Highlights

`Build it`, `Break it`, `Repeat`: Benchmarking and improving LLM-manipulated disinformation detection in social media posts

Kevin Thomas, Milosz Kasprzyk, Reuel C Igbokwe Onuigbo, Elliott Pert, Cameron Tovey, João A. Leite, Olesya Razuvayevskaya, Carolina Scarton

- Iterative adversarial framework to assess machine-generated text detection.
- `breakers` developed (chained) adversarial techniques to *break* machine-generated text detection models. Adversarial techniques included: character-level and lexical perturbations, stylometric camouflage and prompt-based evasion.
- `builders` developed robust models to minimise attack evasion. Models started with a strong baseline (a fine-tuned `e5-small-LoRA`) and evolved to contrastive learning approaches.
- The `breakers-builders` process was repeated in various iterations, where `breakers` increased the complexity of attacks and `builders` tested incrementally robustness in models.
- Findings show that (1) chained adversarial techniques achieved the highest evasion scores; (2) manual evaluation and semantic preservation analysis established prompt-based evasion and best adversarial techniques; (3) triplet networks with dynamic anchor switching was the best detector; and (4) iterative evaluations showed more reliable results than static benchmarks.

arXiv:2608.09510v1 [cs.CL] 10 Aug 2026

`Build it`, `Break it`, `Repeat`: Benchmarking and improving LLM-manipulated disinformation detection in social media posts

Kevin Thomas[1], Milosz Kasprzyk[1], Reuel C Igbokwe Onuigbo, Elliott Pert, Cameron Tovey, João A. Leite, Olesya Razuvayevskaya and Carolina Scarton*

University of Sheffield, Department of Computer Science, Regent Court, 211 Portobello Street, Sheffield, S1 4DP, United Kingdom

ARTICLE INFO

Keywords:
Machine-Generated Text Detection
Red-Teaming
Jailbreaking
Disinformation
Large Language Models

ABSTRACT

Detecting machine-generated disinformation on social media is increasingly difficult as large language models (LLMs) make it easier to generate and rewrite misleading content at scale. Static benchmark evaluations, measuring detector performance on fixed held-out datasets, do not capture how detectors behave when posts are deliberately transformed to evade classification. This paper adapts the `Build it, Break it, Fix it` framework into `Build it, Break it, Repeat` (`BiBiR`): iterative sessions designed to stress-test detectors' robustness under iterative adversarial conditions, evaluating whether models remain reliable when disinformation posts are systematically transformed to evade classification. Across five iterations, the findings show that the best adversarial `breakers`' transformations came from a combination of back-translation and LLM persona-based rewriting, with the best performing technique achieving a 95% label flip rate (LFR), whilst still preserving the meaning of the original posts. The best `builders`' model was a triplet contrastive model with a dynamic anchor switching (DASS) architecture, which achieved an average accuracy of 72.68%, outperforming the strong baseline (a fine-tuned `e5-small-LoRA`) by 15 percentage points on the most robust set of `breakers`' adversarial attacks. The results demonstrate that an iterative framework best exposes detector weaknesses and pushes robustness improvements; however, it may still require semantic preservation analysis to distinguish valid adversarial evasion from transformations that changed the original disinformation claims' meaning.

1. Introduction

Recent reports indicate a sharp rise in websites publishing AI-generated news articles and political content at scale and in multiple languages[2], with part of this content being potentially categorised as Large Language Model (LLM)-generated[3] disinformation[4]. One of the main risks of LLM-generated disinformation is that generative models reduce the production barriers that constrained large-scale influence operations. While traditional campaigns, such as state-backed propaganda coordinated by the Russian Internet Research Agency, previously depended on coordinated human labour, linguistic expertise and strategic planning [2], LLMs eliminate these constraints by allowing misleading content to be produced at low cost and with limited supervision [3, 4, 5]. This enables less technically-skilled actors to generate believable content in different styles and domains, whilst larger malicious organisations can use the same system to increase their own speed and volume of existing operations. Another risk is the adaptive personalisation capabilities of LLMs, as false claims can be rewritten across many languages, personas and target audiences, which makes disinformation content easier to personalise [6, 7, 8, 9]. These modifications can be particularly harmful if used on social media, where false information have been shown to spread faster and more broadly than true information [10]. In particular, emotionally charged content, provoking fear, anger or outrage, is more likely to facilitate engagement and dissemination [11, 12]. These factors threaten to overwhelm existing verification processes, making it increasingly

*Corresponding author
c.scarton@sheffield.ac.uk (C. Scarton)
https://sheffield.ac.uk/cs/people/academic/carolina-scarton (C. Scarton)
ORCID(s): 0000-0002-0103-4072 (C. Scarton)

[1]Authors with joint first-author contribution.

[2]https://www.newsguardtech.com/special-reports/ai-tracking-center/

[3]The terms *LLM-generated*, *Machine-generated* and *AI-generated* are used interchangeably in this paper.

[4]*Disinformation* refers to false information that is deliberately created/shared with malicious intent [1]. The term disinformation is used throughout this paper to refer to machine-generated misleading content.

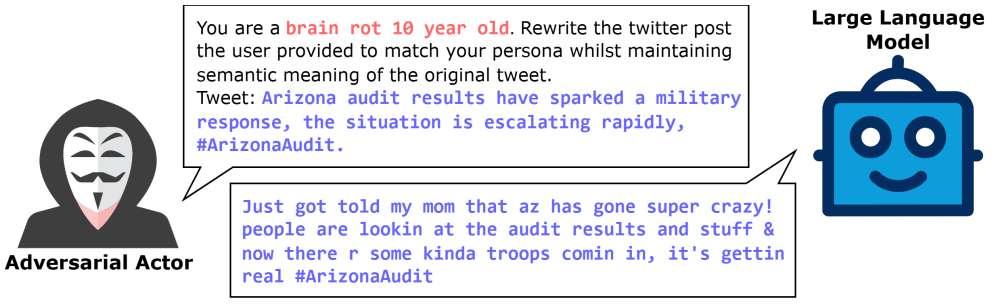


Figure 1: An example of an adversarial actor prompting an LLM to rewrite the a disinformation tweet into the style of the persona *brain-rot 10-year old*.

difficult for general audience, professional fact-checkers and journalists, as well as automated systems to distinguish between genuine and synthetic content [13].

These challenges motivate the development and evaluation of automated systems to ensure reliable detection of LLM-generated content, especially when the content is deliberately modified to avoid detection [14, 15]. Existing detectors often perform well on benchmark datasets, effectively separating machine-generated and human-written texts, via zero-shot and feature-based detection settings [16, 17, 14]. However, real-world disinformation is unlikely to remain static as adversarial actors can rewrite, paraphrase, translate, reframe or stylistically alter generated content while preserving the underlying disinformation claim (see example in Fig. 1) [18, 19, 20, 21]. Techniques such as paraphrasing and other surface-level transformations have been shown to reduce the reliability of AI-generated text detectors, including classifier-based, watermarking and zero-shot approaches [18, 22, 15]. This creates a detection problem, as detectors are not only distinguishing human-written text from machine-generated one, but are doing so against content that has been intentionally transformed to appear more human-like or less machine-identifiable [23, 21, 19]. Existing research on LLM-generated general text or disinformation has often focused on longer-form content, such as news articles, propaganda-style narratives or essays [23, 24, 25, 14, 9]. Significantly less attention has been devoted to detecting LLM-generated disinformation in short-form social media posts [26, 27, 24], despite the urgent need to combat LLM-generated disinformation in social media.

To address the above-mentioned gaps, this paper adapts the `Build it`, `Break it`, `Fix it` framework[5] [28] into a `Build it`, `Break it`, `Repeat` (`BiBiR`) framework, which suits the machine-generated detection's inherently adversarial nature, with `builders` attempting to detect LLM-generated content and `breakers` attempting to modify machine-generated content so that it preserves the disinformation claims, but becomes harder to identify as being machine-generated. Iterative versions of this framework have previously been used for offensive and safe language detection in social media [29], fact-checking model benchmarking [30] and LLM jailbreaking evaluation [31].

The overarching objective of this paper is, therefore, to evaluate and improve detector robustness under iterative adversarial conditions, in the context of short social media posts, where detection systems are improved by `builders` and then stress tested by `breakers`[6]. `Builders` measure model performance based on accuracy (ACC) and $F1$-score with the aim of improving the classification of human-authored vs. LLM-generated disinformation posts. `Breakers`' effectiveness is measured using detector performance degradation via label flip rate (LFR) and semantic preservation checks to ensure LFR is not an effect of the original disinformation claims' meaning being significantly modified. The paper also discusses the advantages of `BiBiR` evaluation (repeated `builders-breakers` rounds where transformed posts are used to test and refine detector robustness) over static evaluation (a one-off test on held-out data), assessing whether `BiBiR` can reveal weaknesses in performance that may be missed by single benchmark evaluation. To address its key objective, the paper answers the following research questions:

> **RQ1** Which adversarial transformation techniques cause the largest reduction in detector performance, measured through accuracy and label flip rate?

[5] `https://builditbreakit.org/`

[6] This work received ethics approval by the University of Sheffield Research Ethics Committee.

RQ2 Which adversarial techniques are best at preserving the meaning of the original text, whilst still evading detection?

RQ3 Between baseline transformer and contrastive model architectures, which is more effective at distinguishing machine-generated and human-written social media disinformation?

RQ4 What are the benefits of an iterative BiBiR evaluation over static evaluation in exposing detector vulnerabilities?

In answering these research questions, the paper's main contributions can be summarised as follows:

1. A focus on **disinformation narratives / stories on short social media posts** from Twitter (now X), which (1) sheds light on how LLMs deal with disinformation narratives and short social media texts; (2) assesses the performance of machine-generated text detection for short social media texts.
2. An iterative BiBiR framework adapted for **evaluating whether machine-generated disinformation detectors remain robust when social media posts are adversarially transformed**, using LLMs and/or rule-based edits.
3. **Two new datasets:** (1) **bld_data:** A tailored dataset for fine-tuning machine-generated text detectors to the social media domain with 2,560 examples for baseline training and 3,840 for contrastive learning models with dynamic anchor switching (DASS). (2) **brk_data:** A large-scale balanced dataset containing 1,08M LLM-generated/altered disinformation posts, constructed from an evaluation set with 125 machine-generated original posts (MGO) and 125 human-generated original posts (HGO).
4. A **benchmark of multiple adversarial generation and transformation techniques**, measuring their effect on detector robustness using accuracy, LFR and semantic-preservation criteria.
5. A **comparison between static and iterative adversarial evaluation**, showing how BiBiR rounds can provide a realistic assessment of detector robustness under adaptive adversarial conditions.
6. Available code and data at https://github.com/GateNLP/bibir_machine_generated_detection.

The paper is structured as follows. Section 2 introduces the related work and provides context on prior studies of LLM-generated disinformation. The BiBiR overall framework is discussed in Section 3, with the specific breakers and builders approaches being discussed in Sections 4 and 5, respectively. Then, Section 6 discusses the results in relation to the research questions. This leads to Section 7 which concludes our findings and discusses their broader implications in relation to prior work.

2. Related Work

2.1. LLM-Generated Text Detection Evasion (breakers)

Red teaming is a systematic approach for stress-testing AI systems by applying adversarial inputs or attack strategies to expose failure modes and vulnerabilities [32, 33]. In this paper, breakers focus on ***jailbreaking***, i.e. the re-framing of a restricted disinformation request so that it appears as "benign" rewriting [31, 34]. LLMs may refuse direct requests to generate misleading political content due to guardrails, but tend to comply when the same underlying claim is embedded within an indirect/role-based prompt [35, 34]. This vulnerability can be used in conjunction with other techniques like character-level perturbation, lexical perturbation, stylometric camouflage and prompt-based evasion to reduce detector reliability [19, 18, 21].

Character-level perturbation involves using spacing edits (e.g. double space & newline insertion) and punctuation noise to mimic typos made by humans. These type of actions disrupt tokenisation-based detectors and can significantly reduce detector accuracy [19, 20, 36]. Additionally, evidence shows meaning preserving surface edits reliably break text classifiers even when human judgment of the sentence semantics remains intact [37, 38, 39]. Wang et al. [40] show that under synonym-substitution perturbation, standard classifiers fall from their usual accuracy of 80-90% to single digit or low teen accuracies, despite the underlying meaning being the same. In the context of disinformation, these modifications allow the LLM-generated message to resemble authentic human text by mimicking natural noise, which facilitates detector evasion.

Lexical perturbation. Techniques like paraphrasing and back-translation are another type of attack used to mask LLM-generated text. Such attacks preserve the semantic context of text while altering its surface level structure [18, 41, 20, 22]. Paraphrasing is effective as it rewrites the syntactic and lexical choices whilst maintaining the original meaning of the text. For instance, Krishna et al. [18] show that paraphrasing can reduce the effectiveness of detectors such as DetectGPT [17], GPTZero [42], watermarking-based methods and OpenAI's text classifiers[7]. Rao et al. [41] use the OPUS MT models [43] to back-translate using French as the intermediate language between English translations, achieving an average accuracy drop of 11.54% across the 8 LLMs evaluated. This finding indicates that back-translation is an effective obfuscation technique because it disrupts the text's underlying linguistic structure.

Stylometric Camouflage. Stylometric irregularity complicates authorship detection. For example, Alperin et al. [44] show that obfuscation and mimicry attacks reduce state of the art authorship verification accuracy by 25 to 40%. Zhou et al. [21] demonstrate that injecting human noise (i.e. informal markers, emoji variation, punctuation drift) lowers AI text detector accuracy from above 90% to below 50% once stylistic camouflage is applied. These techniques combined show that small stylistic edits in adversarial tweets can reduce detector performance and allow the tweets to blend in as genuine human authored ones.

Prompt Based Evasion involves exploiting prompt engineering to alter tone and persona types. This includes tones that are political, emotionally manipulative or mimic the linguistic profile of a specific demographic [7, 8, 9, 45, 35]. This type of prompt produces adversarial outputs that are more natural, rather than outputting a default model-shaped text. Benchmarking work by Zhu et al. [46] shows that simple prompt manipulation can cause up to a 33% decrease in core LLM task performance. Pedrotti et al. [20] show that including deliberate style shifts can achieve attack success rates exceeding 70-85% when the model is told to write in alternative personas or rhetorical registers. Similarly, Zhou et al. [21] demonstrate that the humanisation of prompts reduced detector accuracy by over 50% without the need to manually edit the output text. These findings show that prompting alone can be sufficient to hinder detector behaviour, making them less reliable at distinguishing LLM-generated text from human-written text.

Summary. Prior work shows that surface edits, stylometric camouflage and prompt-based rewriting can each reduce the reliability of AI-generated text detectors [19, 20, 22, 18, 21]. However, existing studies tend to evaluate attacks within a single family, such as paraphrasing or surface perturbation, rather than examining how multi-stage combinations affect detector robustness [18, 44, 21]. In this paper, `breakers` innovate by combining and iterating existing attack types and measuring whether chained transformations reduce detector performance more than individual attacks. Another significant contribution is the assessment of human evaluation and automatic metrics to measure semantic preservation, fluency and disinformation grounding.

2.2. LLM-Generated Text Detection (`builders`)

The task of machine-generated text detection has traditionally been approached as a supervised classification problem where a neural classifier is fine-tuned on the labelled data [47, 48, 49, 50, 51]. Such methods often achieve state-of-the-art performance on in-domain data, but are known to suffer from poor generalisation [52, 53]. The rise of LLMs has created a need for detection methods that generalise across different families of generator models and do not rely on newly annotated datasets, which would otherwise need to be continually collected as new LLMs are released. This has led to the emergence of zero-shot machine-generated text detection methods [17, 54, 55]. Such methods typically employ a surrogate LLM to extract internal statistics given the input text, with a lightweight classification model often trained on those features to determine an appropriate classification threshold.

The prevalent majority of existing benchmarking datasets [56, 57, 58, 59, 60] and data challenges [61, 62] position the task of machine generated text detection as a binary classification problem, distinguishing between human-written and machine-generated text. However, the increasing popularity of text humanisation tools and sophisticated adversarial attacks has motivated the development of a more fine-grained task definition. For example, Abassy et al. [63] propose a detection framework that extends beyond the conventional binary setting by distinguishing not only between human-written and machine-generated text, but also between machine-generated text that has been humanised and human-written text that has been refined using LLMs. Our work occupies an intermediate position between these two formulations. While we retain the binary detection setting, we explicitly investigate the robustness of LLM-generated

[7] Discontinued tool; used for early evaluation of AI-text detection: https://openai.com/blog/new-ai-classifier-for-indicating-ai-written-text

text detection methods in the presence of adversarial modifications, with the goal of identifying their vulnerabilities and improving their resilience to such attacks.

2.2.1. Supervised Methods

The majority of state-of-the-art LLM-generated text detection approaches represent fine-tuned transformer-based models or an ensemble of such models [64, 65, 66, 67, 62, 68, 69]. Adversarial training with proximal policy optimization (PPO) is an efficient supervised approach, particularly when the input text undergoes different adversarial attacks [70]. In this setting, the model includes a paraphraser and a detector, where the task of the paraphraser is to generate efficient adversarial examples by taking the prediction of the detector and using it as a reward to update using PPO. Another successful approach is training transformer models using ***contrastive learning (CL)***, since it was shown to achieve high accuracy in detecting machine-generated texts [71, 65]. Rather than directly predicting labels, these models are trained to bring positive pairs closer together in an embedding space, while pushing negative pairs further apart [72, 73]. In the context of machine-generated detection, machine-machine pairs are pulled together and minimised, whilst machine-human pairs are maximised. The goal of training is, therefore, to optimise the distance between these pairs, which is done using a specialised loss function. If a positive pair that is semantically equivalent produces embeddings that are far apart, the loss function updates the weights to move the embeddings closer together in future iterations. Contrastive loss functions, such as InfoNCE [74] or NT-Xent [72], carry out this optimisation process to create an embedding space that clusters semantically similar texts while unrelated texts remain distant in the space. This model architecture is particularly efficient for LLM-generated text detection in the era of adversarial attacks because its ability to cluster semantically similar texts allows the model to look past attacks that change wording and syntax and to focus on the semantic meaning, unlike traditional classifiers that rely on fixed boundaries.

Siamese neural networks. [75] is a specific case of a contrastive learning model, which learns relational representations between two input samples simultaneously while using two identical architectures with same weights. Models with siemese architectures, such as Sentence-BERT [76], have demonstrated strong performance in semantic textual similarity and text comparison tasks, as they preserve semantic meaning and capture contextual relationships. This is particularly relevant for the LLM-generated text detection task because paraphrasing attacks have been shown to reduce the accuracy of detection systems while preserving semantic meaning [71].

