machine learning & knowledge extraction

IMPACT FACTOR 8.4

CITESCORE 12.7

Article

Algorithmic Diffusion on YouTube: A Machine Learning Analysis of Channel-Level Information Spread and Its Cross-Platform Generalisability

Dana Tyulemissova, Aigul Shaikhanova, Oleksandr Kuznetsov, Aigerim Sambetova, Kainizhamal Iklassova and Aisanim Sarsenbayeva

MDPI

https://doi.org/10.3390/make8090272

machine learning & knowledge extraction

MDPI

Article

Algorithmic Diffusion on YouTube: A Machine Learning Analysis of Channel-Level Information Spread and Its Cross-Platform Generalisability

Dana Tyulemissova [1], Aigul Shaikhanova [1,*], Oleksandr Kuznetsov [2,3,*], Aigerim Sambetova [4], Kainizhamal Iklassova [5] and Aisanim Sarsenbayeva [6]

[1] Department of Information Security, L.N. Gumilyov Eurasian National University, Satpayev 2, 010008 Astana, Kazakhstan; tyulemissova_db_3@enu.kz
[2] Department of Theoretical and Applied Sciences, eCampus University, Via Isimbardi 10, 22060 Novedrate, Italy
[3] Department of Intelligent Software Systems and Technologies, School of Computer Science and Artificial Intelligence, V.N. Karazin Kharkiv National University, 4 Svobody Sq., 61022 Kharkiv, Ukraine
[4] Department of Mathematical and Computer Modeling, International Information Technology University, 34/1 Manasa Str., 050040 Almaty, Kazakhstan; asambetova@iitu.edu.kz
[5] Department of Information and Communication Technologies, Manash Kozybayev North Kazakhstan University, Pushkin Str. 86, 150000 Petropavlovsk, Kazakhstan; keiklasova@ku.edu.kz
[6] Department of Informatics and Educational Informatization, Abai Kazakh National Pedagogical University, 13 Dostyk Avenue, 050010 Almaty, Kazakhstan; a.sarsenbayeva@abaiuniversity.edu
* Correspondence: shaikhanova_ak@enu.kz (A.S.); oleksandr.kuznetsov@uniecampus.it (O.K.)

Abstract

(1) Background: Information diffusion models developed for graph-based platforms such as Reddit and broadcast architectures such as Telegram identify temporal features—particularly the timing of peak spread—as dominant predictors of coverage. Whether these predictors generalise to platforms where content is distributed through algorithmic recommendation rather than social-graph contagion remains an open question. (2) Methods: We analyse the YouNiverse dataset, comprising 133,364 English-language YouTube channels observed weekly from January 2015 to September 2019 (18.9 million observations). We derive channel-level diffusion features—including time-to-peak, post-peak decay rate, diffusion volatility, and upload frequency—and train three machine learning models (Linear Regression, Random Forest, and LightGBM) on two tasks: predicting peak weekly view growth (regression) and identifying viral channels (classification). A single-feature naive baseline (subscriber count alone) establishes the marginal contribution of the broader feature set beyond subscriber count alone, and a temporal split experiment (training on channels peaking before 2018, testing on 2018–2019) assesses cross-temporal stability. Because subscriber count and subscriber rank are measured at the October 2019 crawl, this is a retrospective characterisation rather than a strict real-time forecasting design. (3) Results: LightGBM achieves $R^2 = 0.776$ (5-fold CV: 0.778 ± 0.003) compared with $R^2 = 0.548$ for the naive baseline, a net gain of $+0.228$ R^2. Because subscriber rank and subscriber count are near-perfectly collinear, we interpret them jointly as a channel-size dimension (42.2% of total mean absolute SHAP attribution), rather than as independent effects. Time-to-peak ranks fourteenth (1.1%), in contrast to its dominant role on Reddit ($r = 0.995$, rank #1). For virality classification, LightGBM achieves ROC-AUC = 0.967. Under the temporal split, Random Forest ($R^2 = 0.703$) outperforms LightGBM ($R^2 = 0.683$), showing greater cross-temporal stability within this retrospective split. (4) Conclusions: Within the 2015–2019 data, the results are consistent with algorithmic recommendation weakening the relationship between temporal diffusion dynamics and coverage magnitude at the channel

Check for updates

Academic Editor: Fabrizio Riguzzi

Received: 4 July 2026
Revised: 12 August 2026
Accepted: 3 September 2026
Published: 6 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

level. Time-to-peak is weakly informative in this setting, while generalisation to the current recommendation system requires validation on newer data.

Keywords: information diffusion; YouTube; machine learning; LightGBM; Random Forest; SHAP; time-to-peak; platform architecture; channel-level analysis; temporal validation; viral content; social media analytics; big data

1. Introduction

The study of information diffusion in social media has produced a substantial body of models grounded in the assumption that spread follows the logic of social contagion: content propagates through network ties, and the speed of early propagation is a reliable predictor of eventual reach [1–3]. Leskovec et al. [4] showed that a cascade structure in blog networks is predictable from early observations. Cheng et al. [5] demonstrated that whether a cascade on Facebook will double in size can be determined from structural features of its first few hours. A companion study in this research series confirmed the assumption at scale on Reddit: Bakenova et al. [6] showed that a single temporal feature—peak speed time—achieves $r = 0.995$ with coverage using gradient boosting ($R^2 = 0.847$) across synthetic and real Reddit thread data under three delay mechanisms (constant, uniform, exponential).

A subsequent study examined Telegram [7], a broadcast platform with star-topology forwarding cascades (depth = 1) and heavy-tailed delay distributions. The timing structure there still plays a structural role: the delay distribution governs the forwarding model, achieving median prediction errors below 10%.

YouTube represents a third architectural regime. Unlike Reddit (social-graph cascade) or Telegram (broadcast without algorithmic curation), YouTube distributes content through a recommendation engine that learns from historical engagement patterns and surfaces content to users with no prior connection to the creator [8]. The recommendation system can surface content weeks or months after upload, creating growth waves structurally uncorrelated with the timing of the first.

This raises the central question: *does time-to-peak retain its predictive power in an algorithmically mediated environment?* A secondary motivation concerns the unit of analysis. Most prior YouTube work predicts *video*-level popularity from upload-time features [9–12]. We study *channel*-level diffusion over multi-year windows—a perspective that captures how creators build persistent audiences through algorithmic accumulation rather than individual content virality.

Our contributions are as follows:

- Channel-level diffusion features with a naive single-feature baseline that quantifies the marginal value of the broader structural, temporal, behavioural, and categorical feature set over subscriber count alone.
- Demonstration that time-to-peak has near-zero association with peak coverage on YouTube ($\rho = 0.056$; SHAP rank #14, 1.1%) compared with $r = 0.995$ on Reddit.
- $R^2 = 0.776$ for LightGBM (CV: 0.778 ± 0.003), a net $\Delta R^2 = +0.228$ over the naive baseline.
- Temporal-split experiment revealing that Random Forest is more robust to distributional shift than LightGBM (Section 4.8).
- A three-platform comparison used as a cautious, conceptual test of the architecture hypothesis rather than as a direct numerical benchmark: graph diffusion is time-sensitive, broadcast is delay-driven, and algorithmic diffusion is size- and shape-driven (Section 4.9).

2. Related Work

2.1. Information Diffusion Models and Temporal Predictors

Classical epidemic models were adapted for online social networks following the work of Kermack and McKendrick [1] and the demonstration by Gruhl et al. [13] that blog propagation follows contagion-like dynamics. Pastor-Satorras et al. [2] review epidemic processes on complex networks. Goel et al. [3] analysed one million diffusion cascades and found most viral content reaches broad audiences within one or two degrees of the seed. Cheng et al. [5] demonstrated cascade doubling prediction from early Facebook features.

Bakenova et al. [6] tested constant, uniform, and exponential delay types across five network topologies and identified peak speed time as the overwhelmingly dominant predictor ($r = 0.995$; LightGBM $R^2 = 0.847$). The Telegram companion study [7] showed platform architecture produces categorically different structures—star topology, depth = 1, heavy-tailed forwarding delays—requiring platform-specific models.

2.2. Machine Learning for Content Popularity Prediction

Gradient-boosted decision trees consistently outperform linear models on structured social media prediction tasks [14,15]. Random forests [16] provide a complementary baseline whose bagging-based regularisation offers robustness under distributional shift, as demonstrated in our temporal split experiment (Section 4.8). SHAP values [17,18] provide theoretically grounded attributions that correct the known bias of gain-based importance toward high-cardinality features.

Liu et al. [19] survey feature-fusion-based popularity prediction methods, noting a key gap: most methods focus on short-term prediction at upload time and do not model long-term diffusion trajectories. Jia et al. [20] analysed BiliBili (a YouTube-like platform), finding that subscription-based social features improved prediction beyond content features—consistent with our observation that subscriber rank dominates SHAP attribution.

2.3. YouTube-Specific Popularity Studies

YouNiverse [21] has been used to study political radicalisation [22] and audience migration, but not for channel-level diffusion characterisation over multi-year windows.

At the video level, temporal features retain predictive power. Szabo and Huberman [9] showed a log-normal relationship between early and long-term view counts. Chen and Chang [11] proposed an ensemble using upload-time features only. Figueiredo et al. [12] introduced TrendLearner, achieving F1 improvements of 38% while retaining substantial residual views. Halim et al. [23] examined content-agnostic features for trending prediction across seven regional markets, finding substantial market-to-market variation—motivating our category-stratified analysis.

Recent work has moved toward multi-modal deep learning. Chen and Wan [24] proposed FPPEV, jointly modelling video popularity and engagement through contrastive feature decomposition, reducing MAE by 17.6%. Le et al. [25] developed EnTube integrating titles, audio, thumbnails, and content tags. Cho et al. [26] introduced AMPS, an attention-based model for short-form YouTube popularity using full video frame representation.

