. These are fed into a 2-layer MLP with layer normalization and dropout, projecting them into a learnable 512-dimensional space.

Late fusion We concatenate the 768-dim text embedding, 768-dim video embedding, and 512-dim tabular output into a single 2048-dimensional vector, passed through a shared fusion MLP mapping to d_{model}=512.

Multi-head targets We employ a multi-head architecture: two regression heads predict the continuous engagement and view velocity scores, and one classification head predicts virality (the 95th percentile of the combined distribution).

Loss targets We experimented with two loss formulations. **Weighted BCE:** Huber loss Huber (1964) for regression and weighted BCE for classification, with $\lambda_{\text{cls}}=0.7$ and $\lambda_{\text{reg}}=0.3$; the minority class is upweighted by $\frac{p}{1-p}$ where $p=0.95$. **Focal loss** Lin et al. (2020)**:** replaces weighted BCE with an adaptive mechanism that down-weights easy negatives, with $\gamma=2.0$ and $\alpha=p_{\text{virality\_threshold}}$.

3 Experiments and Results

We track ROC-AUC (general class separability), PR-AUC (our primary metric, as it does not reward correct prediction of the non-viral majority), and F1-Score at the 0.5 threshold.

3.1 Training dynamics

We compared two training regimes: a multi-task **weighted loss** baseline with linear learning rate decay over 15 epochs, and a **focal loss** experiment with cosine scheduler and 10% warmup phase. As shown in Figure 4, both models exhibited steady convergence. The focal loss model reached optimal performance earlier (around epoch 8), suggesting it could ignore easy negatives and converge faster on the viral decision boundary.

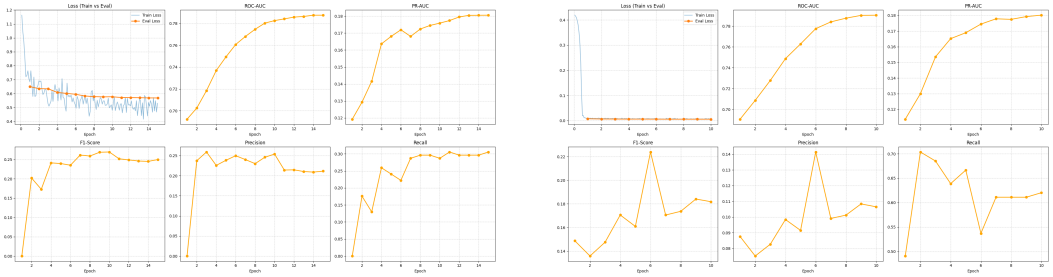

Figure 4: Training dynamics for weighted loss (left) and focal loss (right).

3.2 Analysis of results

Table 2: Comparative performance between weighted BCE and focal loss targets.

Metric	Weighted loss	Focal loss	Δ
ROC-AUC	0.7875	**0.8012**	+1.7%
PR-AUC	**0.1805**	0.1788	-0.94%
F1-Score	0.2452	**0.2620**	+6.8%
Precision	0.2092	**0.2479**	+18.4%
Recall	**0.2963**	0.2777	-4.5%

The transition to focal loss yielded a **18.4% improvement in precision**. While the baseline weighted BCE model achieved higher recall by aggressively upweighting the minority class, it suffered from a high false-positive rate. Focal loss mitigated this by forcing the model to focus on "hard" examples, resulting in a more balanced F1 of 0.2620.

Despite the marginal improvement in ROC-AUC, PR-AUC stayed mostly across both experiments, confirming that the absolute confidence in viral classification is likely bounded by factors we did not model:

1. **Unmeasurable stochasticity:** Too many ongoing dynamics (socio-political discourse, randomness) influence popularity in ways a finite dataset cannot capture.
2. **Gaps in dataset:** Missing input sources like audio features, platform-specific nuances (e.g. current trends), or richer video dynamics.

3.3 Feature importance

We run permutation importance on the 19 tabular features (shuffling each across 300 test examples and measuring F1 drop) through two lenses: the **classification head** and the **regression heads** (Figure 5).

The two heads reveal complementary strategies. The classification head relies almost entirely on **creator reputation**: `follower_count` (+0.41), `heart_count` (+0.23), and `friend_count` (+0.14) dominate, while content features have near-zero impact. The regression heads instead prioritize **content and temporal properties**: `duration` (+0.13), `sin_hour` (+0.10), `height` (+0.09), and `vq_score` (+0.08), with creator signals contributing moderately. This suggests the classification

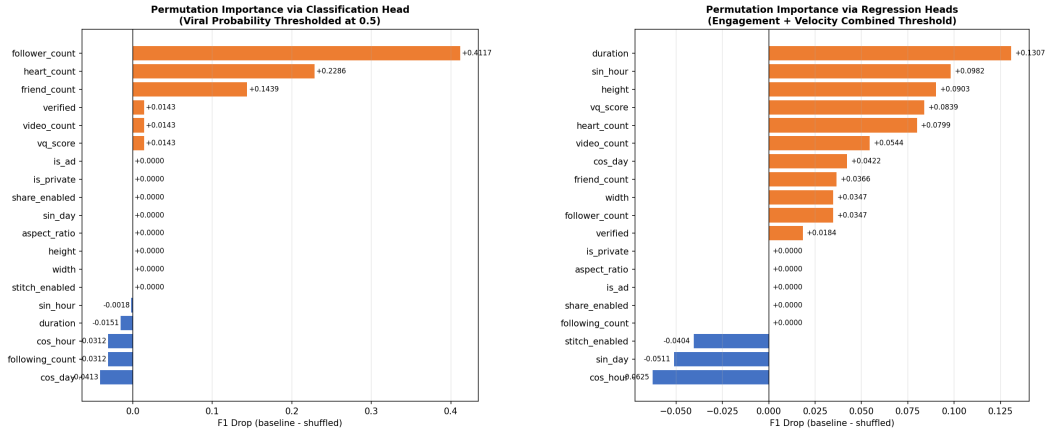

Figure 5: Permutation importance via classification head (left) and regression heads (right).

head learns a shortcut that large accounts are likely to go viral, while the regression heads, tasked with predicting continuous scores, must attend to finer-grained signals that influence *how much* engagement a video receives.

