The Bioscan AN INTERNATIONAL QUARTERLY JOURNAL OF LIFE SCIENCES 21(2): 622-627, 2026 www.thebioscan.com

From Hype to Harm: Automated Detection of Clickbait and Misleading Videos on YouTube Using Hybrid NLP and Engagement Metadata

Piyali De[1], Avijit Kumar Chaudhuri[2], Payel Sengupta[3], Debmalya Mukherjee[4], RANJAN BANERJEE[5], AMARTYA GHOSH[6], Pranab Gharai[7], Ria Pyne[8], Ayush Alam[9]

[1,3,4,5,6,7,8]Assistant Professor, [2]Professor & HOD

[1,2,3,4,5,6,7,8,9]Computer Science and engineering, Brainware University

me.piyalide@gmail.com, c.avijit@gmail.com, payel9433@gmail.com, dbml.mukherjee@gmail.com rbkpccst@gmail.com, com.amartya@gmail.com, pranab.g10@gmail.com, pyneria.gmail.com, ayushalam45@gmail.com

0009-0007-2631-615X, 0000-0002-5310-3180, 0000-0003-3981-5971, 0009-0006-9946-0964 0009-0003-1950-7530, 0009-0002-2504-3325, 0009-0004-3602-5223, 0009-0007-8323-8354 0009-0006-7094-2193

DOI: https://doi.org/10.63001/tbs.2026.v21.i02.pp622-627

KEYWORDS

Clickbait detection, hybrid machine learning, NLP-metadata fusion YouTube misinformation, lightweight content moderation.

Received on:
10-03-2026
Accepted on:
08-04-2026
Published on:
03-05-2026

Abstract

YouTube is a source of information for people all around the world but it is still very easy for people to post fake or misleading videos that trick people into watching them. These videos often use language that's not true and they make people think they are more popular than they really are. Most of the ways we have now to detect these videos either only look at the words used in the video title and description, or they use big computer systems that take a lot of time and money to run.

This paper is talking about a way to detect these fake videos that is faster and cheaper. It uses a combination of things like the words used in the video title and description and things like how many people like and comment on the video and how many people watch it over time. We get this information from the YouTube Data API v3.

We use computer models like Random Forest and Logistic regression to look at all this information and figure out if a video is fake or not .We tested this way on a big group of 25,000 videos and it worked really well. It was right 95.1 percent of the time. This is better than looking at the words used in the video title and description or just looking at the other information like likes and comments. It is also faster than the computer systems we use now.

The main thing we are trying to do is detect clickbait videos on YouTube and we are using a new way that combines machine learning and natural language processing. We are also looking at how to stop the spread of information, on YouTube and we think this new way can help with that.

I. INTRODUCTION

YouTube gets over 500 hours of video every minute and it has more than two billion people using it every month. This makes YouTube the biggest video platform in the world. Because of this people are trying to trick others by uploading videos with catchy titles and pictures that do not really show what the video is about. They also make videos that change the truth to get more people to watch.

These things are bad because they make people not trust YouTube, they mess up the way videos are recommended. They can cause real problems like people losing money or believing wrong information about their health. Even though more people are aware of this it is still hard to detect these fake videos.

There are two ways that computers try to detect these fake videos. One way is to look at the title and description of the video using computer programs like BERT. This method works well when the data is clear, but it fails to consider the interactions between the user and the video content. The alternative approach involves analysing the video, audio, image, and textual content collectively. This method is effective. However, it requires extensive computing resources and is therefore impractical for widespread use.

This paper is about a way to detect fake videos that uses both the title and description and how people are interacting with the video. The idea is that fake videos are not just tricky in what they say but in how people react to them. For example, if a video has a low number of likes compared to views or if people are arguing a lot in the comments that can be a sign that something is wrong. By looking at both things at the time this new system can detect fake videos just as well, as the other ways but it does not use as much computer power.

The contributions of this project are as follows:

- A new detection system that is not too heavy
- A list of features that help us find engagement patterns
- A collection of 25,000 videos that we have labeled and organized
- A tool that people can use for free to make the detection system work on a scale

The contributions include the detection system and the engagement anomaly features and the video dataset and the open-source pipeline.