Our work differs in three respects: (1) channel level vs. video level; (2) purely temporal and structural features with no content modality; (3) windows of up to five years vs. hours to weeks. Ahmad et al. [27] and Oh et al. [28] demonstrated that YouTube engagement signals influence real-world outcomes (box-office revenue; stock market power), underscoring the practical significance of understanding channel-level reach.

3. Materials and Methods

3.1. Dataset Description

We use the YouNiverse dataset [21] (DOI: 10.5281/zenodo.4650046), covering English-language YouTube channels with more than 10,000 subscribers and more than 10 videos as of October 2019. Two files are used: `df_channels_en.tsv.gz` (136,470 channels; category, creation date, subscriber count, video count, subscriber rank) and `df_timeseries_en.tsv.gz` (18,872,499 weekly observations; cumulative views, subscribers, videos, weekly deltas, and upload activity).

Table 1 reports missing-value rates; values are below 0.11% and no imputation is applied.

Table 1. Missing-value rates after daylight-saving-time deduplication.

File	Column	Missing (*n*)	(%)
df_channels_en	category_cc	128	0.09
df_channels_en	name_cc	10	0.01
df_timeseries_en	category	20,584	0.11

3.2. Preprocessing

Daylight saving time artefact. At each seasonal clock change, timestamps appear at both `00:00:00` and `01:00:00` for the same calendar day. We remove duplicates by flooring datetime values to day precision and retaining the first occurrence per channel–day pair.

Negative delta removal. Rows with negative `delta_views` (0.01%) are removed as data collection errors.

After preprocessing, the dataset contains 18,872,499 observations for 133,516 unique channels spanning 2015-01-05 to 2019-09-30 (Table 2).

Table 2. Time-series dataset overview after preprocessing.

Metric	Value
Total rows (clean)	18,872,499
Unique channels	133,516
Mean weeks per channel	141.4
Median weeks per channel	156.0
P5–P95 weeks per channel	35–185
Period	5 January 2015 to 30 September 2019

The channel counts reported at different stages refer to successive filters in the pipeline: 136,470 channels are present in the raw channel-metadata file; 133,516 remain in the cleaned weekly time series; 133,364 have sufficient cleaned observations for diffusion-feature extraction; and 131,105 complete cases remain in the final modelling matrix after excluding rows with missing model features or targets.

Scope. All features are derived from channel-level aggregates over the complete observed time window. This is a *characterisation* study: we ask what distinguishes channels with high peak coverage given their complete observed trajectory, not a forecasting task where predictions must be made from early observations only. Implications for temporal leakage are discussed in Section 4.8. Accordingly, the reported model scores quantify retrospective explanatory/discriminative performance and should not be interpreted as deployable early-forecast accuracy.

3.3. Feature Engineering

For each channel, we derive 10 scalar features and 14 category dummies. Table 3 provides the complete listing.

Table 3. Feature set used in machine learning models.

Feature	Group	Description	Transform
time_to_peak	Temporal	Weeks to peak Δviews	None
decay_rate	Temporal	Mean post-peak/peak views	Clip [0, 1]
coeff_variation	Temporal	Std/mean Δviews	Clip p99.9
zero_frac	Temporal	Fraction of zero-view weeks	None
n_weeks	Temporal	Total weeks observed	None
log_upload_freq	Behavioural	Mean videos per week	$\log_1 p$
log_subs	Structural	Subscribers (October 2019 crawl)	$\log_{10}$
log_rank	Structural	Subscriber rank (October 2019 crawl)	$\log_{10}$
log_sub_growth	Structural	Relative subscriber growth	$\log_1 p$
channel_age_days	Structural	Days from obs. start to peak week	None
cat_* (14)	Categorical	Content category dummies	OHE

Structural features are post hoc for channels peaking before 2019; see Section 4.8.

Temporal diffusion features. *Time-to-peak* (t_{peak}): weeks from the first observation to the week of maximum `delta_views`:

$$t_{\text{peak}} = \text{argmax}_{t \in \{1,\ldots,n\}} \Delta v_t \tag{1}$$

This feature is directly analogous to the *peak speed time* used in Bakenova et al. [6] and is the primary temporal predictor whose cross-platform generalisability we assess. *Decay rate* (d): mean post-peak weekly views divided by peak value:

$$d = \frac{1}{n - t_{\text{peak}} - 1} \sum_{t=t_{\text{peak}}+1}^{n} \frac{\Delta v_t}{\Delta v_{t_{\text{peak}}}} \tag{2}$$

Coefficient of variation (CV): $\sigma(\Delta v)/\mu(\Delta v)$. *Zero-week fraction*: proportion of weeks with zero view growth. *Observation length* (n): total weeks observed.

Behavioural feature. *Upload frequency*: $\log_1 p$ (mean new videos per week).

Structural features. $\log_{10}$ (subscriber count) and $\log_{10}$ (subscriber rank) are both measured at October 2019 crawl time and are near-perfectly collinear ($\rho = -0.994$), representing a single latent dimension of channel size. We therefore treat these two variables as alternative proxies for the same channel-size dimension and do not interpret their individual SHAP shares as independent effects; their grouped contribution is reported in Section 4.6. $\log_1 p$ (subscriber growth) and channel age at peak complete the set.

Categorical feature. A total of 14 binary dummies for YouTube content categories (OHE; *Autos & Vehicles* as reference). Table 4 lists all 15 categories.

Table 4. Channel distribution across the 15 content categories.

Category	Channels	(%)
Music	24,285	17.81
Entertainment	22,951	16.83
Gaming	20,143	14.77
People & Blogs	18,413	13.51
Howto & Style	11,875	8.71
Education	7803	5.72
Film and Animation	6875	5.04
Sports	5148	3.78
Science & Technology	4864	3.57
Comedy	3767	2.76
Autos & Vehicles	3705	2.72

Table 4. *Cont.*

Category	Channels	(%)
News & Politics	2263	1.66
Travel & Events	1989	1.46
Pets & Animals	1292	0.95
Nonprofits & Activism	969	0.71

Figure 1 shows the Spearman correlation of each feature with both target variables. Structural features dominate the regression target; temporal shape features provide a largely independent signal for the classification target.

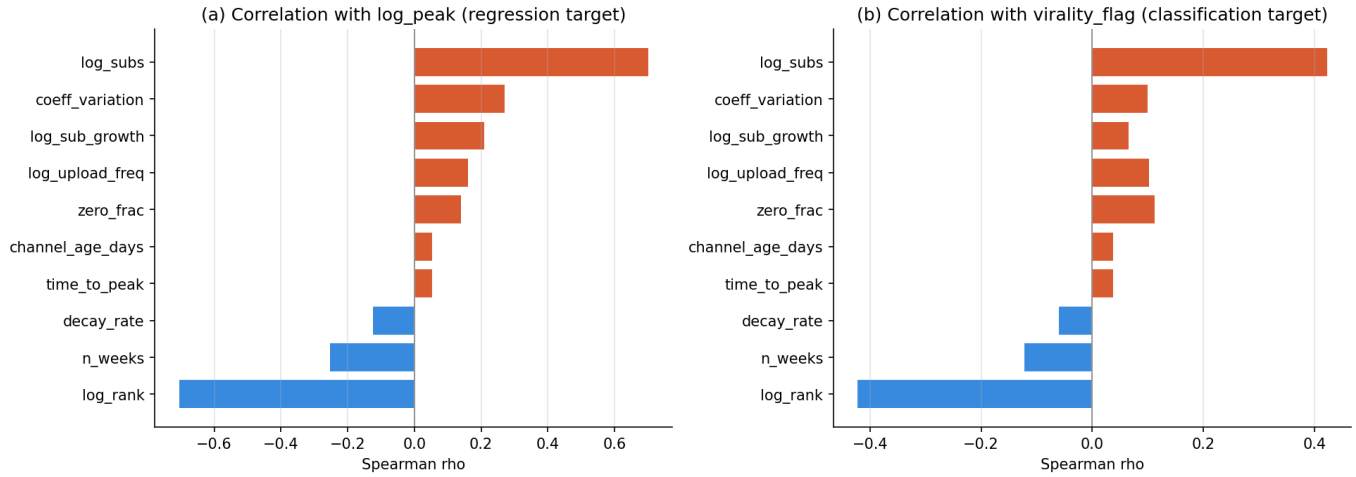


Figure 1. Feature–target Spearman correlations. (**a**) Correlation with the regression target $\log_{10}(\Delta v_{\text{peak}})$: structural size features (log_rank, log_subs) dominate. (**b**) Correlation with the virality classification target: signal is more distributed, with temporal shape features (coeff_variation, decay_rate) contributing independently of size. Orange bars indicate positive Spearman correlation; blue bars indicate negative correlation.

3.4. Target Variables

Regression: $y_{\text{reg}} = \log_{10}(\Delta v_{\text{peak}})$.

Classification: virality flag $y_{\text{clf}} \in \{0, 1\}$, equal to 1 if the channel's peak falls in the top 10% within its content category, 0 otherwise. Within-category thresholds ensure virality is measured relative to the channel's peer group. The resulting class balance is approximately 90:10 across 131,105 channels (Table 5).

Table 5. Virality flag class balance ($n = 131{,}105$).

Class	Count	Percentage
Non-viral (0)	118,004	90.01%
Viral (1)	13,101	9.99%

3.5. Models and Baselines

Naive baseline. A linear model using $\log_{10}$ (subscribers) alone, quantifying what channel size alone can explain.

Linear Regression/Logistic Regression. Full-feature linear model with StandardScaler normalisation, implemented using scikit-learn [29]. Logistic Regression uses `class_weight=balanced`.

Random Forest [16]. A total of 300 trees, maximum depth 12, minimum 10 samples per leaf.

LightGBM [15]. A total of 500 estimators, learning rate 0.05, 63 leaves, feature and sample sub-sampling rates of 0.8.