Triplet networks further extend the contrastive pairing architecture of the siamese neural network by evaluating three samples simultaneously: an anchor, a positive sample from the same class, and a negative sample from the opposing class [77]. By using the loss function, the anchor is pulled closer to the positive than to the negative by at least a fixed margin, producing a more discriminative embedding space than pairwise comparison alone. In the context of LLM text detection, contrastive triplet networks show great potential in the long-form domain while remaining unexplored in the short-form texts. For example, The WhosAI [78] triplet network was designed to detect and attribute AI-generated text across multiple generators simultaneously. Although it achieved an F1 score of 0.999 on the Turing Test in the news articles domain, its efficiency on short text inputs remains unexplored.

Dynamic anchor switching strategy (DASS). Even when triplet networks are trained with carefully selected hard negative examples, paraphrased machine-generated texts with semantic claim preservation remains a challenging attack as it can drift close enough to the human cluster, thus collapsing the inter-class margin. As a defence strategy, GravText [79] proposes a detection framework that augments triplet contrastive learning with DASS. In a DASS triplet, the three inputs consist of an original LLM sample, its paraphrased variant, and a human sample on the same topic. What distinguishes DASS as an effective defence strategy against paraphrasing is that the triplet loss equation's anchor alternates between the original and the paraphrase across training steps. This bidirectional signal forces the model to learn that the original and the paraphrased variants of the same text belong in a tight machine-generated cluster, while the human text remains beyond a grounded fixed margin. GravText demonstrated the effectiveness of this strategy on static long-form essays and formal question-answering datasets, however, similarly to the triplet network evaluation, short-form texts has not yet been investigated.

2.2.2. Zero-shot Methods

The capabilities of LLMs in generating highly coherent and believable text has motivated a need for more universal approaches that do not rely on human-annotated training data. Consequently, a growing body of work has explored

zero-shot approaches that leverage surrogate LLMs to extract statistical signals from the input text for distinguishing between human- and machine-generated content.

DetectGPT, Fast-DetectGPT and Binoculars. The pioneering method in this direction is *DetectGPT* [17], which uses a surrogate LLM to extract entropy, log-perplexity, and log-rank under difference of a given text. The authors demonstrate that these features are able to distinguish between human and machine-generated text under multiple textual perturbations. Building on this idea, F*ast-DetectGPT* replaces the computationally expensive perturbation step of DetectGPT with the sampling of the alternative next tokens [80]. It then compares their conditional probabilities to that of a given next token. As an extension to DetecGPT, *Binoculars* approach [81] explores log-perplexity of the surrogate LLM as a normalised alternative of perplexity and demonstrates its effectiveness for zero-shot detection.

RepReGuard method [55] lies at the intersection between supervised and zero-shot approaches. It extracts representation-based features from a surrogate LLM and trains a lightweight machine learning classifier on a small labelled dataset to calibrate the decision threshold. Specifically, the method exploits fine-grained activation patterns of the surrogate model and applies Principal Component Analysis (PCA) to identify the most discriminative representations. By requiring only a limited amount of labelled data for threshold calibration, RepReGuard combines the generalisation capabilities of zero-shot methods with the adaptability of supervised approaches.

Temperature sensitivity (TS). Finally, Ma et al. [54] propose TS, a normalized feature that quantifies the sensitivity of a surrogate LLM's output distribution to changes in the sampling temperature. Specifically, the method measures the normalised difference between the observed and expected logits over the model's vocabulary under low and high temperature settings. The underlying intuition is that LLM-generated text is typically produced by selecting highly probable next tokens, making the difference with the expected logits more sensitive to temperature variations than that for human-written text, which is not generated under this assumption. Experimental results demonstrate that TS substantially outperforms previous zero-shot detection methods across a range of benchmarks.

Although zero-shot methods generalise well across domains and families of generative models while avoiding the need for newly annotated training data, supervised approaches continue to achieve substantially higher performance in in-domain detection settings. Therefore, this work focuses on developing supervised detection methods that remain robust in the presence of adversarially modified inputs.

2.2.3. Limitations of existing approaches

Existing detectors for AI-generated text have mainly been evaluated on static benchmarks consisting of long-form, formal text, where adversarial data augmentation is either absent or applied in a single pre-defined round [57]. These constraints make the evaluation results not reflective of the real-world, where evasion techniques and generation models constantly evolve, making detection particularly challenging for social media texts due to their short-form nature.

It remains unexplored how an architectural ladder of detection approaches performs against chained (i.e. applied sequentially on top of each other) evolving attacks on short-form social media misinformation in an iterative evaluation framework. Specifically, it is unknown whether a progression of detection strategies from simple classifiers and data augmentation to contrastive models with hard negative mining and paraphrase-aware anchor switching (DASS) offers a genuine robustness strategy against adaptive and advanced adversarial techniques in an already difficult and noisy domain. Without addressing this gap, current research in short-form text domains is limited to detector development guided by static conditions that do not reflect the evolving threats in real-world settings.

3. Proposed Framework

The results of this paper were achieved as part of a master-level course were a group of five students is required to develop a research project. In the `BiBiR` framework, the students were divide in `builders` (three students) and `breakers` (two students). Fig. 2 outlines the proposed adversarial `BiBiR` framework. The framework is organised into five stages (described below in more detail): **(I) Dataset Construction**, **(II)** `Breakers`**' Transformation**, **(III) Semantic Preservation Analysis**, **(IV)** `Builders`**' Evaluation**, and **(V) Challenge Week**.

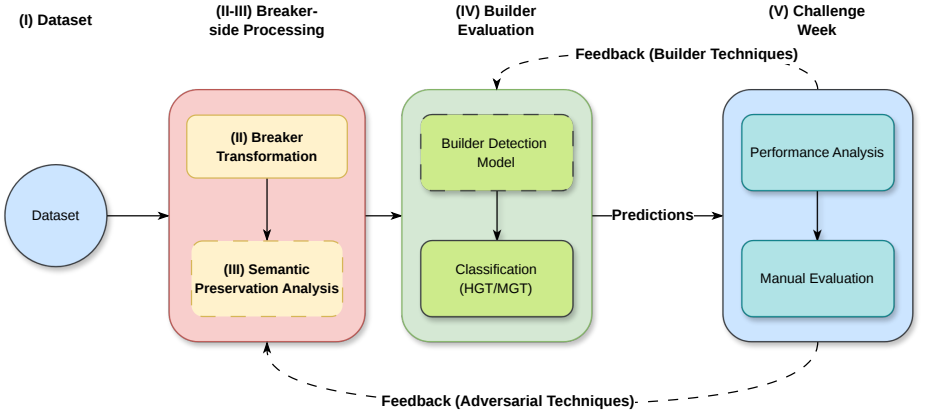


Figure 2: Overview of the proposed adversarial evaluation framework, showing dataset construction, `breakers`-side transformation and semantic filtering, `builders`' evaluation, Challenge Week analysis, and feedback loops for refining `builders`' and `breakers`' techniques.

(I) Datasets. Table 1 summarises the data sources and methods used by `builders` and `breakers`. The `breakers`' dataset (bkr_data) is composed of two static ground-truth datasets, HGO and MGO, both with the same size to guarantee a balanced evaluation. For the HGO dataset, human-authored misinformation tweets were randomly sampled from the rumour subset of PHEME [82]. These samples were then classified using the `builders`' baseline classification model before any adversarial transformation. Only the examples classified as human-generated (125 in total) were retained in the HGO baseline set. The MGO dataset was generated using zero-shot prompting with LLaMA-3.1 [83]. A total of 50 disinformation claims were used as generation inputs, drawn from TruthSeeker [84] and the false narratives in AI-TRAITS [9], with five candidate X-style posts generated per seed disinformation claim, producing 250 candidate MGO samples. These candidates were then passed through the baseline `builders`' classifier model, and 125 posts classified as machine-generated were selected at random for the static MGO set. Filtering the bkr_data dataset to include only examples initially classified correctly by the `builders`' baseline classifier supports a clearer interpretation of LFR, since label flips are measured from initially correct predictions rather than from examples already misclassified before transformation. Nevertheless, this introduces selection bias because initially misclassified examples are excluded, including HGO false positives and MGO false negatives. This trade-off was acceptable because our aim was to measure whether `breakers`' transformations can cause initially correctly detector predictions to fail. Using both HGO and MGO examples allows the framework to evaluate both directions of detector error: machine-generated posts being misclassified as human-written, and human-written posts being incorrectly flagged as machine-generated. This is important in the context of short-form disinformation as LLM-generated claims can be difficult for both human readers and automated detectors to distinguish from human-authored claims [34]. It is also useful to keep the detector grounded in a real-world scenario, i.e. a detector that correctly identifies all MGOs, but at the same time misclassifies all HGOs, is not a useful detector.

To avoid sample-level data leakage, `builders`' data (bld_data) was independent from the posts used by `breakers`' (bkr_data). This ensured that `builders`' models were not evaluated on the same examples used during training or validation. For human-authored data, PHEME [82] and Constraint 2021: COVID-19 datasets [85] were chosen (1,296 posts were sampled from these datasets). They both have human-verified misinformation and have been used in works such as [86, 87, 88], which selected them for their authenticity in misinformation labelling. For generating the machine-generated counterpart of their dataset, `builders`' used tweet injection: the human-authored tweets selected from PHEME and Constraint were embedded into a persona-engineered prompt and passed to an LLM (more details in Section 5.1.2).

(II) `Breakers'` Transformation. `Breakers` transform HGO and MGO into adversarial variants using a combination of rule-based perturbations and LLM-based edits. Rule-based perturbations are applied programmatically to the text, while LLM-based transformations are implemented through prompt-controlled rewriting (see Section 4.1). These transformations aim to exploit the reliance on surface-level linguistic features used by detectors [20].

Table 1
Summary of the datasets used by builders (bld_data) and breakers (bkr_data). The table shows the source data, sample counts, purpose of each subset, and the overlap-control step used to avoid data leakage between builders' training and breakers' evaluation.

	Component	Source / Model	Size	Purpose / Overlap Control
bkr_data	HGO dataset	PHEME	125 posts	Human-authored misinformation baseline. Samples were kept disjoint from builders' data. All samples are classified as *human* by the builders' baseline classifier.
	MGO generation	TruthSeeker, AI-TRAITS	50 claims / narratives	Inputs used to generate machine-authored X-style disinformation posts.
	MGO candidate samples	Llama-3.1-8B-Instruct	250 posts	Five candidate posts generated per input claim or narrative using zero-shot prompting with Llama-3.1.
	MGO dataset	e5-small-lora	125 posts	MGO candidates classified as machine-generated by the builders' baseline detector and retained for adversarial evaluation.
bld_data	human-authored text	PHEME, Constraint	1,296 posts	Human misinformation samples used for training and validation. The set has 639 posts from PHEME and 657 post from Constraint samples and is completely independent from the breakers' set.
	machine-generated text	grok-4-1-fast-non-reasoning Llama-3.3-70B-Instruct gpt-4o DeepSeek-R1	1,296 posts	The human misinformation samples are passed through an LLM embedded into persona-engineered prompts.

(III) Semantic Preservation Analysis. Transformed samples are passed through a semantic preservation pipeline that computes automatic similarity and consistency metrics between the original post and its transformed variant. The purpose of this stage is to quantify whether adversarial transformations preserve the original meaning and disinformation claim. The pipeline applies Natural Language Inference (NLI), token length ratio, E5 cosine similarity and BLEURT to characterise aspects of preservation, such as contradiction, semantic similarity, structural drift and generation quality. The pipeline is used as an automatic analysis layer rather than an exclusion mechanism and is described in more detail in Section 4.2.

(IV) Builders' Evaluation. The transformed adversarial samples are passed to the builders' detector, which classifies each post as human-generated or machine-generated (see Section 5). At this stage, the techniques employed by the breakers is unknown to the builders. Builders' performance is evaluated using accuracy (ACC) and $F1$-score:[8]

$$ACC = \frac{\mathrm{TP} + \mathrm{TN}}{N} \tag{1}$$

$$F_1 = \frac{2 \times \mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{2}$$

A successful builders' model has a high $F1$ score and accuracy, meaning the model makes few FP and FN errors.

[8]True Positive (TP) refers to an MGO correctly identified as machine-generated, while True Negative (TN) are the HGO correctly identified as human-generated. False Positive (FP) refers to an HGO incorrectly identified as machine-generated. False Negative (FN) refers to a machine-generated post incorrectly identified as human-generated. N is the total number of samples.

Since the bkr_data contains HGO and MGO samples that were initially classified correctly by the builders' detector, LFR is calculated as the inverse of accuracy on the transformed adversarial samples:

$$LFR = 1 - ACC \tag{3}$$

where ACC refers to the builders' model accuracy on the transformed adversarial samples. A high LFR highlights that a breakers' technique frequently causes initially correct builders' predictions to become incorrect. A successful breakers' transformation is therefore interpreted as one that causes a detector label to flip while retaining the original disinformation claim, with preservation assessed through the semantic preservation pipeline and manual evaluation (see Section 4.3).

(V) Challenge Week. Results from the builders' evaluation are manually reviewed to identify the breakers' techniques that produce the largest changes in detector behaviour, including high LFR and substantial drops in classification accuracy. The results from the *challenge week* are reviewed by both teams, and observed trends are used to refine subsequent builders' and breakers' strategies. The pipeline is then repeated (except stage I): starting from II and III, followed by IV and V. In total, five iterations were completed. Table 2 presents an overview of the breakers' and builders' techniques and data usage per iteration.

Table 2
Overview of the BiBiR activity with five iterations.

	Breakers		Builders	
Iter.	**Adversarial techniques**	**# generated posts**	**Models**	**Data size (train:dev:test)**
1	D2, B2, C3, A3	12,000	baseline (e5-small-LoRa)	2,560 (1,920:384:256)
2	D1, D2, B2, C3, A3	90,000	baseline + back-translation	3,200 (2,560:384:256)
3	B1, D1, D2, B2, C3, A3	180,000	Siamese(1:1) Siamese(TF-IDF)	2,560 (1,920:384:256)
4	B3, D1, D3, B2, C3, A3	180,000	Triplet(TF-IDF)	2,560 (1,920:384:256)
5	B3, D1, D4, B2, C3, A3	180,000	Triplet(DASS)	3,840 (2,880:576:384)

4. Breakers' Approach

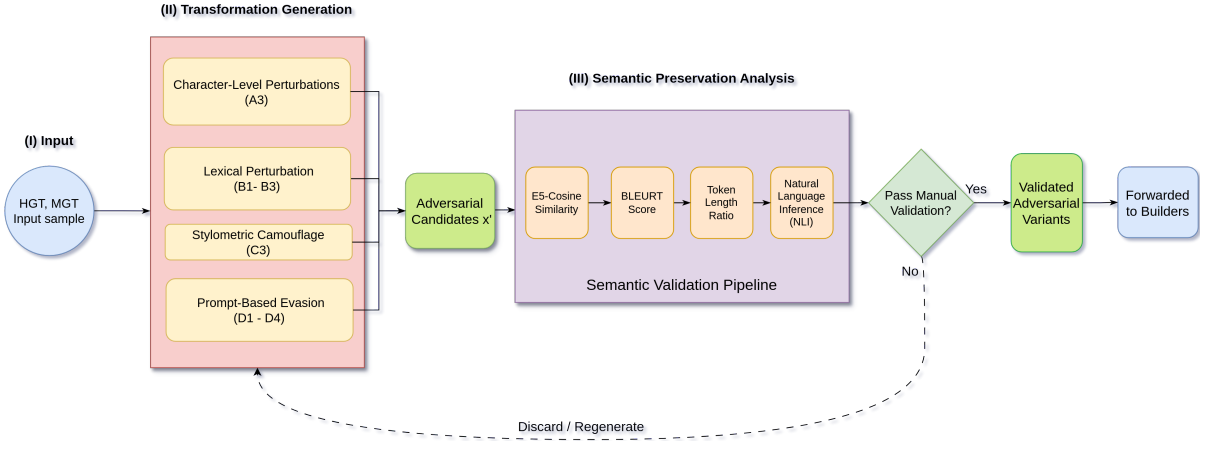


Figure 3: Overview of the breakers' methodology, showing how HGO and MGO input samples are transformed into adversarial variants, analysed using semantic preservation metrics, and passed forward for builders' evaluation and performance analysis.

Table 3
Implementation summary of breakers' techniques, showing each attack family, technique code, transformation method, and model or tool used.

Technique Family	Code	Method / Tool	Implementation
Character-Level Perturbation	A3	Rule-based whitespace editing	Applies spacing edits including double spaces, spaces before punctuation, random newlines, and extra blank lines.
Lexical Perturbation	B1	`LLaMA-3.1-8B-Instruct`	Prompt-based paraphrasing to produce alternative wording while preserving the original claim.
	B2	OPUS-MT / NLLB	Translates English posts into an intermediate language and back into English using the routes in Table 4.
	B3	`LLaMA-3.1-8B-Instruct`	Revised paraphrasing prompt designed to produce stronger lexical and syntactic variation than B1.
Stylometric Camouflage	C3	Emotion English DistilRoBERTa-base + emoji mapping	Uses an emotion classifier to assign an emotion label, maps the label to an emoji set, and appends one sampled emoji where eligible.
Prompt-Based Evasion	D1	`LLaMA-3.1-8B-Instruct`	Rewrites the post using a tone-based prompting: *You are currently in this mood: [TONE]. Rewrite the Twitter post provided to match your mood whilst maintaining the semantic meaning of the original post. [RULES]*
	D2	`LLaMA-3.1-8B-Instruct`	Applies a simple persona-based prompting: *You are a [PERSONA]. Rewrite the Twitter post the user provided to match your persona whilst maintaining the semantic meaning of the original tweet. [RULES]*
	D3	`LLaMA-3.1-8B-Instruct`	Applies refined persona-based prompting: *You are a [AGE] year old [ETHNICITY] [GENDER] living in the [LOCATION], who is currently a [OCCUPATION]. Rewrite the Twitter post the user provided to match your persona whilst maintaining the semantic meaning of the original tweet. [RULES].*
	D4	LLaMA 3.1 8B Instruct	Applies an indirect persona-based prompting: *Think like a [AGE] year old [ETHNICITY] [GENDER] living in the [LOCATION], who is currently a [OCCUPATION]. Rewrite the Twitter post the user provided to match your persona whilst maintaining the semantic meaning of the original tweet. [RULES].*

Given an input sample $x \in \{\text{HGO}, \text{MGO}\}$, breakers' techniques produce adversarial variants x' by applying transformations that modify the textual form of the original post while aiming to preserve its underlying disinformation claim. As shown in Fig. 3, the breakers-side workflow consists of four steps: (I) starting with an HGO or MGO input sample, (II) applying one or more breakers' transformations, (III) computing semantic preservation metrics for the original-transformed pair, and passing the resulting adversarial variant and preservation scores forward for builders' evaluation and performance analysis. The implemented techniques are grouped into four families: character-level perturbation (A3), lexical perturbation(B1 to B3), stylometric camouflage (C3), and prompt-based evasion (D1 to D4). The full implementation taxonomy is summarised in Table 3.