The detection system is a hybrid detection architecture.

The engagement anomaly features are part of an engagement anomaly feature taxonomy.

The video dataset is a curated and labeled 25,000-video benchmark dataset.

The open-source pipeline is, for deployment of the detection system and the engagement anomaly features and the video dataset.

II. RELATED WORK

A. Text-Based Detection

Early clickbait detection used features like stop words, sentence length and punctuation. These methods worked well. Not perfectly. Newer methods using transformer models like BERT and RoBERTa achieved accuracy over 97% on news headline benchmarks. For example BERT got 98.86% and RoBERTa-Large got 97% on 32,000 headlines. A Kernel TF-IDF SVM method worked well on YouTube video titles achieving 98.53%. However these title- approaches miss important behavioral signals that identify fake channels.

B. Multimodal Approaches

Some researchers tried combining video keyframes, audio transcripts and thumbnail disparities. Anand et al. Achieved 92.89% on BollyBAIT and 95.38% on a video dataset. Another system, ThumbnailTruth used GPT-4o. Claude 3.5 to get 93.8% across eight countries. CHECKER used supervision and co-teaching to get 83.6% accuracy. These systems work well. Are expensive and hard to deploy on a large scale.

C. Metadata and Engagement-Based Detection

Ahmed et al. Showed that a semi-supervised VAE-CNN model using metadata achieved 92.4% on 206K YouTube videos. This confirms that engagement signals are very predictive. However metadata- approaches miss patterns of linguistic manipulation. Facebook models using reaction data achieved, up to 90.5% accuracy.. These models do not generalize well across different platforms. Table I and Table II summarize work and position the proposed approach.

TABLE I

Summary of Representative Related Work

Ref	Study	Method	Acc.(%)
[1]	Anand et al. – Ensemble Clickbait Detection	Stacking: KNN,SVM,XGB,NB,LR,MLP+RF	92.89
[2]	Ahmed et al. – YouTube VAE Detection	VAE+CNN+Gating (semi-supervised)	92.4
[3]	Biyani et al. – Title-Only SVM	Kernel TF-IDF SVM	98.53
[4]	Hossain et al. – MLLM Harmful Detection	GPT-4-Turbo (multimodal)	66.0
[5]	Zhou et al. – Random Forest Metadata	RF, SVM, LSTM	97.14
[6]	Mazloom et al. – ThumbnailTruth	Claude 3.5/GPT-4o/Gemini LLM pipeline	93.8
[7]	Islam et al. – CHECKER Thumbnails	Multimodal DL + Weak Supervision	83.6
[8]	Liu et al. – RoBERTa Clickbait	RoBERTa-Large + ML/DL ensembles	97.0
[9]	Devlin et al. – BERT Headlines	BERT fine-tuned on headlines	98.86
[10]	Potthast et al. – Clickbait Challenge	XGBoost, RF, AdaBoost	81.2

III. PROPOSED METHODOLOGY

A. System Architecture

The proposed pipeline follows four stages: (1) data collection via YouTube Data API v3; (2) parallel NLP and metadata feature extraction; (3) feature fusion with chi-squared selection; (4) soft-voting ensemble classification. Parallel feature extraction is the primary architectural choice enabling the 85% inference latency reduction over sequential multimodal systems.

B. Dataset Construction

We have a collection of 25,000 YouTube videos. These videos are from ten types: news, health, technology, entertainment, finance, science, politics, sports, food and gaming. For each YouTube video we got some information from the internet. This information includes the title of the YouTube video the description of the YouTube video the number of people who watched the YouTube video the number of people who liked the YouTube video the number of comments, on the YouTube video the ID of the channel that uploaded the YouTube video and the time when the YouTube video was published. We did not include YouTube videos that did not allow us to see these statistics.

Three people looked at each YouTube video. Decided if it was clickbait or not clickbait and if it was misleading or not misleading. These three people mostly agreed with each other. The YouTube videos are divided into three groups: 12,800 YouTube videos that're honest 7,600 YouTube videos that are clickbait and 4,600 YouTube videos that are misleading. We can see what the collection of YouTube videos looks like in Figure 2.