Software. All preprocessing, feature engineering, and modelling were implemented in Python 3.11. Key libraries: pandas $\geq$ 2.0, NumPy $\geq$ 1.26, scikit-learn [29] $\geq$ 1.4, LightGBM [15] $\geq$ 4.3, SHAP [17,18] 0.45.0, matplotlib $\geq$ 3.8, and seaborn $\geq$ 0.13. The complete dependency list is available in the `requirements.txt` file of the code repository (see Data Availability Statement).

3.6. Evaluation Protocol

Random split. A 80:20 stratified split, plus 5-fold cross-validation on the full dataset.

Temporal split. Channels partitioned by peak date: those peaking before 1 January 2018 form the training set (51,084 channels); those peaking in 2018–2019 form the test set (80,021 channels). The test set is larger, reflecting accelerated growth during 2018–2019. Since $\log_{10}$ (subscribers) and $\log_{10}$ (rank) are measured at October 2019 for all channels, the temporal split evaluates cross-temporal robustness of model rankings, not strict forecasting.

Metrics. Regression: R^2, RMSE, and MAE in log space, RMSE in original view counts. Classification: F1, ROC-AUC, precision, and recall. Feature importance: LightGBM gain scores (T16) and SHAP values [17,18] on 5000 held-out instances (T16b). All code is available at https://github.com/KuznetsovKarazin/Algorithmic-Diffusion-on-YouTube (accessed on 2 September 2026).

4. Results

4.1. Descriptive Statistics

Table 6 summarises channel metadata. The subscriber distribution is strongly right-skewed (median: 42,400; mean: 246,602; maximum: 112 million). The four largest categories—Music (17.8%), Entertainment (16.8%), Gaming (14.7%), and People & Blogs (13.5%)—account for 62.9% of channels (Figure 2). The median channel is observed for 156 weeks (P5–P95: 35–185 weeks; Table 2).

Table 6. Channel metadata descriptive statistics (n = 136,470).

Metric	Mean	Std	Min	P25	P50	Max
Subscribers	246,602	1,150,096	10,000	18,886	42,400	112,139,463
Videos	700	4525	10	70	175	461,923
Sub. rank	357,009	271,114	3	116,131	301,567	1,030,844

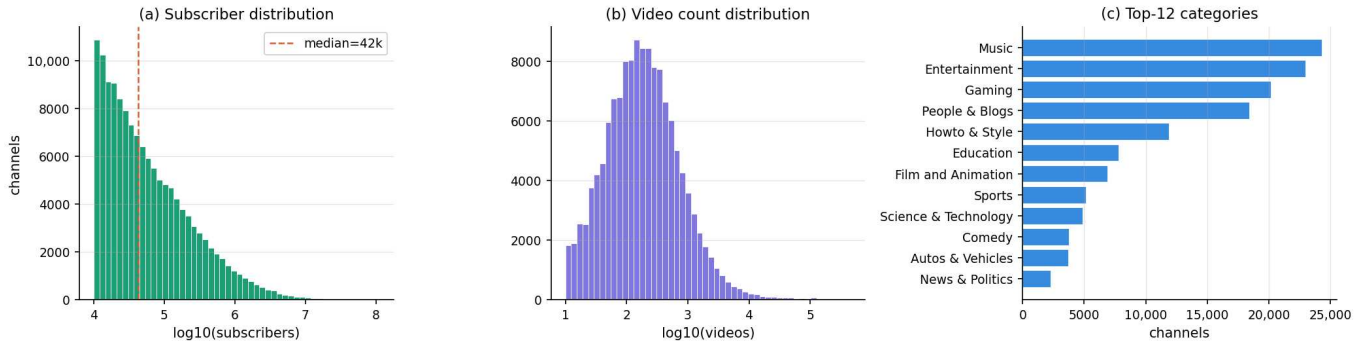


Figure 2. Channel structural characteristics (n = 136,470). (**a**) Log_{10}-transformed subscriber distribution (median = 42k). (**b**) Log_{10}-transformed video count distribution. (**c**) Distribution across the top-12 content categories.

The platform-level aggregate dynamics are shown in Figure 3. View growth increases continuously from 2015 to mid-2018, followed by relative stabilisation. Active channel counts grow throughout, reflecting dataset expansion.

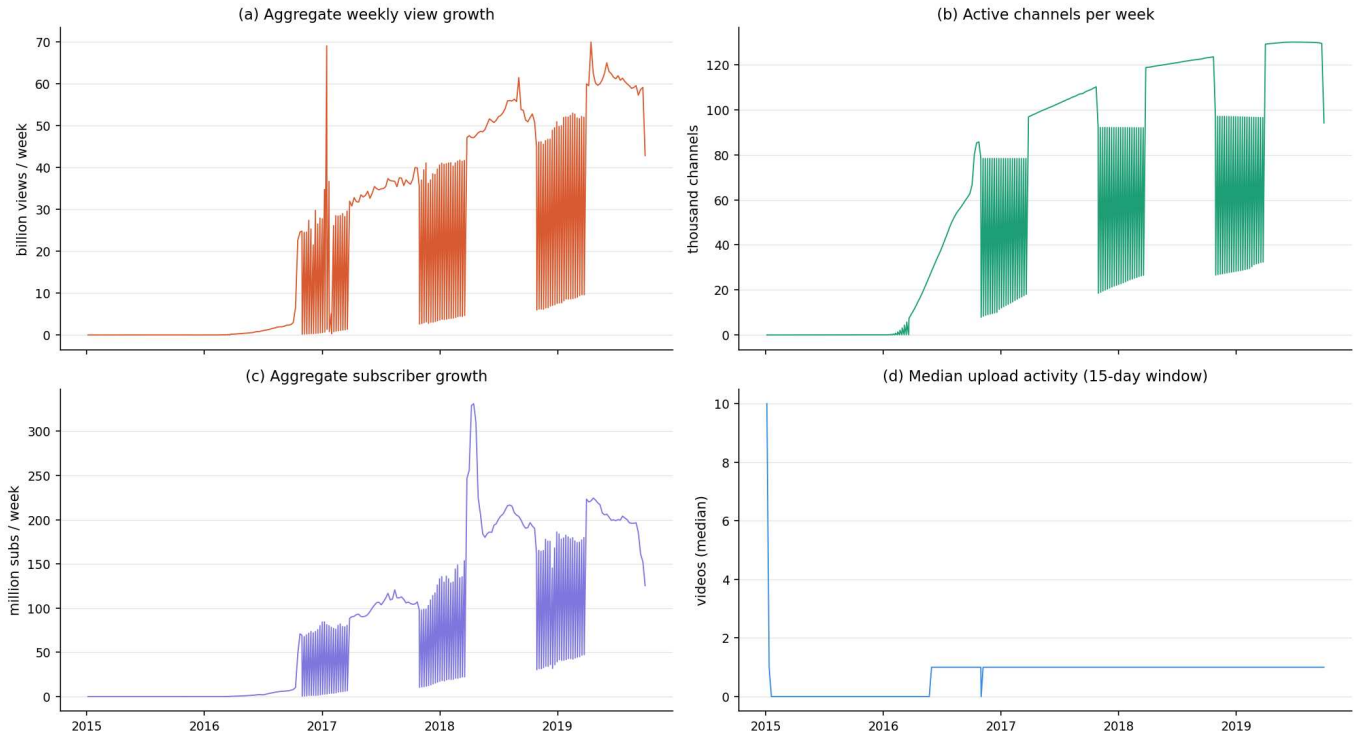


Figure 3. Platform-level weekly dynamics (January 2015 to September 2019) after daylight-saving-time deduplication. (**a**) Aggregate weekly view growth. (**b**) Active channels per week. (**c**) Aggregate subscriber growth. (**d**) Median upload activity (15-day window).

4.2. Diffusion Feature Analysis

Table 7 and Figure 4 present the derived diffusion features.

Table 7. Diffusion feature descriptive statistics (n = 133,364 channels).

Feature	Mean	Std	Min	P25	P50	Max
Time-to-peak (wk)	68.9	50.4	0	25	57	230
Decay rate	0.317	0.205	0.000	0.161	0.277	1.000
Coeff. variation	0.988	0.745	0.056	0.525	0.785	11.53 *
Zero-week fraction	0.026	0.052	0.000	0.006	0.011	1.000
Upload freq (v/wk)	2.22	8.59	0.000	0.268	0.737	644.1 *
Weeks observed	141.5	43.8	4	122	156	248
Peak Δviews	1983 k	22,594 k	0	98 k	284 k	6.66 B
Sub. growth	540.3	6049	−0.40	1.58	8.34	867,499 *

* Clipped at the 99.9th percentile before $\log_1 p$ transformation.

The median **time-to-peak is 57 weeks** (≈14 months), contrasting sharply with hours on Twitter [3] and days on Telegram [7]. Decay rate has a median of 0.277, indicating that channels retain roughly 28% of peak engagement after the peak—substantially more persistent than viral cascades on other platforms. Coefficient of variation (median 0.785) captures substantial heterogeneity in diffusion curve shapes.

Figure 5 illustrates three archetypal diffusion patterns: fast-rise channels (peak $<$ 10 weeks), slow-rise channels (peak $>$ 50 weeks), and flat/sustained channels (CV $<$ 0.5).