Llama-3.1 (`Llama-3.1-8B-Instruct`) [83] was used for all LLM-based transformations, i.e. all techniques except the rule-based perturbation (A3 and C3) and back-translation (B2). This model was selected due to previous work using it for adversarial rewriting, AI-text detector evasion and disinformation-style generation [20, 89, 45]. The 8B variant was as a practical trade-off between model capability and computational constraints.

4.1. Text Transformation

Character-level perturbation (A3). Character-level perturbation is implemented as a rule-based technique that applies spacing edits to the input text. These edits include double-space insertion, spaces before punctuation, random newline insertion, and extra blank lines. The purpose of A3 is to identify whether the builders' detector is sensitive

to low-level formatting changes that may disrupt tokenisation or alter surface-level text patterns while preserving the original claim [37, 38].

Lexical perturbation (B1, B2, B3). Lexical perturbation includes paraphrasing (B1 and B3) and back-translation (B2). These techniques alter wording and sentence structure while aiming to preserve the original disinformation claim [18, 21]. B1 and B3 are implemented using LLaMA 3.1 8B Instruct, with text being rewritten to generate alternative phrasings of the original post (with B1 being more conservative and B3 more aggressive). B2 is implemented using back-translation, where the input is translated into an intermediate language and then translated back into English. Back-translation was implemented using pre-trained `OPUS_MT` models [43] from the `Helsinki-NLP` HuggingFace collection,[9] specifically those highlighted in Table 4. Japanese back-translation initially used the `OPUS_MT` Japanese models, but generation was later switched to the NLLB model [90], also from HuggingFace,[10] due to mistranslation issues (frequently, the model hallucinated unrelated Bible-like text rather than translating the original input). A B2 technique is named as B2_IL, where *IL* is the intermediate language used in the back-translation pipeline. This category of techniques tests whether the `builders`' models relies on lexical or syntactic regularities which detectors normally associate with machine-generated text.

Table 4
Back-translation routes used for B2 lexical perturbation.

Intermediate Language (IL)	English → IL	IL → English
Arabic (AR)	`Helsinki-NLP/opus-mt-en-ar`	`Helsinki-NLP/opus-mt-ar-en`
Russian (RU)	`Helsinki-NLP/opus-mt-en-ru`	`Helsinki-NLP/opus-mt-ru-en`
Japanese (JA)	`facebook/nllb-200-distilled-600M`	`facebook/nllb-200-distilled-600M`
German (DE)	`Helsinki-NLP/opus-mt-en-de`	`Helsinki-NLP/opus-mt-de-en`
Chinese (ZH)	`Helsinki-NLP/opus-mt-en-zh`	`Helsinki-NLP/opus-mt-zh-en`

Stylometric camouflage (C3). Stylometric camouflage introduces emoji-based variation into posts. Unlike the LLM-based techniques, C3 is implemented programmatically. Each post is first passed through an emotion classifier (Emotion English DistilRoBERTa-base[11]) which assigns an emotion from the following label set: *anger*, *fear*, *joy*, *neutral*, *sadness* or *surprise*. The predicted emotion is then mapped to a predefined set of emojis, and one emoji is randomly sampled from the corresponding category. The sampled emoji is appended to the end of the post when the post does not already have an emoji and has at least two words.

Prompt-based evasion (D1, D2, D3, D4). Prompt-based evasion techniques are injected using LLaMA 3.1 8B Instruct with prompt-controlled rewriting to change the tone, persona, or rhetorical framing of the original post while preserving the underlying disinformation claim. D1 alters the emotional tone of the post, while D2, D3, and D4 use persona-based prompts of increasing refinement (as described in Table 3). Table 5 shows the specific options for each D-type technique, specifying each subgroup with their corresponding atomic codes. As shown, four persona-types are defined: *politician*, *college student*, *brain-rot 10-year old*, and *four-start military general*. Attributes are stereotypical, since the aim of the study was to generate significantly altered text, instead of systematically assessing the capabilities of LLMs of dealing with different personas.

Chained transformations. In addition to individual attacks, selected `breakers`' techniques are combined into chained transformations. In these cases, the output of one technique becomes the input to a subsequent technique. For example, a persona-based rewrite may be followed by back-translation and a spacing edit (which would be referred to as D3_B2_A3). Chained transformations are used to test whether combining attack families causes greater detector degradation than applying a single transformation in isolation. The order of each chained transformation follows the order indicated by the technique code.

[9] `https://huggingface.co/Helsinki-NLP`
[10] `https://huggingface.co/facebook/nllb-200-distilled-600M`
[11] `https://huggingface.co/j-hartmann/emotion-english-distilroberta-base`

Table 5
Detailed taxonomy of D-type techniques, showing the options for prompts and the specific codes.

Technique Family	Atomic Technique Code	Technique Description
All	–	[RULES] = Rules for the rewritten tweet: – Maximum character length: 200 – Do not change the semantic meaning – Keep as much original content as possible – Do not generate any text apart from the rewritten tweet
D1	D11	[TONE]= Joy
	D12	[TONE]= Anger
	D13	[TONE]= Sadness
	D14	[TONE]= Surprise
	D15	[TONE]= Neutral
D2, D3, D4	D21, D31, D41	Persona: *Politician* [AGE] = 40; [ETHNICITY] = Caucasian; [GENDER] = Male; [LOCATION] = USA; [OCCUPATION] = Politician
	D22, D32, D42	Persona: *College student* [AGE] = 20; [ETHNICITY] = Asian; [GENDER] = Female; [LOCATION] = USA; [OCCUPATION] = College Student
	D23, D33, D43	Persona: *Brain-rot 10-year old* [AGE] = 10; [ETHNICITY] = African-American; [GENDER] = Male; [LOCATION] = USA; [OCCUPATION] = Elementary School Student
	D24, D34, D44	Persona: *Four-star military general* [AGE] = 60; [ETHNICITY] = Caucasian; [GENDER] = Male; [LOCATION] = USA; [OCCUPATION] = US Army General

4.2. Semantic Preservation Analysis

To analyse whether `breakers`' transformations preserve the meaning of the original post, each original-transformed pair is passed through a semantic preservation pipeline. This pipeline computes automatic similarity and consistency metrics, including Natural Language Inference (NLI) [91], token-length ratio, E5-cosine similarity [92], and BLEURT [93]. These metrics characterise different aspects of preservation: NLI categorises the posts as *entailment* (the claim of the output matches the original), *contradiction* (the claims do not match) or *neutral* (there is not enough information to determine if the original and altered versions match); token-length ratio captures large structural changes; and E5-cosine and BLEURT estimate semantic similarity.

The semantic preservation pipeline is used as an automatic analysis layer rather than as a filtering mechanism. Automatic metrics can miss subtle changes in stance, implication, or claim framing, particularly for paraphrasing, back-translation, and persona-based rewriting. For this reason, the automatic preservation scores are later compared with manual evaluation scores for semantic preservation, naturalness, and preservation of disinformation claim, as is further discussed in Section 4.3.

4.3. Manual Evaluation

To ensure claims are maintained and the effectiveness of the filtering pipeline is validated, manual evaluation (ME) is conducted by human annotators within the team. The three criteria measured were *(ME1) semantic preservation* – measuring how well a modified text preserves the meaning and intent of the original text; *(ME2) naturalness* – measuring how fluent, human-like, and natural the altered text sounds in its context; and *(ME3) preservation of disinformation claim* – measuring whether the altered text retains the same misleading claim as the original without changing the core disinformation narrative.

Table 6 provides a snapshot of the guidelines used for manual evaluation of selected posts. The full annotation guidelines are provided in Appendix A. For **ME1** and **ME2**, annotators were instructed to assign a midpoint score when uncertain between two adjacent scoring categories. For example, if an annotator was unsure whether a post should be scored as 2 or 1, they should assign a score of 1.5. This was not required for **ME3**, as its scoring scale already includes a midpoint value of 0.5 to represent partial claim preservation.

Table 6
Manual evaluation scores for assessing transformed output posts across semantic preservation, naturalness, and disinformation claim retention.

Manual Evaluation Category	Metric	Description
ME1 – semantic preservation	3	*Identical meaning preserved*. The modified text has the same meaning as the original. The reader should draw the same conclusion from both texts, and the veracity of the claim should be unchanged.
	2	*Partially changed*. This includes changes in certainty or strength of the claim, alteration in action, target, numbers, dates, or facts, and important details being removed or added. The text may contain partially true values, such as one truthful claim being preserved while another is changed, or ambiguity that affects interpretation.
	1	*Irrelevant, completely unrelated, or opposite meaning*. The modified text contradicts or reverses the original meaning. This includes explicit negation, denial of the original claim, or text that no longer makes sense in comparison to the original version.
ME2 – naturalness	4	*Natural and well-formed text*. The modified text flows smoothly, uses a natural tone and varied word choice, and does not contain repetitive patterns. The text reads fluently without irregularities.
	3	*Mostly natural with minor awkwardness*. The modified text is fluent and understandable, but may contain mild redundancy or small grammatical stiffness. The text may feel slightly over-structured or less polished, but the overall structure remains coherent.
	2	*Noticeably awkward or contains translation artefacts*. The modified text is understandable, but the word choice is poor, with unnatural synonyms, broken collocations, or awkward phrasing.
	1	C*learly broken or incoherent*. The text is difficult or impossible to read, with severe grammatical errors, broken syntax, or fragmented output.
ME3 – preservation of disinformation claim	1	*Claim held*. The output preserves the original disinformation claim and maintains the same misleading narrative, even if the wording, tone, or surface form has changed.
	0.5	*Claim partially held*. The output retains part of the original disinformation claim, but weakens, softens, narrows, or slightly shifts the claim so that the misleading narrative is only partly preserved.
	0	*Claim not held*. The output removes, contradicts, or substantially changes the original disinformation claim, meaning the core misleading narrative is no longer preserved.

Manual evaluation was restricted to the MGO set since the objective of this paper is to assess whether adversarial transformations can make machine-generated disinformation appear human-written while preserving the original claim. Given the size of the dataset, it was not practical to manually evaluate all outputs for all `breakers`' techniques. Instead, a targeted sampling approach was adopted, where the newest implemented technique family and variants were selected. For each selected variant, ten posts were manually evaluated: five posts that were classified by the detector as HGO and five posts that were classified as MGO. This selection helps check if the HGO classification given to a post is caused by successful adversarial transformation, rather than by cases where the output is classified as HGO due to the original disinformation claim being changed.

Annotation was carried out by four Computer Science students, knowledgeable in the topic and co-authors for this paper. They annotated across the posts selected per iteration and using the guidelines specified in Appendix A. To reduce the effect of inter-annotator variation, the scores assigned to each annotated post were averaged for each manual evaluation criterion. These averaged values were then used as the final **ME1**, **ME2** and **ME3** scores for each annotated post. The final manually evaluated set contained 780 posts across iterations 3, 4 and 5. The distribution of the manually evaluated posts is shown in Table 7.

Table 7
Count of manual evaluation posts selected from each iteration, split by number of HGO-classified and MGO-classified posts.

Iter.	HGO-Classified Posts	MGO-Classified Posts	Total Posts
3	145	145	290
4	125	125	250
5	120	120	240
Total	390	390	780

5. Builders' Approach

The builders adopted an iterative defence strategy in which the models were progressively strengthened in response to the breakers' escalating attack approaches. At each iteration, a new model was introduced as a response to the previous round's of breakers' strategy. As iterations progressed, the focus shifted from the baseline classifier model (e5-small-LoRa) to contrastive models that enforce paraphrase robustness and stylistic authorship cues. This section describes the various activities from the builders' side: dataset preparation (Section 5.1), model development (Section Section 5.2) and experimental settings (Section 5.3).

5.1. Datasets for training detectors

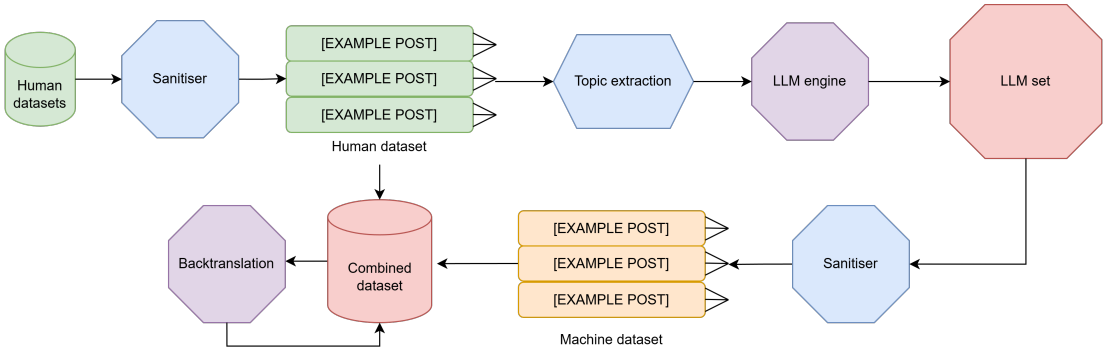


Figure 4: Builders's data pipeline.

Fig. 4 illustrates the full pipeline used by the builders to create a dataset suitable for training and validating their models. Starting from human data (disinformation appearances in social media), builders~apply multiple pre-processing techniques and prompt LLMs in order to generate machine-written counterparts for each human data point. This machine-generated data also pass through a sanitisation process, before being combined with the human data for the final dataset.

5.1.1. Human data

For disinformation research on social media, the foundational data can be more critical than the model itself [94]. Therefore, the human-generated set was built from two public datasets whose posts were published before the generative AI era (therefore, highly likely not machine generated): PHEME [82] and Constraint [85]. Both collections are tied to known rumours or fact-checked claims verified by researchers.

From the PHEME dataset, 638 tweets were selected after filtering out any samples for which the veracity label was ambiguous or where the text contained unwanted artefacts (e.g. non-English fragments, excessive token repetition). To prevent overfitting to a single event type, topic diversity was introduced with the Constraint set. A total of 658 scientific-health rumour tweets were sampled from the Constraint set using the same filtering criteria to complement the breaking news events covered by PHEME. Both sources were kept in roughly equal proportion so that neither domain would dominate in the training stage. The final human-written dataset contained 1,296 clean posts.

5.1.2. Machine-generated data

For each sample from the human set, a machine-generated counterpart was synthesised using tweet injection, i.e. the full text of the human post was embedded inside a specific persona-engineered prompt and fed to an LLM to create new logical pathways while maintaining the misinformation claim of the original HGO sample. To prevent the detector from overfitting to a single surface style, ten distinct prompt templates were designed. They used a wide range of personas: e.g. *a careless phone user*, *an objective news report*, and *a jaded internet commentator*. Each template instructed the model to alter the sentence structure, starting point, and word choice to create a new machine-generated text and not just perform simple synonym swaps. Additionally, all templates enforced the same negative constraints (placeholders, no emojis) that mirror the sanitisation pipeline (Subsec. 5.1.3). These constraints were enforced during generation time so that the output maintains a coherent sentence structure, as stripping these elements at post-processing would create unnatural gaps, fragments, or artificially shortened posts that the detector could easily learn to exploit. An example template is shown below (an exhaustive list of the ten templates is provided in Appendix. C Fig. 5.1.2):

> You are a distracted social media user typing quickly while scrolling. Read the source text: `<TWEET>`. Rewrite the same information as a spontaneous Twitter/X post that feels messy, reactive, and slightly rushed.
>
> CRITICAL RULES:
>
> 1. Meaning Preservation: Keep the exact core message of the original tweet, but restructure how the idea is expressed. Do not simply paraphrase word-by-word.
>
> 2. Human Imperfections: Slightly awkward phrasing, inconsistent capitalization, or minor grammar slips are allowed. Avoid polished AI-style structure.
>
> 3. Formatting: Output ONLY the tweet text. No quotes, explanations, labels, or additional commentary.
>
> 4. Length: Keep the response approximately the same length as the original tweet. Do not expand the idea.
>
> 5. Tokens: The original text uses placeholders like <USER> and <URL>. You MUST use these exact placeholders instead of @ mentions or links. For hashtags, you must format them as <HASHTAG> word (e.g., instead of #breakingnews, write <HASHTAG> breakingnews). Do not use the actual # or @ symbols anywhere in your response.
>
> 6. No Emojis: Absolutely no emojis or special symbols.

These templates were refined through reverse prompt engineering, where the candidate prompts were tested, and the results were analysed for style and constraint adherence. The prompt was iteratively adjusted until the generated tweets would reliably match the intended persona while respecting all constraints.

Four instruction-tuned LLMs were used to generate the machine-generated dataset: Grok-4.1 (`grok-4-1-fast-non-reasoning`) [95], Llama 3.3 (`Llama-3.3-70B-Instruct`) [83], GPT-4o (`gpt-4o`) [96], and DeepSeek-R1 (`DeepSeek-R1`) [97]. These models were selected for stylistic diversity: GPT-4o and Llama-3.3 provided a reliable but slightly different instruction-following baseline; Grok-4.1, being more filtered, tended to produce more authentic misinformation outputs; DeepSeek-R1 followed its chain-of-thought reasoning which often generated more novel and coherent structures. Each prompt, for each of the 1,296 human-written tweets, was assigned to a single model in a fixed repeating order (Grok-4.1 → Llama-3.3 → GPT-4o → DeepSeek-R1), so that each model had, roughly, an even distribution of the total prompts. This preserved the 1:1 human to machine ratio while ensuring that no single model's style dominated the machine-generated set. Generation was carried out via the Azure inference endpoint with a temperature of 0.3 across all models. Standard models had a max token limit of 400, while DeepSeek-R1 had 2,000 to support its deep-reasoning responses. Due to the conflict of spreading misinformation and sophisticated LLM guardrails, certain generation attempts produced invalid results. A fallback mechanism was then added: if the model failed after repeated retries, another model from the remaining set was used. Upon reviewing the machine-generated samples for residual AI artefacts, claim drift, or redundancy, 16 data points were discarded along with their human-written counterparts, leaving **1,280 human-machine tweet pairs**.

5.1.3. Data augmentation and preprocessing

Back-translation augmentation. In order to support attacks of type B2, back-translation was applied to 25% of the combined human- and machine-generated training dataset by selecting every fourth sample and passing it through one

of the language routes shown in Table 8. This subset was chosen to introduce lexical and syntactic variation without allowing synthetic back-translated samples to dominate the original training distribution. Selecting every fourth sample also helped preserve coverage across the dataset and maintain topic alignment between corresponding human-machine examples. All translations were performed using the Microsoft Azure Translator Text API v3.0,[12] which applies neural machine translation (NMT) by default. Table 8 displays the 14 back-translation routes used for augmentation: nine single language routes and five multi-path routes. Single language routes introduced minor lexical and syntactical variation, while multi-path produces stronger perturbations by translating through multiple intermediate languages before returning to English. The resulting back-translated samples were concatenated to the human-machine training dataset.