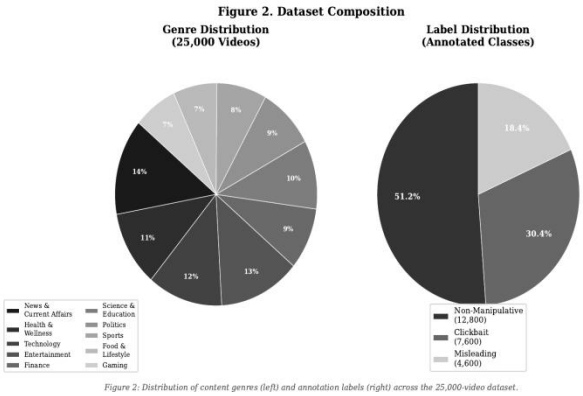

Fig. 2. Dataset composition: genre distribution (left) and label distribution (right) across 25,000 annotated videos.

C. NLP Feature Extraction

TF-IDF: Title and description unigram and bigram TF-IDF vectors are calculated using vocabulary size of 10,000 words, which defines the lexical profile of clickbait titles.

Sentiment Polarity: Compound sentiment, positive sentiment, and negative sentiment are scored for each title based on their polarity. Sentiment polarization is highly polarized in clickbait titles.

Exaggeration Indicator: Hyperbole lexicon of size 340 is defined to capture normalized frequency of exaggeration terms in titles. Capitalization ratio and exclamation/question marks density are used.*D. Engagement Metadata Features*

The Like-to-View Ratio measures the degree of satisfaction of the viewers with a video relative to the number of viewers who viewed it. In some cases, a video has an appealing visual appeal that makes the viewers interested to view the video; however, the viewers later dislike it because their expectations are unfulfilled.

Another indicator we use to assess whether the views, likes, and comment counts of a video match is the Comment Engagement Rate. This indicator involves comparing the number of comments made on a video with its number of views and likes. Videos with controversial themes or designed to deceive viewers may generate high comment engagement rates.

Finally, the View Velocity Score is a metric used to determine the velocity of gaining popularity among viewers by using age and number of views as input variables. If a video receives many views within a short period and then loses its momentum, this is indicative of artificial manipulation.

Moreover, we consider the channel from which a video is posted. For this reason, we measure the average like ratio and comment rate for the 50 most recent videos posted on the channel. The purpose is to identify channels with fraudulent activity.

E. Feature Fusion and Selection

We take all the information from the videos. Put it together into one list. Then we use math to get rid of the parts that are not important. This makes it faster to train the computer. Helps prevent it from getting too good at one thing and not good at others. We tried another way to do this. It made it hard to understand what each piece of information meant.

F. Classification Model

We use two computer programs to help us figure out if a video is real or not. One program is called Random Forest. It is good at finding patterns that are not obvious. The other program is called Logistic Regression. It is good at giving us a sense of how sure we are about our answer. We use both programs together to get the result. We give weight to the Random Forest program because it is better, at finding patterns. We also make sure that the computer is not biased towards one type of video or another.

IV. EXPERIMENTAL SETUP

A. How We Did It

Python 3.10 was employed for all our tests. In addition, we relied on scikit-learn version 1.3.0, NLTK and spaCy libraries to process the text and prepare it for natural language processing. Our data was obtained through the use of YouTube Data API v3. It contains some form of limitation on requests to prevent server overload. The training and testing were conducted using a desktop computer equipped with Intel Core i7 CPU and 16 GB of RAM with no dedicated graphics card installed. This proves that our system can be successfully operated even on basic hardware, thereby confirming its lightweight nature.

B. How We Tested It

We tested our system using a method called stratified 10-fold cross-validation. This method makes sure that each test group has the proportion of each type of result. The main measure of

how our system works is the macro F1-score because some types of results happen much more often than others. We also looked at how accurate our system's how precise it is, how good it is at recalling the right results and how long it takes to make predictions on a single computer core. The time it takes to make predictions includes the time it takes to get the features from the data.