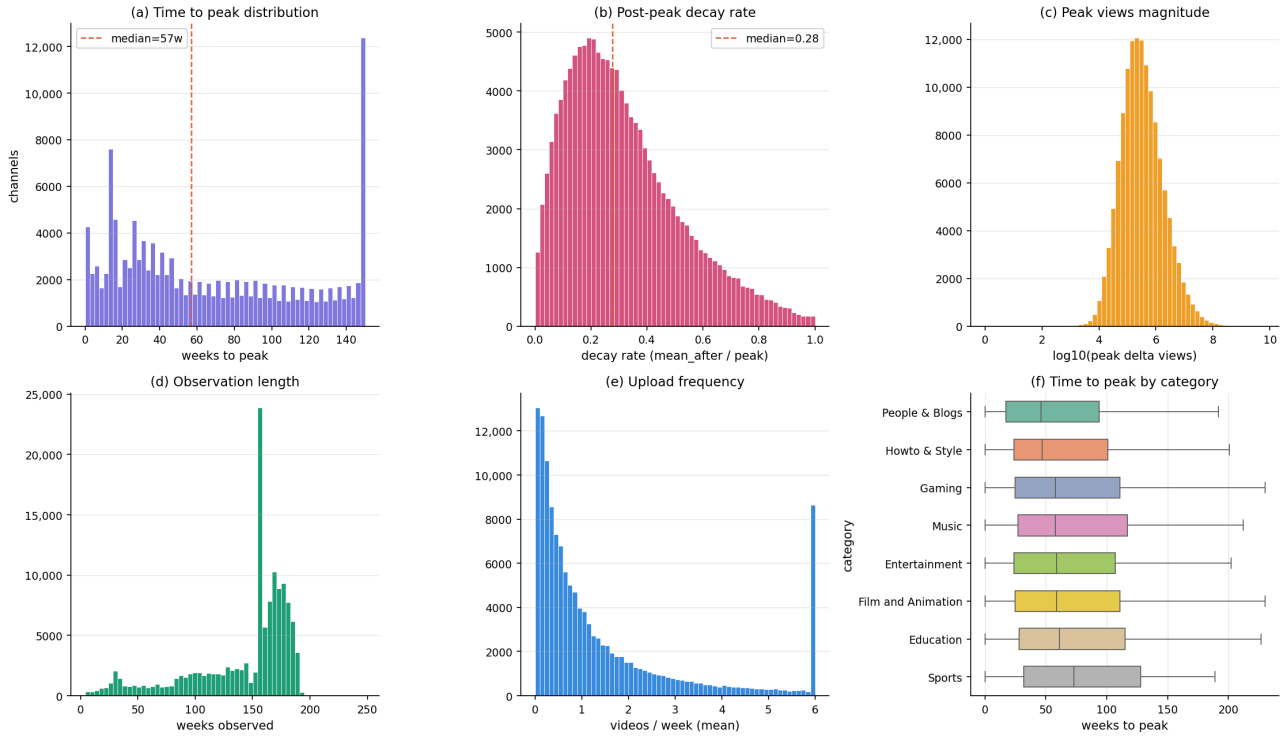


Figure 4. Distributions of derived diffusion features (n = 133,364). (**a**) Time-to-peak (median = 57 weeks). (**b**) Post-peak decay rate (median = 0.277). (**c**) Peak views in $\log_{10}$ scale. (**d**) Observation length. (**e**) Upload frequency. (**f**) Time-to-peak by content category (top-8).

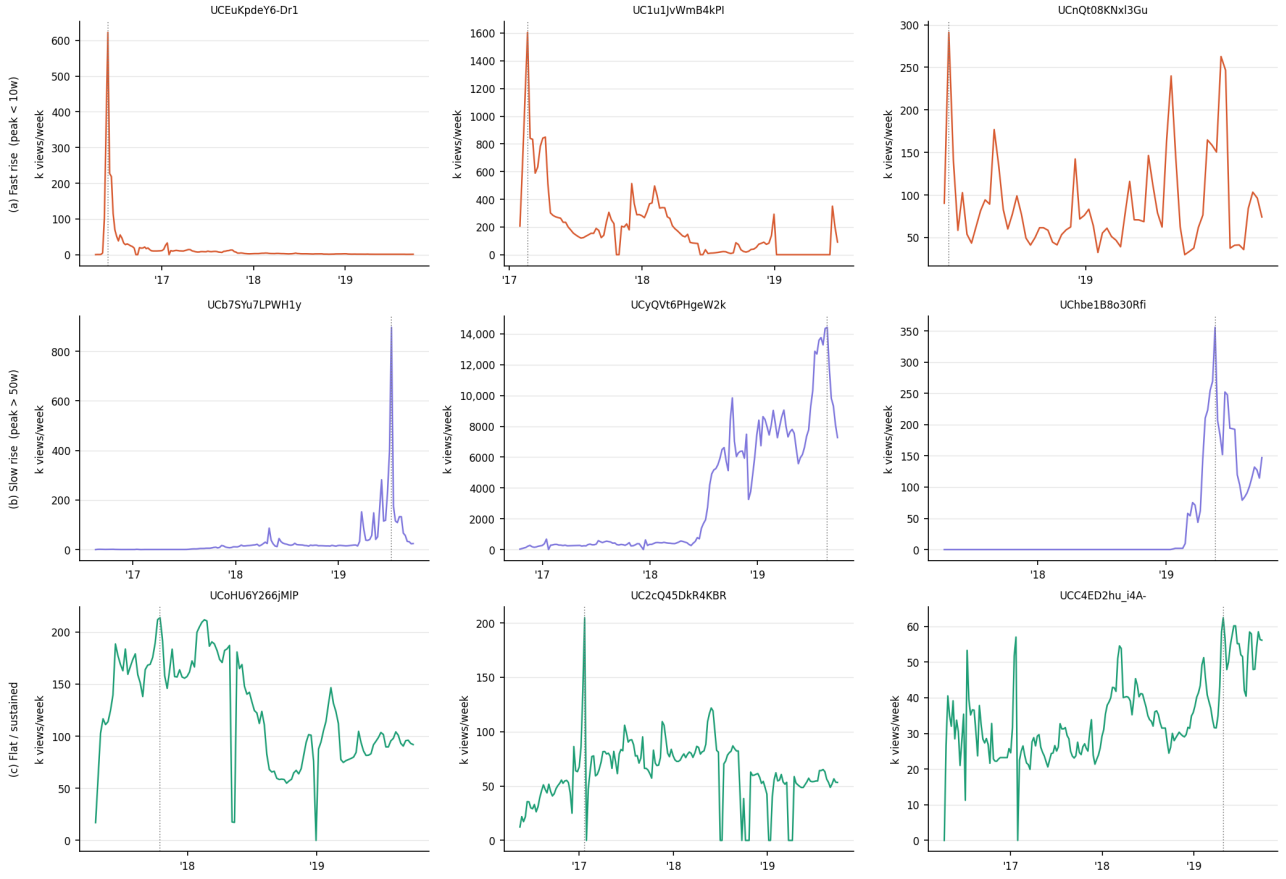


Figure 5. Diffusion curve archetypes (weekly Δviews per channel). (**a**) Fast-rise channels (time-to-peak < 10 weeks). (**b**) Slow-rise channels (time-to-peak > 50 weeks). (**c**) Flat/sustained channels (coefficient of variation < 0.5). Dashed vertical lines mark the peak week.

4.3. Correlation Analysis

Table 8 and Figure 6 report Spearman correlations with the regression target.

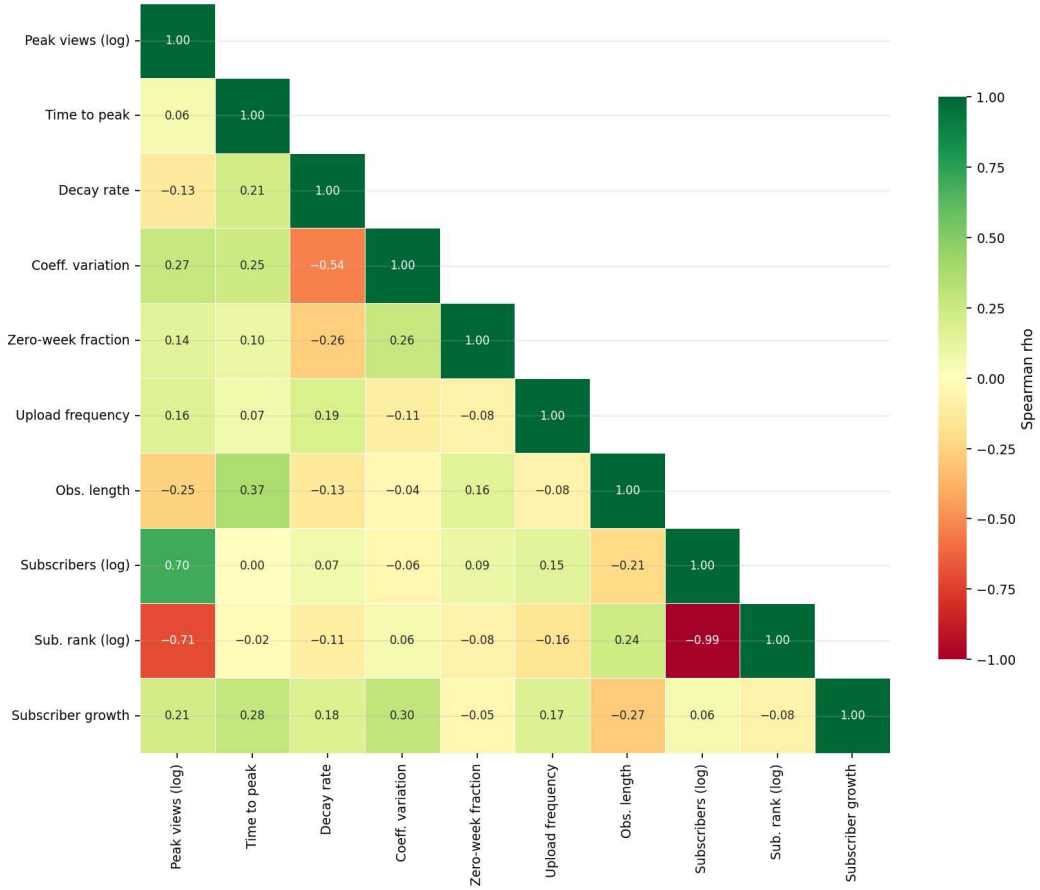


Figure 6. Spearman correlation matrix for diffusion and structural features. log_subs and log_rank are near-perfectly collinear ($\rho = -0.994$). Time-to-peak has near-zero correlation with the regression target ($\rho = 0.056$).

