Enhancing Clickbait Detection with Generative–Discriminative Transformer Models

Farah Husain Salem Alharthi

Abstract

Clickbait headlines, designed to attract clicks through sensational or misleading language, undermine online information credibility and user trust. Detecting such headlines is challenging due to diverse linguistic strategies, including hyperbole, curiosity gaps, and emotional triggers. In this work, we propose a hybrid generative–discriminative framework for automatic clickbait detection that leverages both real and synthetic data to improve classifier performance. A GPT-2 model is fine-tuned using Supervised Fine-Tuning (SFT) to generate high-quality, class-conditional synthetic headlines. These synthetic headlines are combined with real headlines to form an augmented training dataset, which is then used to train a BERT-based classifier. Experiments on the Clickbait Title Classification dataset demonstrate that BERT trained on all real data surpasses a baseline SVM model in all metrics (Accuracy, Precision, Recall, F1-score, and ROC-AUC), and performance further improves when synthetic data is incorporated, particularly in recall and overall generalization. These results highlight the value of generative data augmentation for enhancing discriminative models in text classification. Our study demonstrates that combining generative and discriminative transformer models offers a promising approach to improving automatic clickbait detection while providing insights into effective data augmentation strategies for NLP.

1 Introduction

Clickbait headlines—characterized by misleading and catchy language designed to attract clicks—have become a troubling issue across digital media platforms. This trend undermines the credibility of online information, leading to misinformation and reduced public trust in news sources. For instance, Research on 1.67 million Facebook posts from 153 media organizations found that reliable media sources contained 33.54% clickbait headlines, while unreliable sources reached 39.26%[6].

Detecting clickbait is challenging due to the diverse linguistic strategies employed, including hyperbole (e.g., "This One Trick Will Change Your Life Forever!"), curiosity gaps (e.g., "You Won't Believe What Happened Next…"), and emotional triggers (e.g., "The Heartwarming Story That Will Restore Your Faith in Humanity")[6]. In 2014, Facebook announced plans to reduce clickbait stories users find in their news feeds based on metrics such as the click-to-share ratio and the time spent on articles. However, users continued to report exposure to clickbaits, indicating that these measures were insufficient[2].

There have been measures taken to address this issue like Ad-hoc approaches. These are quick, rule-based solutions designed to detect or block clickbaits but are not flexible or generalizable. An example of such an approach is Downworthy[2]. This tool identifies clickbait headlines by looking for a predefined set of common clickbait phrases (like "You won't believe…" or "This will blow your mind…"). Once detected, it replaces or modifies the headline into something nonsensical or "garbage-ish" so the user is no longer enticed to click. Another example is Clickbait Remover for Facebook[2]: This extension blocks links from a fixed list of domains known for clickbait. It prevents these links from appearing in the user's feed, rather than analyzing each headline individually.

This project aims to address these challenges by proposing a hybrid approach that combines generative and discriminative transformer models. A fine-tuned generative model will produce synthetic clickbait and non-clickbait headlines, augmenting the training dataset. A BERT-based classifier will then be trained on both real and synthetic examples, enhancing the model's robustness and generalization capabilities. Several studies have applied tra-

ditional and transformer-based machine learning models, such as BERT, for clickbait detection. Our approach investigates whether synthetic data generated can improve the representational quality of the training set and more accurately approximate the underlying data distribution.

2 Related Work

A paper[2] proposed an approach to detecting and preventing clickbait headlines in online news media. The authors first collected a large, balanced dataset consisting of clickbait headlines from popular entertainment-oriented websites such as BuzzFeed and Upworthy, and non-clickbait headlines from Wikinews. They performed an extensive linguistic and structural analysis of these headlines, identifying key differences between clickbait and traditional news headlines. These differences included sentence length, word length, the proportion of stop words, syntactic dependency lengths, hyperbolic words, internet slang, punctuation patterns, common clickbait phrases, subjects, determiners, possessive pronouns, and n-gram patterns, including word, part-of-speech, and syntactic n-grams.

Using these features, the authors trained a set of classifiers—including Support Vector Machines (SVM), decision trees, and random forests—demonstrating that SVM achieved the best performance with 93% accuracy in detecting clickbaits. The authors implemented these techniques in a browser extension called Stop Clickbait, which warns users about potential clickbaits and allows them to block specific types of headlines according to their interests.

This combines detailed linguistic analysis with user personalization, addressing the challenge of individual reader preferences. However, challenges remain in capturing evolving clickbait styles, and designing scalable detection systems that generalize across diverse online sources. Our work builds on these insights by exploring data augmentation and generative modeling to enhance headline generation and detection performance.

