# Fine-tuned for the Feed: A Modern Approach for AI Detection Across Social Media Domains

**Kenneth Hahn**
UC Berkeley
hahnkenneth@berkeley.edu

**Matthew Paterno**
UC Berkeley
mpaterno@berkeley.edu

**Victoria Brendel**
UC Berkeley
victoriabrendel@berkeley.edu

## Abstract

The rise of large language models (LLMs) has led to an increase of AI-generated content on social media, raising concerns around misinformation and authenticity. Yet existing AI detectors perform poorly on short-form, informal text typical of platforms like Twitter, Reddit, and Facebook. In this paper, we present a domain-adapted approach to AI-generated detection using ModernBERT, a lightweight encoder-only transformer model fine-tuned on social media data. Our methods achieved 98.6% accuracy on our test sets, outperforming leading baselines such as DetectGPT, SeqXGPT, and DeTeCtive. These results highlight the need for fine-tuning on domain-representative content to build robust, generalizable detectors in the era of generative AI.

## 1 Introduction

The rise of large language models (LLMs) has led to an explosion of AI-generated content across the internet (Sun et al., 2025). On social media platforms, this content is particularly difficult to detect, due to its short, informal, and diverse nature. Posts often contain slang, emojis, or sentence fragments, making them hard to distinguish from human-written text.

Existing detection tools tend to perform well on long-form or formal writing, but struggle in social media settings (Tian et al., 2024). Some detectors analyze statistical irregularities or entropy in language (metric-based approaches), while others employ neural network classifiers fine-tuned to recognize machine-generated text (model-based approaches). These techniques achieve high accuracy on standard benchmarks in controlled settings; however, their effectiveness on the noisy, short-form content prevalent in social platforms remains limited. Short posts provide little context, and simple paraphrasing can often fool AI text detectors (Wu et al., 2024).

In this project, we address this challenge by training a domain-adapted classifier to detect generated social media posts. We fine tune a transformer-based models, ModernBERT, on Reddit, Twitter, and Facebook posts - both human-written and AI generated. We evaluate our model's performance against three existing AI text detection models: SeqXGPT, DeTeCtive, and DetectGPT. Our results show that targeted fine-tuning leads to significant performance improvements on short-form content.

## 2 Background

Approaches to AI text detection typically fall into three broad categories: supervised classifiers, perplexity-based methods, and contrastive retrieval models. Each has demonstrated strengths in various settings, but all face challenges when applied to short, informal social media content.

**Supervised classifiers** like BERT or RoBERTa can be fine tuned on labeled examples of human and AI generated text. These models excel in in-distribution settings, and can learn deep linguistic patterns. However, their performance often drops when applied to new domains or to text generated by unfamiliar LLMs. For example, SeqXGPT (Liu et al., 2023) enhances supervised classification by using token level log probabilities as features, but still struggles when text is too short, or generated by newer models like GPT-4.

**Perplexity-based methods**, such as DetectGPT (Mitchell et al., 2023), operate in a zero shot setting and rely on a language model's probability likelihood to detect AI generated text. These methods perturb input text and assess whether the original text sits at a local maximum of the log-probability function.

**Contrastive detectors** like DeTeCtive (Guo et al., 2023) take a different approach by learning an embedding space, where human and AI writing styles are separable. These models retrieve similar examples from a labeled corpus, and make predic-

tions based on nearest-neighbor similarity.

Despite their promise, most existing detectors have been tested on clean, formal text from academic journals, or synthetic benchmark sources(Wu et al., 2024). Few have been evaluated on noisy, user-generated social media content. Our work addresses this gap by directly training and testing on platform-specific data.

## 2.1 Baseline Models for Comparison

To evaluate the performance of our social media post domain-adapted ModernBERT classifier, we tested it against three AI text detectors: SeqXGPT, DeTeCtive, and DetectGPT. Each model represents a different methodology and design philosophy in the AI-generated text detection landscape.