3.4 Ablation studies

To determine the specific contributions of each modality to our multimodal model, we conduct ablation experiments. Specifically, we zero out one modality's representation at a time using forward hooks and measure metric degradation on 300 test examples (Figure 6).

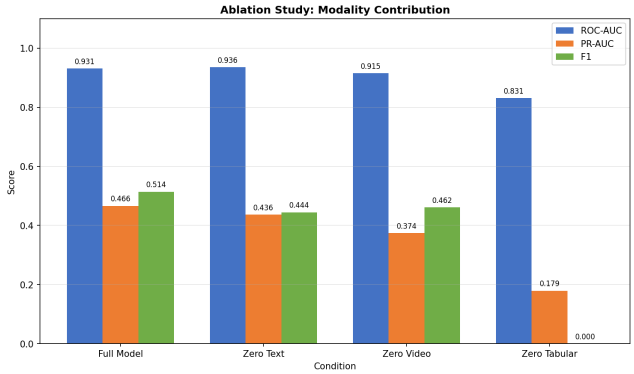

Figure 6: Ablation study: metrics when each modality is individually zeroed. Removing tabular features causes the most severe degradation (F1 drops to 0).

From Figure 6, we see that Tabular Features are critical, and Text has the smallest individual contribution. Zeroing the tabular MLP output collapses F1 to 0.000, meaning no example's viral probability exceeds the 0.5 threshold without creator and metadata signals. The model does retain some ranking ability (ROC-AUC = 0.831) from text and video alone. Moreover, Video provides moderate signal: removing video reduces F1 from 0.514 to 0.462 and PR-AUC from 0.466 to 0.374. Zeroing text causes only a slight F1 drop (0.514 $\rightarrow$ 0.444). Interestingly, ROC-AUC *increases* marginally (0.931 $\rightarrow$ 0.936), suggesting the text branch may introduce slight noise.

4 Conclusion and Future Work

While the model provides a strong baseline for late-fusion multimodal learning, the gap between ROC-AUC and PR-AUC indicates difficulty handling the extreme sparsity of viral signals. Future work could incorporate additional modalities such as **audio features** and richer video dynamics, and explore **Self-Exciting Hawkes Process** models to capture engagement growth over time.

Overall, intrinsic content and creator signals provide predictive value for virality, but outcomes remain highly stochastic. Our multi-task framework stabilizes training of a high-dimensional multimodal model and offers a useful foundation for future work on social media dynamics.

Contributions

- **Oscar Rodriguez:** Implementation of scraping scripts and dataset compilation as well as support with model architecture design and training runs. I also helped putting together the final write up.
- **Mahda Soltani:** Implementation of Temporal CV and assistance with feature importance. I also helped putting together the final write up.
- **Ela Naz Sigin:** Conducted model analysis (ablation studies, feature importance), generating plots and figures for the report. Also experimented with various dataset rebalancing techniques (in practice, training on the rebalanced dataset yielded worse model performance than the original).

References

Peter J Huber. 1964. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1):73–101.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2020. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327. ArXiv:1708.02002.

Oscar Rodriguez. 2025. Viral: A curated tiktok virality dataset. https://huggingface.co/datasets/rodmosc/viral.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

The Data Company. 2024. Tiktok-10m dataset. https://huggingface.co/datasets/The-data-company/TikTok-10M.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. 2022. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*.

A Appendix

A.1 Inference examples

We ran a few inferences through our final model to understand our model's strengths and limitations. We decided to prioritize recall over precision while fine tuning our model hyperparameters. Our reasoning was that we aim to capture as many viral videos as possible (true positives), and in this application, a false positive does not have drastic consequences (unlike a medical diagnosis application for instance). Additionally, by prioritizing recall, we can get more useful information out of our model (recognizing more of the truly viral videos).

Category	Description	Ground Truth	Viral Prob	Prediction
True Positive	Amazing Fishing Video	Viral	60.60%	Viral
True Positive	Inexplicably funny😂#tiktok #fyp #animals	Viral	58.59%	Viral
True Negative	#basketball #trending #repost #fyp #abcxy	Not Viral	14.48%	Not Viral
True Negative	I've had ideas for this song and this one was	Not Viral	14.76%	Not Viral
False Positive	Balde back with his side eye 😒 #fcbarcelona	Not Viral	72.61%	Viral
False Positive	Cheetah Hunt at #buschgardens #rollercoas	Not Viral	59.12%	Viral
False Negative	Raw vid of end game on a big ol bluefin last	Viral	37.89%	Not Viral
False Negative	Venice Beach □☀	Viral	45.54%	Not Viral

Figure 7: Inference examples showing model strengths (true positives/negatives, green) and shortcomings (false positives/negatives, red).