Counter Turing Test (CT^2): Investigating AI-Generated Text Detection for Hindi - Ranking LLMs based on Hindi AI Detectability Index (ADI_{hi})

Ishan Kavathekar[1] **Anku Rani** †[2] **Ashmit Chamoli**[1]
Ponnurangam Kumaraguru[1] **Amit Sheth**[3] **Amitava Das**[3]
[1] International Institute of Information Technology, Hyderabad
[2] Massachusetts Institute of Technology
[3] AI Institute, University of South Carolina
ishan.kavathekar@research.iiit.ac.in

Abstract

The widespread adoption of Large Language Models (LLMs) and awareness around multilingual LLMs have raised concerns regarding the potential risks and repercussions linked to the misapplication of AI-generated text, necessitating increased vigilance. While these models are primarily trained for English, their extensive training on vast datasets covering almost the entire web, equips them with capabilities to perform well in numerous other languages. AI-Generated Text Detection (AGTD) has emerged as a topic that has already received immediate attention in research, with some initial methods having been proposed, soon followed by the emergence of techniques to bypass detection. In this paper, we report our investigation on AGTD for an indic language Hindi. Our major contributions are in four folds: i) examined 26 LLMs to evaluate their proficiency in generating Hindi text, ii) introducing the AI-generated news article in Hindi (AG_{hi}) dataset, iii) evaluated the effectiveness of five recently proposed AGTD techniques: ConDA, J-Guard, RADAR, RAIDAR and Intrinsic Dimension Estimation for detecting AI-generated Hindi text, iv) proposed Hindi AI Detectability Index (ADI_{hi}) which shows a spectrum to understand the evolving landscape of eloquence of AI-generated text in Hindi. The code and dataset is available at https://github.com/ishank31/Counter_Turing_Test

† Work was done when the author was at the AI Institute, University of South Carolina.

1 AGTD - The Necessity

With the rise in the number of LLMs capable of generating text across languages, the risks associated with it have increased. AI-generated text could result in misinformation and fake news (Kreps et al., 2022), online manipulation can be used to create fake reviews and social media posts (Chernyaeva et al., 2022), Phishing and scams where malicious actors may use AI-generated text to generate phishing emails (Basit et al., 2021), etc.

In summary, as generative models grow, we need comparable detection techniques. AI text detection is necessary to safeguard individuals, organizations, and society from the potential negative consequences of malicious or misleading content generated by AI systems. It plays a crucial role in maintaining the integrity of online communication and upholding ethical standards in the use of AI technologies. We are the first to conduct experiments for AI-generated news article generation and detection techniques for the Hindi language. Hindi is the fourth most spoken first language in the world after Mandarin, Spanish, and English (Wikipedia, 2023). Taking inspiration from recent works of AI-generated text detection for English (Chakraborty et al., 2023a) where they discussed 6 detection techniques namely watermarking, perplexity estimation, burstiness estimation, negative log curvature, stylometric variation and classification-based approach. We extend it to regional languages like Hindi and cover five new detection techniques to assess AI-generated text

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 4902–4926
November 12-16, 2024 ©2024 Association for Computational Linguistics

detection for Hindi.

OUR CONTRIBUTIONS: A Counter Turing Test (CT^2) and AI Detectability Index for Hindi (ADI_{hi})

- Introducing the *Counter Turing Test (CT^2)* for Hindi, a benchmark that incorporates methods designed to provide a thorough assessment of the resilience of existing AGTD techniques in Hindi.
- Conducting a thorough examination of **26** LLMs to generate an AI-generated news article in Hindi. (AG_{hi}) dataset
- Presenting the *AI Detectability Index for Hindi (ADI_{hi})* as a metric for Language Models to assess whether their outputs can be identified as generated by artificial intelligence or not.
- Curated datasets and models is made available for open-source research and commercial use.

2 Multilingual LLMs

In this paper, we investigate the effectiveness of AGTD techniques on the Hindi language. This section discusses our selected LLMs and elaborates on our data generation methods.

2.1 LLMs: Rationale and Coverage

We chose a wide gamut of 26 LLMs that have exhibited exceptional results on a wide range of NLP tasks. They are: (i) GPT-4 (OpenAI et al., 2024); (ii) GPT-3.5 (Chen et al., 2023); (iii) GPT-2 (Base, Medium, Large, XL) (Radford et al., 2019); (iv) BARD (now Gemini) (Bard, 2023); (v) Bloom (560M, 3B, 7B) (BigScience, 2022) (vi) Bloomz (560M, 1B, 3B, 7B) (Muennighoff et al., 2022); (vii) mGPT (1.3B) (Shliazhko et al., 2023); (viii) Mistral Instruct 7B (Jiang et al., 2023); (ix) Gemma-1.1 (2B, 7B) (Team et al., 2024); (x) mT0 (Small, Base, Large, XL) (Muennighoff et al., 2022); (xi) mT5 (Small, Base, Large, XL) (Xue et al., 2021) .

As the field is in a constant state of evolution, we acknowledge that this process will never reach its finality but instead will persist in its expansion. Therefore, we intend to maintain the Hindi leaderboard benchmark as an open platform for researchers, facilitating ongoing updates and contributions.

2.2 Criteria of selection for Hindi LLM

We experimented with a total of 26 LLMs including variation in their parameter size. We reject a model if it generates no output, produces gibberish, engages in code-switching or generates output solely in English. Table 1 summarizes the rejection criteria for these dismissed models. Additional details about the selection criteria are provided in Appendix A.1.

Model	No output	Gibberish output	English output	Code-switching
Bloom-560M	✓	–	✓	✓
Bloom-3B	✓	–	–	✓
Bloom-7B	✓	–	✓	✓
Bloomz-560M	✓	–	–	–
Bloomz-1B	✓	✓	–	–
Bloomz-3B	✓	–	–	✓
Bloomz-7B	✓	–	–	–
GPT-2 Base	–	✓	✓	✓
GPT-2 Medium	–	✓	✓	✓
GPT-2 Large	–	✓	✓	✓
GPT-2 XL	–	✓	✓	✓
mGPT-1.3B	–	✓	–	–
Mistral-7B	–	✓	–	–
mT0 models	✓	✓	–	–
mT5 models	✓	✓	–	–

Table 1: Criteria used for rejecting the dismissed models. Bloom and Bloomz models fail to generate outputs for most Hindi prompts. GPT-2 models produce gibberish or English outputs, with occasional instances of code-switching. Encoder-decoder models, mT5 and mT0 either produce no output or generate gibberish.

Through our experimentation and observation of the generated outputs, we rejected 21 models. Some of the outputs from these models are present in Appendix A.2. We have retained the responses for 100 data points from BBC Hindi for the rejected models, thereby providing a valuable resource for future research endeavors. This dataset exemplifies why certain models were deemed unfit for inclusion due to their inability to generate coherent and meaningful text. In summary, out of all the 26 LLMs tested for AI-generated news articles in Hindi, we have considered 5 models (BARD, GPT-3.5 Turbo, GPT-4, Gemma-1.1-2B-it, Gemma-1.1-7B-it).

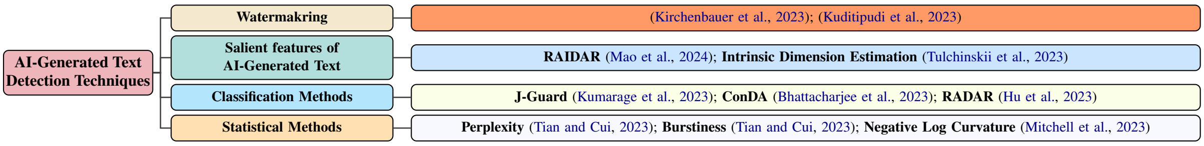


Figure 1: Taxonomy of AI-Generated Text Detection techniques, showcasing various watermarking, feature-related, statistical, and classification-based techniques for detecting AI-generated text.

2.3 Hindi AGTD (AG_{hi}) dataset

In this section, we detail the methodology employed for generating our AG_{hi} dataset.

Human Written Articles: The human-written articles dataset is derived from BBC and NDTV news platforms, encompassing various categories, including India, international affairs, sports, Bollywood, lifestyle, health, and more.

AI-Generated Articles: To obtain AI-generated responses, we employed state-of-the-art 5 LLMs. The headlines collected from the human-written articles were presented as prompts to these LLMs, which generated text responses. The five selected models for the curation of AI-generated articles, resulted in a total of 29,627 AI-generated news articles in Hindi from two Hindi news sources BBC and NDTV as shown in Table 2. The details of the prompts and the hyperparameters used while producing the dataset are detailed in Appendix A.4.

Data Sources	Human Written News Articles	AI Generated News Articles
BBC	1762	7390
NDTV	5281	22237
Total	7043	29627

Table 2: Statistics of human-written and AI-generated news articles in Hindi. The dataset comprises a total of 36,670 news articles.

3 Related Works - SoTA methods

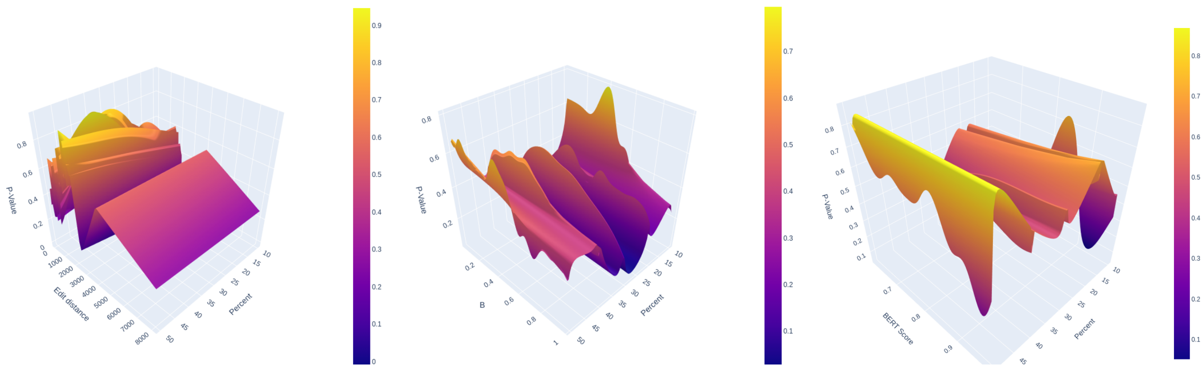
The current AGTD methods can be broadly grouped into four categories: *(i) Watermarking, (ii) Methods based on features of AI-generated text (iii) Classification based methods and (iv) Statistical methods*, as illustrated in Figure 1.