### 2.1.1 SeqXGPT

SeqXGPT is a supervised sequence-level classifier that uses token-level log-probabilities from a language model to detect AI-generated text (Liu et al., 2023). The model transforms sequences into feature representations, and uses a hybrid CNN and Transformer architecture to classify whether each sentence was written by a human or LLM. This method is effective on sentence-level tasks, and the authors reported nearly perfect F1 scores. SeqXGPT was trained on hybrid academic documents, where the first three sentences of a document were used to prompt an LLM to complete the paragraph or document. Various LLMs were used, such as GPT-2 and LLaMa to generate these responses. Their original human-written documents were taken from sources such as XSum, IMDB reviews, and scientific abstracts.

We trained SeqXGPT on our social media training data, but our results were significantly poorer than the reported results of the paper. One suspected reason for this is that SeqXGPT utilizes GPT-2 to extract token-level log-probabilities, which are only meaningful if the extractor model understands the structure and distribution of the generator. In our case, the AI generated social media posts were produced with GPT-4o-mini, DeepSeek v3, and LLaMA 3.1, which are more modern than GPT-2. Furthermore, SeqXGPT's original benchmarks involved longer sentences with full grammatical structure, so we suspect that on shorter text, the model may receive too little information to distinguish writing styles.

### 2.1.2 DetectGPT

DetectGPT is a zero-shot detector and works by perturbing an input text and evaluating the change in log probability under a language model (Mitchell et al., 2023). If the original text sits at a local optimum of the model's likelihood landscape, it is classified as being likely AI-generated. Unlike SeqXGPT and DeTeCtive, DetectGPT does not use training data. Rather, it assumes access to the model that generated the text. In practice, the model is most effective on longer texts (100+ tokens) and performs best in white-box scenarios. Our dataset consists of short-form social media text, which weakens the curvature signal that DetectGPT relies on.

### 2.1.3 DeTeCtive

DeTeCtive is a model that uses a contrastive learning framework to model writing style differences between humans and various LLMs (Guo et al., 2023). Instead of classifying directly, it embeds text samples into a learned style space, and retrieves nearest neighbors from a labeled corpus. The final prediction is based on similarity scores to known human or AI generated examples.

DeTeCtive was trained on a corpus of over 300K human and machine-generated documents across ten domains and 27 LLMs. While the data is highly diverse, it primarily comprises longer-form content such as articles and essays. Of all three baseline models we compared our model's performance to, DeTeCtive produced the highest accuracy, precision, recall, and F1 score. One factor that we suspect contributed to DeTeCtive's success was the size and diversity of the dataset it was trained on. Furthermore, rather than relying on token probabilities like SeqXGPT and DetectGPT, DeTeCtive uses deep style embeddings learned via a pre-trained transformer encoder, which may capture more syntactic and semantic features.

### 2.1.4 Performance Against Baseline Models

For SeqXGPT, we trained the model on 35,000 randomly selected inputs from our training data, and all three models were tested on the same 15,000 randomly selected inputs from our testing data. We ensured that the human/AI ratio in our training and testing data was balanced.

Table 1: Performance Comparison of Baseline Models on Our Social Media Test Set

| Model | Accuracy | Precision | Recall | F1 |
|---|---|---|---|---|
| SeqXGPT | 58.6 | 51.3 | 51.3 | 65.0 |
| DetectGPT | 43.6 | 38.5 | 21.4 | 27.5 |
| DeTeCtive | 91.7 | 87.2 | 97.5 | 92.0 |

## 3 Methods

### 3.1 Task

Our goal was to develop an effective AI text detector, specifically for social media. Unlike previous approaches that focus on long-form or formal content, we aim to build a model that performs well on short, informal, and platform-specific text found on Twitter, Reddit, and Facebook.

The core objective is to classify whether a given post was written by a human, or generated by an LLM. We frame this as a binary classification problem, and approach it using supervised fine-tuning. By training on a custom dataset of human-written and AI-generated social media posts, we seek to improve detection accuracy in this unique domain of short-form social media posts.

### 3.2 Model Architecture

We use **ModernBERT** (Sun et al., 2024), an encoder-only transformer model that builds on the foundational BERT architecture while integrating a number of recent innovations. Like BERT, input text is tokenized into subword units and passed through stacked self-attention layers, but ModernBERT expands the architecture with 22 transformer layers compared to BERT's 12.