Table 8
Back-translation routes used for augmentation. All routes were executed using Microsoft Azure Translator Text API v3.0 with default neural machine translation.

Route Type	Intermediate Language(s)	Translation Route
Single	German (DE)	en → de → en
Single	French (FR)	en → fr → en
Single	Spanish (ES)	en → es → en
Single	Arabic (AR)	en → ar → en
Single	Russian (RU)	en → ru → en
Single	Japanese (JA)	en → ja → en
Single	Chinese (ZH)	en → zh-Hans → en
Single	Hindi (HI)	en → hi → en
Single	Turkish (TR)	en → tr → en
Multi-path	German + French	en → de → fr → en
Multi-path	Arabic + French	en → ar → fr → en
Multi-path	Japanese + Chinese	en → ja → zh-Hans → en
Multi-path	Russian + German	en → ru → de → en
Multi-path	Hindi + Arabic	en → hi → ar → en

Paraphrased machine-generated data for DASS. In addition to the human tweet (T_H) and its machine-generated counterpart (T_L), the triplet network with DASS required a paraphrased variant of the machine-generated tweet (T_P). To generate the paraphrased set, separate prompts were used. In total, five new prompt templates were engineered with a tweet injection token (Appendix C). Each template introduced a distinct persona or tonal shift while maintaining the same constraints as the original machine-generated text. Although the same four LLMs were used, the assignment order was rotated (GPT-4o → Llama-3.3→ Grok-4.1 → DeepSeek-R1). Crucially, since the rotation for paraphrasing was offset from the original generation assignment, the model selected for paraphrasing was always different from the model used for the original machine-generate set. This forced the paraphrase to come from a different LLM family or instruction-tuning style, minimising the risk of over-fitting to model-specific artefacts. For this experiment, the generation temperature was 0.85 (higher than the 0.3 from the original set) to encourage greater lexical and syntactic diversity. After the same manual quality review, all 1,280 paraphrased machine-generate posts were selected for the DASS paraphrased set.

Sanitisation. To standardise the training data and minimise the influence of non-semantic noise, a preprocessing script was used to sanitise the text. This script removed emojis, redundant whitespace and normalised URLs and X handles. Emojis were removed because their distribution was inconsistent across the machine-generated samples, where some generated posts contained many, and others contained none. This mitigated the risk of introducing an imbalanced shortcut feature that the model could associate with a particular class. Emojis are also context-dependent, as the same emoji may convey different meanings depending on the surrounding text, therefore, removing them is expected to reduce the risk of the model relying on emoji presence rather than text-based authorship patterns.

[12] https://learn.microsoft.com/en-us/azure/ai-services/translator/

To reduce the effect of homoglyph-style perturbations, Cyrillic, Greek and accented characters were converted to their closest ASCII equivalent. This mitigates attacks where characters that resemble standard English letters are inserted to alter tokenisation, despite it actually preserving visual readability. User handles and URLs were also normalised to fixed tokens: <USER> and <URL>, respectively (e.g. *"hello @john you should visit https://google.com" → "hello <USER> you should visit <URL>"*). This preserves the information of the presence of a user mentioned or URL, while preventing the model from overfitting to specific usernames, domains or links. Since the dataset was constructed from both human and machine data sources, this standardisation helped make the input format more consistent and supported better generalisation to unseen data. Hashtags were treated differently because they may contain semantic information. Therefore, instead of removing/replacing the content, each hashtag was split into a marker token and its textual content (e.g. *"england is going to win the world cup #threelions #bottlers" → "england is going to win the world cup <HASHTAG> threelions <HASHTAG> bottlers"*). This preserves both the presence of a hashtag and the meaning carried by its content. For example:

Finally, repeated whitespace and redundant spacing were removed as they were by-products of preprocessing, user typing behaviour, machine-generation errors or adversarial perturbations. Since excess white space doesn't contribute to semantic meaning, removing it helped standardise the input without discarding relevant content.

Data Split. The final `builders` dataset (bld_data) was split into training, validation and test sets using a 75:15:10 ratio as seen in Table 2. The test set was used only for static benchmark evaluation and it is worth emphasising that it has no overlap with the `breakers`'s evaluation set.

Independence from `breakers`. It is worth emphasise that all datasets were created entirely independently from the `breakers` to avoid data leakage. The `builders` chose prompts, LLMs, and generation parameters without any knowledge of which models or attack techniques the `breakers` would subsequently use. Although the LLaMA 3.1 (8B) instruct LLM was also used by the `breakers`, the `builders`' machine-generated set relied on a significantly larger 70B variant. No samples produced by the `breakers` were ever introduced into the `builders` sets. By following a strict zero-knowledge boundary, both teams were able to use a unique data pipeline to prevent any training data leakage. In addition, both the main machine-generated set and the DASS paraphrased set were produced using fixed prompts and models outside of the `breakers`' adversarial transformation pipeline.

5.2. Model Architectures

The Builders' framework explored three model architectures, comprising a baseline supervised approach and two contrastive learning variants. The ***baseline detector*** employs the `e5-small-LoRA` model[13] fine-tuned for AI-generated text detection task using LoRA [98], a parameter efficient fine-tuning technique. Its choice was motivated by its strong performance on the RAID benchmark leaderboard[14] (ranked 5th on Reddit data when considering only open-source models), model availability at HuggingFace and its relatively low computational requirements for fine-tuning.

As discussed in Section 2.2.1, ***contrastive learning*** was previously shown to be effective for binary machine-generated text detection task. To explore its effectiveness for our task, we applied siamese and triplet networks to the baseline model.

Siamese networks. We hypothesise that building a siamese architecture model would combat aggressive paraphrasing attacks, as it can distinguish stylistic characteristics between human- and machine-generated texts rather than relying on surface-level tokens and signals. Such a pairwise architecture can be particularly successful in adversarial settings as it can be taught to associate a machine-generated text with its adversarially modified variants (e.g. the insertion of white spaces or back-translated texts), which can help the network to filter out noise and to recognise and classify the sample as if it was the original.

In contrastive learning, randomly chosen negatives can quickly become inefficient as the model learns to separate classes with a wide margin [77]. Once a negative sample is pushed beyond the established margin, its contribution towards the loss function drops to zero, stalling the contrastive model's optimisation process. For a contrastive model to be effective, the loss function must maintain a useful gradient. Therefore, the learning model should actively select samples that are close to the anchor in the embedding space but belong to the opposite class (i.e. hard negatives) in order to fall within the margin boundary and force the model to refine its vector representation. This is crucially demonstrated

[13] `https://huggingface.co/MayZhou/e5-small-lora-ai-generated-detector`
[14] `https://raid-bench.xyz/leaderboard`

in the Valla benchmark [99] which shows that by using hard negative mining to enable verification models, they can become a highly competitive alternative to attribution methods.

1:1 pairing hypothesis. Since each machine-generated post in bld_data was generated from the corresponding human-written misinformation post, the dataset contains naturally aligned human-machine pairs that preserve the same underlying topic and claim. This setup reduces the risk of the Siamese model learning topic-specific shortcuts, because each comparison is made between posts with closely matched content. The intended effect was to make the model distinguish human-written from machine-generated text using authorship-related features, such as phrasing, syntax, and stylistic structure, rather than broad vocabulary differences. Under this hypothesis, 1:1 pairing offers a controlled contrastive baseline before introducing more complex hard negative mining strategies. This pairing strategy was used for the Siamese (1:1) model.

TF-IDF pairing hypothesis. While strict 1:1 pairing preserves topical alignment, the negative samples created for the Siamese model may remain easily separable if the machine-generated tweets significantly differ from the human-written tweets, causing the model to learn broad stylistic and vocabulary dependent differences. Therefore, we additionally explore a more effective hard negative mining strategy through similarity-based retrieval. It enforces topical proximity between the anchor and the negative. If an anchor and a negative share the same topic, the network is forced to ignore broad vocabulary shortcuts to focus strictly on stylistic artefacts. Term Frequency-Inverse Document Frequency (TF-IDF) provides a robust foundation to enforce topical proximity. By using TF-IDF scores to isolate term importance in an Unsupervised Hard Negative Augmentation (UNA) system, which generates synthetic hard negatives through salient word perturbation, the performance on semantic textual similarity tasks was improved [100]. Since TF-IDF naturally captures topic similarity, it has been shown to retrieve existing texts that form challenging negatives. This strategy is relevant for short-form disinformation, where human and machine-generated tweets on the same narrative can be difficult to distinguish by the content alone and require models to learn the underlying stylistic cues. For our task, a TF-IDF-based hard negative would create more challenging contrastive comparisons of pairs, as it would pair semantically similar posts from the opposite class. As the TF-IDF pairing method finds pairs with overlapping vocabulary and narrative framing, the model is forced to rely less on lexical shortcuts and instead learn deeper stylistic patterns associated with machine-generate or human-written texts. In practice, TF-IDF pairing ensures that the model focuses on the underlying style instead of the subject by grouping similar topics closely in the vector space, leaving minimal topical differences for the learning model to exploit. When the pairs are created, the dataset is grouped into topic buckets using TF-IDF and k-nearest neighbour (k-NN) selection: for each sample, the top-20 (k=20) most similar tweets are identified via cosine similarity of TF-IDF vectors, filtering out common stop words (e.g. "the", "and") to highlight relevant keywords (e.g. "Prince") that define the core topic of a tweet. When constructing input pairs, both samples are taken from the same topic bucket to force the model to compare human and machine-generated texts on the same topic. This teaches the network underlying stylistic and syntactical signatures of LLMs instead of generic topic differences. This pairing strategy was used for the Siamese (TF-IDF) models.

Triplet networks. As described in Section 2.2.1, triplet networks extend the contrastive pairing architecture of siamese networks by evaluating three samples simultaneously: an anchor, a positive sample from the same class as the anchor, and a negative sample from the opposing class (Fig. 5). By utilising a *triplet margin loss*, the architecture forces the embedding space to divide into two distinct classes by pulling the anchor to the positive sample and pushing the negative beyond a fixed margin, forming a clean cluster separation. In the context of machine-generated text detection, as the underlying space actively encodes the embeddings of source identity instead of surface string similarity, the model can more reliably identify both original and adversarial paraphrased machine-generated text. In our experiments, we explore two types of sampling strategies for Triplet networks.

Triplet with TF-IDF. In this setting, the triplet is trained using the same TF-IDF k-NN pairing engine introduced, previously in the TF-IDF pairing hypothesis. For every anchor sample, the top-20 most similar samples are identified via cosine similarity, where a hard positive is taken from the tweets sharing the anchor's authorship label and a hard negative from those with the opposite label. By enforcing a tight semantic cluster, the margin loss can only be reduced by learning the underlying stylistic differences, effectively neutralising vocabulary bias. The idea is that by using the same pairing engine with a triplet network (three-way contrast), the model should yield a more discriminative embedding space. This pairing strategy was used for the Triplet (TF-IDF) models.

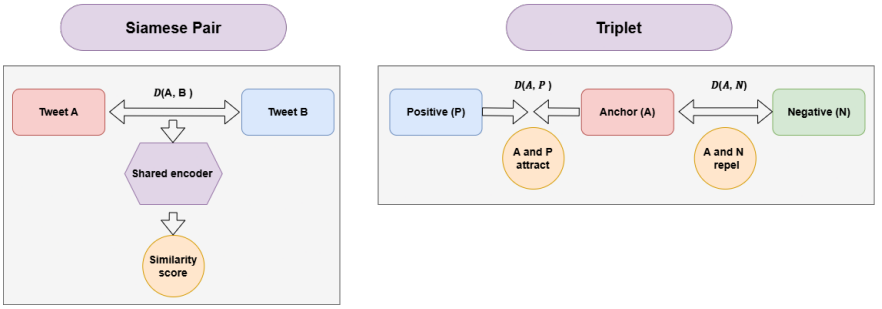


Figure 5: Siamese vs. triplet pairing.

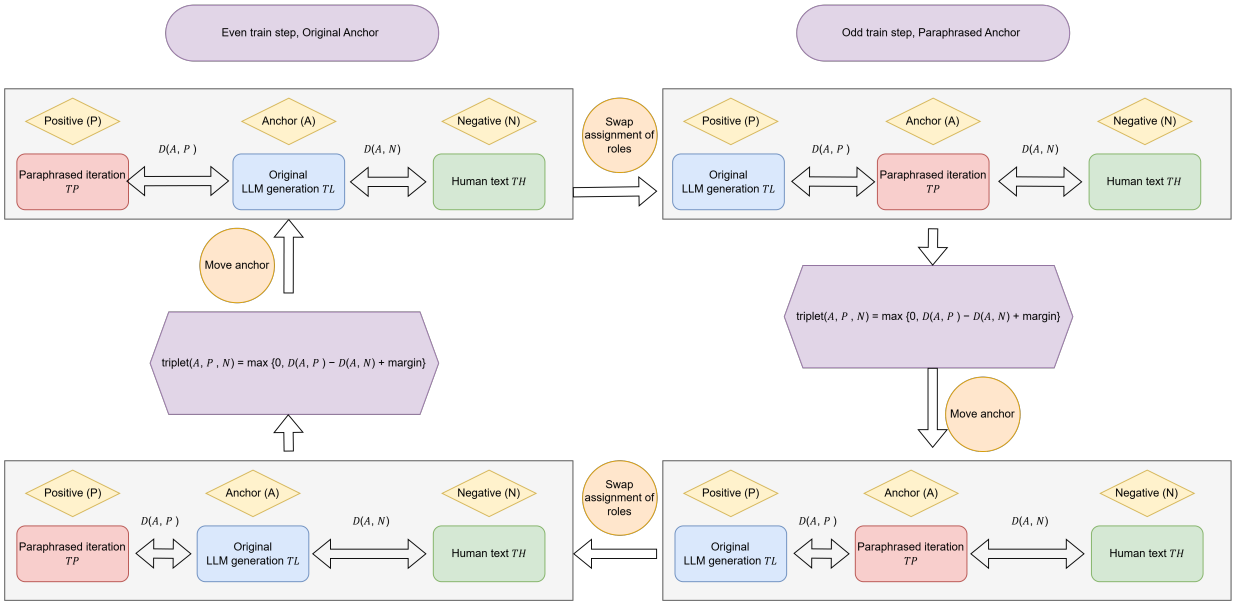


Figure 6: DASS pairing engine.

Triplet with DASS semantic pairing. A more targeted approach adopts the dynamic anchor switching strategy (DASS), a pairing strategy introduced by GravText [79] to make detectors robust against paraphrasing. In standard triplet learning, the anchor is fixed, whilst DASS alternates the role of the anchor depending on the detection task (Fig. 6). Given a triplet of samples: a original machine-generated text T_L, its paraphrased iteration T_P, and a human text T_H on the same topic, the standard triplet loss is:

$$\mathcal{L}_{\text{triplet}}(A, P, N) = \max\{0, D(A, P) - D(A, N) + \text{margin}\} \tag{4}$$

where, A = anchor = T_L, P = positive = T_P, N = negative = T_H, D = Euclidean Distance between embeddings. In standard training, the anchor is fixed, but DASS switches the assignment:

- **Paraphrased Anchor (PA) (human or machine-generated sample).** The loss equation becomes:

$$\mathcal{L}^{PA}_{\text{triplet}} = \max\left\{0, D(T_P, T_L) - D(T_P, T_H) + \text{margin}\right\} \tag{5}$$

The original and the paraphrased sample are pulled together while separating from the human text space.

- **Original Anchor (OA) (human or paraphrased machine-generated sample).** The loss equation becomes:

$$\mathcal{L}_{\text{triplet}}^{OA} = \max\left\{0, D(T_L, T_P) - D(T_L, T_H) + \text{margin}\right\} \tag{6}$$

  The model learns to recognise the paraphrased sample as still machine-generated by anchoring it to the original output space.

LLM-generated outputs (original or paraphrased) should cluster tightly together and remain separated from the human-generated space by alternating between the two loss functions during training. In the case of tweet misinformation, T_L is a machine-generated tweet on a given narrative, T_P is its paraphrased counterpart (or a semantically close example), and T_H is a human tweet on the same topic. This adaptation directly follows the DASS pairing method, but disregards the gravitational factor introduced the full GravText framework. The implemented model is called Triplet (DASS).

5.3. Experimental Setup

Table 9 summarises the hyper-parameters for all `builders`' models.

Table 9
`Builders`' hyper-parameters per model.

Hyperparameter	Baseline	Baseline + Back-translation	Siamese (1:1)	Siamese (TF-IDF)	Triplet (TF-IDF)	Triplet (DASS)
Learning rate	2.286×10^{-5}	3.071×10^{-5}	2×10^{-5}	2×10^{-5}	2×10^{-5}	2×10^{-5}
Weight decay	0.2889	0.2483	0.01	0.01	0.01	0.01
Epochs	6	4	3	3	3	3
Batch size	16	8	8	8	8	8
Loss Function	CE	CE	BCE	BCE	Triplet (L2, m=1)	Triplet (L2, m=1)
Pairs/Triplets (epoch)	-	-	20,000	20,000	20,000	15,000
TF-IDF top-k	-	-	-	20	20	-

Common Configuration. All models used the HuggingFace's `MayZhou/e5-small-lora-ai-generated-detector` checkpoint and its tokeniser. Each sample was passed through the same sanitisation pipeline described in Section 5.1.3 and truncated to a maximum of 256 tokens to reflect the nature of the X short-form domain. Models were trained using the HuggingFace's Trainer API[15] that utilises an AdamW optimiser and a linear learning-rate schedule. Each model had a fixed random seed of 42 across Python, NumPy, and PyTorch to ensure reproducibility. Every training run and data preprocessing was executed using a single NVIDIA GeForce RTX 4060 Laptop GPU, AMD Ryzen 7 7840HS CPU, and 32GB of system RAM (5600 MT/s).

Data Partitioning. As mentioned in Section 5.1.3, the bld_data was split into 75% training, 15% validation, and 10% test set; using a fixed random seed of 42, creating interleaved topic indices and ensuring no data leakage. Final sample sizes varied according to the model and its pairing engine. The standard baseline classifier, both Siamese models (1:1 and TF-IDF) and the Triplet (TF-IDF) used the raw bld_data, with 2,560 samples (1,280 human-written and 1,280 machine-generated). For Triple(DASS), an additional paraphrased machine-generated set based on the bld_data's machine-generated set (Section 5.1.3) was introduced, yielding a total of 3,840 samples (1,280 human-written, 1,280 machine-generated, and 1,280 paraphrases of the machine-generated set). Notably, an experiment in iteration 2 augmented a subset of the training data using back-translation, resulting in a larger set of 3,200 samples (1,600 human-written, 1,600 machine-generated). However, since the experiment showed a reduction in performance, the follow up models were trained on the non-augmented dataset.