Watermarking has long been an established method in computer vision to identify the source and ownership of content. Kirchenbauer et al. (2023) were the first to present watermarking models for LLMs, though their initial proposal faced criticism. Studies by Sadasivan et al. (2024) and Krishna et al. (2023) demonstrated that paraphrasing can effectively eliminate the watermark, rendering this method ineffective. In response, Kirchenbauer et al. (2024) introduced a more resilient method, which was more robust to paraphrasing. However, this method can be circumvented by a combination of replacing high entropy words and paraphrasing (Chakraborty et al., 2023a). In this paper, we are the first to discuss the balance between the distortion and detectability of the watermark in Section 4.

Recent studies suggest that salient features of AI-generated text and operational characteristics of the LLMs can be utilized to effectively detect AI-generated content. In Section 5, we explore two methods that leverage these features: (i) Intrinsic Dimension Estimation (Tulchinskii et al., 2023) and (ii) RAIDAR (Mao et al., 2024)

Classification methods address the problem of AI-generated text detection by framing it as a binary classification task. Present studies utilize the texts generated from LLMs to train the classifiers (Li et al., 2023; Mao et al., 2024). We discuss three such methods: (i) RADAR (Hu et al., 2023); (ii) J-Guard (Kumarage et al., 2023) and (iii) ConDA (Bhattacharjee et al., 2023) in Section 6.

Statistical methods leverage the discrepancies in statistical characteristics of texts. They assess the deviations in measures such as perplex-



(a) Edit Distance (b) BLEU Score (c) BERTScore

Figure 2: Distortion of text vs detectability of watermark. We observe that the p-value increases even when the percentage of watermarked tokens increases. (a) We see variation in p-values for edit distances below 2000. However, for distances above 2000, p-values become constant and are not influenced by the percentage of tokens watermarked (b) Higher similarity to the original text, indicated by a high BLEU score, correlates with lower p-values (c) Semantic similarity does not influence p-values. However, p-values increase after 30% watermarked tokens, reducing watermark detection reliability.

ity, burstiness (Tian and Cui, 2023), entropy and n-gram frequency to differentiate between human and AI-generated texts. DetectGPT (Mitchell et al., 2023) makes use of the observation that the AI-generated text lies in the negative curvature region of an LLM's log probability space to differentiate between human and AI-generated text. Recent studies, however, have criticized these methods for their unreliability (Chakraborty et al., 2023a). Therefore, we do not investigate them further.

There has been significant exploration of ways to circumvent the detection techniques. Krishna et al. (2023) and Sadasivan et al. (2024) have shown that the watermarking technique is vulnerable to paraphrasing attacks. Lu et al. (2024) proposed a Substitution-based In-Context example Optimization (SICO) method which can evade AI detectors without relying on an external paraphraser.

4 Testing the tradeoffs of distortion vs. detectability in watermarking

To embed a watermark in text, targeted alterations of specific text units are required. While it is intuitive that increasing the number of alterations enhances the strength of the watermark, excessive changes can significantly distort the original text. Therefore, an effective watermarking method requires a delicate balance between distortion and detectability. To our knowledge, no prior work has addressed this issue comprehensively. Although Kuditipudi et al. (2023) discussed distortion in the watermarked text at a high level, they refrained from quantifying this phenomenon. In this paper, we empirically study the balance between distortion and detectability based on the watermarking methods proposed by Kirchenbauer et al. (2023). We propose using Minimum Edit Distance to calculate lexical distortion, BLEU score (Papineni et al., 2002) for syntactic distortion, and BERTScore (Zhang et al., 2019) for semantic distortion. For detectability, we utilize z-score and p-value as proposed by Kirchenbauer et al. (2023).

In our evaluation, we employ the Gemma-2B model for paraphrasing responses by Gemma-7B. We observe that after paraphrasing, the watermark present in the text becomes undetectable, evidenced by p-values greater than 0.01 in Figure 2. Intuitively, one would expect that a higher number of watermarked tokens would result in

AGTD Technique	Performance	Pros	Cons
Intrinsic Dimension Estimation	- Different LLMs exhibit distinct MLE and PHD values. - GPT-4, GPT-3 and BARD showcase intrinsic dimension similar to human text. - Difference in intrinsic dimension of Gemma outputs and human text make the responses more discernible.	- Invariant property of text - Language agnostic technique - No training required	- Models producing human-like responses share similar intrinsic dimension as human text, making their outputs harder to detect.
RAIDAR	- Responses of Gemma models are highly detectable - The method fails to detect GPT-4, GPT-3, and BARD responses, with a significant performance drop of 24-32%.	- Language agnostic technique - No training required	- Computational resources vary depending on the LLM used for text rewriting. - Performance is sensitive to both the model chosen for rewriting and the prompt provided.
RADAR	- Detection accuracy of GPT responses drops below 50%, performing worse than a random classifier. - Higher precision than recall suggests the model classifies human-written text well but struggles to detect AI-generated text. - Consistently low accuracy and F1-scores indicate RADAR's difficulty in accurately identifying AI-generated text.	- Identifies human text with great precision - Trained on paraphrased data along with training data	- Not trainable
J-Guard	- Outperforms other methods, likely due to its focus on journalistic features and news article data. - Cross-model analysis shows a 10-27% performance dip when trained on the BBC dataset and tested on NDTV. - The model is less efficient when trained on data from Gemma models compared to GPT or BARD data.	- Computationally easy to train - Performs better than other models considered in the study	- Specifically designed to detect AI-generated news articles. - Performance is sensitive to the training data.
ConDA	- ConDA's performance metrics, all below 50%, highlight its difficulty in handling the task effectively. - Low precision and recall indicate frequent misclassification of AI-generated and human-written text.	- Utilizes unsupervised domain adaptation and self-supervised contrastive learning to leverage labeled data from the source domain and unlabeled data from the target domain effectively.	- Significantly low performance on Hindi text

Table 3: A brief description of performance, pros and cons of each AGTD technique.

paraphrasing having a lower impact on the watermark. However, our observations indicate that samples with the highest percentage of watermarked tokens (50%) still exhibit high p-values, indicating almost complete elimination of the watermark. The semantic distortion of the text, as quantified by BERTScore, does not significantly affect watermark detectability. Additionally, we noticed that as the BLEU score increases, indicating that the paraphrased text is syntactically similar to the original, the p-value decreases, suggesting more reliable watermark detection compared to samples with lower BLEU scores.

5 Methods based on salient properties of AI-generated texts

This section discusses the methods which leverage the distinct features of the AI-generated text for detection.

5.1 Intrinsic Dimension Estimation

Intrinsic Dimension estimation (Tulchinskii et al., 2023) introduces an invariant property for human-written text—namely, the intrinsic dimension of the underlying embedding manifold. The authors focus on the *Persistence Homology Dimension (PHD)* which belongs to the class of fractal dimension approaches. They chose PHD due to its ability to capture both local and global dataset properties efficiently and robustly against noise. The hypothesis is that the human-texts exhibit higher PHD than that of AI-generated texts enabling a clear differentiation between the two. Tulchinskii et al. (2023) show that the PHD of most European languages is approximated to be **9**$\pm$**1**. However, our experiments reveal that Hindi texts have a lower PHD, ranging from **6 to 7**. Moreover, the maximum likelihood estimation (MLE) values lie in the range of **9 to 10**.

5.2 RAIDAR

Generative AI Detection via Rewriting (RAIDAR) (Mao et al., 2024) method suggests that text generated by auto-regressive generative models typically maintains a consistent structure, often leading other such models to perceive this AI-generated text as high quality. RAIDAR observes that generative models alter AI-generated text less frequently compared to human-written text during rewriting. RAIDAR focuses on the symbolic word outputs of large language models (LLMs) over other features, leveraging the minimal character edit distance between original and rewritten text. In our experiments, we utilized six prompts and applied Gemma-2B to rewrite samples from the dataset. Additional details on the prompts are available in Appendix B.4.

6 Classification Based Methods

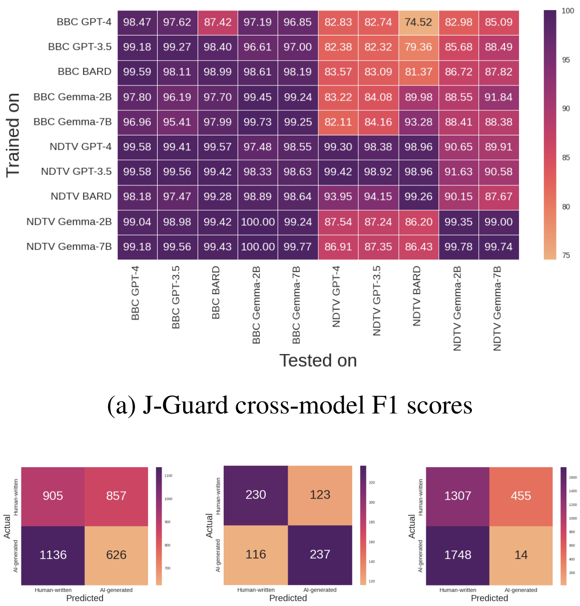
This section discusses classification-based methods for detecting AI-generated text. These methods utilize a range of techniques such as adversarial learning, self-supervised learning, and stylometry while training on both human-written and AI-generated text.

6.1 RADAR

Robust AI-text detector via adversarial learning (RADAR) (Hu et al., 2023) is a novel framework that employs adversarial training to enhance AGTD. RADAR presents a paraphraser and a detector as two opposing agents inspired by adversarial machine learning techniques. The paraphraser aims to generate realistic content that can bypass AI-text detection, whereas the detector is trained to enhance the detectability of the AI-generated text. The paraphraser rewrites the text generated by the LLMs to evade detection as AI-generated. Conversely, the detector learns to distinguish between human and AI-generated text using both the training data and the paraphraser's output.

6.2 J-Guard

Journalism Guided Adversarially Robust Detection of AI-generated News (J-Guard) (Kumarage et al., 2023) is a framework designed to tackle the growing issue of AI-generated news. J-Guard leverages stylistic cues derived from journalistic features to distinguish human-written articles from AI-generated news articles. The premise is that deviation from the Associated Press (AP) Stylebook standards can indicate that an article is AI-generated. The framework extracts various journalistic features to quantify these deviations including, organization and grammar standards such as mean



(a) J-Guard cross-model F1 scores

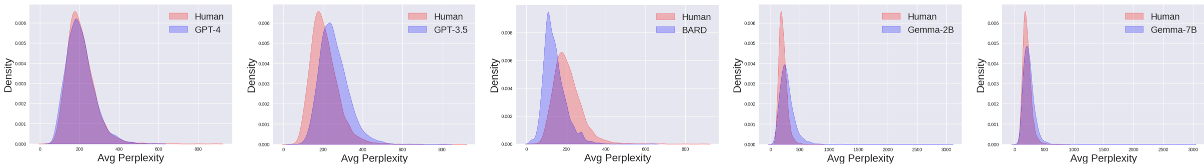
(b) ConDA (c) RAIDAR (d) RADAR

Figure 3: (a) Models trained on BBC dataset and tested on NDTV dataset show a significant drop of 10-15% in F1 score. ConDA and RAIDAR (b, c) misclassifies AI-generated text as human-written and vice versa, leading to high number of false positives and false negatives. RADAR (d) frequently classifies a given text as human-written, leading to misclassification of majority AI-generated text.

word count, word count of leading paragraph, punctuation use and standard formatting violations such as date, time and number formats.