ModernBERT incorporates several architectural enhancements that make it more efficient, expressive, and better suited for downstream tasks such as text classification:

- Deep, narrow architecture**:** ModernBERT uses 22 slimmer layers, totaling 149 million parameters - approximately the same parameter count as BERT-base, but redistributed to support deeper hierarchy. Deeper architectures with narrower layers have been shown to improve hierarchical representation learning (Lu et al., 2017).

- Feed-forward network with GeGLU: Replacing BERT's GELU activation, ModernBERT adopts Gated GELU (GeGLU) (Shazeer, 2020). GeGLU introduces an elementwise gating mechanism that allows dynamic modulation of information flow. This simple modification has been empirically shown to enhance transformer performance without increasing the parameter count.

- Rotary Positional Embeddings (RoPE): Instead of absolute position embeddings as used in BERT, ModernBERT employs RoPE (Su et al., 2021). RoPE encodes token positions as complex-valued rotations applied to the query and key vectors in the attention mechanism. This allows the model to encode relative positional information inherently, with a native context window of up to 8,192 tokens.

- Unpadding and FlashAttention**:** ModernBERT eliminates redundant padding through unpadding batching, enabling efficient packing of multiple short texts into a shared attention window. In addition, it uses the FlashAttention algorithm (Dao et al., 2022), a memory-efficient method for computing exact attention that significantly accelerates training and inference.

- Training data: ModernBERT is trained on over 2 trillion tokens from a mixture of web content, source code, academic papers, and social discourse. The broader linguistic exposure is expected to yield better generalization across formal and informal domains, including the internet slang and variability characteristic of social media.

In summary, ModernBERT provides a strong architectural backbone for our domain-adapted binary classification task, offering improved performance potential over legacy transformer baselines through architectural and training-scale improvements.

### 3.3 Data

To train our model, we will generate our own datasets utilizing different LLMs, crafted from existing social media post datasets. Specifically, we will use Twitter data from the Sentiment140 dataset (Go et al., 2009), Facebook posts data from a repository of scrapings from the site (Diwan, 2017), and Reddit data from the "Explain Like I'm Five" (otherwise known as r/ELI5) subreddit provided by Meta (Fan et al., 2019). An assumption we make is that the collected social media posts are entirely

Table 2: Datasets used for Fine Tuning our model with quantities of human and generated AI posts

| Dataset | Num. Posts | Num. AI Posts |
|---|---|---|
| Reddit | 272,397 | 240,745 |
| Twitter | 28,309 | 59,317 |
| Facebook | 90,000 | 80,900 |

written by humans and not generated by AI. We can make this assumption due to the general timeframe of when these datasets were created. The r/ELI5 dataset was compiled prior to 2019, the tweet dataset from 2009, and the facebook data from 2017 - all well before the emergence of LLMs in society.

As shown in Table 2, there are significantly more Reddit posts compared to both Twitter and Facebook posts. This was mainly due to the accessibility of the data from the sources. However, with our total data set amounting to more than 771K AI generated human responses, we believe that we have a sufficient collection of data from which our model can learn.

For generating the posts themselves, we utilized three different large language models: ChatGPT-4o-mini, Deepseek v3, and LLaMa 3.1-8B-Instruct. We chose these three LLMs in order to balance API cost, speed of generation (where all three methods allow for batched processing of requests via API or parallelization packages, such as vLLM), and modernity of the model itself. As mentioned in the Background, it was found that utilizing a diverse dataset of models and sources can help our model pick up on overall commonalities of machine generated text.

We randomly split the data such that each LLM would generate $\frac{1}{3}$ of the generated posts. To generate the text, we took three different approaches for the datasets described in the processes below:

- Reddit: We used the r/ELI5 dataset because it is a long-form question answer dataset. As a result, we created generated responses to the questions and not based on the answers such that the machine generated text is independent of the human responses, as suggested by Wu et. al. (Wu et al., 2024)

- Twitter: We took on an extractive summarization approach where we would show an LLM the individual tweets and requested a rephrased tweet with similar meaning and writing style.

- Facebook: This data was generated using an abstractive approach, where a model was shown a set of Facebook posts and requested to generate new ones based on the batch of human posts it was given. This is also the reason why there are more Facebook generated posts than there are human posts overall.