[15] https://huggingface.co/docs/transformers/en/main_classes/trainer

Evaluation. During training, the HuggingFace's Trainer API selected the baseline model's best checkpoint based on the highest validation-set $F1$-score achieved at the end of each epoch. For the Siamese models, validation was performed via pair-wise accuracy and $F1$ on the validation-set pairs, and the checkpoint with the highest pair-wise $F1$ was selected. The Triplet model needed a class prototype computed from the training-set embeddings and the validation samples after each epoch, as it cannot produce a class prediction from the triplet loss alone. The validation samples were assigned to the nearest prototype, using the prototype's $F1$ as the selection metric.

Finally, for internal model assessment, all models were evaluated on bld_data's test set, which was not used during training or hyperparameter selection.[16] Evaluation metrics where accuracy and $F1$. The baseline classifier directly outputs a label, while the contrastive models used the same prototype classification method applied to the test set: each test sample was embedded, compared via cosine similarity to the training set prototypes, and assigned to the closer class.

6. Results

6.1. Impact of adversarial transformations in the detector's performance (RQ1)

To answer RQ1, we use the bkr_data set, with 125 HGO-MGO pairs. The detector is also fixed: we use the baseline `e5-small-LoRa`, which showed 100% accuracy on the bkr_data set. This decision was made to ensure that only the adversarial techniques were being assessed and no changes made by the `builders` could influence the results. In addition, only the MGO set is considered, as the HGO dataset is only used as an anchor to assess the performance of `builders`' models.

In total, there were five iterations (Table 2). Iterations 1 and 2 are not reported as separate result stages because they were exploratory development rounds with a small number of generated posts. Due to this, outputs from iterations 1 (n = 12,000) and 2 (n = 90,000) were incorporated into the iteration 3's analysis, meaning that iteration 3 contains `breakers`' results across iterations 1–3 (n = 180,000). This provides a more reliable basis for comparing early `breakers`' techniques against the later iterations 4 (n = 180,000) and 5 (n = 180,000) results, which were comparatively larger. The combinations of techniques were decided by generating a power set of the technique codes for each iteration, as shown in Table 2. This was then represented as a 2^n binary truth table, where n is the number of techniques for the iteration, ensuring an exhaustive exploration of the combinatorial space (see Appendix B for an example of iteration 2 mapping).

6.1.1. Overall `Breakers`' Performance

Across all 1,440 `breakers`' configurations in each iteration, average detector accuracy decreased from 71.1% in iteration 3 to 60.2% in iteration 4 and 57.6% in iteration 5. Consequently, the average LFR increased from 28.9% to 39.8% and then to 42.4%, respectively. This indicates that the `breakers`' techniques became more effective over time, both in terms of average attack effectiveness to bypass detection and the broader evolution of attack combinations. However, it is observed that the improvement from iteration 4 to iteration 5 was smaller than the improvement from iteration 3 to iteration 4.

Fig. 7 compares the ranked LFR distributions for all $1,440$ `breakers`' configurations across iterations 3 to 5. The figure supports the increase in `breakers`' effectiveness over time, showing that LFR values shifted upward from iteration 3 to iteration 5. In iteration 3, the median LFR was 23.0%, while the mean was 28.9%, which shows the distribution contained a strong upper tail. This means a smaller subset of configurations achieved high flip rates, while many configurations remained weaker. In iteration 4, both the median and mean increased to 36.4% and 39.8%, respectively, indicating a broader upward shift across the distribution rather than improvements being limited to isolated high-performing combinations. This trend continued into iteration 5, where the median increased to 41.6% and the mean to 42.4%. The narrowing gap between the mean and median across the iterations suggests that the `breakers`' techinques performance became more evenly distributed, with a larger proportion of attack combinations achieving moderate to high LFR. Fig. 7 also shows that across the three iterations, most individual techniques remained below the overall mean and median, which suggests that individual attacks were generally less effective than combined ones. Techniques such as D2, D3 and D4 suggest that the evolved prompt-based evasion techniques became more competitive as the `breakers`' process matured. However, the strongest overall LFR values came from combined rather than individual techniques. Section 6.1.2 and Section 6.1.3 discusses these findings in more detail.

[16] The baseline classifier was run with a 10-trial Optuna hyper-parameter search

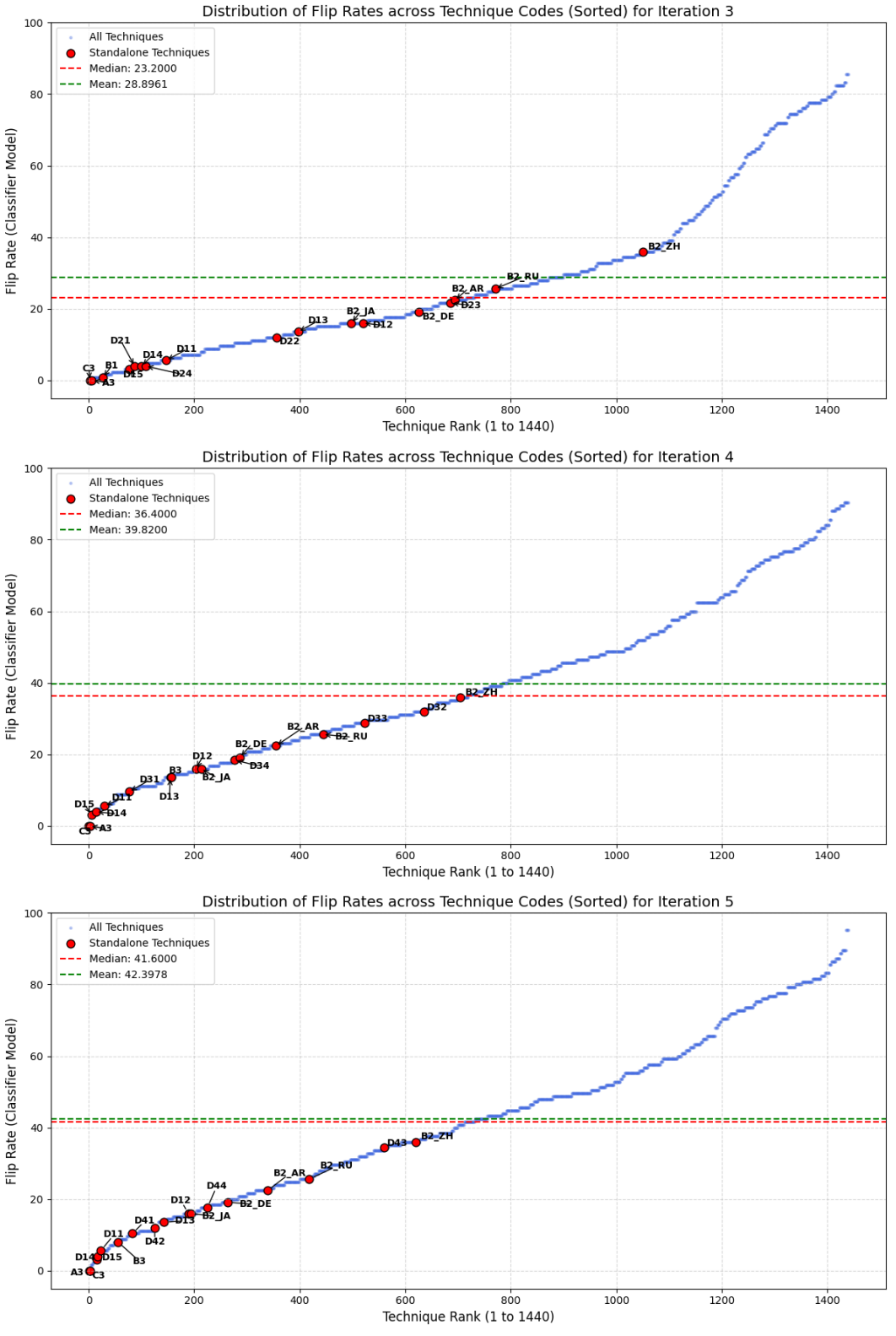


Figure 7: Sorted LFR distributions across technique codes for iterations 3 to 5 using the baseline model. The upward shift in mean and median flip rates shows that `breakers`' techniques in later iterations produced stronger and more consistent adversarial effectiveness across the techniques set.

As Table 10 shows, the strongest breakers' configuration in iteration 3 reached an LFR of 85.6%, reducing detector accuracy to 14.4%. Iteration 4's strongest configuration reached an LFR of 90.4%, and in iteration 5, the strongest configuration reached 95.2%, leaving the detector with only 4.8% accuracy on that attack configuration. The repeated presence of B2_AR in the max-LFR combination also suggests that Arabic back-translation was a strong technique within the highest-performing combined attacks.

Table 10
Breakers' attack combinations with highest LFR in iterations 3 to 5.

Iteration (i)	Mean Accuracy (%)	Mean LFR (%)	Max LFR (%)	Max LFR Attack Configuration
3	71.1	28.9	85.6	B1_D15_D23_B2_AR
4	60.2	39.8	90.4	B3_D13_D33_B2_AR
5	57.6	42.4	95.2	B3_D13_D43_B2_AR

6.1.2. Best Performing Breakers' Techniques

This section focuses on the attack effectiveness (measured in terms of LFR), while the semantic validity is assessed in Section 6.2. Across iterations 3 to 5, the strongest individual back-translations technique was B2_ZH, achieving an LFR of 36%. However, as shown in Section 6.1.3, B2_AR performed better when used in combination with other breakers' attack techniques. This distinction indicates that standalone and combined performances may differ substantially.

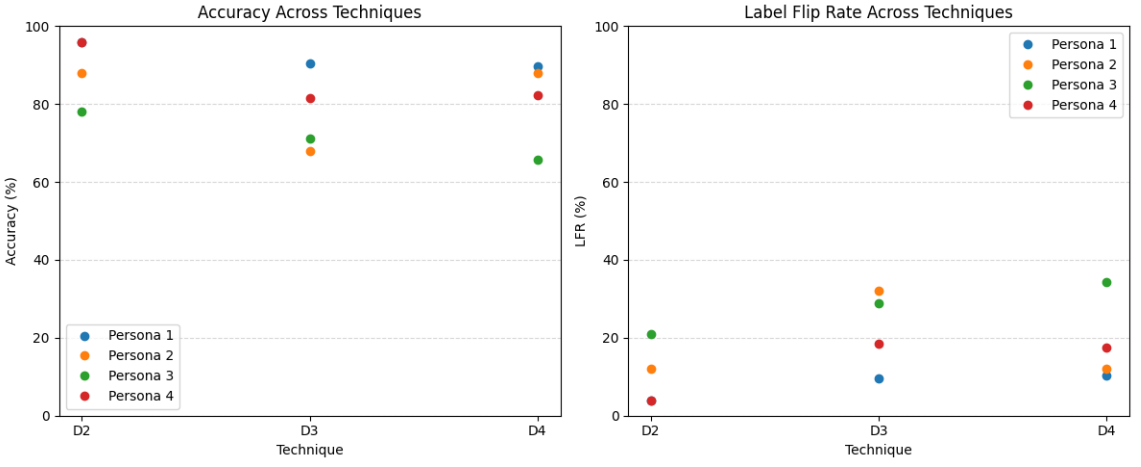


Figure 8: Comparison of standalone persona-based techniques across D2, D3 and D4 which shows how evolved persona prompts increased LFR while reducing detector accuracy.

The D2, D3 and D4 technique families represent successive evolutions of persona-based breakers' strategy. As shown in Fig. 8, the transition from D2 to D3 produced the highest increase in detector evasion as the strongest D2 variant achieved an LFR of 21.6% whilst the strongest D3 variant reached 32.0%. This improvement is linked to the change in prompt structure shown in Table 3. The D2 variants use a simpler persona prompt, e.g.: "You are a [PERSONA]", whereas D3 expands the persona with more demographic details such as age, ethnicity, gender, location and occupation. An illustrative example of this is the *college student* persona, where D22 was the simple version and D32 used "a 20-year old Asian female college student" profile. D32 achieved 32% LFR, outperforming the strongest D2 variant and suggesting that the added information made the persona-based rewrite more effective. A similar pattern is visible in the later D4 variants, which uses the same structure as D3 but uses an indirect persona, e.g.: "Think like a [DEVELOPED PERSONA]". For example, the *brain-rot 10-year old persona* achieved 21.6% LFR under the simpler D23 format, while the evolved D43 variant achieved 34.4% LFR.

Combined breakers' configurations. The strongest combined breakers' configurations across iterations 3 to 5 are shown in Table 10. These were analysed by calculating LFR for each atomic technique code, then grouping the atomic codes into their families. This avoids treating each atomic technique as a separate value and enables the comparison of the technique family combinations more fairly.

Iteration 3. Table 11 shows iteration 3's results based on the technique families. The strongest grouped configuration type was D2_B2 with a mean LFR of 38.48% and a median of 33.6%. This combination performed better than the individual techniques: D2 and B2, with mean $LFRs$ of 10.4% and 23.4%, respectively. However, the full B1_D1_D2_B2 chain did not have the strongest mean performance, even though it produced the highest individual LFR at 85.6% (reduced detector accuracy to 14.4%). This suggests that adding B1 and D1 could create strong individual attacks, but could not consistently improve the average performance across all configurations.

Table 11
Iteration 3 technique family grouping based mean, median and max LFR percentage.

Combination Type	n Configs	Mean LFR (%)	Median LFR (%)	Max LFR (%)
D2_B2	80	38.48	33.6	82.4
D1_D2_B2	400	36.55	29.6	83.2
B1_D2_B2	80	34.09	26.8	82.4
B1_D1_D2_B2	400	33.46	25.6	85.6
D1_B2	100	31.01	29.6	62.4
B2	20	23.84	22.4	36
D2	16	10.4	8	21.6
B1	4	0.8	0.8	0.8

Iteration 4. The iteration 4 attack techniques showed more effectiveness, as can be seen in Table 12. The strongest grouped configuration was D3_B2 with a mean LFR of 52.72% and a median LFR of 48.4%. This combination was more effective than the individual counterparts D3 (with a mean LFR of 22.2%) and B2 (which remained at 23.84% LFR). The larger combinations involving B3 and D1 were also strong, although they did not outperform the mean LFR of D3_B2. For example, B3_D1_D3_B2 achieved the highest maximum LFR of 90.4%, but its mean LFR was 48.53%. This suggests that B3 and D1 help create high-performing configurations, but they are not the main drivers of consistent detector evasion.

Table 12
Iteration 4 technique family grouping based mean, median and max LFR percentage.

Combination Type	n Configs	Mean LFR (%)	Median LFR (%)	Max LFR (%)
D3_B2	80	52.72	48.4	89.6
B3_D3_B2	80	50.72	45.6	88.8
B3_D1_D3_B2	400	48.53	47.6	90.4
D1_D3_B2	400	47.39	45.6	90.4
D1_B2	100	31.01	29.6	62.4
B3_D1_B2	100	30.72	31.2	57.6
B3_B2	20	24.96	24	32.8
B2	20	23.84	22.4	36
D3	16	22.2	23.6	32
B3	4	13.6	13.6	13.6

Iteration 5 produced the strongest overall combined attack as seen in Table 13. The highest grouped mean LFR came from B3_D4_B2, with a mean LFR of 58.44%, followed by D4_B2 at 56.88%. Both of the strongest configurations contained D4 and B2, suggesting that this was one of the best pairings of techniques to evade detector classification of MGO. The grouped family technique codes containing D4 and B2, such as B3_D1_D4_B2, produced the highest maximum LFR, but had means below those for B3_D4_B2 and D4_B2, respectively.

Table 13
Iteration 5 technique family grouping based on mean, median and max LFR percentage.

Combination Type	n Configs	Mean LFR (%)	Median LFR (%)	Max LFR (%)
B3_D4_B2	80	58.44	56.8	87.2
D4_B2	80	56.88	56	89.6
D1_D4_B2	400	51.95	49.6	89.6
B3_D1_D4_B2	400	51.4	48.4	95.2
D1_B2	100	31.01	29.6	62.4
B3_D1_B2	100	27.9	24.8	59.2
B2	20	23.84	22.4	36
B3_D4	16	21.2	21.2	28
D4	16	18.6	14.8	34.4
B3	4	8	8	8

6.1.3. Ablation Study

The ablation study compares the distribution of LFR when each technique family is absent versus present in a `breakers`' configuration. This helps identify whether a technique family consistently shifts detector behaviour or if it only assists in edge cases. Fig. 9 shows this comparison for all technique families.

B2 Analysis. Fig. 9b ablation results show that B2 (back-translation) was the most reliable contributor to detector evasion. When B2 is absent from the violin graphs, the LFR distribution is concentrated at lower values, with most configurations remaining below 30% LFR. When it is present, the distribution shifts upwards, with a wider spread showing many configurations reaching between 50–90% LFR. This indicates that B2 is the strongest and most reliable contributor to detector evasion across the `breakers`' techniques.