6.3 ConDA

The Contrastive domain adaptation framework (ConDA) (Bhattacharjee et al., 2023) addresses the problem of AI-generated text detection by framing it as an unsupervised domain adaptation task where the domains are different LLMs. The framework assumes access to labeled source data and unlabeled target data. This framework blends standard domain adaptation techniques with the representation power of contrastive learning to learn domain invariant representations that are effective for the final unsupervised detection task. ConDA leverages



(a) GPT-4 (b) GPT-3.5 (c) BARD (d) Gemma-2B (e) Gemma-7B

Figure 4: Perplexity estimation for models. The perplexities of the responses generated by these LLMs are nearly identical to those of human texts. This similarity in perplexity makes it an unreliable factor for distinguishing between human and AI-generated text.

the power of both, unsupervised domain adaptation and self-supervised representation learning.

7 Overall Analysis

Table 4 provides a summary of the results for the AGTD techniques. Our experiments highlight the fragility of existing AGTD methods. RADAR and ConDA exhibit poor performance in effectively detecting AI-generated text. While intrinsic dimension estimation can effectively differentiate the responses of certain models as AI-generated, its performance is not consistent across models. This is evident from the variation in intrinsic dimensions across models. J-Guard outperforms the other techniques, but struggles in cross-model scenarios, revealing its limitations, as seen in Figure 3. Additional results are provided in Appendix B.

We found that responses generated by black-box LLMs such as GPT-4, GPT-3.5 and BARD are particularly challenging to detect, possibly due to their large parameter sizes. In contrast, outputs from open-source models like Gemma are easier to identify. Interestingly, we did not observe a significant difference in detectability across open-source models with varying parameter sizes. As the LLMs increasingly generate human-like text, detection becomes more difficult, since most techniques rely on comparing AI-generated text to human-written content.

8 AI Detectability Index for Hindi (ADI_{hi})

Given the rapid advancements in LLMs, the existing AGTD techniques may prove to be ineffective for the newer models. We propose AI Detectability Index for Hindi (ADI_{hi}) as a benchmark to assess and rank LLMs according to the detectability of the model's responses.

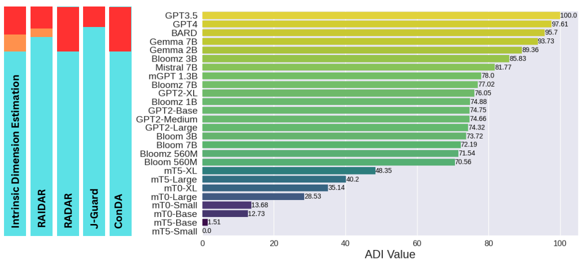


Figure 5: Right: ADI Spectrum of a diverse set of LLMs based on their detectability. Left: Bars represent AI-generated text detectability by a specific method. Blue indicates fully detectable, orange indicates unreliable detection, and red indicates not detectable.

8.1 Limitations of ADI proposed by Chakraborty et al. (2023a)

Previous work by Chakraborty et al. (2023a) focuses on perplexity and burstiness as the factors to quantify the detectability of the model responses. However, the text generated by the newer LLMs is indistinguishable from human-written text. Furthermore, the perplexity and burstiness of larger models like GPT-4, GPT-3.5 and BARD resemble human-written text with a small variance, as illustrated in Figure 4. Liang et al. (2023) and Chakraborty et al. (2023b) have shown that perplexity and burstiness are not reliable detectors of human writing. To overcome this, we assess the divergence between the AI-generated text and

Detection Techniques	Models	News Source 1 [BBC Data]				News Source 2 [NDTV Data]			
		Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
RADAR	GPT-4	37.486	2.985	0.795	1.255	49.205	25.862	0.852	1.650
	GPT-3.5	37.089	0.000	0.000	0.000	48.959	12.838	0.360	0.699
	BARD	72.211	14.634	50.000	22.642	53.380	79.024	9.203	16.486
	Gemma-2B	50.748	52.448	16.026	24.55	53.557	61.423	19.125	29.169
	Gemma-7B	62.103	71.616	40.098	51.411	64.415	80.287	38.213	51.781
J-Guard	GPT-4	98.440	99.718	97.245	98.466	99.242	99.229	99.229	99.229
	GPT-3.5	99.291	99.128	99.417	99.272	98.958	99.606	98.249	98.923
	BARD	99.007	99.709	98.281	98.990	99.290	99.505	99.016	99.260
	Gemma-2B	99.467	99.454	99.454	99.454	99.344	98.996	99.711	99.352
	Gemma-7B	99.237	99.246	99.246	99.246	99.733	99.529	99.947	99.738
ConDA	GPT-4	43.445	42.212	35.528	38.582	51.856	52.736	35.770	42.627
	GPT-3.5	45.658	45.099	39.955	42.371	50.587	50.899	33.232	40.211
	BARD	47.645	47.456	43.927	45.623	55.245	57.030	42.548	48.736
	Gemma-2B	53.739	53.700	54.274	53.985	59.679	63.607	45.248	52.879
	Gemma-7B	52.353	52.323	52.995	52.657	57.491	59.537	46.762	52.382
RAIDAR	GPT-4	66.147	65.833	67.134	66.48	69.584	67.814	74.551	71.023
	GPT-3.5	64.589	64.345	65.439	64.888	60.549	60.43	61.116	60.771
	BARD	74.22	74.085	74.504	74.294	89.64	88.582	91.012	89.781
	Gemma-2B	98.404	98.925	97.872	98.396	96.939	96.532	97.376	96.952
	Gemma-7B	98.476	99.688	97.256	98.457	94.712	95.146	94.231	94.686
		MLE		PHD		MLE		PHD	
Intrinsic Dimension	Human written	10.016		6.967		9.592		6.781	
	GPT-4	9.541		7.002		9.416		6.900	
	GPT-3.5	9.796		6.882		9.549		6.720	
	BARD	7.272		3.120		7.061		3.105	
	Gemma-2B	4.368		3.004		4.537		3.118	
	Gemma-7B	5.354		3.597		5.577		3.744	

Table 4: Results showcasing the efficacy of various AI-Generated Text Detection (AGTD) methods. Results compare performance metrics across different techniques, highlighting their effectiveness in accurately identifying AI-generated text versus human-written text

human-written text to quantify the detectability of the model.

8.2 ADI - Proposed by us

For every pair of human-written article and AI-generated article in our AG_{hi} dataset, we identify the common set of words between them, denoted as V. For each word w_i in V, we extract the sentences S_h and S_{ai} from the human-written and AI-generated text respectively, containing w_i. Using these sentences, we generate the co-occurrence vectors C_h and C_{ai}.

This helps us capture the semantic, syntactic, and lexical features of the text. These co-occurrence vectors are then transformed into probability distributions by normalizing the frequency counts. A detailed explanation of this can be found in Appendix C.1 To quantify the divergence between these distributions we initially employed KL-divergence (KLD) (Csiszár, 1975). However, KLD lacks the capability to handle zero values in the probability distribution. This results in the divergence values escalating to infinity, causing the models to cluster closely together and overlap, rendering the spectrum discernible. To address this we adopt Jenson-Shannon divergence (JSD)

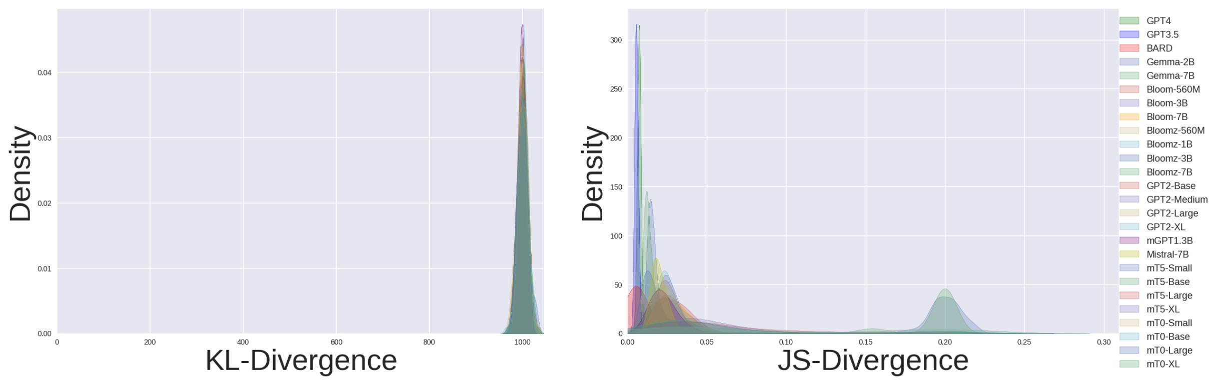


Figure 6: Comparison of Divergence values using KL-Divergence and Jenson-Shannon Divergence. When employing KL-divergence, all the divergence values cluster around a point. In contrast, using JS-Divergence reveals a more distributed spectrum of values. *Note: The divergence values of 1000 in the KL-divergence case are merely indicative; the actual value approaches infinity.*

(Jenson-Shannon-Divergence), a more robust and symmetric version of KL-divergence. Figure 6 illustrates the comparison between the performance of KLD and JSD. To assess overall divergence, summation has been taken over all the data points U as depicted in Equation 1. After calculating the mean divergence of all 26 models initially, we adopt Yeo-Johnson power transformation (Yeo and Johnson, 2000) to make the data more normally distributed. This is crucial for balanced and unbiased scaling. Finally, the values are scaled between 0-100 using the min-max normalization (Normalization) for better readability and interpretability. The resulting ADIs are then ranked and scaled providing a comparative spectrum as presented in Figure 5. A higher value of ADI_{hi} corresponds to greater difficulty in detecting the model's responses as AI-generated.

$$ADI_x = \frac{1}{U} * \sum_{j=1}^{U} \left[\{ \sum_{i=1}^{|V|} JSD_j(P_h^i \mid\mid P_{ai}^i) \} * \frac{1}{|V|} \right] \quad (1)$$

Three groups of models can be observed from the ADI spectrum, namely: easy-to-detect, detectable and difficult-to-detect. The mT0 and mT5 models are situated in the realm of easy-to-detect range while models like Bloom, Bloomz and GPT-2 are detectable. The remaining models are regarded as nearly undetectable using the existing SoTA AGTD techniques.