3 Proposed Method

3.1 Overview

The proposed method introduces a **hybrid generative–discriminative framework** for automatic clickbait detection. The goal is to enhance the classifier's generalization and robustness by augmenting the training data with synthetic examples.

Our approach integrates two main components:

1. **Generative model** fine-tuned using **Supervised Fine-Tuning (SFT)** to produce class-conditional synthetic headlines, and
2. **BERT-based discriminative classifier** trained on both real and synthetic data.

3.2 Generative Model: SFT Fine-Tuned Headline Generator

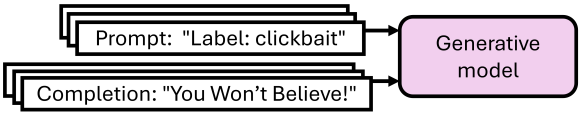

Figure 1: The training of generative model.

For the generative task, I employed a pre-trained causal language model, specifically GPT-2[5]. Supervised Fine-Tuning (SFT) was applied to this model to adapt it for text generation task.

SFT[3] extends the pretrained model's capabilities by training it on a smaller, task-specific dataset that contains explicit examples of the desired input–output behavior. Through this process, the model learns to align its text generation behavior with the patterns present in the labeled data while preserving its general language understanding. In this study, the dataset was reformatted into prompt–completion pairs, where each prompt specified the label category ("clickbait" or "non-clickbait"), and each completion contained the corresponding news headline or title. This structure enables the model to learn a conditional generation mapping—from a given label to a suitable text completion.

During fine-tuning, the model was trained using a causal language modeling (CLM) objective, in which it learns to predict the next token in the completion sequence based on both the preceding tokens and the contextual information from the prompt.

3.3 Discriminative Model: BERT Classifier

For classification, we fine-tune a **BERT-base**[4] model with a binary classification head. I froze all of the model's weights except for the classification head. This means that the transformer layers retained their pre-trained parameters, and only the

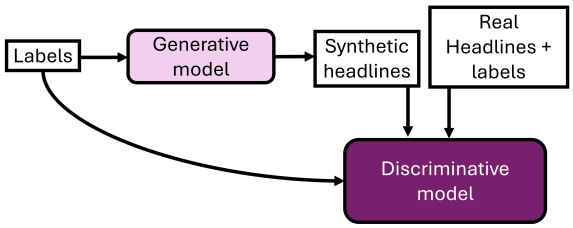

Figure 2: The training of discriminative model.

weights of the classification head were trained from scratch.

Every input text (like a headline) is converted into a sequence of tokens. BERT adds a special token called [CLS] (classification token) at the very beginning of every input. When the input passes through BERT's transformer layers, each token (including [CLS]) gets transformed into a contextual embedding vector. This encodes information not just about the token itself but about the entire sentence context. The embedding of the [CLS] token is often used as a summary representation of the whole input sentence. Then, this embedding vector [CLS] is fed into a linear (fully connected) layer, which projects it into two output values — one for each class (e.g., clickbait and non-clickbait). Finally, a softmax activation function is applied to these two values to convert them into probabilities that sum to 1.

The classifier is trained on a combination of **real headlines** from the Clickbait Title Classification dataset and **synthetic headlines** generated by the SFT-trained model. This hybrid training set aims to reduce overfitting and improve the model's generalization to unseen headline patterns.

4 Experiments and Evaluation

4.1 Dataset

For this project, we used the **Clickbait Title Classification** dataset, which is publicly available on Kaggle[1]. The dataset contains approximately 32,000 news headlines, balanced between clickbait and non-clickbait categories. For the non-clickbait category, 18,513 headlines were extracted from 'WikiNews', 'New York Times', 'The Guardian', and 'The Hindu'. For the clickbait category, the creators manually identified domains known for publishing clickbait content, including BuzzFeed, Upworthy, ViralNova, ScoopWhoop, and ViralStories. Each entry consists of a short headline and a binary label: `1` for clickbait and `0` for non-clickbait. This is the dataset used in the related work.

For the discriminative model, experiments were conducted using two datasets of equal size: one comprising only real data, as described above, and another composed of a balanced mix of 50% synthetic data and 50% real data. In the generative model, the dataset was divided into training data (90%) and validation data (10%). For discriminative model, I employed cross-validation.

4.2 Evaluation Metrics

The models are evaluated using the following standard classification metrics:

- **Accuracy**: The overall proportion of correctly classified instances.
- **Precision**: The fraction of true clickbait predictions among all predicted clickbait headlines.
- **Recall**: The fraction of true clickbait headlines correctly identified by the model.
- **F1-Score**: The harmonic mean of Precision and Recall.
- **ROC-AUC**: The Area Under the Receiver Operating Characteristic Curve, which measures discrimination performance across different thresholds.