Table 8. Spearman correlations with $\log_{10}(\Delta v_{\text{peak}})$. Full matrix in Figure 6.

Feature	Spearman ρ	$\lvert\rho\rvert$
log_rank	−0.708	0.708
log_subs	+0.703	0.703
coeff_variation	+0.273	0.273
n_weeks	−0.255	0.255
sub_growth	+0.212	0.212
upload_freq	+0.163	0.163
zero_frac	+0.142	0.142
decay_rate	−0.126	0.126
time_to_peak	**+0.056**	**0.056**
log_subs vs. log_rank collinearity:		$\rho = -0.994$

Bold denotes time_to_peak, the feature of central comparative interest in this study.

The two features most correlated with the target are subscriber rank ($\rho = -0.708$) and subscriber count ($\rho = 0.703$), which are near-perfectly collinear ($\rho = -0.994$) and represent a single latent dimension of channel size. The key comparative result is:

$$\rho\Big(t_{\text{peak}},\ \log_{10}(\Delta v_{\text{peak}})\Big) = 0.056 \tag{3}$$

compared with $r = 0.995$ on Reddit [6].

4.4. Naive Baseline and Full-Model Comparison

The subscriber-only naive model achieves $R^2 = 0.548$ (CV: 0.545 ± 0.005), confirming that channel size is a strong predictor but that the full model adds a substantial $\Delta R^2 = +0.228$. Because subscriber count and subscriber rank are near-duplicate proxies of channel size, this subscriber-only model also provides a transparent size-only reference using one representative of that latent dimension. It is not used to claim independent effects for the two correlated proxies. Table 9 and Figure 7 report all four models on the held-out test set.

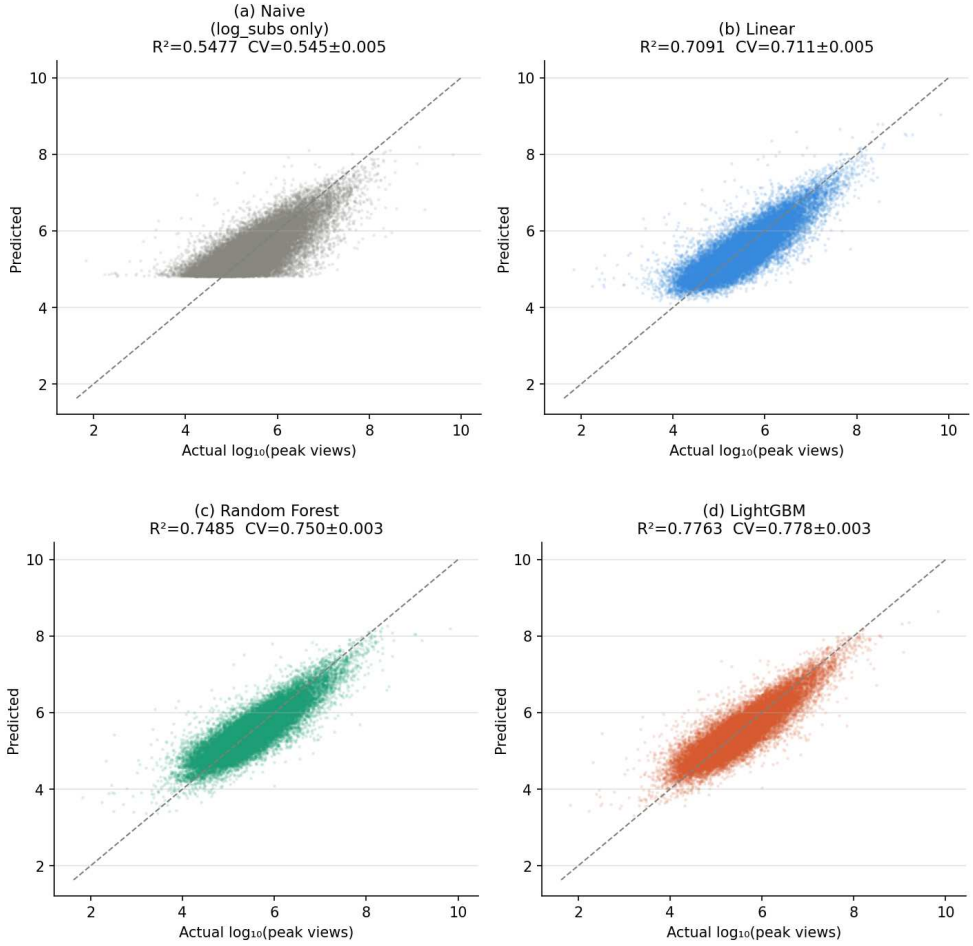


Figure 7. Regression: predicted vs. actual $\log_{10}(\Delta v_{\text{peak}})$ on the held-out test set. (**a**) Naive baseline (log_subs only; $R^2 = 0.548$): predictions compressed to a narrow range, illustrating the limitation of a single predictor. (**b**) Linear Regression ($R^2 = 0.709$). (**c**) Random Forest ($R^2 = 0.749$). (**d**) LightGBM ($R^2 = 0.776$). Dashed diagonal = perfect prediction; CV mean $\pm$ std shown in each panel.

Table 9. Regression results: random 80/20 split and 5-fold CV. Target: $\log_{10}(\Delta v_{\text{peak}})$.

Model	R^2	RMSE_{log}	MAE_{log}	$\text{RMSE}_{\text{views}}$	CV R^2	CV RMSE
Naive (log_subs)	0.548	0.499	0.388	43.4 M	0.545 ± 0.005	0.500 ± 0.002
Linear	0.709	0.400	0.308	36.9 M	0.711 ± 0.005	0.399 ± 0.003
Random Forest	0.749	0.372	0.290	42.7 M	0.750 ± 0.003	0.371 ± 0.001
LightGBM	**0.776**	**0.351**	**0.270**	40.9 M	$\mathbf{0.778 \pm 0.003}$	$\mathbf{0.349 \pm 0.002}$
Gain vs. naive (LGB)	+0.228	−0.148	−0.118			

Bold indicates the best-performing model. $\text{RMSE}_{\text{views}}$: dominated by channels with peak $> 10^8$ weekly views; RMSE_{log} is the primary metric.

LightGBM achieves $R^2 = 0.776$ (CV: 0.778 ± 0.003). The 7.1-pp gap below the Reddit result ($R^2 = 0.847$, [6]) is attributable to the unpredictability of algorithmically amplified bursts. The $\text{RMSE}_{\text{views}}$ anomaly—where the linear model (36.9 M) outperforms LightGBM (40.9 M)—is a scale artefact: a small number of channels with extreme peak values dominate $\text{RMSE}_{\text{views}}$ but are down-weighted in log-space training.

4.5. Classification Results

Table 10 and Figure 8 report virality prediction performance. LightGBM achieves ROC-AUC = 0.967 (CV: 0.966 ± 0.001), compared with 0.909 for the naive baseline—a gain of +0.058 AUC. Precision = 0.545 and recall = 0.874 indicate that the model errs toward over-predicting virality, which is preferable in audience-discovery applications.

Table 10. Classification results: random split and 5-fold stratified CV. Target: virality_flag.

Model	F1	ROC-AUC	Prec.	Recall	CV F1	CV AUC
Naive (log_subs)	0.496	0.909	0.355	0.826	0.495 ± 0.004	0.908 ± 0.005
Linear	0.604	0.957	0.455	0.899	0.603 ± 0.004	0.957 ± 0.001
Random Forest	0.638	0.961	0.502	0.876	0.637 ± 0.004	0.960 ± 0.002
LightGBM	**0.671**	**0.967**	0.545	0.874	$\mathbf{0.668 \pm 0.002}$	$\mathbf{0.966 \pm 0.001}$

Bold indicates the best-performing model.

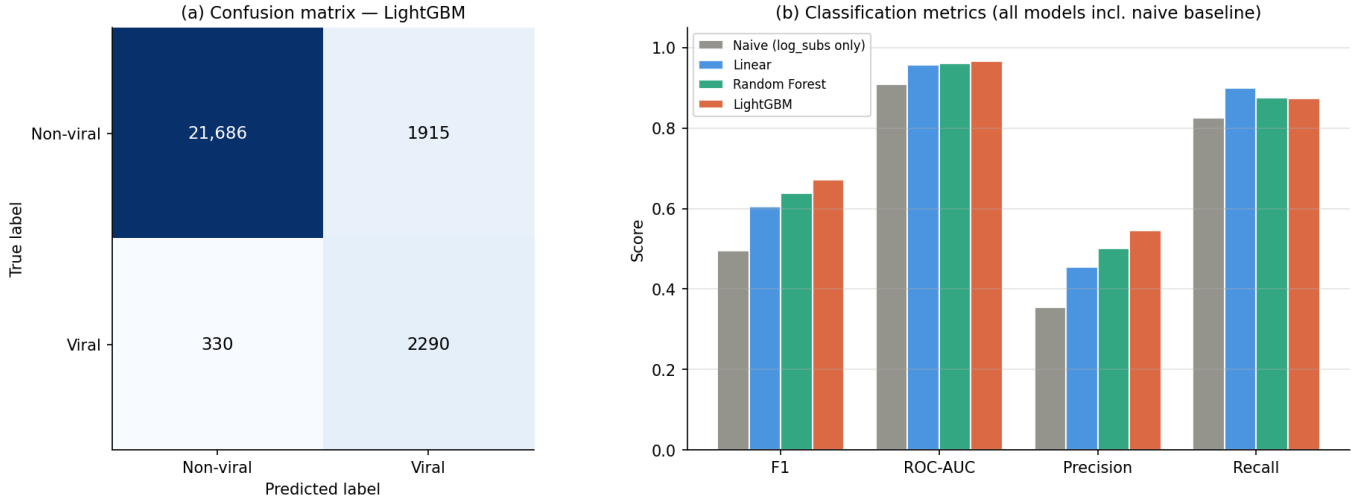


Figure 8. Classification performance (virality_flag = top-10% within category). (**a**) Confusion matrix for LightGBM. (**b**) F1, ROC-AUC, Precision, and Recall across all four models.