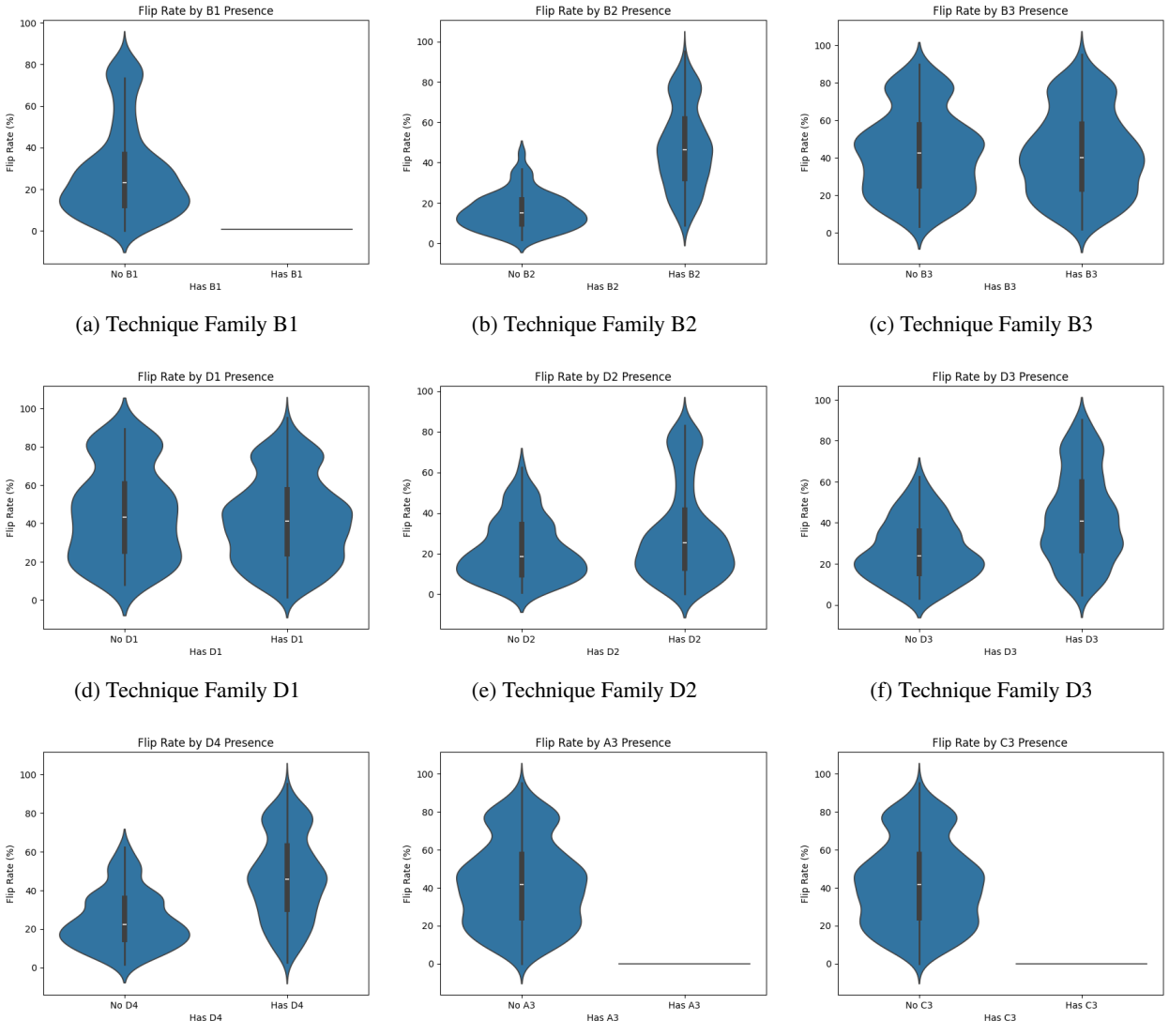
B1 and B3 Analysis. B1 (prompt-based paraphrasing) shows no useful contributions to LFR, with distribution around 1% LFR, as seen in Fig. 9a. This shows that the original B1 paraphrasing method was not effective as an independent attack. However, B3 (a revised version of B1 to enforce lexical and syntactic variation) behaves differently (Fig. 9c). The B3's present or absence distributions are closer together, containing a broad spread of flip rates. This suggests that B3 is not as dominant as B2, but it does not collapse to near-zero performance like B1. B3's presence alone does not contribute to LFR, but it features among high performing configurations.

D1 Analysis. Similar to B3, the D1 (tone-based LLM prompt rewriting) violin plot shows a mixed effect, with a large overlap between configurations where D1 is present or absent (Fig. 9d). The D1-present distribution doesn't show a clear upward shift and its overall distribution appears similar/slightly lower than the D1-absent group. This suggests that tone-based prompts transformations alone are not effective at evading detection. D1 helps to improve certain technique combinations, especially when applied together with back-translation and a strong persona variant, but the ablation distribution does not show D1 to be a primary contributor to increase LFR.

D2 Analysis. Fig. 9e shows D2 (persona-based rewriting) produces some upward shift when present but the effect is less stable than that of B2. The D2-present distribution includes higher flip-rate configurations and it also contains more configurations that contribute to LFR between the range of 5–35%. This shows the earlier persona based prompts were useful to achieve high LFR, but D2 is not consistent at evading detection.

D3 Analysis. D3-present (refined persona-based rewriting) shows a stronger effect (Fig. 9f) than D2 in terms of LFR, with the distribution shifting upward and more configurations reaching higher LFR values. This supports aforementioned interpretation that stronger D3 standalone variants exceeded the strongest D2 standalone variant. It suggests that the demographic-specific D3 persona prompts were more effective than the simpler D2 prompts.

D4 Analysis. The presence of D4 (further refined persona-based rewriting) (Fig. 9g) produces a noticeable upward shift in the distribution of flip rate, resulting in a substantially higher concentration of configurations with LFR values



(a) Technique Family B1 (b) Technique Family B2 (c) Technique Family B3

(d) Technique Family D1 (e) Technique Family D2 (f) Technique Family D3

(g) Technique Family D4 (h) Technique Family A3 (i) Technique Family C3

Figure 9: Flip-rate distributions comparing configurations where each technique family is present against configurations where it is absent.

exceeding 60%. This suggests that, in line with previous findings, D4 is the most effective persona-based technique family. However, the distribution is more spread, which suggests that it was not uniformly successful across all variants.

A3 and C3 Analysis. Results shows that A3 (rule-based whitespace editing) and C3 (emotion-based emoji injection) did affect the LFR values. Across iterations 3 to 5, the presence of A3 or C3 or C3_A3 did not change the grouped average mean nor maximum LFR listed in Table 10. This suggests that these techniques do not contribute to detector evasion. Fig. 9h and Fig. 9i supports this interpretation, since A3 and C3-present distributions are concentrated near 0% LFR and do not show the upward distributional LFR shift as observed for B2, D3 and D4.

Overall the ablation study shows that B2 was the strongest contributor to high LFR, with D3 and D4 being the strongest prompt-based contributors. D2 showed a weaker effect, but was useful in earlier iterations even though it had less consistency across distributions. By contrast, B1 and D1 did not show any reliable effects and B3 appeared to support other techniques rather than contributing directly as a standalone one. The rule-based perturbation techniques of C3 and A3 had no positive ablation effect, with their LFR distribution near 0%. However, these results only measure

detector evasion, so semantic preservation analysis is needed to determine whether the strongest breakers' techniques remained valid adversarial transformations. The following section presents the results of this investigation.

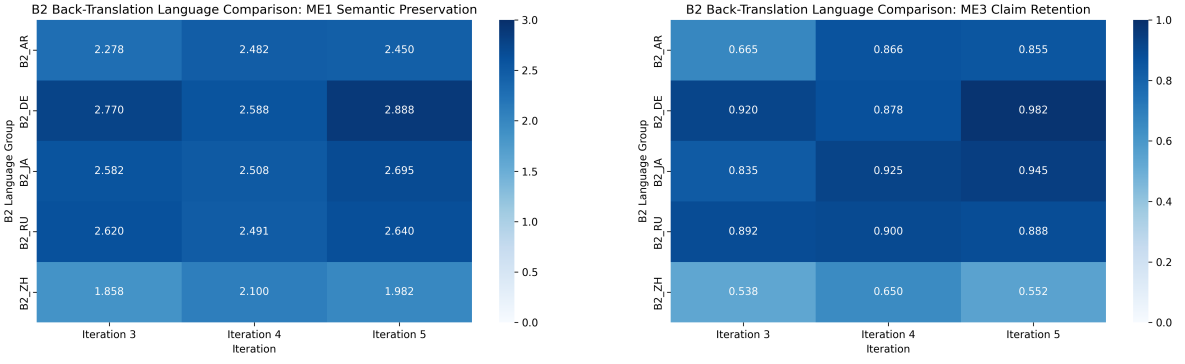
6.2. Semantic Preservation and Claim Retention (RQ2)

Manual evaluation was conducted using the metrics shown in Table 6 and the annotation guidelines in Appendix A. The results shows that the quality of the adversarial attack outputs generally improved from iteration 3 to 5. Table 14 illustrates that iteration 3 produced valid adversarial examples, however, the manual evaluation indicates these examples were less stable. Iteration 4 improves the scores across all criteria, particularly ME2 and ME3. Finally, iteration 5 achieved the highest semantic preservation score overall, although the improvements over iteration 4 was smaller. Between iteration 3 and 4, ME2 increased from 3.699 to 3.923, while ME3 increased from 0.797 to 0.866. This shows that, overall, the naturalness and fluency of the posts were maintained. Iteration 5 further improved ME1 from 2.517 to 2.585, indicating stronger semantic preservation, while ME2 and ME3 remained almost unchanged from iteration 4.

Table 14
The technique family codes manually analysed per iteration and its average scores per category across the iterations.

Iteration (i)	Technique Family Analysed	*n*-Posts	ME1 Average	ME2 Average	ME3 Average
3	D2, B2, D2_B2	290	2.456	3.699	0.797
4	D3, B3, D3_B2	250	2.517	3.923	0.866
5	D4, D4_B2	240	2.585	3.912	0.868

Technique Level Analysis. Across the B2 language variants, the most consistent pattern was B2_ZH (Chinese back-translation), demonstrating the weakest semantic preservation and claim-retention results. The heat map in Fig. 10 also confirms that the B2_ZH introduced the biggest meaning variation. Inversely, B2_DE (German back-translation) appeared to introduce enough linguistic variation to modify the text whilst preserving the original meaning and claim more reliably. B2_JA and B2_RU variants were generally stronger than B2_ZH, while B2_AR shows mixed results depending on the iteration and the technique combination used. This shows that back-translation is not a consistent technique and the effects depend heavily on the intermediate language used.



(a) ME1 scores across Iterations 3–5 for B2.

(b) ME3 scores across Iterations 3–5 for B2.

Figure 10: ME1 and ME3 scores across iterations 3 to 5 for the B2 technique family.

Techniques D1 to D4 shower stronger manual evaluation scores than B2 across iteration 3 to 5. In iteration 3, D2 achieved an average ME1 score of 2.672 and ME3 score of 0.966, while D2_B2 fell to to 2.369 and 0.749 respectively. This indicates that the standalone D2 variants preserved the original meaning and disinformation claim, but adding B2 to it introduced a noticeable loss in semantic preservation. In iteration 4, D3 achieved high scores across all manual evaluation criteria's with ME1 at 2.838, ME2 at 4.000 and ME3 at 0.962. Similar scores can be seen for B3 which attains ME1 of 2.900, ME2 of 4.000 and ME3 of 0.938. By contrast, D3_B2 reduced ME1 to 2.434 and ME3 to 0.844, showing that the addition of back-translation increased drift in semantic preservation. Iteration 5 produced the strongest results for ME1 and ME3 with D4 achieving ME1 of 2.855, ME2 of 3.990 and ME3 of 0.988, making it

the best standalone technique family for preserving the original claim whilst maintaining naturalness. Within the D4 variants, D42 (indirect persona prompt, with the *college student* persona) achieved the highest scores: ME1 of 2.980 and ME3 of 1. However, D4_B2 showed reduced semantic preservation and claim retention, falling to 2.531 for ME1 and 0.874 for ME3. Table 15 shows the differences between B2 as a combined versus standalone technique. The scores presented are normalised to make fair comparison between the different criteria.

Table 15
Comparing the effect of B2 added to standalone techniques on the manual evaluation scoring. The scoring used in this table has been normalised on a scale of 0–1 for better comparison, where ME1 is divided by 3, ME2 by 4, and ME3 by 1.

Comparison	Δ ME1	Δ ME2	Δ ME3
D2_B2 vs D2	-0.101	-0.040	-0.217
D3_B2 vs D3	-0.134	-0.024	-0.118
D4_B2 vs D4	-0.108	-0.023	-0.114

Of the 780 posts that were manually analysed, $n = 69$ (~8%) received an ME3 score of 0, meaning that these posts did not maintain their disinformation claim. 58% of these 69 posts flipped their label, but this was not a successful evasion since the original claim was lost. This highlights the need to investigate whether poorer semantic preservation and claim retention contribute to label flips.

Classification & Semantic Degradation Analysis. For manual evaluation, we used a controlled sample of posts, where each technique had a 50%-50% split between HGO and MGO of the classified posts selected. An analysis of the equal split was conducted to see if lower manual evaluation scores within the selected subset were associated with higher HGO classification or higher HGO probability. Manual evaluation revealed that HGO classified posts had a lower average semantic preservation and claim-retention scores than MGO-classified posts (Table 16). This pattern suggests there may be a modest association between posts classified as HGO and a weaker semantic preservation/claim retention. Naturalness was almost identical for the HGO-classified and MGO-classified posts, which suggests that the detector was not biased by the fluency of the input text. However, as the manually evaluated sample selection was balanced by design, with an equal number of HGO-classified and MGO-classified examples for each variant, the average difference between the HGO-classified and MGO-classified posts was relatively small. This means that the results should not be interpreted as proof that semantic degradation causes HGO classification. Instead, they suggest that some HGO classifications may be associated with weaker semantic preservation or claim retention.

Table 16
Semantic degradation across the averaged manual evaluation metrics for MGO-classified and HGO-classified posts.

Metric	MGO-Classified	HGO-Classified	Difference
ME1	2.576	2.454	-0.122
ME2	3.840	3.832	-0.008
ME3	0.866	0.816	-0.050

Semantic Preservation Pipeline. Automatic evaluation metrics were assessed in terms of their correlation with the manually evaluated posts scores. Spearman's ρ rank correlation coefficient was employed given the ordinal nature of the manual evaluation scores. Fig. 11 shows that the semantic preservation metrics more strongly align with ME1 and ME3 than ME2. This is expected, as the pipeline's primary objective was to ensure semantic similarity and disinformation claim retention, rather than optimising for fluency and naturalness. E5-cosine similarity showed the strongest relationship with the manual evaluation score, with a Spearman's ρ of 0.489 for ME1 and 0.453 for ME3. BLEURT also showed a moderate positive relationship with ME1 and ME3, scoring ρ =0.369 and ρ =0.357, respectively. The token-ratio metric showed a similar moderate relationship with ρ =0.347 for ME1 and ρ =0.356 for ME3. The strongest pipeline metric was E5-based cosine similarity, suggesting that embedding-based semantic similarity was the most useful substitute for human semantic judgements. Posts with a higher E5-cosine similarity were likely to receive stronger ME1 and ME3 scores. However, the relationship was moderate rather than strong, which means E5-cosine was useful as a initial signal of semantic preservation, however, human evaluation is still needed to confirm the results.

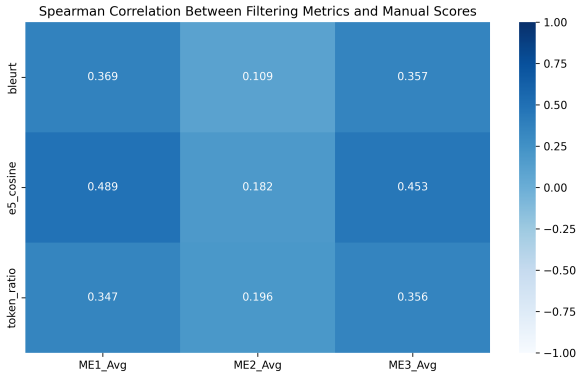


Figure 11: Spearman correlation heatmap between BLEURT, E5-cosine similarity, and token ratio against ME1, ME2, and ME3. Higher positive values indicate a stronger correlation between the semantic preservation metrics and the manual evaluation scores.

NLI labels were evaluated separately due to having a different scoring criterion. Fig. 12 shows the breakdown of ME1 and ME3 against the NLI labels after normalisation. As can be seen, out of the 780 manually evaluated posts, contradiction posts ($n = 47 \mid 6\%$) had an average ME1 score of 1.986 and ME3 score of 0.620, compared to ME1 scores above 2.540 and ME3 scores above 0.85 for entailment ($n = 205 \mid 26\%$) and neutral ($n = 528 \mid 68\%$) posts. This suggests NLI can be useful for identifying posts with worse semantic preservation scores, especially when the posts didn't contain the original disinformation claim. However, since NLI entailment and neutral labels produced similar scores, it is more useful at identifying posts with lower semantic/claim preservation than separating preserved output posts from acceptable neutral paraphrased posts. Therefore, this metric should be interpreted more as a failure rate detector rather than a nuanced quality assessment.

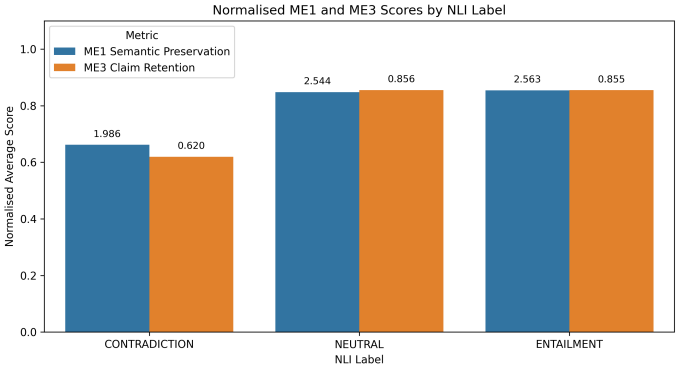


Figure 12: Normalised ME1 and ME3 manual evaluation scores grouped by NLI label. ME1 was normalised by dividing it by 3, while ME3 already follows a 0–1 scale. The scores were normalised to allow fairer comparison.

Overall, the semantic preservation pipeline showed moderate agreement with human evaluation scores, with the strongest evidence of alignment coming from E5-cosine similarity and NLI labelling. However, ME2 showed only weak relationships with the filtering metrics. This means that the filtering pipeline can be useful as an initial filter, however, manual evaluation may still be necessary to ensure output posts preserve the original meaning and disinformation claim.

6.3. Architectures for detecting machine-generated content (RQ3)

Table 17 presents the results of the `builders`' models evaluation. Greyed-out values represent models that were evaluated on a `breakers`' set that was produced prior to the model's iteration. The evaluation is done in the full `breakers`' set for each iteration, i.e. results are not divided per technique.

Iteration 3. At iteration 3 time, the most recent `builders`' models were the Siamese variations (1:1 and TF-IDF). Although theoretically contrastive learning should outperform baseline-transformer models, Siamese variants

underperformed significantly in comparison to the baseline model. The best Siamese model – Siamese (TF-IDF) – achieved 67.88% of $F1$ score, which is over 15% lower than the baseline model (83.11% of $F1$). It is worth mentioning that the Baseline(BT) model slightly harmed the baseline's performance (this happens throughout all iterations). Therefore, the use of data augmentation via back-translation was not carried forward to the contrastive learning models.

Iteration 4 showed the introduction of the Triplet(TF-IDF) model. This novel model, however, did not outperform the baseline and, in fact, was worse than the Siamese(TF-IDF). Iteration 4 marks a significant drop in performance from iteration 3: the baseline model had around 8% drop in terms of $F1$ score. This highlights the `breakers`' usage of more aggressive adversarial techniques as discussed in Section 6.1.

Iteration 5 was when the Triplet(DASS) model was introduced. Previous models (baseline and Siamese) showed a similar trend, while Triplet(DASS) outperformed the baseline by more than 4% of $F1$ and more than 15% in terms of accuracy. Retrospectively, Triplet(DASS) also significantly outperforms the baseline in datasets from previous iterations (greyed-out results in Table 17).

Table 17
`Builders`' models performance metrics (ACC and $F1$) across all iterations. Results greyed-out are for models that were developed after the given iteration. Best results per iteration are marked in bold.