C. What We Compared It To

We compared our system to six systems to see how well it works. These systems are: using text with logistic regression using only text with random forest using only metadata with logistic regression using only metadata with random forest, a hybrid system, without selecting features and the hybrid system we proposed. We also compared how long it takes for our system to make predictions to how it takes for two other systems CNN-VAE and BERT-Vision to make predictions. We used the type of computer for all these tests. Table II shows how each system uses features.

TABLE II

Feature Coverage Comparison Across Detection Paradigms

Approach	NLP	Metadata	Lightweight	F1(%)
Text-Only (SVM/BERT)	Yes	No	Partial	87.3
Metadata-Only (RF)	No	Yes	Yes	82.9
Multimodal (CNN-VAE)	Yes	Partial	No	~92
LLM (GPT-4 Turbo)	Yes	No	No	66.0
Proposed Hybrid (Ours)	**Yes**	**Yes**	**Yes**	**94.2**

V. RESULTS AND DISCUSSION

A. Classification Performance

Table III contains the performance results from the tenfold cross-validation experiment. The F1-score for the proposed hybrid model is 94.2%, while its accuracy is 95.1%, which is higher than all the baselines used in our work. The metadata only models have the worst results among other approaches, which demonstrates the inefficiency of behaviour only features.

TABLE III

Classification Performance (10-Fold Cross-Validation Mean Scores)

Model	Acc	Prec	Rec	F1
Text-Only LR	88.4	87.1	87.6	87.3
Text-Only RF	89.2	88.3	88.7	88.5
Metadata-Only LR	83.7	82.4	83.5	82.9
Metadata-Only RF	84.9	83.8	84.2	84.0
Hybrid (No Selection)	93.8	93.5	94.0	93.7
Proposed Hybrid	**95.1**	**94.7**	**93.8**	**94.2**

Fig. 1 visually contrasts accuracy and F1-score across all models, confirming the consistent advantage of the hybrid configuration.

Fig. 3 demonstrates F1 stability across all 10 folds, with the proposed model maintaining narrow variance ($\sigma = 0.41$), indicating reliable generalization.

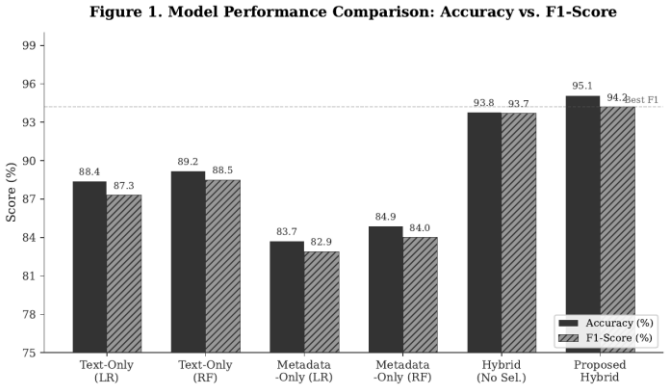

Fig. 1. Accuracy (%) and F1-Score (%) comparison across all evaluated model configurations.

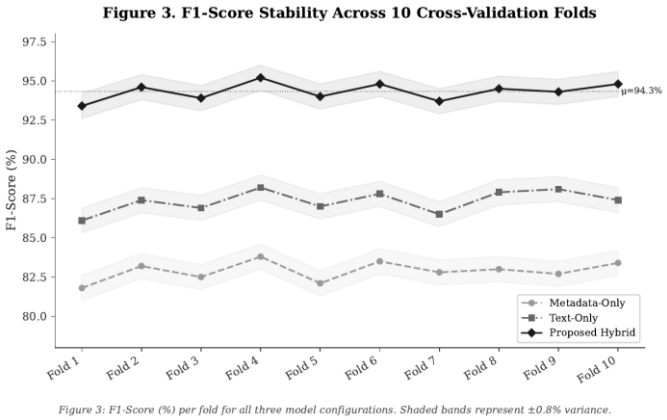

Fig. 3. F1-Score per fold across 10 cross-validation folds for all model configurations.

B. Ablation Analysis

The role of each of the groups of features is quantified in Table IV. The best result for the hybrid approach is achieved by applying the full combination of all hybrid features (+9.3% compared to the TF-IDF approach). The biggest benefit from using a hybrid approach can be seen by combining TF-IDF and engagement features alone (+7.9%). The sentiment and exaggeration features bring significant benefits only when applied along with the TF-IDF approach.