4.3 Experimental Setup

We fine-tuned all models using the Hugging Face `Trainer` API, which handled the training loop, gradient updates, and evaluation. The tokenized training and validation datasets were passed to the trainer along with the model and training configuration.

The GPT-2 model was trained for **five epochs** with a **per-device batch size** of 4. To simulate a larger effective batch size while remaining within GPU memory constraints, **gradient accumulation** was applied over 4 steps. A **learning rate** of 5×10^{-4} was used. Training progress was monitored by logging every 10 steps, while both **evaluation** and **checkpoint saving** were performed at the end of each epoch. The parameter `load_best_model_at_end=True` ensured that the best-performing model, based on the lowest validation loss, was retained for final evaluation.

The BERT model was trained for **three epochs** with a **batch size** of 16 for both training and evaluation. The **learning rate** was set to 2×10^{-4}, and a **weight decay** of 0.01 was applied to reduce overfitting by penalizing large parameter values.

Evaluation and checkpoint saving were performed at the end of each epoch. This model also used the `load_best_model_at_end=True`. Logging was conducted every 10 steps to monitor the training process. Mixed-precision training (`fp16=True`) was enabled.

4.4 Cross-Validation Strategy

In the classifier model, we employ a 10-fold cross-validation strategy to ensure robust and unbiased evaluation. The dataset was randomly partitioned into ten equal-sized subsets, or folds. During each iteration, nine folds were used for training and validation: within these nine folds, eight folds were used for training and one fold for validation, enabling hyperparameter tuning and early stopping. The remaining fold was reserved for testing to assess performance. This process was repeated ten times so that each fold served exactly once as the test set, providing a comprehensive assessment of the model's generalization ability across the entire dataset.

The final performance metrics—including Accuracy, Precision, Recall, F1-score, and ROC-AUC—are computed as the average across all folds, providing a reliable estimate of the model's generalization capability.

4.5 Baseline Models

To evaluate the performance of our proposed model, we benchmarked it against a baseline SVM model from related work[2], which was trained on the same dataset.

4.6 Experiment

We evaluated the performance of our BERT-based classifier using two datasets to investigate the effect of synthetic data generated by our GPT-2 SFT model. First, the classifier was trained on the complete real dataset of headlines using a 10-fold cross-validation strategy. For comparison, we also trained the classifier on an augmented dataset consisting of 50% real headlines and 50% synthetic headlines generated by the GPT-2 model, to assess whether adding synthetic examples improves classification performance. In both cases, the BERT model was trained using the same parameters mentioned in the experimental setup section. Classifier performance was measured using accuracy, precision, recall, and F1 score across all folds, allowing us to directly compare the impact of incorporating synthetic data into the training process.

4.7 Results and Discussion

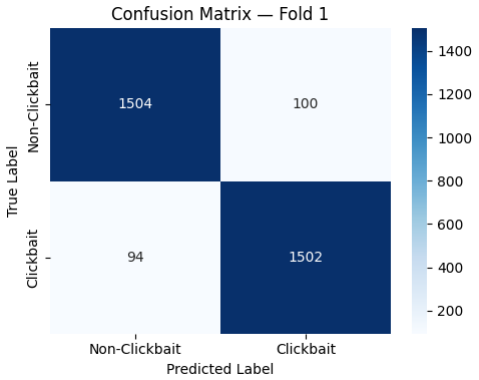

Figure 3: Confusion matrix for one of the folds.

The results in Table 1 highlight the performance improvements achieved by the BERT classifier compared to the SVM baseline. The SVM model, drawn from previous work, achieves a strong performance with an accuracy of 0.93 and an F1 score of 0.93, demonstrating the effectiveness of traditional feature-based approaches for clickbait detection. However, the BERT model trained on the full real dataset surpasses the SVM across all metrics, achieving an accuracy of 0.9425, F1 of 0.9424, and ROC AUC of 0.9864. This demonstrates the advantage of deep contextual representations in capturing nuanced linguistic patterns present in headlines.

Further improvements are observed when the BERT classifier is trained on the mixed dataset containing 50% synthetic data generated by the generative model. In this case, accuracy increases to 0.9504 and F1 rises to 0.9506, while ROC AUC reaches 0.9892. Notably, the recall also improves from 0.9405 to 0.9543, indicating that the model is better at identifying clickbait headlines without sacrificing precision. These results suggest that augmenting the training data with high-quality synthetic examples enhances the model's generalization capability and robustness, confirming the potential of generative data augmentation in downstream discriminative tasks.

Overall, Table 1 illustrates that transformer-based classifiers, particularly when combined with synthetic data augmentation, can outperform traditional machine learning baselines in the task of clickbait detection.