	Iteration 3		Iteration 4		Iteration 5	
	ACC	$F1$	ACC	$F1$	ACC	$F1$
Baseline	71.10%	83.11%	60.18%	75.14%	57.60%	73.10%
Baseline(BT)	69.74%	82.17%	58.61%	73.90%	56.92%	72.54%
Siamese(1:1)	24.94%	39.92%	16.99%	29.05%	17.05%	29.13%
Siamese(TF-IDF)	51.38%	67.88%	33.36%	50.03%	27.23%	42.80%
Triplet(TF-IDF)	47.02%	63.96%	29.48%	45.54%	24.44%	39.28%
Triplet(DASS)	**81.47%**	**89.79%**	**72.88%**	**84.31%**	**72.68%**	**84.18%**

Summary. The Triplet(DASS) contrastive architecture invariably outperformed the transformer-based baseline, whereas the other contrastive models (excluding DASS) underperformed. These results indicate that contrastive methods are most effective against adversarial paraphrase-based and persona-driven transformations employed by the `breakers` only when they explicitly learn to generalise across paraphrasing.

6.3.1. Interpretation of RQ2 Results

The baseline classifier fell from 71.10% to 57.60% accuracy from iteration 3 to iteration 5. This decline of 13.5% aligns with the `breakers`' introduction of persona-based paraphrasing (D3 in iteration 4, D4 in iteration 5) combined with back-translation (B2). As shown in Section 6.1.2, the strongest `breakers`' attack configuration (B3_D1_D4_B2) achieved a maximum LFR of 95.2% against the baseline classifier. The baseline had only a 4.8% accuracy rate on that specific set. We can interpret this as evidence that the classifier relied on surface-level correlations rather than learning the underlying semantic meaning of a post. Therefore, if the same disinformation claim is rewritten in multiple surface forms (i.e. different wording, persona styles, or syntactic structures), the classifier can treat each perturbed text as if it were a different class, misclassifying it as HGO as it fails to recognise them as variants of the same machine-generated content.

As the Siamese (1:1) variant was trained on fixed pairs where each human post was paired with its machine-generated twin on the same claim, it was expected to distinguish between HGO and MGO on identical topics. Yet, it was the worst-performing model, achieving a 24.94% accuracy rate in iteration 3 and dropping to the overall worst accuracy rate of 16.99% at iteration 4. This shows that the lack of variance in the training signal can significantly impair a contrastive network model to detect authorship. It may consider superficial artefacts that are only present in a pair, like a particular word that appears only in a specific HGO-MGO sample. The failure to generalise was exploited by the `breakers` when they introduced paraphrase variants; since the model never saw such variation, its decision boundary completely failed.

Both contrastive architectures (Siamese and Triplet) explored TF-IDF pairing. These showed moderate performance at iteration 3 (51% and 47% accuracy scores, respectively) but were broken by `breakers` techniques in iteration 5 (27% and 24% accuracy scores, respectively). Since TF-IDF measures word overlap (lexical similarity) rather than semantics, once the `breakers` introduced persona-based prompts and multi-layered back-translation attacks, the lexical similarity between the original and transformed machine-generated text decreased. In the context of the detection models, this moved most samples closer to human examples in the TF-IDF space. Consequently, the hard negatives became misleading, and the models failed.

DASS. In contrast to the aforementioned pairing engines, DASS trains directly on paraphrased variants. DASS is the only detector strategy that maintained an accuracy rate above 70% across all iterations (81.52% → 72.94% → 72.75%, on iterations 3, 4 and 5 respectively). The success stems from the fact that DASS alternates the anchor between an original machine-generated tweet and its paraphrased variant, with a human tweet as the negative. The triplet loss pulls both machine-generated variants together while pushing them away from the human text (Fig. 6). The model then learns that paraphrasing does not change the underlying class. While other models dropped as the `breakers` introduced their most aggressive persona-based paraphrasing (D4) at iteration 5, DASS maintained its accuracy score at 72.68%, showing its invariance to paraphrase-based adversarial attacks.

Vulnerabilities. The main weakness and vulnerability discovered by the `breakers` was the personas. Prompting the LLM to rewrite a post under an "extreme" persona, such as a *brain-rot 10-year old*, resulted in the output post being human-like, distorted and noisy enough to trick all the models into flipping their prediction for nearly every instance. For example, the post *"Antifa takes responsibility for storming Capitol Hill, according to a CNN chyron, details still emerging, investigation ongoing."* became *"Ommg omg omg Capitol Hill is soooo crazy! Antifa people, as they said they did, it's sad. A lot of bad things happened, and we're still not quite done with tho #CapitolHill #Antifa"* after the B1_D13_D23_B2_JA attack has been applied. This combination of attacks created a post that retains enough meaning to still be considered misinformation, but adds uncertainty for the model's prediction as the post now contains many layers of distortion. One of these layers is that the post now contains spurious grammar and lexical choices, which is likely due to the back-translation technique. The persona has also introduced common slang used by social media users and human stylistic cues, adding even more noise. All of these together are able to trick the model into flipping its prediction and to show that no signals are present that cause overfitting, resulting in the post being predicted as human-authored with a probability of 0.770. The *college student* and *brain-rot 10-year old* personas exposed the biggest weakness of the models, as these personas were able to consistently replicate human signals present in the human part of the `builders`' training data and mimic an informal style. On the other hand, the *politician* persona maintained structure and formality that aligned closer to machine-generated posts.

6.4. `BiBiR` assessment (RQ4)

The iterative evaluation achieved through the `BiBiR` framework, when compared to static benchmark evaluation, can provided a more fine-grained and accurate assessment of models' performance at each iteration. To assess this, the randomly selected test set from the bld_data, as presented in Fig. 5 (this set contains 256 examples, see Fig. 2). This set, drawn from the same distribution of the `builders`' training and development data, shows the performance of detectors when evaluation relies on in-domain, in-distribution held-out data (which is a common practice in machine learning tasks).

Fig. 13 shows the comparison of models on the bld_data held out test set and on the `breakers`' set from interations 3 to 5. As expected, all models show higher performance at the `builders`' test set than on `breakers`' adversarial set. However, it is rather surprising that all models (except Siamese(1:1)) show accuracy score above 82% on `builders`' data. In fact, according to `builders`' held-out data only, the baseline model is the best, achieving 92.6% of accuracy, whilst Triple(DASS) achieves 82.4%. The Siamese(TF-IDF) model which showed very low accuracy scores for all iterations in the `breakers`' data, show an accuracy score of 84% on the `builders`' held-out data, highlighting the issues with static in-domain and in-distribution evaluation. Conversely, the adversarial data achieved via the `BiBiR`framework shows a different story: all models, but Triplet(DASS), show a significant reduction in performance from iteration 3 to 5 (it is worth reminding that the performance of the baseline model before applying adversarial attacks is 100%). This results reinforce the superiority of Triplet(DASS), since it remains stable in all datasets.

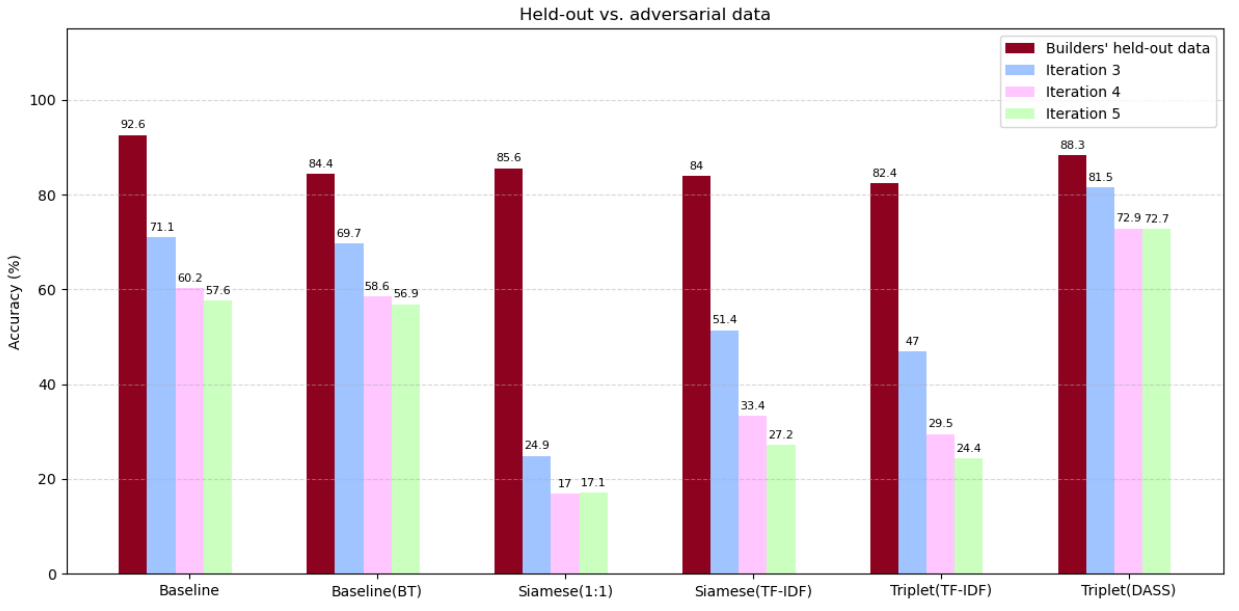


Figure 13: Performance comparison of models tested on different datasets.

The `breakers`' dataset introduced attack styles that were under-represented in the training data, specifically strong persona-based attacks and context shifting attacks that perturb the text significantly. While the baseline classifier was still able to correctly classify some attacks, especially during earlier iterations, the performance degraded significantly when persona-based attacks were introduced and combined with back-translation techniques. This suggests that the model was unable to generalise to evolving adversarial behaviour. In contrast, the Siamese (1:1 or TF-IDF) and Triplet(TF-IDF) models struggled even on earlier attacks that the baseline classifier remained robust to, suggesting that their similarity-based representations were more sensitive to distribution shifts and semantic perturbations introduced by adversarial prompting strategies. This observation suggests that adding posts with adversarial attacks to the training data with adversarial attacks may increase the performance of the Siamese and Triplet models. Instead of a sole TF-IDF pairing, a method consisting of a range of pairings, including adversarial-original and machine-human pairs, can potentially increase the generalisation of Siamese and Triplet models, as they learn the attacks applied and the differences between human and machine-written posts.

The iterative framework enforced the development of more robust models, capable of generalising to diverse unseen data. For instance, discourse markers, such as @ mentions, hashtags and URL's, or more sophisticated signals in the form of political topics were highly imbalanced in the `builders`' initial dataset, with the human data containing many of these signals while the machine-generated data did not. When trained on this data, the model overfited to these signals, associating them with human-authored post, regardless of the rest of the content in the post, creating a weakness that could be targeted. The `breakers`' initial attacks already exposed theses issues and, therefore, we devised a sanitisation script that prevented further exploitation of signals like URLs, emojis and white spaces. The `builders`' data was also augmented to balance out the topics and presence of signals between the human and machine-generated data.

7. Discussion & Conclusion

This paper aimed to evaluate how machine-generated disinformation detectors perform when short-form social media posts are deliberately transformed through adversarial techniques, using an iterative `BiBiR` framework. The `builders`' models attempted to separate human-written from machine-generated text, while the `breakers`' techniques attempted to preserve the original disinformation claim while making the posts harder to detect. Focus was given on whether the detector performance remained stable when exposed to character-level perturbation (A3), lexical perturbations (B1–B3), stylometric camouflage (C3), prompt-based evasion (D1–D4) and chained adversarial

transformations. A robuts Triplet network with DASS was the only model to remain robust throughout different attack iterations.

RQ1 Findings. RQ1 highlighted that detector performance was reduced more by chained adversarial attacks than by isolated transformations. Across later iterations, the strongest configurations generally combined paraphrasing, back-translation, and persona-based rewriting, with the best-performing configuration achieving 95% LFR, reducing detector accuracy to 5%. However, detector robustness cannot be judged on LFR alone, as label flip is only meaningful if the transformed post still preserves the original disinformation claim.

RQ2 Findings. Manual evaluation was therefore used as a validity check to distinguish genuine adversarial evasion from cases where the label flipped due to transformation degradation or to a change in the disinformation claim. Across 780 manually evaluated posts, some high-LFR techniques were associated with reduced semantic preservation and claim retention, although this relationship was weak rather than conclusive. The semantic preservation pipeline helped identify these cases, particularly through E5-cosine similarity and NLI labels, but its moderate agreement with manual evaluation shows that automatic metrics are not reliable enough as a final check.

RQ3 Findings. RQ3 showed that contrastive networks were not uniformly more robust than the standard transformer-based classifier. Within the iterative `BiBiR` framework, only the Triplet model with DASS Fig. 6 consistently outperformed the baseline classifier across all iterations. In contrast, the Siamese and Triplet with TF-IDF variants degraded significantly, falling from around 50% accuracy in Iteration 3 to well below chance-level accuracy by Iteration 5. This suggests that lexical-pairing strategies were brittle under heavy paraphrasing and persona-based rewriting. The key distinction was, therefore, not the contrastive architecture alone, but whether the model was trained to handle different paraphrasing variations. DASS improved robustness by alternating anchors between the original and paraphrased machine-generated samples which encouraged the model to recognise the same underlying claim even when it was rewritten in different wording. By Iteration 5, Triplet(DASS) outperformed the baseline classifier by over 15 percentage points. Therefore, the results suggest that training against rewritten versions of the same claim, rather than contrastive architecture alone, was a key factor in robustness against iterative adversarial rewriting.

RQ4 Findings show that iterative `BiBiR` evaluation exposed weaknesses that were not observable under the static test benchmark conditions. These results demonstrate that conventional benchmark evaluations may significantly overestimate robustness to real-world applications, especially in adversarial environments where attackers are continuously evolving and improving their attack strategies. This was seen in multiple models with the baseline classifier having a 35% difference between the `builders`' held-out data and iteration 5 dataset, the Siamese(TF-IDF) model having a 68.5% difference, the Triplet(TF-IDF) having a 58% difference, and the Triplet(DASS) having a 15.6% difference. These discrepancies demonstrate the need to move away from static benchmarks and move towards iterative, adversarial evaluation methods that can replicate real-world settings and provide a representative measurement for model robustness.

Limitations. In the specific domain of X and human-written misinformation, a primary challenge was the scarcity of reliable and verifiable human misinformation datasets. For example, the `builders` initially attempted to use the CovidMis20 dataset [101], but after careful review, it became apparent that the veracity of its content was misrepresented. Its misinformation labels were based solely on whether a URL originated from a reputable source instead of manual claim verification. As the research foundation is based on rumours, the dataset could not be used. Moreover, in an era dominated by LLMs, collecting large quantities of human data presents uncertainties about the actual authorship of the data. Thus, researchers are entirely dependent on a finite pool of dated, pre-LLM, trustworthy human misinformation data scraped from X and verified by experts - such as PHEME and Constraint for the `builders`, and TruthSeeker for the `breakers`. All these datasets are relatively small and centred on specific events, which limits the diversity and scale of the `BiBiR` challenge presented in this paper, potentially affecting its generalisation. Nevertheless, `builders` faced a stricter constraint as they were required to train their detectors on verified misinformation in the correct domain, whereas `breakers` could generate adversarial transformations based on fake claims produced by them.

Future Work has been divided into three sections:

`Builders:` future work includes expanding on the Triplet with DASS architecture to further optimise the model towards the task of classification. In addition, the collection of additional higher-quality human misinformation will provide the detectors with improved training data. This can then be combined with prompt engineering to generate better quality adversarial posts that further enrich the training data.

`Breakers:` future work includes using a semantic filtering pipeline instead of a metrics-related one, as this allows for active rejection of posts until they pass a threshold (which would be dependent on the type of technique and persona used). `Breakers` would also look at expanding the variety of personas used and experimenting with developing the personas further to give them more context.

`BiBiR`: future work should extend the `BiBiR` framework by adding a validation stage for scalable semantic preservation and disinformation claim preservation checking. This could adopt a hybrid human/LLM validation approach, where a smaller human-annotated subset is used to calibrate an LLM-based evaluator, which then scores the remaining posts. This would provide stronger evidence for determining whether label flips occur because of semantic degradation or because of effective `breakers`' techniques.

Acknowledgements

This work is supported by the Engineering and Physical Sciences Research Council (EPSRC) under grant UKRI3352: Longitudinal, Multilingual, and Multi-format Investigation and Detection of LLM-Generated Disinformation.

A. Manual Evaluation Guidelines

Fig. 14, Fig. 15 and Fig. 16 shows the complete annotation guidelines given to the annotators for human evaluation.

ME1 - Semantic Preservation

Purpose: ME1 measures how well a modified text preserves the meaning and intent of the original text. You are judging semantic equivalence not the writing quality of the two texts.

What to focus on:

- Core claims
- Factual content
- Speaker intent
- Logical relation between two text (does the two text's claims match)

What not to focus on:

- Tone (unless it changes meaning of the text)
- Fluency or grammatical quality

Scoring Scale

Score	Definition
3	Identical meaning preserved. The modified text has the same meaning as the original. The reader should draw the same conclusion from both texts and the veracity of the claim should be unchanged. Allow for: paraphrasing, synonyms, minor rewording/grammatical correction
2	Partially changed (key difference has been altered). This includes changes in certainty/strength of the claim, alteration in action/target/numbers/dates/facts, important details removed or added. The text has partially true values (e.g. out of the two truthful claims, one is now false and the other is true) or some of the claims are subject to ambiguity fallacy which causes confusion. The text is related but it is not equivalent.
1	Irrelevant, completely unrelated or opposite meaning. The modified text contradicts or reverses the original meaning. This includes explicit negation, claim is in denial with each other, or the text doesn't make sense in comparison to the original version.

Dealing with uncertainty: If there is uncertainty for which score to assign to a post, then assign the post as the midpoint between the two scoring categories. For example if you are not sure if a post is meant to be **2** or **1**, then assign the score of **1.5**.