4.6. Feature Importance and Interpretability

Table 11 reports gain-based (T16) vs. SHAP (T16b) importance for the top features.

Discrepancy between gain and SHAP. Gain places `log_upload_freq` first (12.6%) while SHAP places `log_rank` first (29.3%). Two effects explain this. First, gain is biased toward high-cardinality features with many split points [18]. Second, the collinearity of log_subs and log_rank ($\rho = -0.994$) causes gain to assign shared information to whichever feature splits first, making the other appear unimportant. SHAP provides additive local attributions, but under strong predictor dependence the shared signal can be distributed between correlated variables in a model- and background-dependent way. With the near-perfect dependence observed here, the allocation between the two variables is therefore not uniquely interpretable as two independent scientific effects. We avoid treating the 29.3% for `log_rank` and 12.9% for `log_subs` as separate mechanisms.

Table 11. LightGBM gain vs. SHAP mean absolute attribution, top-10 by SHAP rank.

SHAP Rank	Feature	Gain Rank	Gain (%)	SHAP (%)
1	log_rank	9	7.8	**29.3**
2	coeff_variation	3	11.5	13.7
3	log_subs	7	8.0	12.9
4	log_sub_growth	2	12.0	8.9
5	decay_rate	4	10.3	8.3
6	log_upload_freq	**1**	**12.6**	6.1
7	n_weeks	5	10.0	5.4
8	cat_Howto&Style	13	1.1	3.2
9	cat_People&Blogs	15	0.9	2.3
10	cat_Education	14	0.9	2.0
14	**time_to_peak**	8	7.9	**1.1**

Bold marks: the top SHAP-ranked feature (row 1) and the top gain-ranked feature (row 6, Gain Rank/Gain % columns), for visual contrast between the two importance metrics; and time_to_peak (last row), the feature of central comparative interest in this study. SHAP is the primary interpretability metric; see text for the gain/SHAP discrepancy.

SHAP findings (Figure 9). Taken together, subscriber rank (29.3%) and subscriber count (12.9%) account for 42.2% of total mean absolute SHAP attribution. The separate values are retained for transparency, but the scientifically interpretable quantity is their grouped channel-size contribution rather than either value in isolation. Temporal and behavioural features add the remaining signal: coefficient of variation (13.7%), subscriber growth (8.9%), and decay rate (8.3%). **Time-to-peak ranks fourteenth (1.1%)**—below several category dummies—confirming that peak timing carries almost no information about peak magnitude on YouTube.

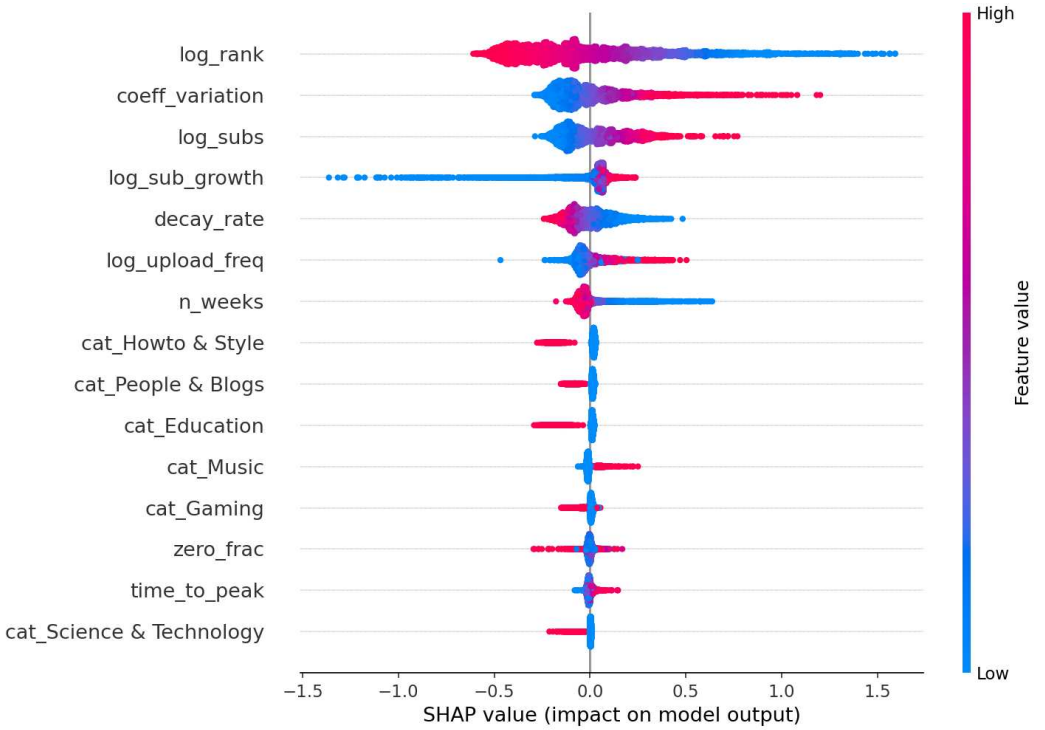


Figure 9. SHAP summary plot for LightGBM regression (5000-sample subsample). Each row is a feature; each point is a channel. Colour encodes feature value (red = high, blue = low). log_rank accounts for 29.3% of total mean absolute SHAP attribution; time_to_peak ranks 14th (1.1%). Because log_rank and log_subs are near-perfectly collinear, their individual SHAP percentages should be read as a shared channel-size signal rather than as independent effects.

4.7. Category-Stratified Analysis

Table 12 reports LightGBM performance by content category. R^2 ranges from 0.846 (Pets & Animals) to 0.678 (Nonprofits & Activism). Notably, the best-predicted categories are not the largest: Pets & Animals ($R^2 = 0.846$, $n = 267$), Howto & Style ($R^2 = 0.831$, $n = 2293$), and News & Politics ($R^2 = 0.814$, $n = 459$) outperform the two largest—Music ($R^2 = 0.736$, $n = 4627$) and Gaming ($R^2 = 0.736$, $n = 3829$). This pattern suggests that internal heterogeneity, not sample size, determines predictability: large categories contain extreme diversity in content strategy and audience type (live-streaming, tutorials, music videos, shorts), reducing within-category homogeneity. This is consistent with Halim et al.'s [23] finding that trending dynamics differ substantially across content markets. Nonprofits & Activism ($R^2 = 0.678$, $n = 189$) is the worst-predicted category, combining a small sample with event-driven upload patterns.

Table 12. LightGBM performance stratified by content category (test set; $n < 20$ excluded).

Category	n	R^2	F1	AUC
Pets & Animals	267	0.846	0.710	0.977
Howto & Style	2293	0.831	0.652	0.971
News & Politics	459	0.814	0.752	0.974
Sports	991	0.810	0.717	0.977
Education	1515	0.803	0.665	0.967
Entertainment	4420	0.801	0.717	0.973
Film and Animation	1330	0.777	0.682	0.973
Comedy	747	0.775	0.718	0.969
Science & Technology	907	0.771	0.689	0.967
Autos & Vehicles	734	0.740	0.735	0.972
Gaming	3829	0.736	0.651	0.966
Music	4627	0.736	0.668	0.966
Travel & Events	398	0.730	0.653	0.975
People & Blogs	3515	0.716	0.616	0.952
Nonprofits & Activism	189	0.678	0.571	0.928

Music and Gaming—the two largest categories—rank 11th and 12th, reflecting internal content diversity.

4.8. Temporal Split Experiment

Table 13 and Figure 10 report R^2 under random and temporal splits.

Table 13. Temporal split regression results (train: peak < 2018, $n = 51{,}084$; test: peak ≥ 2018, $n = 80{,}021$).

Model	R^2 (Random)	R^2 (Temporal)	Drop (pp)	RMSE_{log} (Temporal)
Naive (log_subs)	0.548	0.482	−6.6	0.530
Linear	0.709	0.681	−2.8	0.416
Random Forest	0.749	**0.703**	−4.6	**0.401**
LightGBM	0.776	0.683	−9.3	0.414

Bold indicates the best-performing model under the temporal split. Rank reversal: RF ($R^2 = 0.703$) > LightGBM ($R^2 = 0.683$) under temporal split.

Random Forest outperforms LightGBM under the temporal split. Under the random split, LightGBM leads by 2.7 pp; under the temporal split, this reverses: RF leads by 2.0 pp. LightGBM's drop is 9.3 pp—the largest among all models—while RF drops only 4.6 pp. This is consistent with the known property that boosted models, by minimising residuals iteratively, fit more tightly to the training distribution and are more susceptible to covariate shift. RF's bagging-based regularisation makes it more conservative and robust. Within this retrospective design, RF therefore shows greater cross-temporal stability. This result should

not be interpreted as validated future-deployment forecasting, because both temporal partitions still include October 2019 crawl-time subscriber metadata.