4.7.1 t-SNE Embedding Visualization

To evaluate the semantic similarity between real and synthetic headlines, we computed contextual embeddings using the BERT encoder and projected

them into two dimensions using PCA followed by t-SNE (figure 5). The resulting scatter plot shows that real and synthetic data form two heavily overlapping clusters. This indicates that the generative model learned to produce headlines that are semantically close to real clickbait data.

This overlap suggests that the generator successfully captured the high-level structure, tone, and semantic patterns characteristic of clickbait headlines. Thus, despite being generated artificially, the synthetic examples appear semantically consistent with the real dataset.

4.7.2 Word Cloud Comparison

While t-SNE captures semantic similarity, the word cloud analysis highlights differences at the lexical level. The word clouds (figure 4) for real and synthetic headlines show that although both data sources share many frequent terms typical of clickbait (e.g., "you", "this", "why"), there are noticeable discrepancies. In particular, some words appear prominently in the synthetic data but not in the real headlines.

4.7.3 Limitations and Potential Failure Modes of Synthetic Data Generation

Although synthetic data can significantly enhance training sets, several limitations and failure modes must be considered:

- The model may repeatedly generate specific words or patterns that appear disproportionately in synthetic data. This can introduce biases and reduce the overall richness of the dataset.

- The t-SNE overlap suggests that real and synthetic clickbait and non-clickbait headlines share similar embedding space. This indicates that the generator captures global semantics but may not preserve subtle class-specific distinctions needed for classification.

5 Conclusion

In this work, I presented a hybrid generative–discriminative framework for automatic clickbait detection. Our approach leverages a GPT-2 model fine-tuned with Supervised Fine-Tuning (SFT) to generate high-quality synthetic headlines, which are then used to augment the training dataset of a BERT-based classifier. Experimental results demonstrate that BERT trained on the mixed dataset of real and synthetic headlines outperforms both the traditional SVM baseline and BERT trained on real data alone, achieving improvements across all evaluation metrics, particularly in recall and overall F1 score. These findings highlight the advantage of incorporating generative data augmentation to enhance model robustness and generalization in the presence of limited labeled data.

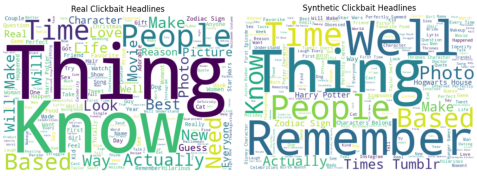

Figure 4: Word clouds for both synthetic and real clickbait headlines.

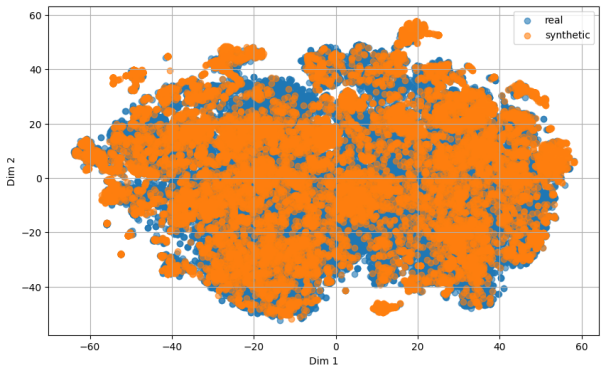

Figure 5: T-SNE of real vs synthetic headlines.

Table 1: Comparison of Model Performance on Clickbait Classification

Metric	SVM	BERT (All Real Data)	BERT (50% Synthetic)
Accuracy	0.93	0.9425	0.9504
Precision	0.95	0.9442	0.9469
Recall	0.90	0.9405	0.9543
F1	0.93	0.9424	0.9506
ROC AUC	0.97	0.9864	0.9892

References

[1] Click bait title classification. https://www.kaggle.com/datasets/aryansinha2003/click-bait-title-classification.

[2] Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. Stop clickbait: Detecting and preventing clickbaits in online news media. In *2016 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM)*, pages 9–16. IEEE.

[3] Hugging Face. Sft trainer. https://huggingface.co/docs/trl/en/sft\{_trainer.

[4] Hugging Face. 2025. Bert — transformers documentation. https://huggingface.co/docs/transformers/en/model_doc/bert.

[5] Hugging Face. 2025. Gpt-2 — transformers documentation. https://huggingface.co/docs/transformers/en/model_doc/gpt2.

[6] Amara Muqadas, Hikmat Ullah Khan, Muhammad Ramzan, Anam Naz, Tariq Alsahfi, and Ali Daud. 2025. Deep learning and sentence embeddings for detection of clickbait news from online content. *Scientific Reports*, 15(1):13251.