### 3.4 Evaluation Metrics

For evaluation of our model, we split the data mentioned above into train, validation, and test sets, at a 60/20/20 split of our data, ensuring that we stratified based on the social media source, to preserve the distribution of our long (Reddit/Facebook) and short (Twitter) form mediums. With the test set, we will evaluate this as a binary classification problem, where 0 represents a human-generated sample, and 1 an AI generated sample. These labels were used to evaluate the following metrics:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

Where $TP$, $FP$, $TN$, and $FN$ are the number of true positives, false positives, true negatives, and false negatives, respectively.

We also wanted to evaluate how our model generalized overall, since fine-tuned detectors have been known for high in-distribution accuracy and low generalizeability outside of the target distribution. As a result, we also compiled new datasets from RAID, a benchmark for Machine-Generated Text Detectors to evaluate out-of-domain and adversarial robustness of these detectors (Dugan et al., 2024). Additionally, we generated two new datasets: one to test on Reddit data using the Mistral-7B-Instruct-v0.3 model (generated text from a model that the detector has never seen) as well as another dataset using LLaMa-3.1-8B-Instruct on LinkedIn social media posts (generated text from a new social media site that the detector has never seen). These different evaluation datasets will allow us to quantify the robustness of our model to various different conditions and

test the eligibility of this detector for varying social media posts.

### 3.5 Fine-tuned Architecture

We fine-tuned the ModernBERT model using a range of learning rates, weight decay values, and batch sizes, while adhering to architectural constraints and training recommendations from the original authors. After empirical experimentation, the configuration that yielded the best performance is shown in the table:

Table 3: Final training configuration for fine-tuned ModernBERT

| Hyperparameter | Value |
|---|---|
| Per-device train batch size | 16 |
| Gradient accumulation steps | 4 |
| Weight decay | 0.01 |
| Number of training epochs | 3 |
| Learning rate | 2e-5 |
| Mixed precision (fp16) | True |
| Dropout rate | 0 |

## 4 Results and Discussion

Our fine-tuned ModernBERT classifier outperformed all baseline models, achieving 98.6% accuracy on the held-out test set of human- and AI-generated social media posts, as seen in Table 4. This confirms the effectiveness of domain-specific fine-tuning for detecting AI-generated content in short-form, informal contexts.

### 4.1 Performance Against Baseline Models

#### 4.1.1 Model's ability to generalize

To evaluate generalization, we tested our model on the RAID benchmark (Dugan et al., 2024), using 50,000 labeled rows, which is designed to assess out-of-domain and adversarial robustness of AI detectors. The poor performance on the RAID data is likely due to distributional mismatch: RAID includes a wider variety of text lengths and writing styles, in contrast to our training data, which consists primarily of brief, informal social media posts. As a secondary experiment, we included 50,000 samples from the RAID benchmark into our fine-tuning dataset. This inclusion led to a substantial performance improvement, with the model achieving 99% accuracy on RAID evaluations. These results demonstrate the importance of diverse training data in improving a model's ability to detect out-of-domain samples.

Next, we evaluated the model on a dataset of 600 LinkedIn posts and 600 AI-generated posts from LLaMA 3.1. These posts were stylistically different and longer than our training data. The model achieved 71.2% accuracy, indicating moderate transferability to new domains. This is in comparison to our Baseline models, which consistently saw a significant drop in performance to out-of-distribution text. The drop in performance may again be attributed to stylistic differences in the training set, which was majority consisted of Reddit posts due to data availability.

On the other hand, when tested on a dataset of Reddit-style posts generated using Mistral-7B-Instruct—a model not seen during training—our classifier achieved 99.26% accuracy, suggesting that the model has learned the underlying patterns of machine generated text for social media platforms it has been adapted to.