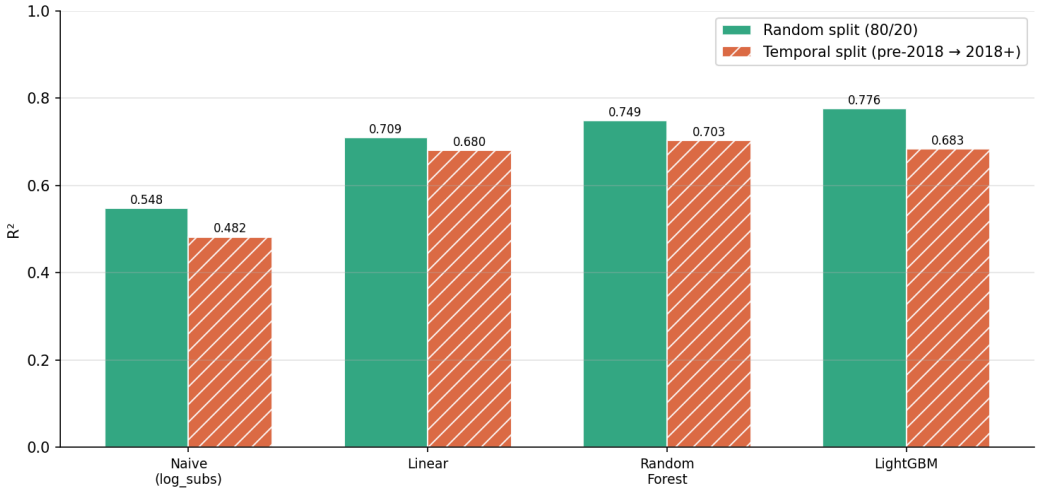


Figure 10. R^2 comparison: random split (teal) vs. temporal split (hatched coral; train $<$ 2018; test 2018–2019). The rank reversal of Random Forest ($R^2 = 0.703$) over LightGBM ($R^2 = 0.683$) indicates greater cross-temporal stability of the more conservative ensemble within this retrospective split; it is not a strict future-forecasting test.

4.9. Cross-Platform Comparison

The three studies are not a controlled cross-platform benchmark: they use different datasets, prediction targets, units of analysis, and evaluation metrics. Table 14 and Figure 11 are therefore intended as a conceptual comparison of diffusion patterns rather than a direct numerical comparison of model performance. Descriptively, time-to-peak is strongly associated with coverage in the Reddit study and weakly associated with peak channel growth here; the Telegram study occupies an intermediate, partially structural role. This pattern is consistent with, but does not establish, an architecture hypothesis concerning algorithmic mediation. Median time-to-peak also differs substantially across the three studies, from hours–days on Reddit to days on Telegram and 57 weeks on YouTube.

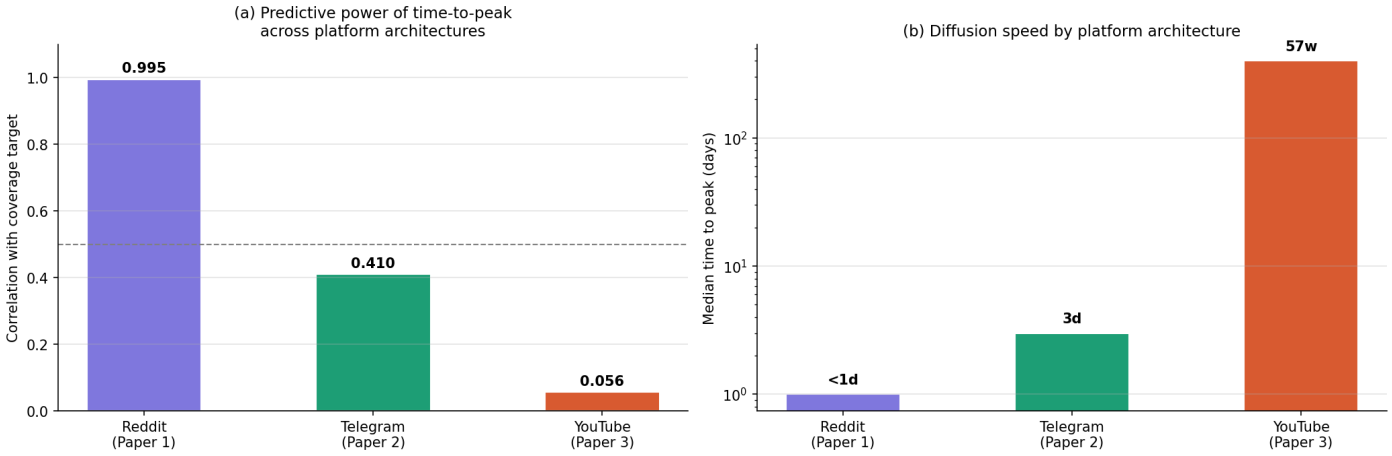


Figure 11. Cross-platform comparison. (**a**) Bivariate association of time-to-peak with coverage target. (**b**) Median time-to-peak on a logarithmic scale. The dashed line in panel (**a**) marks the 0.5 correlation threshold for reference.

Table 14. Cross-platform comparison across the three studies in this series. Metrics are task-specific and are not presented as a common performance scale.

Property	Reddit [6]	Telegram [7]	YouTube (This Study)
Platform type	Graph (thread cascades)	Broadcast (star, depth = 1)	Algorithmic recommendation
Best metric	$R^2 = 0.847$	Med. error $< 10\%$	$R^2 = 0.776$; AUC = 0.967
Naive R^2	N/A	N/A	0.548
Temporal robust	N/A	N/A	RF > LightGBM
Median TTP	Hours–days	Days	57 weeks
TTP bivariate	$r = 0.995$	Partial	$\rho = 0.056$
TTP SHAP rank	#1	N/A	#14 (1.1%)
Top predictor	peak_speed_time	Delay distribution	Size group (42.2% SHAP)

5. Discussion

5.1. Main Findings and Theoretical Implications

The central finding—near-zero SHAP attribution of time-to-peak on YouTube (1.1%, rank #14) vs. its dominant role on Reddit ($r = 0.995$, rank #1 [6])—is consistent with a platform-level distribution process in which recommendation can reduce dependence on early timing. On Reddit, a post either achieves engagement quickly or it does not. YouTube's recommendation algorithm can surface a channel weeks or months after publication, creating growth waves structurally uncorrelated with early timing. The 57-week median time-to-peak is consistent with such delayed, recommendation-mediated growth, although it is not by itself causal evidence of the recommendation mechanism.

This has broad implications for diffusion theory. Most models encode the assumption that early timing is informative: threshold models, cascade models, and epidemic models all assign predictive value to initial spread speed [1,3,5]. Within the 2015–2019 YouNiverse data, our results are consistent with the hypothesis that this assumption is more informative when contagion is primary and less informative when algorithmic curation mediates exposure. Because platform, period, prediction task, and unit of analysis also differ, the cross-platform evidence is descriptive rather than causal. Whether the same pattern persists in current recommendation systems requires direct validation on post-2019 data.

5.2. Comparison with Video-Level YouTube Studies

Our work differs from the majority of YouTube prediction literature in the unit of analysis (channel vs. video) and temporal horizon (years vs. days).

At the video level, temporal features retain predictive power. Szabo and Huberman [9] reported a strong correlation between early and long-term view counts. Chen and Chang [11] achieved strong performance using upload-time features only. Figueiredo et al. [12] reported F1 improvements of 38% while retaining substantial residual views.

This contrast is not contradictory. Individual videos are released at discrete moments and their fate is partly determined by early sharing energy. Channels are judged by the recommendation algorithm based on accumulated engagement history. A channel that consistently publishes trains the algorithm to recommend it widely, regardless of when any individual upload first attracted attention. This interpretation is consistent with the long 57-week median time-to-peak and the strong grouped SHAP attribution of the channel-size dimension (42.2% across subscriber rank and subscriber count).

Regarding multi-modal approaches: Chen and Wan [24], Le et al. [25], and Cho et al. [26] achieved strong results combining content features with metadata for video-level predictions. Our channel-level approach uses no content features and achieves $R^2 = 0.776$ from purely structural and temporal engagement signals. At the channel level, the consistency

and scale of engagement may matter more than content type—consistent with Jia et al.'s [20] finding on BiliBili that social features improved prediction beyond content.

The category-stratified results add nuance: niche categories (Pets & Animals, Howto & Style) are better predicted than the largest, most heterogeneous categories (Music, Gaming), consistent with Halim et al. [23] on market-to-market trending variation.

5.3. The Random Forest Robustness Finding

The reversal of model rankings under the temporal split—RF ($R^2 = 0.703$) outperforming LightGBM ($R^2 = 0.683$) when trained on pre-2018 channels and tested on 2018–2019—provides evidence about cross-temporal stability within the retrospective dataset. LightGBM's larger drop (9.3 pp vs. 4.6 pp for RF) is consistent with the general property that boosted models adapt more tightly to the training distribution [16] and are more susceptible to covariate shift. Accordingly, RF is the more stable model in this particular temporal split, but the experiment does not establish its superiority for real-time future forecasting because crawl-time subscriber metadata is available in both partitions.

5.4. Limitations

Temporal scope. The dataset covers 2015–2019; YouTube's algorithm has changed substantially since 2019. Consequently, exact feature rankings and model-performance values should be regarded as period-specific. The qualitative interpretation concerning algorithmic mediation may remain plausible, but it requires validation on post-2019 data before being generalised to the current recommendation system. **Channel vs. video level.** Individual video dynamics are not resolved. **Post-hoc structural features.** Subscriber count and rank are measured at October 2019 for all channels; for strict forecasting, they should be replaced by time-appropriate measurements. This prevents interpreting the reported R^2/AUC values or the temporal-split results as real-time forecast accuracy. **Cross-platform comparability.** The three platforms differ in data collection methodology, unit of analysis, and task definition; direct numerical R^2 comparison is not warranted. The comparison is therefore conceptual and hypothesis-generating, not a controlled benchmark or a basis for causal attribution to platform architecture alone. **Causal interpretation.** All associations are observational; causality would require controlled experiments or instrumental variable approaches.

5.5. Future Work

Natural extensions include (1) genuine forecasting using only features measurable in the first k weeks; (2) a Hawkes process model [30] for the weekly view time series as a probabilistic baseline; (3) application to TikTok and Instagram Reels; (4) incorporation of video-level metadata from YouNiverse to disentangle channel-level and video-level effects.