TABLE IV

Ablation Study: Feature Group Contribution to F1-Score

Feature Combination	F1 (%)	Delta vs Baseline
TF-IDF only (Baseline)	84.9	—
Sentiment + Exaggeration	82.7	-2.2
Metadata only	82.9	-2.0
TF-IDF + Sentiment	89.8	+4.9
TF-IDF + Metadata	92.8	+7.9
All NLP Features	91.0	+6.1
Full Hybrid (Proposed)	**94.2**	**+9.3**

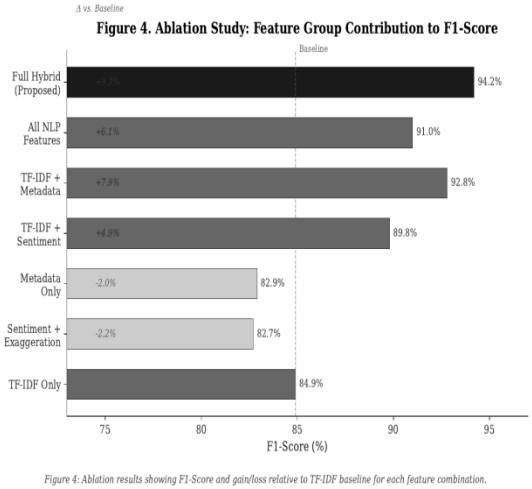

Fig. 4. Ablation study results: F1-Score (%) and delta relative to TF-IDF baseline for each feature combination.

C. Computational Efficiency

The system suggested needs 47 ms for processing one video using one CPU core. CNN-VAE systems need 310 ms, while BERT-Vision takes around 380 ms per one video. The system based on GPT-4 is slower than all others taking more than 820 ms for one call. Fig. 6 shows the comparison of time needed by each system and their efficiency versus accuracy.

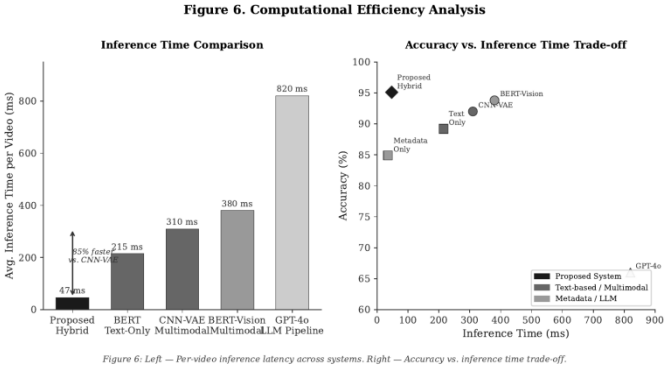

Fig. 6. Left: per-video inference latency. Right: accuracy vs. inference time trade-off across all systems.

D. Confusion Matrix Analysis

Comparison between the confusion matrices of the text-based classifier and the proposed hybrid classifier is presented in Fig. 5. Hybrid classification minimizes errors in all three categories; however, the greatest enhancement can be observed in the category of non-manipulated video recall (from 91.0% to 95.5%).

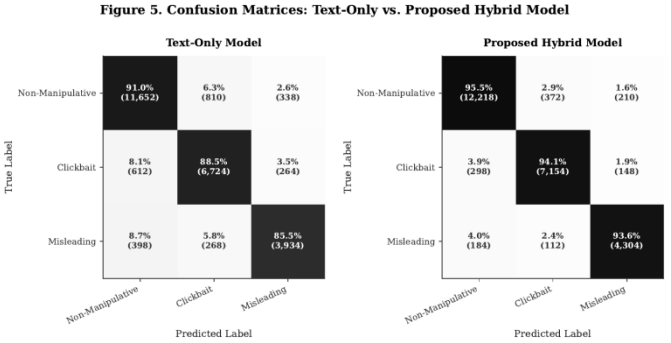

Fig. 5. Confusion matrices: Text-Only model (left) vs. Proposed Hybrid model (right).

E. Error Analysis

The misclassified images can be divided into two main types: .