#### 4.1.2 Token Length Analysis

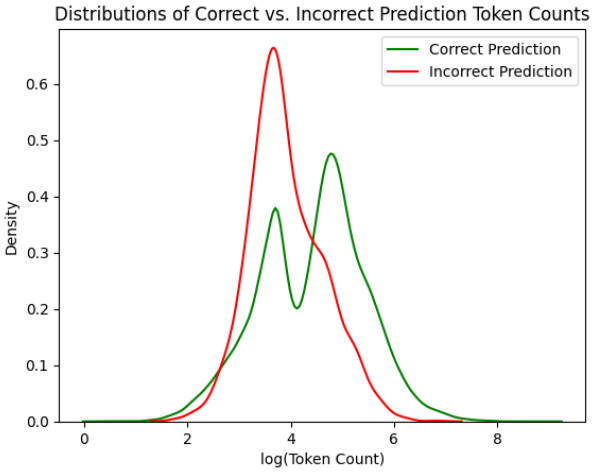


Figure 1: Token Length Kernel Density Distributions of Correct and Incorrect predictions

Across all evaluations, we observed a consistent trend linking token length to model performance. Correct predictions had substantially higher average token lengths compared to incorrect ones, suggesting that shorter texts are disproportionately harder to classify accurately. This can be observed by Figure 1, where our main distribution of token lengths (which can be approximated by the "Correct Predictions" distribution due to the high accuracy) is bimodal, likely due to difference in token length between the Reddit data versus the Twit-

Table 4: Performance of ModernBERT and baseline models across test datasets

| Model / Dataset | Accuracy | Precision | Recall | F1 Score |
|---|---|---|---|---|
| *Baseline Models on Our Test Dataset* | | | | |
| SeqXGPT (GPT-2 logprobs) | 58.6% | 51.3% | 51.3% | 65.0% |
| DetectGPT (zero-shot, GPT-2) | 43.6% | 38.5% | 21.4% | 27.5% |
| DeTeCtive (contrastive) | 91.7% | 87.2% | 97.5% | 92.0% |
| *ModernBERT (Fine-Tuned Model)* | | | | |
| Main Social Media Test Set | **99.4%** | **99.3%** | **99.5%** | **99.4%** |
| RAID (Out-of-domain benchmark) | 42.8% | 4.90% | 99.2% | 9.30% |
| LinkedIn + LLaMA 3.1 | 71.2% | 89.9% | 47.9% | 62.4% |
| Reddit + Mistral-7B (unseen generator) | 98.9% | 99.3% | 98.3% | 98.9% |
| *ModernBERT (Fine-Tuned with RAID + Social Media Data)* | | | | |
| ModernBERT + RAID | 99.4% | 99.6% | 99.6% | 99.7% |

Table 5: Average Token Length Statistics by Evaluation Context

| Dataset / Source | Correct | Incorrect | Gap |
|---|---|---|---|
| Main Test Set (Reddit/Twitter/Facebook) | 155.27 ($\pm$53.2) | 96.78 ($\pm$30.9) | +58.49 |
| LinkedIn + LLaMA 3.1 | 299.77 ($\pm$127.4) | 283.83 ($\pm$155.3) | +15.94 |
| Reddit + Mistral-7B | 102.45 ($\pm$40.3) | 53.60 ($\pm$22.4) | +48.85 |
| Avg. Token Length (Reddit) | 134.81 ($\pm$54.3) | | |
| Avg. Token Length (Twitter) | 24.36 ($\pm$11.7) | | |
| Avg. Token Length (Facebook) | 42.78 ($\pm$44.9) | | |

ter/Facebook data as seen in Table 5. The detector tends to incorrectly detect token lengths nearest to the first mode of the bimodal distribution, which is the Facebook and Twitter data. This could be due to an imbalance in the classes as there is far more Reddit data than Facebook and Twitter and it could be due to the fact that there are fewer patterns and nuances that can be tracked in the short-form media. Future work should incorporate more Facebook and Twitter data to assess whether a more balanced dataset improves overall performance.

This pattern held in the out-of-domain Reddit+Mistral evaluation, where correct predictions had nearly double the average number of tokens as incorrect predictions. The LinkedIn results, on the otherhand, shows us that the average token lengths are similar between the correct and incorrect distributions (300 tokens vs. 284 tokens) This suggests that the model's challenges are more related to the inherent properties of LinkedIn data rather than to variations in text length.

## 5 Conclusion

We present a domain-adapted ModernBERT classifier fine-tuned for detecting AI-generated social media content. By training on platform-specific, short-form datasets and leveraging a diverse set of LLM generators, our model achieved 98.6% accuracy—outperforming several state-of-the-art detectors.