From the methods we considered, it is unlikely that any of them would be effective for models with high ADI, as shown by our experiments and results. With advancements in LLM technology, the current AGTD methods would become more ineffective. Recognizing this, the ADI spectrum will serve as a crucial tool for assessing the detectability of LLMs.

9 Conclusion

Our research contends that SoTA AGTD techniques are susceptible to fragility. These techniques, while effective in English often struggle or perform poorly when applied to Hindi, highlighting the need for language-specific considerations in AI-generated text detection. We experimented with 26 distinct LLMs to create the AG_{hi} dataset and support the assertion. We introduce the AI Detectability Index for Hindi (ADI_{hi}), and present a means to assess and rank LLMs based on their detectability levels. The excitement and success of LLMs have resulted in their extensive proliferation, and this trend is anticipated to persist regardless of the future course it takes. In light of this, the CT^2 benchmark and the ADI_{hi} will continue to play a vital role in catering to the scientific community.

10 Discussion And Limitations

In this paper, we address the critical issue of AI-generated text detection in the context of the Hindi language. Despite the valuable contributions, there are certain limitations inherent in this work as discussed in the following points.

- Exploring the temperature hyperparameter: We experiment with temperature hyperparameters while selecting the LLMs. However, we generate AG_{hi} considering a constant temperature. Investigating the influence of temperature on the detectability of the generated text would provide valuable insights.

- Text consistency in experiments: We generate only a single response per headline while forming the dataset. However, future work can involve generating multiple responses for each headline and evaluating the detectability of these responses.

- Temporal Limitations: Due to the absence of an archive feature on the BBC and NDTV websites, we opted to compile a varied set of headlines without being bound by temporal limitations. However, our selection criteria for LLMs focuses on the quality of the text generated. Furthermore, none of the AGTD techniques evaluated in the study assess the text based on its factuality. Therefore, this decision does not affect the validity of our results.

- Generalization to other languages: The study primarily focuses on the Hindi language, and the findings may not be directly applicable to other languages with distinct linguistic characteristics. Future research could explore the extension of these insights to a broader range of languages.

- Evolution of LLMs: The rapidly evolving nature of LLMs raises the possibility that newer models, not included in the study, may exhibit different behaviors. As such, the generalizability of the findings to future LLMs may be limited.

- Dynamic AI-generated text detection landscape: The research evaluates AGTD techniques based on the current state of detection methods. However, the dynamic nature of the AI-generated text detection methods suggests that new strategies may emerge, potentially impacting the long-term efficacy of the proposed techniques.

11 Ethical Considerations

Our experiments reveal the constraints of AGTD methods in Hindi. It is crucial to note that while we envision ADI_{hi} as a tool for constructive purposes, there exists the potential for misuse by malicious entities, especially in generating AI-generated text like fake news that is indistinguishable from human-written content. We strongly caution against any such misuse of our findings.

Acknowledgments

We would like to thank Harshavardhan Nemani, Mann Khatri, Prashant Kodali, Ritwik Mishra and other members of Precog group, IIIT-Hyderabad for their valuable feedback.

References

Google AI Bard. 2023. An important next step on our ai journey. [Online; accessed 06-December-2023].

Abdul Basit, Maham Zafar, Xuan Liu, Abdul Rehman Javed, Zunera Jalil, and Kashif Kifayat. 2021. A comprehensive survey of ai-enabled phishing attacks detection techniques. *Telecommunication Systems*, 76:139–154.

Amrita Bhattacharjee, Tharindu Kumarage, Raha Moraffah, and Huan Liu. 2023. Conda: Contrastive domain adaptation for ai-generated text detection.

BigScience. 2022. Bigscience language open-science open-access multilingual (bloom) language model. `https://huggingface.co/bigscience/bloom`. International, May 2021-May 2022.

Megha Chakraborty, SM Tonmoy, SM Zaman, Krish Sharma, Niyar R Barman, Chandan Gupta, Shreya Gautam, Tanay Kumar, Vinija Jain, Aman Chadha, et al. 2023a. Counter turing test ct^ 2: Ai-generated text detection is not as easy as you may think–introducing ai detectability index. *arXiv preprint arXiv:2310.05030*.

Souradip Chakraborty, Amrit Singh Bedi, Sicheng Zhu, Bang An, Dinesh Manocha, and Furong Huang. 2023b. On the possibilities of ai-generated text detection.

Xuanting Chen, Junjie Ye, Can Zu, Nuo Xu, Rui Zheng, Minlong Peng, Jie Zhou, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. How robust is gpt-3.5 to predecessors? a comprehensive study on language understanding tasks. *arXiv preprint arXiv:2303.00293*.

O Chernyaeva, TH Hong, YK Park, YH Kim, and G Ren. 2022. Ai generating and detecting manipulated online customers reviews. pages 270–275.

Imre Csiszár. 1975. I-divergence geometry of probability distributions and minimization problems. *The annals of probability*, pages 146–158.

Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. 2023. Radar: Robust ai-text detection via adversarial learning.

Jenson-Shannon-Divergence. [link].

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 17061–17084. PMLR.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Manli Shu, Khalid Saifullah, Kezhi Kong, Kasun Fernando, Aniruddha Saha, Micah Goldblum, and Tom Goldstein. 2024. On the reliability of watermarks for large language models.

Sarah Kreps, R Miles McCain, and Miles Brundage. 2022. All the news that's fit to fabricate: Ai-generated text as a tool of media misinformation. *Journal of experimental political science*, 9(1):104–117.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2023. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. In *Advances in Neural Information Processing Systems*, volume 36, pages 27469–27500. Curran Associates, Inc.

Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. 2023. Robust distortion-free watermarks for language models. *arXiv preprint arXiv:2307.15593*.

Tharindu Kumarage, Amrita Bhattacharjee, Djordje Padejski, Kristy Roschke, Dan Gillmor, Scott Ruston, Huan Liu, and Joshua Garland. 2023. J-guard: Journalism guided adversarially robust detection of ai-generated news.

Linyang Li, Pengyu Wang, Ke Ren, Tianxiang Sun, and Xipeng Qiu. 2023. Origin tracing and detecting of llms.

Weixin Liang, Mert Yuksekgonul, Yining Mao, Eric Wu, and James Zou. 2023. Gpt detectors are biased against non-native english writers.

Ning Lu, Shengcai Liu, Rui He, Qi Wang, Yew-Soon Ong, and Ke Tang. 2024. Large language models can be guided to evade ai-generated text detection.

Chengzhi Mao, Carl Vondrick, Hao Wang, and Junfeng Yang. 2024. Raidar: generative ai detection via rewriting.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: zero-shot machine-generated text detection using probability curvature. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, et al. 2022. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*.

Normalization. [link].

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, and Diogo Almeida et al. 2024. Gpt-4 technical report.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2024. Can ai-generated text be reliably detected?

Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Vladislav Mikhailov, Anastasia Kozlova, and Tatiana Shavrina. 2023. mgpt: Few-shot learners go multilingual.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, and Laurent Sifre et al. 2024. Gemma: Open models based on gemini research and technology.

Edward Tian and Alexander Cui. 2023. Gptzero: Towards detection of ai-generated text using zero-shot and supervised methods.

Eduard Tulchinskii, Kristian Kuznetsov, Laida Kushnareva, Daniil Cherniavskii, Serguei Barannikov, Irina Piontkovskaya, Sergey Nikolenko, and Evgeny Burnaev. 2023. Intrinsic dimension estimation for robust detection of ai-generated texts.

Wikipedia. 2023. List of languages by number of native speakers — Wikipedia, the free encyclopedia. Article Link. [Online; accessed 06-December-2023].

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mt5: A massively multilingual pre-trained text-to-text transformer.

In-Kwon Yeo and Richard A. Johnson. 2000. A new family of power transformations to improve normality or symmetry. *Biometrika*, 87(4):954–959.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Appendix

This section provides supplementary material in the form of additional examples, implementation details, etc. to bolster the reader's understanding of the concepts presented in this work.

A Model Selection and Data Generation

In this section we provide additional information about criteria for model selection, methods used for data generation and hyperparameters applied in generating the dataset.

A.1 Acceptance and Rejection criteria

The criteria used to determine acceptance or rejection of a model are as follows:
Language Consistency: If the response is only in English, the model is rejected.
Code-Switching: If the response starts in Hindi but later switches to English, the model is rejected.
Gibberish Output: Models that produce unintelligible or gibberish responses are rejected.
No Output: Models producing no output are trivially rejected.

The news headline along with an instruction is prompted to the LLM. We assess a hundred responses from the LLM manually and reject a sample if it produces only English responses, engages in code-switching, generates gibberish output, or fails to produce any output. Besides these criteria, the model must generate five unique Hindi sentences. Therefore, responses that repeat sentences are excluded. If over 70 out of 100 responses meet these rejection criteria, the model is rejected.

A.2 Examples from AG_{hi} dataset

We present articles generated by both accepted and dismissed models in Fig 14, 14, 15 and Fig 16, showcasing various types of rejection criteria along with specific examples.

A.3 Prompts for Data Generation

For better responses we add an instruction along with the headlines while prompting the LLMs. We experimented with various prompts for generating news articles in Hindi. Some examples include:

1. Expand this headline into a Hindi news article.
2. Write a Hindi news article for the headline.
3. Consider the given headline and write a news article for it in Hindi.
4. Generate a Hindi news article from the given headline.

We also experiment with Hindi instructions but observe that while larger models like GPT-4, GPT3.5 and BARD were able to generate the desired responses, other models either failed to generate any response or produced responses falling mainly in the rejection criteria mentioned in Section 2.2. The use of Hindi instructions increased the number of responses falling under the aforementioned rejection criteria, therefore, we chose to use English instructions exclusively for our experiments.
Although GPT-4, GPT-3.5 and BARD were able to generate all responses in Hindi, Gemma models could not. Hence, we only consider Hindi responses from these models in our experiments. Table 5 summarizes the statistics for each model's responses included in AG_{hi} dataset.

	Data source	
Model	**BBC**	**NDTV**
GPT-4	1762	5280
GPT-3.5	1762	5280
BARD	1762	5280
Gemma-2B	468	1715
Gemma-7B	1636	4679
Total	7390	22234

Table 5: Data samples statistics for individual models. The combined BBC and NDTV datasets contain a total of 29,624 AI-generated data points, providing a substantial basis for evaluating the performance and generalization capabilities of the models.