6. Conclusions

We present a characterisation study of channel-level information diffusion on YouTube using 133,364 channels from 2015 to 2019 (18.9 million observations). The main conclusions are as follows:

1. A naive subscriber-count baseline achieves $R^2 = 0.548$; LightGBM achieves $R^2 = 0.776$, a net gain of $\Delta R^2 = +0.228$ from the broader structural, temporal, behavioural, and categorical feature set.
2. Time-to-peak, the dominant predictor on Reddit ($r = 0.995$, rank #1 [6]), ranks **fourteenth in SHAP attribution** (1.1%) on YouTube. This is the central empirical finding of this study.

3. Subscriber rank (29.3%) and subscriber count (12.9%) are near-perfectly collinear proxies of one channel-size dimension and together account for 42.2% of mean absolute SHAP attribution. Their individual shares are therefore not interpreted as independent effects; the grouped channel-size contribution is the relevant quantity.
4. LightGBM achieves ROC-AUC = 0.967 for virality classification vs. 0.909 for the naive baseline (+0.058).
5. **Under the temporal split, Random Forest ($R^2 = 0.703$) outperforms LightGBM ($R^2 = 0.683$)**. This indicates greater cross-temporal stability within the retrospective split, not validated real-time forecasting superiority.
6. Niche categories (Pets & Animals, Howto & Style) are better predicted than the largest, most heterogeneous ones (Music, Gaming).
7. The cross-platform comparison is consistent with, but does not establish, the architecture hypothesis: graph diffusion appears time-sensitive, broadcast diffusion delay-driven, and algorithmic diffusion more strongly associated with size and shape. Because the studies differ in data, tasks, and units of analysis, this comparison is descriptive rather than a direct benchmark.

Author Contributions: Conceptualisation, D.T., A.S. (Aigul Shaikhanova), and O.K.; methodology, D.T., A.S. (Aigul Shaikhanova), and O.K.; software, D.T. and A.S. (Aigul Shaikhanova); validation, A.S. (Aigul Shaikhanova), A.S. (Aigerim Sambetova), and K.I.; formal analysis, O.K. and A.S. (Aigerim Sambetova); investigation, D.T., A.S. (Aigul Shaikhanova), and O.K.; data curation, D.T. and A.S. (Aigul Shaikhanova); writing—original draft preparation, D.T. and O.K.; writing—review and editing, A.S. (Aigul Shaikhanova), A.S. (Aigerim Sambetova), and K.I.; visualisation, D.T. and A.S. (Aisanim Sarsenbayeva); supervision, O.K. and A.S. (Aisanim Sarsenbayeva); project administration, O.K. and A.S. (Aisanim Sarsenbayeva). All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The YouNiverse dataset is publicly available at https://zenodo.org/record/4650046 (DOI: 10.5281/zenodo.4650046) (accessed on 2 September 2026). Analysis code is available at https://github.com/KuznetsovKarazin/Algorithmic-Diffusion-on-YouTube (accessed on 2 September 2026).

Acknowledgments: The authors thank Manoel Horta Ribeiro and Robert West (EPFL) for making the YouNiverse dataset publicly available.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

AUC	Area under the ROC curve
CV	Coefficient of variation; cross-validation (context-dependent)
DST	Daylight saving time
GBDT	Gradient-boosted decision trees
MAE	Mean absolute error
ML	Machine learning
RF	Random forest
RMSE	Root mean squared error
SHAP	SHapley Additive exPlanations
TTP	Time-to-peak
UGC	User-generated content

References

1. Kermack, W.O.; McKendrick, A.G. A contribution to the mathematical theory of epidemics. *Proc. R. Soc. Lond. A* **1927**, *115*, 700–721. [CrossRef]
2. Pastor-Satorras, R.; Castellano, C.; Van Mieghem, P.; Vespignani, A. Epidemic processes in complex networks. *Rev. Mod. Phys.* **2015**, *87*, 925–979. [CrossRef]
3. Goel, S.; Anderson, A.; Hofman, J.; Watts, D.J. The structural virality of online diffusion. *Manag. Sci.* **2016**, *62*, 180–196. [CrossRef]
4. Leskovec, J.; McGlohon, M.; Faloutsos, C.; Glance, N.; Hurst, M. Patterns of cascading behavior in large blog graphs. In Proceedings of the 7th SIAM International Conference on Data Mining (SDM 2007), Minneapolis, MN, USA, 26–28 April 2007; pp. 551–556.
5. Cheng, J.; Adamic, L.; Dow, P.A.; Kleinberg, J.; Leskovec, J. Can cascades be predicted? In Proceedings of the 23rd International Conference on World Wide Web (WWW 2014), Seoul, Republic of Korea, 7–11 April 2014; pp. 925–936.
6. Bakenova, K.; Kuznetsov, O.; Artyshchuk, I.; Shaikhanova, A.; Shevchuk, R.; Orobchuk, O. Information diffusion modeling in social networks: A comparative analysis of delay mechanisms using population dynamics. *Appl. Sci.* **2025**, *15*, 6092. [CrossRef]
7. Bakenova, K.; Kuznetsov, O.; Shaikhanova, A.; Cherkaskyi, D.; Khrushkov, B.; Chernushevych, V. Delay-driven information diffusion in Telegram. *Big Data Cogn. Comput.* **2026**, *10*, 30. [CrossRef]
8. Covington, P.; Adams, J.; Sargin, E. Deep neural networks for YouTube recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems (RecSys 2016), Boston, MA, USA, 15–19 September 2016; pp. 191–198.
9. Szabo, G.; Huberman, B.A. Predicting the popularity of online content. *Commun. ACM* **2010**, *53*, 80–88. [CrossRef]
10. Pinto, H.; Almeida, J.M.; Goncalves, M.A. Using early view patterns to predict the popularity of YouTube videos. In Proceedings of the 6th ACM International Conference on Web Search and Data Mining, Rome, Italy, 4–8 February 2013; pp. 365–374.
11. Chen, Y.-L.; Chang, C.-L. Early prediction of the future popularity of uploaded videos. *Expert Syst. Appl.* **2019**, *133*, 59–74. [CrossRef]
12. Figueiredo, F.; Almeida, J.M.; Goncalves, M.A.; Benevenuto, F. TrendLearner: Early prediction of popularity trends of user generated content. *Inf. Sci.* **2016**, *349–350*, 172–187. [CrossRef]
13. Gruhl, D.; Guha, R.; Liben-Nowell, D.; Tomkins, A. Information diffusion through blogspace. In Proceedings of the 13th International Conference on World Wide Web (WWW 2004), New York, NY, USA, 17–22 May 2004; pp. 491–501.
14. Chen, T.; Guestrin, C. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
15. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. LightGBM: A highly efficient gradient boosting decision tree. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 3146–3154.
16. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
17. Lundberg, S.M.; Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 4768–4777.
18. Lundberg, S.M.; Erion, G.; Chen, H.; DeGrave, A.; Prutkin, J.M.; Nair, B.; Katz, R.; Himmelfarb, J.; Bansal, N.; Lee, S.-I. From local explanations to global understanding with explainable AI for trees. *Nat. Mach. Intell.* **2020**, *2*, 56–67. [CrossRef] [PubMed]
19. Liu, A.-A.; Wang, X.; Xu, N.; Guo, J.; Jin, G.; Zhang, Q.; Tang, Y.; Zhang, S. A review of feature fusion-based media popularity prediction methods. *Vis. Inform.* **2022**, *6*, 78–89. [CrossRef]
20. Jia, A.L.; Shen, S.; Li, D.; Chen, S. Predicting the implicit and the explicit video popularity in a user generated content site with enhanced social features. *Comput. Netw.* **2018**, *140*, 112–125. [CrossRef]
21. Ribeiro, M.H.; West, R. YouNiverse: Large-scale channel and video metadata from English-speaking YouTube. In Proceedings of the 15th International AAAI Conference on Web and Social Media, Virtual, 7–10 June 2021. [CrossRef]
22. Ribeiro, M.H.; Ottoni, R.; West, R.; Almeida, V.A.F.; Meira, W. Auditing radicalization pathways on YouTube. In Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, 27–30 January 2020; pp. 131–141.
23. Halim, Z.; Hussain, S.; Hashim Ali, R. Identifying content unaware features influencing popularity of videos on YouTube: A study based on seven regions. *Expert Syst. Appl.* **2022**, *206*, 117836. [CrossRef]
24. Chen, Y.; Wan, Y. A framework for predicting popularity and engagement of educational videos based on contrastive feature decomposition. *Expert Syst. Appl.* **2026**, *307*, 131027. [CrossRef]
25. Le, T.; Nguyen-Thi, M.-V.; Le, M.-T.; Nguyen-Thi, H.-V.; Le, T.; Nguyen, H.T. EnTube: Exploring key video features for advancing YouTube engagement. *Entertain. Comput.* **2025**, *53*, 100934. [CrossRef]
26. Cho, M.; Jeong, D.; Park, E. AMPS: Predicting popularity of short-form videos using multi-modal attention mechanisms in social media marketing environments. *J. Retail. Consum. Serv.* **2024**, *78*, 103778. [CrossRef]
27. Ahmad, I.S.; Bakar, A.A.; Yaakub, M.R. Movie revenue prediction based on purchase intention mining using YouTube trailer reviews. *Inf. Process. Manag.* **2020**, *57*, 102278. [CrossRef]
28. Oh, D.; Kim, T.; Jang, J.; Park, S.-H. YouTube sentiment and the redistribution of market power among investors. *Pac.-Basin Financ. J.* **2026**, *98*, 103148. [CrossRef]

29. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
30. Hawkes, A.G. Spectra of some self-exciting and mutually exciting point processes. *Biometrika* **1971**, *58*, 83–90. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.