While performance was strong on in-domain and generator-shift settings, generalization to out-of-domain tasks (e.g., RAID, LinkedIn) remained limited. These results emphasize the importance of domain alignment in detection and highlight a key limitation of current supervised approaches.

The benefits of the fine-tuning approach is that we can iteratively expand the scope of our detector with more data. As newer LLMs are released and more social media data is extracted, we can refine our detector by feeding more varied data.

Future work should further investigate the extent to which our detector can be fine-tuned within the ModernBert framework to improve performance on more diverse data. For instance, it would be valuable to examine the minimum volume of data required for effective fine-tuning, as well as assess the impact of input diversity on maintaining performance levels.

## References


Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher R
'e. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. ArXiv:2205.14135.

Parikshit Diwan. 2017. fb-posts-dataset: Scraped facebook posts. Github. Accessed: 2025-03-01.

Liam Dugan, Alyssa Hwang, Filip Trhlik, Josh Magnus Ludan, Andrew Zhu, Hainiu Xu, Daphne Ippolito, and Chris Callison-Burch. 2024. Raid: A shared benchmark for robust evaluation of machine-generated text detectors.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019. Eli5: Long form question answering.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*.

Wenxuan Guo, Yihan Yu, Haibin Huang, Chuhan Zhang, Maosong Sun, and Zhiyuan Liu. 2023. DeTeCtive: Contrastive detection of machine-generated text. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Jiawei Liu, Chang Xu, and Xu Han. 2023. SeqXGPT: Detecting AI-generated text via sequences of log probabilities. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*. ArXiv:2301.11305.

Yao Lu, Kihyuk Sohn, and Honglak Xu. 2017. Beyond depth: Do more layers in deep neural networks hurt performance? In *Proceedings of the 34th International Conference on Machine Learning*.

Eric Mitchell, Yoonki Lin, Luke Melas-Kyriazi, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot detection of AI-generated text via probability curvature. In *International Conference on Learning Representations (ICLR)*.

Noam Shazeer. 2020. Glu variants improve transformer. ArXiv:2002.05202.

Jiaming Su, Hongxin Shi, Shaohan Huang, Derek F. Wong, Yue Zhang, Yi Ren, Lei Cui, and Furu Wei. 2021. Roformer: Enhanced transformer with rotary position embedding. ArXiv:2104.09864.

Zhen Sun, Zongmin Zhang, Xinyue Shen, Ziyi Zhang, Yule Liu, Michael Backes, Yang Zhang, and Xinlei He. 2025. Are we in the ai-generated text world already? quantifying and monitoring aigt on social media.

Zhiqing Sun, Shuming He, Zichao Zhang, Mu Li, Xiangrui Lin, Zhouhan Lin, Caiming Xiong, and Richard Socher. 2024. Modernbert: Transformer architectures for modern hardware. *arXiv preprint arXiv:2412.13663*.

Yuchuan Tian, Hanting Chen, Xutao Wang, Zheyuan Bai, Qinghua Zhang, Ruifeng Li, Chao Xu, and Yunhe Wang. 2024. Multiscale positive-unlabeled detection of ai-generated texts. In *International Conference on Learning Representations (ICLR)*.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Derek F. Wong, and Lidia S. Chao. 2024. A survey on llm-generated text detection: Necessity, methods, and future directions.


## 6 Author's Contribution

This section is to demonstrate the contribution that each author played in this report:

**Kenneth Hahn**: Gathered and pre-processed the Reddit data from r/ELI5. Created the generated datasets for Mistral and LinkedIn datasets as well. Fine-tuned on a different model MayZhou/e5-small-lora-ai-generated-detector, but found that the ModernBert performed significantly better.

**Matthew Paterno**: Gathered and pre-processed the Facebook human and AI-generated dataset. Fine-tuned ModernBERT model and hyperparameter selection. Ran the Results evaluations related to our final model. Performed token length analysis and out-of-domain tests.

**Victoria Brendel**: Gathered human tweet dataset, and generated tweets using gpt-4o-mini, llama, and deepseek. Found git repos and academic papers for three baseline models (SeqXGPT, DetectGPT, DeTeCtive), and trained/tested our data on the baseline models.