A.4 Hyperparameters for models

We list the hyperparameters employed during text generation for the included models. Various hyperparameters were tested to evaluate the rejected models, but their outcomes did not meet our criteria, resulting in exclusion from further consideration. Table 6 provides a comprehensive overview of all the hyperparameters for the models.

Model	Hyperparameters
GPT-4	temperature: 1 max_tokens: 500 frequency_penalty: 0
GPT-3.5 Turbo	temperature: 1 max_tokens: 500 frequency_penalty: 0
BARD	-
Gemma-2B	temperature: 1 max_tokens: 500
Gemma-7B	temperature: 1 max_tokens: 500

Table 6: Hyperparameters used to generate text from different models. No hyperparameters are available for BARD as the data was collected directly from the website.

B Results

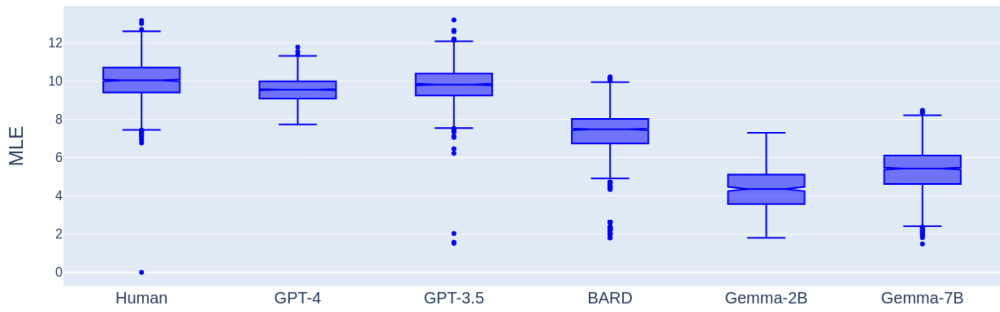
In this section, we discuss additional results from the aforementioned AI-Generated Text Detection techniques.

B.1 Main Results

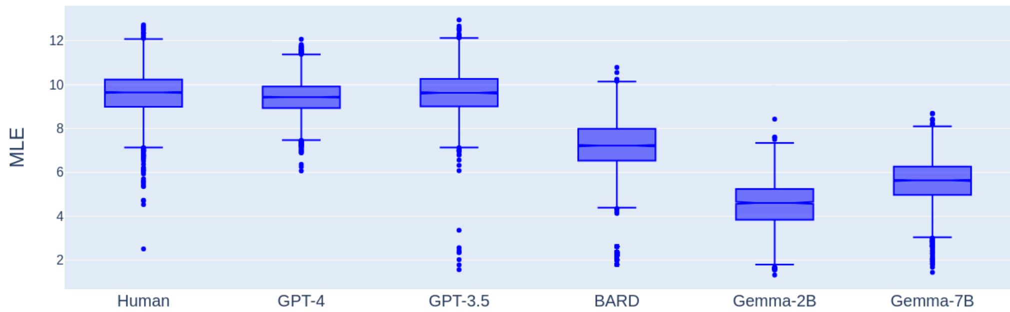
The confusion matrices of all the methods can be found in Fig 9, Fig 10, Fig 11 and Fig 12.

B.2 Results from Intrinsic Dimension Estimation

The results from intrinsic dimension estimation are presented as box plots in Fig 7 and 8. We present the distribution of MLE and PHD estimations across datasets.



(a) BBC dataset



(b) NDTV dataset

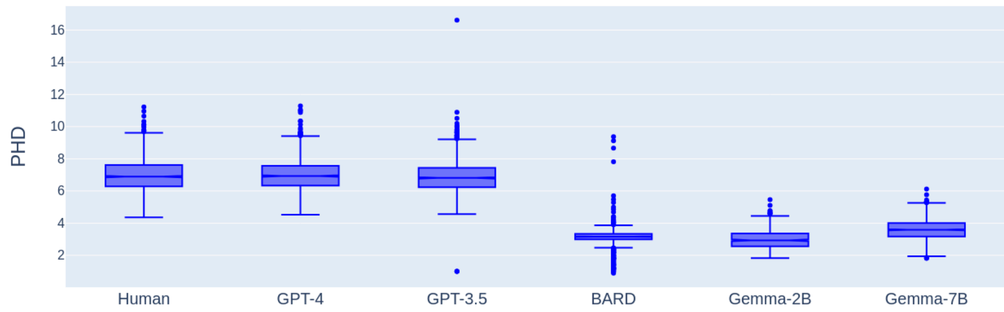
Figure 7: Maximum Likelihood estimation (MLE) of various models across datasets. MLE values of GPT models align closely with human MLE values, while BARD responses are slightly lower. In contrast, MLE values of Gemma models significantly differ from human values, facilitating easier identification of Gemma responses as AI-generated by MLE estimation.

B.3 Results from J-Guard

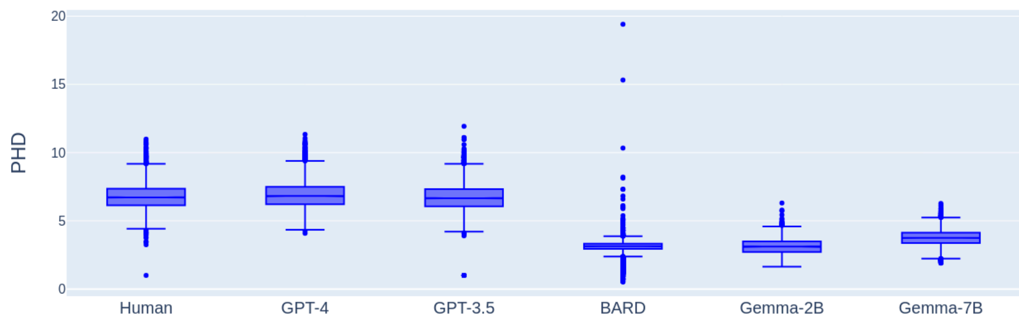
We present the cross-domain performance metrics like accuracy, precision, recall and F1 score for the J-Guard framework in Fig 7, Fig 8, Fig 9 and Fig 10 respectively. In this evaluation, the model undergoes training on a specific dataset and is subsequently tested on each distinct dataset. This method provides insights across various domains, exhibiting the model's ability to generalize to a dataset not encountered during the training phase.

B.4 Results from RAIDAR

We experiment with the prompts used by (Mao et al., 2024) for rewriting the text samples from the dataset. We employ Gemma-2B for this purpose. To rewrite the articles in Hindi itself, we modify the



(a) BBC dataset



(b) NDTV dataset

Figure 8: Persistence Homology Dimension (PHD) estimation of various models across datasets. Two distinct clusters can be observed: one comprising GPT-4, GPT-3.5, and human-written PHD values, and the other comprising BARD, Gemma-2B, and Gemma-7B. The stark difference between these two groups, making the second group easily distinguishable from human text.

prompts accordingly. We observe that the prompts used significantly affect the output language of the rewritten text. We add an instruction along with the text while prompting it to the LLM. The following instructions were effective in generating rewritten articles in Hindi:

1. Concise this for me in Hindi only and keep all the information.
2. Help me polish this in Hindi only.
3. Make this fluent in Hindi only while making minimal changes.
4. Refine the following paragraph for me in Hindi only.
5. Revise this in Hindi only with your best efforts.

			Testing Dataset									
			BBC Dataset					NDTV Dataset				
			GPT-4	GPT-3.5	BARD	Gemma-2B	Gemma-7B	GPT-4	GPT-3.5	BARD	Gemma-2B	Gemma-7B
Training Dataset	BBC Dataset	GPT-4	98.44	97.731	88.963	97.333	96.87	81.013	81.297	73.674	81.341	83.44
		GPT-3.5	99.149	99.291	98.44	96.8	97.023	70.403	79.64	76.752	84.475	87.42
		BARD	99.574	98.156	99.007	98.667	98.168	81.013	80.919	79.072	85.714	86.699
		Gemma-2B	97.73	96.312	97.73	99.467	99.237	83.807	84.706	90.009	88.12	91.587
		Gemma-7B	96.879	95.603	98.014	99.733	99.237	83.617	85.369	93.466	88.557	88.916
	NDTV Dataset	GPT-4	99.574	99.433	99.574	97.6	98.55	99.242	98.438	99.006	91.327	90.598
		GPT-3.5	99.574	99.574	99.433	98.4	98.626	99.432	98.958	99.006	92.128	91.159
		BARD	98.156	97.589	99.291	98.933	98.626	94.366	94.602	99.29	90.743	88.622
		Gemma-2B	99.007	99.007	99.433	100	99.237	86.127	86.127	84.991	99.344	98.985
		Gemma-7B	99.149	99.574	99.433	100	99.771	85.227	86.08	85.038	99.781	99.733

Table 7: J-Guard Cross-model accuracy. J-Guard trained on GPT-3.5 data demonstrates high accuracies when testing on text generated by other models and slightly outperforms the J-Guard model trained on GPT-4 data.

			Testing Dataset									
			BBC Dataset					NDTV Dataset				
			GPT-4	GPT-3.5	BARD	Gemma-2B	Gemma-7B	GPT-4	GPT-3.5	BARD	Gemma-2B	Gemma-7B
Training Dataset	BBC Dataset	GPT-4	99.718	99.696	100	100	98.594	74.557	75.099	69.726	76.847	78.413
		GPT-3.5	98.904	99.128	99.706	100	99.057	71.069	71.296	69.259	80.126	82.656
		BARD	99.724	97.96	99.709	100	98.336	72.701	73.063	71.165	81.633	82.113
		Gemma-2B	97.796	96.755	97.983	99.454	99.545	84.8	85.215	86.961	86.183	90.547
		Gemma-7B	97.23	96.988	97.994	100	99.246	88.616	88.949	92.293	90.347	94.548
	NDTV Dataset	GPT-4	100	100	100	100	99.538	99.229	99.404	99.9	99.312	98.865
		GPT-3.5	100	100	100	100	99.692	99.613	99.606	99.8	98.829	98.759
		BARD	100	99.695	100	100	99.087	99.462	99.566	99.505	97.32	97.552
		Gemma-2B	98.901	99.123	99.712	100	99.545	78.37	78.96	77.283	98.996	98.745
		Gemma-7B	98.904	99.133	99.145	100	100	76.969	78.318	76.677	99.568	99.529

Table 8: J-Guard Cross-model Precision. Models trained on the NDTV dataset and tested on the BBC dataset consistently demonstrate high precision, with many achieving 100%. This indicates a low rate of misclassifying human text as AI-generated.