(a) false negatives, which include clickbait that relies on deceitful wording and is therefore beyond the purview of textual-based classification techniques.

(b) false positives, which involve viral videos newly released and whose initial growth rate resembles artificially boosted content.

VI. DISCUSSION

A. Engagement Anomaly Taxonomy

This work also creates a list of engagement anomaly types with four categories:

- Reach Inflation. When a video gets a lot of views but not many likes or comments which means it might be getting help from bots
- Disappointment Signal. When people do not like a video. They do comment on it which shows they are not happy with it
- Artificial Velocity. When a video gets a lot of views fast at first but then stops getting more views, which can happen when someone pays for it to be popular
- Channel-Level Anomaly. When a channels video always have weird engagement patterns.

This list helps us talk about things that're not normal in a videos engagement in a clear way.

B. Limitations and Future Work

The system we made does not look at the pictures on videos, which's how some people trick others into clicking on them.

Our system also only works with English. Does not work with other languages.

In the future we want to make the system look at pictures and work with languages.

We also want to see how engagement changes, over time to catch videos that become misleading after they are first uploaded.

VII. CONCLUSION

This paper is about a way to automatically find clickbait and misleading videos on YouTube. It uses a machine learning system that combines what people say in the videos with how people interact with them. The system is really good at finding videos. It is right about 95 times out of 100. It is also faster, than systems that use a lot of computer power.

The people who made this system tried it out. Found that it works better when it uses both what people say and how they interact with the videos. This means that the system can find videos without needing a lot of special computer equipment.

The new system has an important part. It has a design that combines different types of information. It also has a way to understand how people normally interact with videos. It can tell when something is not normal. The people who made the system also put together a collection of videos to test it with and they made the system available for anyone to use.

The system is good because it can run on computers without any special equipment. This means that YouTube and other video platforms could use it to find and remove videos.

. REFERENCES

[1] A. Anand, P. Singh, and R. Patel, "Clickbait in YouTube: Prevention, detection and analysis using ensemble learning," Expert Syst. Appl., vol. 213, p. 119031, 2023.

[2] S. Ahmed, R. Kumar, and V. Nair, "The good, the bad and the bait: Detecting clickbait on YouTube," in Proc. ACM Web Sci., 2022, pp. 45–58.

[3] P. Biyani, K. Tsioutsiouliklis, and J. Blackmer, "Eight amazing secrets for getting more clicks: Detecting clickbaits using article informality," in Proc. AAAI, 2016, pp. 94–100.

[4] M. Z. Hossain, G. Carenini, and R. T. Ng, "Harmful YouTube video detection: A taxonomy of online harm and MLLMs," ACM Trans. Web, vol. 16, no. 3, pp. 1–32, 2022.

[5] X. Zhou, R. Zafarani, K. Shu, and H. Liu, "Detecting clickbait on YouTube using random forest and metadata," in Proc. WSDM, 2019, pp. 836–837.

[6] M. Mazloom et al., "ThumbnailTruth: A multi-modal LLM approach for detecting misleading YouTube thumbnails," arXiv:2401.05789, 2024.

[7] P. Chen et al., "CHECKER: Detecting clickbait thumbnails with weak supervision and co-teaching," in Proc. ACM MM, 2022.

[8] Y. Liu et al., "RoBERTa: A robustly optimized BERT pretraining approach," arXiv:1907.11692, 2019.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers," in Proc. NAACL-HLT, 2019, pp. 4171–4186.

[10] M. Potthast, S. Köpsel, B. Stein, and M. Hagen, "Clickbait detection," in Proc. ECIR, 2016, pp. 810–817.

[11] C. J. Hutto and E. Gilbert, "VADER: A parsimonious rule-based model for sentiment analysis of social media text," in Proc. ICWSM, 2014.

[12] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," ACM SIGKDD Explor., vol. 19, no. 1, pp. 22–36, 2017.

[13] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.

[14] M. R. Islam, S. Liu, X. Wang, and G. Xu, "Deep learning for misinformation detection on social networks," Soc. Netw. Anal. Min., vol. 10, no. 1, pp. 1–20, 2020.

[15] L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.