6. Rewrite this in Hindi only.

C AI Detectability Index for Hindi (ADI_{hi})

C.1 Probability Distribution Generation

In this section, we outline the process to calculate the probability distributions essential to calculate the JSD.

For each word w_i in V, we extract sentences S_h and S_{ai} from the human-written and AI-generated texts, respectively, containing w_i. To evaluate the contextual similarity of these words, we create V_{comb} by taking the union of the word sets from S_h and S_{ai}, which helps assess the surrounding context of w_i in both types of texts. We then generate co-occurrence vectors C_h and C_{ai}, which record the frequency of words from V_{comb} in their respective sentences. These vectors are normalized to form probability distributions P_h and P_{ai}, and we calculate the Jensen-Shannon Divergence (JSD) between the two distributions. The detailed algorithm for calculating ADI_hi can be found in Algorithm 1.

$$V_{\text{comb}} = V(S_h) \cup V(S_{ai}) \tag{2}$$

$$C_h(w) = \text{frequency of } w \text{ in } S_h \quad \forall w \in V_{\text{comb}} \tag{3}$$

$$C_{ai}(w) = \text{frequency of } w \text{ in } S_{ai} \quad \forall w \in V_{\text{comb}} \tag{4}$$

			Testing Dataset									
			BBC Dataset					NDTV Dataset				
			GPT-4	GPT-3.5	BARD	Gemma-2B	Gemma-7B	GPT-4	GPT-3.5	BARD	Gemma-2B	Gemma-7B
Training Dataset	**BBC Dataset**	GPT-4	97.245	95.627	77.65	94.536	95.173	93.16	92.121	80.02	90.173	93.007
		GPT-3.5	99.449	99.417	97.135	93.443	95.023	97.977	97.374	92.913	92.052	95.216
		BARD	99.449	98.251	98.281	97.268	98.039	98.266	96.304	94.98	92.486	94.374
		Gemma-2B	97.796	95.627	97.421	99.454	98.944	81.696	82.977	93.209	91.04	93.165
		Gemma-7B	96.694	93.878	97.994	99.454	99.246	76.493	79.864	94.291	86.561	82.965
	NDTV Dataset	GPT-4	99.174	98.834	99.14	95.082	97.587	99.229	97.374	98.032	83.382	82.44
		GPT-3.5	99.176	99.125	98.854	96.721	97.587	99.229	98.249	98.13	85.405	83.649
		BARD	96.419	95.335	98.567	97.814	98.19	89.017	89.3	99.016	83.96	79.6
		Gemma-2B	99.174	98.834	99.14	100	98.944	99.133	97.471	97.441	99.711	99.264
		Gemma-7B	99.449	100	99.713	100	99.548	99.807	98.735	99.016	100	99.947

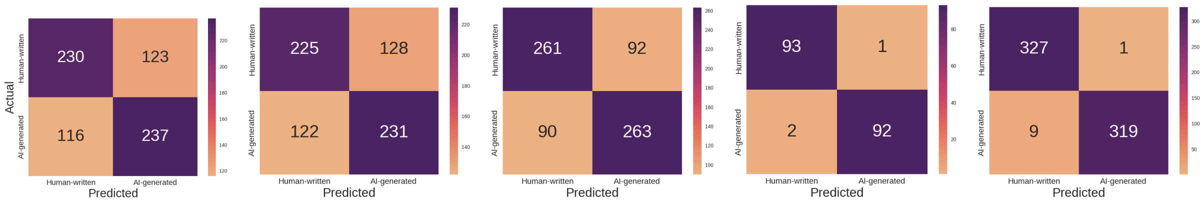
Table 9: J-Guard Cross-model Recall. We observe consistently high recall values across training and testing on both BBC and NDTV datasets, with only a few exceptions. This indicates that the model effectively minimizes missed detections of AI-generated text, irrespective of the dataset used. This robust performance suggests that the model's ability to accurately identify AI-generated text.

			Testing Dataset									
			BBC Dataset					NDTV Dataset				
			GPT-4	GPT-3.5	BARD	Gemma-2B	Gemma-7B	GPT-4	GPT-3.5	BARD	Gemma-2B	Gemma-7B
Training Dataset	**BBC Dataset**	GPT-4	98.466	97.619	87.419	97.191	96.853	82.827	82.744	74.519	82.979	85.089
		GPT-3.5	99.176	99.272	98.403	96.61	96.998	82.382	82.319	79.361	85.676	88.493
		BARD	99.586	98.108	98.99	98.615	98.187	83.572	83.089	81.366	86.721	87.818
		Gemma-2B	97.796	96.188	97.701	99.454	99.244	83.219	84.081	89.976	88.545	91.837
		Gemma-7B	96.961	95.407	97.994	99.726	99.246	82.11	84.162	93.281	88.413	88.379
	NDTV Dataset	GPT-4	99.585	99.413	99.568	97.479	98.553	99.299	98.378	98.957	90.652	89.908
		GPT-3.5	99.585	99.561	99.424	98.333	98.628	99.421	98.923	98.958	91.628	90.578
		BARD	98.177	97.466	99.279	98.895	98.636	93.95	94.154	99.26	90.147	87.666
		Gemma-2B	99.037	98.978	99.425	100	99.244	87.537	87.244	86.199	99.352	99.004
		Gemma-7B	99.176	99.565	99.429	100	99.773	86.913	87.349	86.426	99.784	99.738

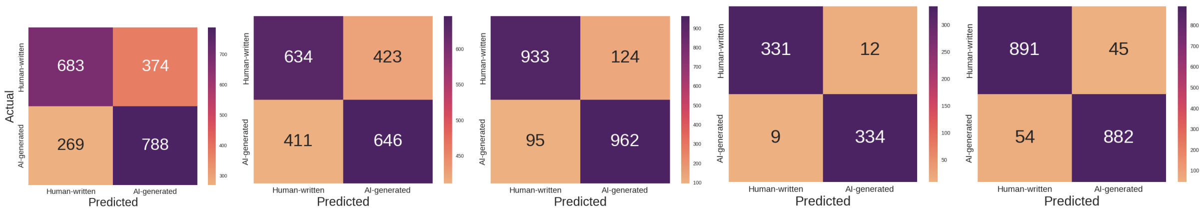
Table 10: J-Guard Cross-model F1 score. We observe that models trained on the BBC dataset and tested on the NDTV dataset exhibit lower performance compared to other combinations. Futhermore, models trained on Gemma responses struggle in detecting responses from GPT models and BARD.

$$P_h(w) = \frac{C_h(w)}{\sum_{w' \in V_{\text{comb}}} C_h(w')} \quad \forall w \in V_{\text{comb}} \tag{5}$$

$$P_{ai}(w) = \frac{C_{ai}(w)}{\sum_{w' \in V_{\text{comb}}} C_{ai}(w')} \quad \forall w \in V_{\text{comb}} \tag{6}$$

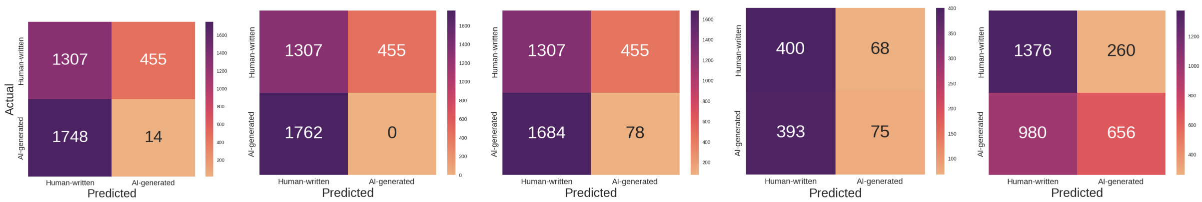


(a) BBC GPT-4 (b) BBC GPT-3.5 (c) BBC BARD (d) BBC Gemma-2B (e) BBC Gemma-7B

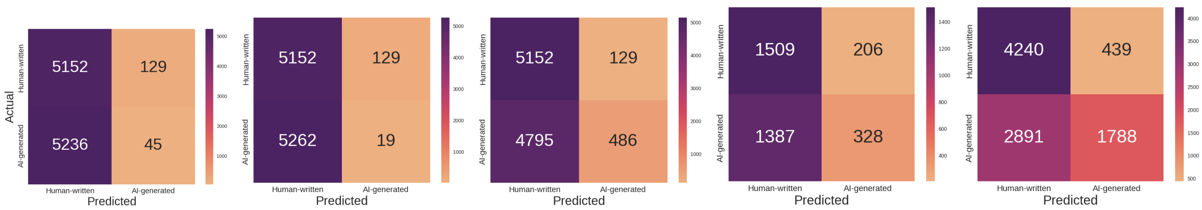


(f) NDTV GPT-4 (g) NDTV GPT-3.5 (h) NDTV BARD (i) NDTV Gemma-2B (j) NDTV Gemma-7B

Figure 9: RAIDAR Confusion Matrices. GPT models exhibit higher misclassification rates compared to other models. Similar numbers of false positives (FP) and false negatives (FN) across models and datasets indicate a trade-off-aware approach of the model.

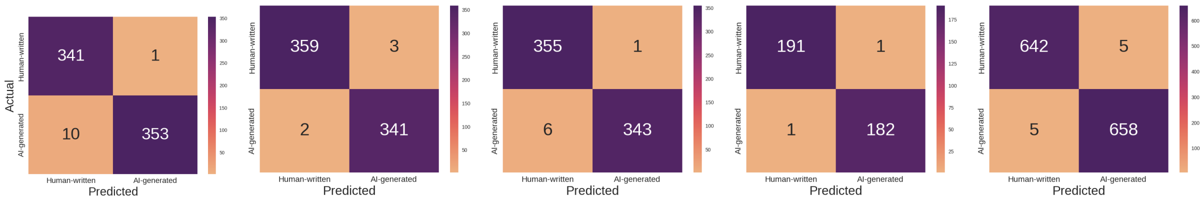


(a) BBC GPT-4 (b) BBC GPT-3.5 (c) BBC BARD (d) BBC Gemma-2B (e) BBC Gemma-7B

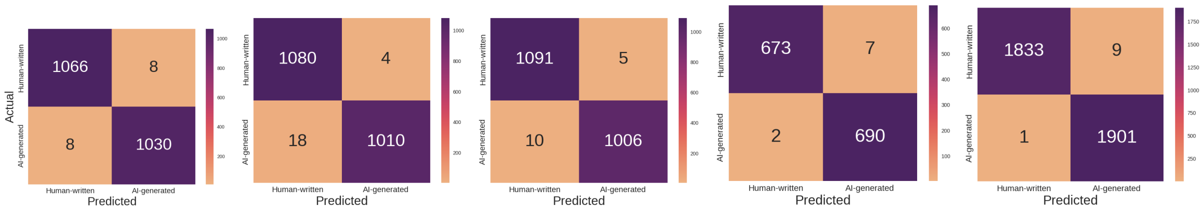


(f) NDTV GPT-4 (g) NDTV GPT-3.5 (h) NDTV BARD (i) NDTV Gemma-2B (j) NDTV Gemma-7B

Figure 10: RADAR Confusion Matrices. A significant disparity can be observed in predictions between AI-generated and human-written classes, with AI-generated classes being predicted much less frequently. This suggests that the model exhibits a bias towards identifying text as human-written rather than AI-generated, reflecting a potential challenge in accurately distinguishing between the two classes.

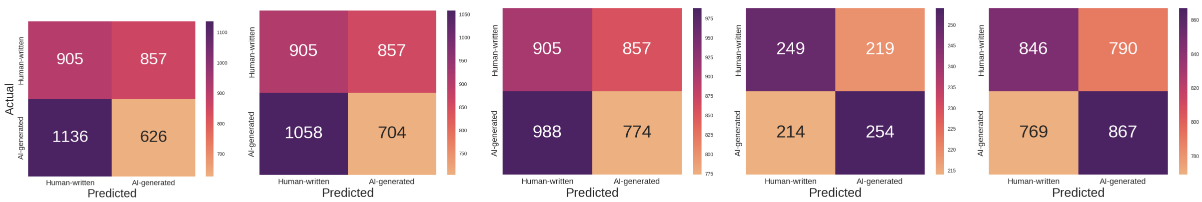


(a) BBC GPT-4 (b) BBC GPT-3.5 (c) BBC BARD (d) BBC Gemma-2B (e) BBC Gemma-7B

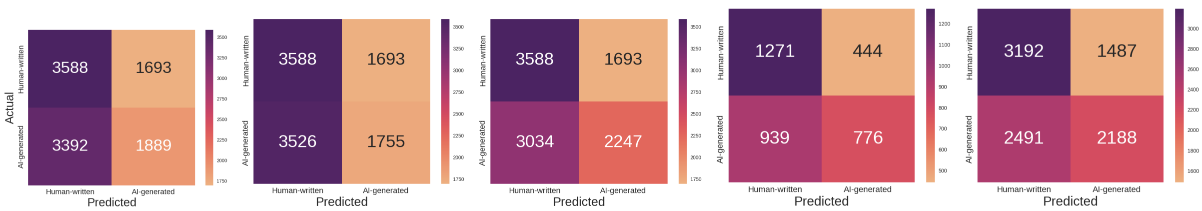


(f) NDTV GPT-4 (g) NDTV GPT-3.5 (h) NDTV BARD (i) NDTV Gemma-2B (j) NDTV Gemma-7B

Figure 11: J-Guard Confusion Matrices. The models trained and tested on the same model responses are able to distinguish AI-generated text from human-written text with a few exceptions. Instances of false positives and false negatives are minimal.



(a) BBC GPT-4 (b) BBC GPT-3.5 (c) BBC BARD (d) BBC Gemma-2B (e) BBC Gemma-7B



(f) NDTV GPT-4 (g) NDTV GPT-3.5 (h) NDTV BARD (i) NDTV Gemma-2B (j) NDTV Gemma-7B

Figure 12: ConDA Confusion Matrices. The confusion matrices reveal various misclassifications of text. Notably, Gemma model responses are slightly more detectable compared to those generated by GPT models and BARD.

Algorithm 1 *ADI_{hi}* Calculation

Require: Human-written text T_h, AI-generated text T_{ai}

Ensure: ADI_{hi} spectrum

1: Extract vocabularies V_h from T_h and V_{ai} from T_{ai}

2: Compute word intersection $V = V_h \cap V_{ai}$

3: Extract sentences S_h from T_h and S_{ai} from T_{ai} containing w_i

4: Compute $V_{comb} = V(S_h) \cup V(S_{ai})$

5: Create co-occurrence vector C_h for S_h and C_{ai} for S_{ai} with respect to V_{comb}

6: Normalize C_h and C_{ai} to create probability distributions P_h and P_{ai}

7: Compute the Jensen-Shannon Divergence:

$$\text{JSD}(P_h, P_{ai}) = \frac{1}{2} D_{\text{KL}}(P_h \| M) + \frac{1}{2} D_{\text{KL}}(P_{ai} \| M)$$

where $M = \frac{1}{2}(P_h + P_{ai})$

8: Repeat steps 3-7 for each word w_i in V

9: Compute average JSD across all words in V:

10: Apply Yeo-Johnson power transformation to ensure normal distribution

11: Perform min-max normalization to scale values between 0 and 100

12: Output the ADI_{hi} spectrum

Field	Content
Headline	पीएम मोदी के ग्रीस दौरे के बाद अदानी के नाम की चर्चा क्यों?
Human-written article	कांग्रेस पार्टी ने एक बार फिर पीएम मोदी पर अदानी समूह को फ़ायदा पहुँचाने के लिए विदेश यात्रा पर जाने का आरोप लगाया है. बीजेपी के प्रवक्ता ने इन आरोपों को बेबुनियाद बताया है जबकि अदानी समूह ने इसे कयासबाज़ी बताया है. पीएम मोदी पिछले हफ़्ते ग्रीस दौरे पर गए थे. पिछले 40 सालों में किसी भारतीय प्रधानमंत्री का ये पहला दौरा था. इससे पहले साल 1983 में भारत की तत्कालीन प्रधानमंत्री इंदिरा गांधी ग्रीस पहुंची थीं जो भूमध्य सागर क्षेत्र में भारत के लिए रणनीतिक रूप से काफ़ी अहम है. पीएम मोदी ने ग्रीस पहुँचकर कहा था, "चालीस सालों बाद भारत के किसी प्रधानमंत्री का ग्रीस आना हुआ है. फिर भी, ना तो हमारे संबंधों की गहराई कम हुई है, ना ही रिश्तों की गर्मजोशी में कोई कमी आई है." इंदिरा गांधी 23 सितंबर को ग्रीस में स्थानीय नेताओं के साथ लेकिन इसके बाद भी पीएम मोदी ने ग्रीस में सिर्फ़ कुछ घंटे ही गुज़ारे. पीएम मोदी दक्षिण अफ़्रीका में आयोजित हुए ब्रिक्स सम्मेलन से लौटते हुए ग्रीस पहुंचे थे. और इसके तुरंत बाद उन्हें बेंगलुरू पहुंचकर इसरो के वैज्ञानिकों से मुलाक़ात करनी थी. लेकिन इतने संक्षिप्त दौरे और व्यस्त कार्यक्रम के बावजूद भारत में काम कर रही कई कंपनियों के शीर्ष अधिकारी ग्रीस पहुंचकर पीएम मोदी के साथ उस बिज़नेस लंच में शामिल हुए जिसे ग्रीक पीएम ने आयोजित किया था.इस मौके पर ग्रीस के व्यापारिक जगत के शीर्ष अधिकारियों ने भारतीय व्यापारिक हस्तियों से मुलाक़ात की.
GPT-4	प्रधानमंत्री नरेंद्र मोदी के हालिया ग्रीस दौरे के बाद व्यापारिक जगत में अदानी समूह के नाम की खूब चर्चाएँ हो रही हैं। इस दौरे का मुख्य आकर्षण था द्विपक्षीय संबंधों की मजबूती और व्यापारिक सहयोग के नए आयामों का विस्तार। ग्रीस के साथ भारत के संबंध पारंपरिक रूप से मजबूत रहे हैं, और इस दौरे के द्वारा उन्होंने नई ऊर्जा का संचार किया। प्रधानमंत्री मोदी ने ग्रीस के राजनीतिक नेतृत्व और व्यापारिक प्रतिनिधियों के साथ गहन वार्ता की, जिसमें कई रणनीतिक साझेदारियां और समझौते सामने आए। लेकिन इन सबके बीच जो मुख्य मुद्दा उभर कर आया, वह था अदानी समूह की व्यापारिक विस्तार योजनाएँ। अदानी समूह, जो कि भारत के प्रमुख उद्योगपतियों में से एक है, वह ग्लोबल इंफ्रास्ट्रक्चर, लॉजिस्टिक्स, ऊर्जा सेक्टर में अपनी पहुंच बढ़ाने की तलाश में है। इस दौरे के दौरान, ग्रीस के साथ हुए समझौतों में अदानी समूह को कुछ महत्वपूर्ण परियोजनाओं के लिए विचार–विमर्श के अवसर प्राप्त हुए। जानकारी के अनुसार, अदानी समूह ग्रीस में नवीन ऊर्जा, पोर्ट डेवलपमेंट और अन्य बुनियादी ढांचे की परियोजनाओं में निवेश करने का इच्छुक है

Figure 13: Samples of articles generated by accepted models (GPT-4, GPT-3.5, BARD, Gemma-2B and Gemma-7B).

Field	Content
Headline	पीएम मोदी के ग्रीस दौरे के बाद अदानी के नाम की चर्चा क्यों?
GPT-3.5	प्रधानमंत्री नरेंद्र मोदी के सम्राटी ग्रीस दौरे सफलतापूर्वक समाप्त हो चुके हैं। इस दौरे ने उपनगरीयों के दिलों में एक तरंग उठाई है जिसके कारण देश भर में यह सवाल उठ रहा है कि पीएम मोदी के इस ग्रीस दौरे के बाद क्यों अदानी के नाम की चर्चा हो रही है। भारतीय व्यापारी और उद्योगपति मुकेश अनिल अम्बानी, जिन्हें देशभर में व्यापार का सर्वश्रेष्ठ ध्येय बताया जाता है, अपनी बदौलत एक आदर्श नागरिक के रूप में मान्यता प्राप्त कर चुके हैं। उनकी सफलता की कहानियाँ बच्चों और युवाओं के बीच मोटिवेशनल कहानियों के रूप में फैल चुकी हैं। इन नागरिकों के लिए, प्रधानमंत्री मोदी और मुकेश अम्बानी के नाम समानरूप से जुड़े हुए हैं। इस ध्यासी और सतत कामकाज के बीच पीएम मोदी ने हाल ही में एक अत्याधुनिक और नवीनतम ग्रीस यात्रा की है, जिसने लोगों के आंदोलन साधारित हो दिए हैं। इस यात्रा के कारण, अखबारों, मीडिया में एक नई परिचर्चा पर चर्चा हो रही है। पीएम मोदी के इस ग्रीस दौरे से होते हुए उनके बड़े और प्रमुख समर्थक, मुकेश अम्बानी का नाम जुड़ा हुआ आया है। विपक्षी दलों ने प्रमुख व्यापारी और उद्योगपति मुकेश अम्बानी के नाम से योग्यानुयोग प्रस्ताव रखने का आरोप लगाया है। वे दावा करते हैं कि प्रधानमंत्री ने यात्रा के दौरान अदानी के साथ व्यापारिक उद्योगों को बढ़ावा दिया है। इसके अलावा, अदानी के कुछ सौभाग्यशाली व्यापारिक उद्योगों के मध्य कई समझौतों की खबरें भी आ रही हैं। प्रधानमंत्री नरेंद्र मोदी ने इस आरोप का कड़ी निंदा करते हुए कहा है कि ये एक गलत आरोप है और जो भी समझौते हुए हैं, वे अमनचंद्र अदानी तथा उनके व्यापारिक उघारों के खातिर हुए हैं, बिना किसी राजनीतिक दबाव के। इन सारे कार्यक्रमों, समारोहों और सम्मेलनों के बावजूद, प्रधानमंत्री विदेश में जाते ही पर्यटन, वाणिज्य, उभरने वाली शिक्षा, वित्तीय खातों का मामला, उद्योग जगत को मिल रहे लाभ, बेरोजगारी दूर करने के बारे में सवाल उठाया गया है। अदानी के नाम की चर्चा होने से पता चलता है कि नागरिक समाज में कई राजनीतिक और आर्थिक मुद्दों पर चर्चा हो रही है। प्रधानमंत्री मोदी के ग्रीस दौरे की सफलता और अदानी के नाम से जुड़े तर्कों को ध्यान में रखते हुए, देशभर में यह भी मान्यता प्राप्त हो रहा है कि अदानी भारतीय उद्योग में एक बड़ी किंवदंती हैं। इसलिए, प्रधानमंत्री के नाम से जुड़े ग्रीस दौरे के बाद अदानी के नाम की चर्चा स्वाभाविक रूप से बढ़ी है। संक्षेप में कहें तो, पीएम मोदी के ग्रीस यात्रा के बाद क्यों अदानी के नाम की चर्चा हो रही है, इसका मुख्य कारण उनके द्वारा संचालित व्यापारिक उद्योगों में भारी सफलता होने से है। इसके अलावा, प्रधानमंत्री और उद्योगपति मुकेश अम्बानी के बीच के गहरे रिश्ते ने इस चर्चा को और भी तेजी से बढ़ाया है।

Figure 14: Samples of articles generated by accepted models (GPT-4, GPT-3.5, BARD, Gemma-2B and Gemma-7B).

Field	Content
Headline	पीएम मोदी के ग्रीस दौरे के बाद अदानी के नाम की चर्चा क्यों?
BARD	भारत के प्रधानमंत्री नरेंद्र मोदी ने 20 जुलाई, 2023 को ग्रीस की यात्रा की। इस यात्रा के दौरान, उन्होंने ग्रीस के राष्ट्रपति काकलिस और प्रधानमंत्री मित्सोटाकिस से मुलाकात की। इन मुलाकातों के दौरान, दोनों देशों ने कई क्षेत्रों में सहयोग बढ़ाने पर सहमति व्यक्त की।इन सहमतियों में से एक अदानी समूह की ग्रीस में निवेश करने की संभावना पर चर्चा शामिल थी। अदानी समूह एक भारतीय बहुराष्ट्रीय समूह है जो बुनियादी ढांचा, ऊर्जा, परिवहन, और अन्य क्षेत्रों में काम करता है। अदानी समूह के ग्रीसमें निवेश करने की संभावना पर चर्चा होने के बाद, अदानी के नाम की चर्चा फिर से शुरू हो गई। यह चर्चा कई कारणों से हो रही है। एक कारण यह है कि अदानी समूह भारतका सबसे बड़ा निजी निवेशक है। समूह की कुल संपत्ति लगभग 150 अरब डॉलर है। अदानी समूह ने पिछले कुछ वर्षों में भारत में कई बड़ी परियोजनाओं में निवेश किया है,जिसमें बंदरगाह, हवाई अड्डे, और बिजली स्टेशन शामिल हैं। दूसरा कारण यह है कि अदानी समूह पर कुछ विवादों का सामना करना पड़ रहा है। इन विवादों में अदानी समूह के परियोजनाओं के लिए सरकार से अनुचित लाभ प्राप्त करने के आरोप शामिल हैं। पीएम मोदी के ग्रीस दौरे के बाद अदानी के नाम की चर्चा के कुछ संभावित कारण इस प्रकार हैं: अदानी समूह के ग्रीस में निवेश करने की संभावना पर चर्चा होने से, अदानी समूह को भारत के बाहर एक बड़ी भूमिका निभाने के अवसर मिल सकते हैं। अदानी समूह पर विवादों के बीच, अदानी के नाम की चर्चा से विवादों को और बढ़ाने का खतरा है। अदानी समूह के ग्रीस में निवेश करने की संभावना पर चर्चा होने से, भारत और ग्रीस के बीच संबंधों को मजबूत करने में मदद मिल सकती है। अदानी समूह के ग्रीस में निवेश करने की संभावना का भारत और ग्रीस दोनों देशों के लिए महत्वपूर्ण आर्थिक और राजनीतिक निहितार्थ हो सकते हैं।
Gemma-2B	पीएम मोदी के ग्रीस दौरे के बाद अदानी के नाम की चर्चा क्यों? देश में पीएम मोदी के ग्रीस दौरे के बाद अदानी नाम की चर्चा में कमी आ रही है। इस दौरे में, अदानी को भारत में ग्रीस के खिलाड़ी के रूप में मानने का आरोप लगाया गया है। इस दौरे के बाद अदानी के नाम को कई लोगों में आकर्षित किया है। हमें से कुछ लोग अदानी को एक स्वच्छ और शिक्षित व्यक्ति के रूप में मानते हैं, जबकि अन्य लोग अदानी को एक अविताम और शक्त व्यक्ति के रूप में मानते हैं। इस दौरे के बाद अदानी के नाम को आकर्षित करने में सहायक में शामिल हैं: अदानी का पारस्परिक संबंध पीएम मोदी से, अदानी का शिक्षा और अनुभव, अदानी का मान्यता और विश्वास, अदानी का आत्मविश्वास और स्वच्छता इस दौरे के बाद अदानी को भारत में ग्रीस के खिलाड़ी के रूप में मानने का आरोप लगाया गया है। हमें से कुछ लोग अदानी को एक स्वच्छ और शिक्षित व्यक्ति के रूपमें मानते हैं, जबकि अन्य लोग अदानी को एक अविताम और शक्त व्यक्ति के रूप में मानते हैं।
Gemma-7B	पीएम मोदी के ग्रीस दौरे के बाद अदानी के नाम की चर्चा क्यों? अनकंडा, 10 नवंबर – प्रधानमिन मोदी के ग्रीस दौरे के बाद अदानी समूह के नाम की चर्चा शुरू हुई है। दौरे के दौरान, मोदी और ग्रीस के प्रधानमिन के बीच अदानी समूह के लिए एक समझौदा हुआ था। दौरे के बाद, अदानी समूह के शेयरों में उछौल देख गया है। कुछ विशेषजों का मानना है कि अदानी समूह के लिए ग्रीस दौरे के बाद के अवसरों में सुधार लाए हैं। अदानी समूह के लिए ग्रीस दौरे के बाद के अवसरों में सुधार लाए हैं। प्रधानमिन मोदी के ग्रीस दौरे के बाद अदानी के नाम की चर्चा के कुछ कारण हैं: अदानी और ग्रीस के बीच समझौदा: दौरे के दौरान, मोदी और ग्रीस के प्रधानमिन के बीच अदानी समूह के लिए एक समझौदा हुआ था। समझौदा के अनुसार, अदानी समूह ग्रीस में एक बड़े प्रौद्योगिकी के लिए संयंत्र स्थापित करेगा। अदानी के शेयरों में उछौल: दौरे के बाद, अदानी समूह के शेयरों में उछौल देख गया है। विशेषजों का मानना: कुछ विशेषजों का मानना है कि अदानी समूह के लिए ग्रीस दौरे के बाद के अवसरों में सुधार लाए हैं। अदानी समूह के लिए ग्रीस दौरे के बाद के अवसरों में सुधार लाए हैं।

Figure 15: Samples of articles generated by accepted models (GPT-4, GPT-3.5, BARD, Gemma-2B and Gemma-7B).

Field	Content
Headline	मधुमिता शुक्ला हत्याकांड: अमरमणि त्रिपाठी और उनकी पत्नी के जेल जाने की कहानी.
Human-written article	अगस्त 2023 नौ मई, 2003 को लखनऊ के तक़रीबन सभी वरिष्ठ पुलिस अधिकारी एक ख़ास बैठक में व्यस्त थे. शाम के चार बजे से चल रही इस बैठक में अगले दिन यानी 10 मई को शहर में होने वाले चुप ताज़िया के जुलूस की तैयारियों पर चर्चा हो रही थी. मोहर्रम के दौरान निकलने वाले हज़ारों लोगों के इस जुलूस की सुरक्षा व्यवस्था लखनऊ पुलिस के लिए हमेशा से चुनौती रही थी.
GPT-2 (English output)	May 15, 2017 The article was written by a journalist who was not a journalist. The article was published on May 15 and was published in the Hindustan Times. In the article, the journalist said that the BJP government has been trying to get the Congress to give the Congress a platform to run the country. @jamesharan_nagar @jameshaan_nao
GPT-2 XL (Code-switching)	मधुमिता शुक्ला हत्याकांड: अमरमणि त्रिपाठी और उनकी पत्नी के जेल जाने की कहानीं का क क । क सर्।ि।ी।\n\nThe article is written in Hindi and the headline is written as : In the above example, the Hindi word is used to indicate the noun . The word is used for the adjective . The adjective is also used for . The noun is used in the article to indicate a noun.
Mistral-7B instruct (Gibberish output)	मधुमिता शुक्ला हत्याकांड: अमरमणि त्रिपाठी और उनकी पत्नी के जेल जाने की कहानी मधुमाता शुक्ला हत्याकांड वालों की उनको जेल जाने की कहानी है। उनकें नहीं उनको शुचिती हमारी पुरानों की संस्करण पहले अवश्यक होता है। उनको जेल जाने की कहानी बहुती है। आजकल समस्यों को अलग उनको जेल जाने को अलग नहीं किया जाता है।

Figure 16: Examples illustrating outputs corresponding to different rejection criteria. We show outputs from GPT-2, GPT-2 XL and Mistral-7B instruct.