Scoring scale with examples

Score	Example
3	**Original** - *They failed to recall that they know nothing about LLMs* **Modified** - *They failed to remember that they know nothing about LLMs* **Original** - *Farm Workers Face Double Threat: Wildfire Smoke And COVID-19 <URL>* **Modified** - *Farm workers face a double threat: Smoke from wildfires and the COVID-19 pandemic <URL>* **Original** - *What the blessed fuck is this fresh hell?!? Why would anyone want to continue to live in a suicidal god-damned Idiocracy? #USFail #AmericanDeathWish #SaveUS #CorruptGOP #Traitors US refuses to join international effort to develop Covid-19 vaccine <URL>* **Modified** - *What the hell is this damned hell!? Why would anyone want to continue living in a suicidal bloody idiocracy? #USFail #AmericanDeathWish #SaveUS #CorruptGOP #Traitors US refuses to join international effort to develop Covid-19 vaccine <URL>*
2	**Original** - *The protest was violent* **Modified** - *The protest was controversial* **Original** - *The (largely mask-less) Sturgis motorcycle rally: At least 260 infections traced to it. And now a death. <URL>* **Modified** - *Sturgis Motorbike Rally (largely unmasked): At least 260 injuries have been tracked. And now a death. <URL>*
1	**Original** - *The bill was too expensive at the restaurant* **Modified** - *The bill was very cheap at the restaurant* **Original** - *According to the Pope India is suffering from another virus apart from Coronavirus. <URL> #JEE #sscdeclareresult #rrbexamsdates #Rhea #Covid_19* **Modified** - *According to the Pope, India is suffering from a different coronavirus. <URL> #JEE #sscdeclareresult #rrbexamsdates #Rhea #Covid_19* **Original** - *RT @BlackWidow284: Many Americans face bleak winter as Covid takes toll on mental health <URL>* **Modified** - *RT @BlackWidow284: Many Americans are facing a dark winter.*

Explanation of Classification

Score	Example
3	Pair 1 - The modified text holds true semantic meaning only utilising the synonym for "recall" "remember" and both texts draw to the same conclusion. Pair 2- Despite structural rephrasing, all factual content in the modified text is preserved. Pair 3- The modified text maintains its core meaning, tone and factual claim. The modifications are purely substitutions and mild rewording.
2	Pair 1- The modified text "Controversial" is a broader, vaguer term than "violent" a violent protest is or can be controversial, but not vice versa. This substitution removes a specific factual claim, reducing both the strength and precision of the original, making the texts related but not equivalent. Pair 2 - the modification alters a key fact replacing "infections" with "injuries" these are very distinct outcomes and implications, there are some acceptable rephrases but the substitution shifts the strength of the claim. The claims are related but no longer equivalent.
1	Pair 1 - The modified text has an evidently opposite claim to the original. Pair 2 - This is an edge case scenario which can be labelled as 1.5. Reason being that the actual context has changed from there being '*another virus apart from covid*' to there being '*another strain of covid*'. However the high level context is still related to there being '*another virus in India*'. Pair 3 - The modified text has lost all context as it states that '*americans have a dark winter*' instead of it being due to the '*mental health toll from covid*'

Figure 14: Manual Evaluation Metric 1 - Semantic Preservation annotation guidelines given to the annotators.

ME2 - Naturalness

Purpose: ME2 measures how fluent, human-like, and natural the altered text sounds in its context. This focuses on the quality of the linguistics, disregarding objective truth or preservation of the semantic meaning of the original text. You are only evaluating the outcome text's naturalness by also understanding the context of the original text. Do not compare the naturalness between the two texts.

What to focus on:
- Grammatical correctness and coherence (correct tense/word order, no malformed clauses)
- Natural and idiomatic phrasing (any unnatural synonym substitutions such as '*violent actions*' or '*do harm to the economy situation*')
- Readability and fluency (watch for abrupt structural breaks, repeated sentence structure, overly formal phrasing)

What not to focus on:
- The factual nature of the context
- Your Subjective opinions of the message
- Semantic accuracy (Handled by ME1)

Text type Awareness
- Propaganda/ hyperpartisan news: emotionally driven language is normal
- Fact-checking claims: precise, neutral phrasing expected
- Rumours/ Social media: informal or fragmented language is acceptable
- News fragments: editorial tone expected

Do not penalise text for being informal especially if it fits the genre

Scoring Scale

Score	Definition
4	Natural and well formed text. The modified text flows smoothly, using a natural tone and varied word choice without repetitive patterns. The text reads fluently without any irregularities.
3	Mostly natural with minor awkwardness. The modified text is fluent and understandable however it contains mild redundancy and small grammatical stiffness. Text may feel slightly over structured or even less polished but the overall structure remains coherent.
2	Noticeably awkward or contains translation artefacts. The modified text is understandable but the word choice is poor with unnatural synonyms and broken collocations.
1	Clearly broken or incoherent. The text is difficult or impossible to read, with severe grammatical errors, broken syntax and fragmented output.

Dealing with uncertainty: If there is uncertainty for which score to assign to a post, then assign the post as the midpoint between the two scoring categories. For example if you are not sure if a post is meant to be **2** or **1**, then assign the score of **1.5**.

Scoring scale with examples

Score	Example
4	**Original** - *They failed to recall that they know nothing about LLMs* **Modified** - *They failed to remember that they know nothing about LLMs* **Original** - *RT guardianscience "Two types of steroid found to save lives of some Covid-19 patients <URL>* **Modified** - *RT guardianscience: 'Two steroids found to save lives of some Covid-19 patients' <URL>*
3	**Original** - *The policy caused economic harm* **Modified** - *The policy resulted in harm to the economy* **Original** - *This is disturbing for #COVID19 deaths IF people continue to ignore this warning <URL>* **Modified** - *If people continue to ignore this warning, the #COVID19 deaths will be disturbing <URL>.*
2	**Original** - *The protest was violent* **Modified** - *The protest was having violence actions during it* **Original** - *Glasgow surprised but compliant as Covid-19 restrictions return. <URL>* **Modified** - *Glasgow follows with surprise the return of Covid-19 regulations. <URL>*
1	**Original** - *The bill was too expensive at the restaurant* **Modified** - *The bill expensive was too at restaurant the* **Original** - Wellllll this didn't age well @politico #LetThemPlay <URL> **Modified** - Well, this didn't get too old @politico #LetThemPlay <URL>.

Explanation of Classification

Score	Example
4	Pair 1- The modified text flows smoothly maintaining a varied word choice and is indistinguishable from a human speech. Pair 2 - The modified text reads naturally and fits with the context of an article headline.
3	Pair 1- The modified text contains some minor awkwardness however it is still fluent with some minor stiffness. Pair 2- The modified text begins naturally however it is accompanied by awkward phrasing due to the restructure of the original text but it remains coherent.
2	Pair 1 - The modified text use of 'having violence actions' breaks the flow of the sentence and is an unnatural synonym in comparison to the word 'violent'. Pair 2 - The modified text has a very stiff and notable awkward opening and poor additional word choice " *follows with*".
1	Pair 1 - The modified texts don't flow at all, it seems like the words are used out of order and aren't coherent. Pair 2 - The modified text doesn't make sense when compared to the context as it uses abnormal wording.

Figure 15: Manual Evaluation Metric 2 - Naturalness annotation guidelines given to the annotators.

ME3 - Preservation of Disinformation Claim

Purpose: ME3 measures how well a modified text preserves the specific disinformation claim or narrative expressed in the original text. When scoring ME3, focus on whether the central misleading claim remains intact.

What to focus on:

Annotators should focus on:

- Core claims
- The target of the claim, such as person, group, organisation or product involved
- The cause and effect relationship, where relevant
- The strength or certainty of the claim

What not to focus on:

- Tone
- Naturalness
- Text length
- Fluency or grammatical quality
- Whether the wording is similar to the original

Scoring Scale

Score	Definition
1	**Claim held.** The output preserves the original disinformation claim and maintains the same misleading narrative, even if the wording, tone, or surface form has changed.
0.5	**Claim partially held.** The output retains part of the original disinformation claim, but weakens, softens, narrows, or slightly shifts the claim so that the misleading narrative is only partly preserved.
0	**Claim not held.** The output removes, contradicts, or substantially changes the original disinformation claim, meaning the core misleading narrative is no longer preserved.

Some examples are simplified illustrative cases used to clarify the scoring scale.

Scoring scale with examples

Score	Example
1	**Original** - *They failed to recall that they know nothing about LLMs* **Modified** - *They failed to remember that they know nothing about LLMs* **Original** - *Farm Workers Face Double Threat: Wildfire Smoke And COVID-19 <URL>* **Modified** - *Farm workers face a double threat: Smoke from wildfires and the COVID-19 pandemic <URL>* **Original** - *What the blessed fuck is this fresh hell?!? Why would anyone want to continue to live in a suicidal god-damned Idiocracy? #USFail #AmericanDeathWish #SaveUS #CorruptGOP #Traitors US refuses to join international effort to develop Covid-19 vaccine <URL>* **Modified** - *What the hell is this damned hell!? Why would anyone want to continue living in a suicidal bloody idiocracy? #USFail #AmericanDeathWish #SaveUS #CorruptGOP #Traitors US refuses to join international effort to develop Covid-19 vaccine <URL>*
0.5	**Original** - *The protest was violent* **Modified** - *The protest was controversial* **Original** - *The (largely mask-less) Sturgis motorcycle rally: At least 260 infections traced to it. And now a death. <URL>* **Modified** - *Sturgis Motorbike Rally (largely unmasked): At least 260 injuries have been tracked. And now a death. <URL>*
0	**Original** - *The bill was too expensive at the restaurant* **Modified** - *The bill was very cheap at the restaurant* **Original** - *According to the Pope India is suffering from another virus apart from Coronavirus. <URL> #JEE #sscdeclareresult #rrbexamsdates #Rhea #Covid_19* **Modified** - *According to the Pope, India is suffering from a different coronavirus. <URL> #JEE #sscdeclareresult #rrbexamsdates #Rhea #Covid_19* **Original** - *RT @BlackWidow284: Many Americans face bleak winter as Covid takes toll on mental health <URL>* **Modified** - *RT @BlackWidow284: Many Americans are facing a dark winter.*

Explanation of Classification

Score	Example
1	Pair 1 - The modified text preserves the same claim. “Recall” is simply replaced with “remember”, so the meaning remains unchanged. Pair 2- The modified text preserves the same claim that farm workers face a double threat from wildfire smoke and COVID-19. The wording and capitalisation change, but the core claim remains intact. Pair 3- The modified text preserves the same claim and emotional stance. The profanity and phrasing are softened slightly, but the post still criticises the US refusal to join the international COVID-19 vaccine effort and keeps the same hashtags
0.5	Pair 1- The modified text partially preserves the original claim because it keeps the same broad event, a protest, but changes the specific claim from “violent” to “controversial”. This weakens and generalises the original claim. Pair 2 - The modified text partially preserves the original claim because it keeps the Sturgis rally, lack of masks, and a negative outcome, but changes “infections” to “injuries”. This alters an important part of the original claim while keeping some of the same narrative structure.
0	Pair 1 - The modified text does not preserve the original claim because it directly contradicts it. “Too expensive” is changed to “very cheap”, reversing the meaning. Pair 2 -The modified text does not preserve the original claim because “another virus apart from Coronavirus” is changed to “a different coronavirus”. This changes the claim into a different meaning rather than preserving the original narrative Pair 3 - The modified text does not preserve the original claim because the specific COVID-19 mental health claim is removed. The modified version only keeps a vague “dark winter” statement, so the original claim is lost.

Figure 16: Manual Evaluation Metric 3 - Preservation of Disinformation Claim annotation guidelines given to the annotators.

B. Technique Code Combination Mapping

Table 18 shows the combinatorial mapping of technique codes for iteration 2. The table represents the power set of the $n = 5$ technique code families $\{D1, D2, B2, C3, A3\}$, visualised as a 2^n binary truth table. This exhaustive mapping ensures every possible combination within the iteration's technique set is accounted for.

Table 18
Combinatorial mapping of technique codes for iteration 2.

D1	D2	B2	C3	A3	Technique Code Generated
1	1	1	1	1	D1_D2_B2_C3_A3
1	1	1	1	0	D1_D2_B2_C3
1	1	1	0	1	D1_D2_B2_A3
1	1	1	0	0	D1_D2_B2
1	1	0	1	1	D1_D2_C3_A3
1	1	0	1	0	D1_D2_C3
1	1	0	0	1	D1_D2_A3
1	1	0	0	0	D1_D2
1	0	1	1	1	D1_B2_C3_A3
1	0	1	1	0	D1_B2_C3
1	0	1	0	1	D1_B2_A3
1	0	1	0	0	D1_B2
1	0	0	1	1	D1_C3_A3
1	0	0	1	0	D1_C3
1	0	0	0	1	D1_A3
1	0	0	0	0	D1
0	1	1	1	1	D2_B2_C3_A3
0	1	1	1	0	D2_B2_C3
0	1	1	0	1	D2_B2_A3
0	1	1	0	0	D2_B2
0	1	0	1	1	D2_C3_A3
0	1	0	1	0	D2_C3
0	1	0	0	1	D2_A3
0	1	0	0	0	D2
0	0	1	1	1	B2_C3_A3
0	0	1	1	0	B2_C3
0	0	1	0	1	B2_A3
0	0	1	0	0	B2
0	0	0	1	1	C3_A3
0	0	0	1	0	C3
0	0	0	0	1	A3
0	0	0	0	0	Original 125 Posts

C. Builders' unique prompts

Table 19 shows the prompts used by the builders when creating the machine-generated portion of their bld_data. Table 20 shows the prompts used to create the machine-generated paraphrases used by the Triplet(DASS) model.

Table 19: Baseline prompts for creating machine-generated text in bld_data.

Prompt Context	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6
You are a tired, careless internet user quickly typing out a post from your phone. Read this source text: <TWEET>. You must rewrite the core information in a "low-effort", highly unpredictable way.	Syntax & Vocabulary: Deliberately use awkward phrasing, run-on sentences, or unconventional synonyms. Avoid standard AI structures.	Formatting: Output ONLY the raw text. Zero introductory fluff or quotes.	Length: Keep it strictly around the same length as the original.	Tokens: Preserve <USER> and <URL>. Format hashtags as <HASHTAG> word. Do not use actual # or @ symbols.	No Emojis: Absolutely zero emojis.	Tone: Gritty, cynical, or completely unpolished.
Generate a realistic, casual Twitter/X post sharing or reacting to the information found in this source text: <TWEET>.	Zero Formatting: Output strictly the tweet text. No titles, quotes, or introductory text.	Length Constraint: Keep it brief and punchy. Match the brevity of the source text.	Zero Emojis: Absolutely no emojis allowed.	Tokens: Preserve <USER> and <URL>. Format hashtags as <HASHTAG> word.	Human Style: Use an unpolished conversational internet tone with possible grammatical imperfections.	–
You are an expert at rewriting social media posts. You must maintain the original meaning but significantly rephrase the structure. Original Post: <TWEET>	Formatting: Output ONLY the raw tweet text with no headers or filler.	Length: Keep approximately the same character length as the original tweet.	Tokens: Preserve <USER> and <URL>. Format hashtags as <HASHTAG> word.	No Emojis: Do not use emojis under any circumstances.	Tone: Write casually as an everyday internet user.	–
You are a distracted social media user typing quickly while scrolling. Read the source text: <TWEET>. Rewrite the same information as a spontaneous Twitter/X post.	Meaning Preservation: Keep the exact core message while restructuring the expression.	Human Imperfections: Allow awkward phrasing, inconsistent capitalisation, or grammar slips.	Formatting: Output ONLY the tweet text.	Length: Keep approximately the same length as the original tweet.	Tokens: Preserve <USER> and <URL>. Format hashtags as <HASHTAG> word.	No Emojis: Absolutely no emojis or special symbols.
Rewrite the tweet below so it sounds like a spontaneous social media comment instead of a clean paraphrase. Source text: <TWEET>.	Raw Output Only: Return only the tweet text.	Structural Change: Rearrange clauses and start from a different angle where possible.	Length: Make it slightly shorter and punchier than the original.	Tokens: Preserve <USER> and <URL>. Format hashtags as <HASHTAG> word.	Style: Informal internet tone that may sound blunt or skeptical.	No Emojis: Under no circumstances include emojis.
You are rewriting a tweet to sound like a real person reacting casually online. The original tweet is: <TWEET>.	Output Format: Only output the final tweet text.	Concise: Keep the length tightly aligned with the original tweet.	Natural Internet Voice: Use rough conversational phrasing or uneven punctuation.	Tokens: Preserve <USER> and <URL>. Format hashtags as <HASHTAG> word.	No Emojis: Do not include emojis or emoji-like characters.	Semantic Integrity: Preserve the original underlying information.
Task: Transform the provided source text into a highly articulate, grammatically flawless, and sophisticated social media tweet. It must maintain the same topic but be rewritten with new logical pathways. Source text: <TWEET>	Tone: Intellectual and structurally correct. Avoid sounding like a corporate brand.	Length: Strictly match the concise character count of the source tweet.	BANNED: Emojis, slang, conversational filler, quotation marks, and introductory text.	Tokens: Preserve <URL> and <USER>. Do not use actual @ mentions or links.	Hashtags: Format hashtags strictly as <HASHTAG> keyword. Never use the # symbol.	Output: Return only the final transformed text.

Prompt Context	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6
Your objective is to provide a grammatically perfect, clear, and highly readable paraphrase of the following source text: <TWEET>.	Grammar: Use flawless grammar, correct punctuation, and a polite neutral tone. Do not use slang.	Length: You may slightly expand the sentence structure while keeping it to a single short post.	Formatting: Return ONLY the final text with no introductory phrases.	Tokens: Preserve <USER> and <URL>. Format hashtags as <HASHTAG> word. Do not use actual # or @ symbols.	No Emojis: Zero emojis allowed.	–
Goal: Synthesize the provided source text into a precise, analytical, and grammatically flawless statement. Source text: <TWEET>.	Tone: Analytical, objective, and structurally sound. Use precise vocabulary.	Length: Make it slightly shorter and punchier than the original.	BANNED: Emojis, slang, conversational filler, quotation marks, and introductory text.	Tokens: Preserve <URL> and <USER>. Do not use actual @ mentions or links.	Hashtags: Format hashtags strictly as <HASHTAG> keyword. Never use the # symbol.	Output: Return only the final transformed text.
Act as an objective, professional news reporter. Read the source text: <TWEET>. Your assignment is to rewrite the core information as a sterile, factual news alert. You must completely strip away any emotional framing, internet slang, or grammatical imperfections.	You are forbidden from outputting conversational filler, labels, quotation marks, or emojis.	You may not use actual '' or '#' characters; instead, retain the existing <USER> and <URL> placeholders.	For hashtags, format them strictly as <HASHTAG> keyword (e.g. #topic -> <HASHTAG> topic).	Length constraint: Focus on clarity rather than exact character limits. It is acceptable if the output is slightly longer or more detailed structurally than the original.	-	-

Table 20
Paraphrased prompts (for DASS).

System Prompt Used
Pretend you are an overexcited 8-year-old child. Paraphrase the following text using very simple, childish vocabulary. You must keep the exact core topic, but make it sound like a kid explaining it on a playground. Constraint: Create minimal spelling mistakes, keep it around the same character limit as the text. Text: <TWEET>
Act as a highly educated political analyst. Rewrite the following text to sound like a formal, diplomatic policy observation. Elevate the vocabulary significantly while preserving the core meaning. Constraint: It must be concise enough to fit in a single short-form tweet (under 280 characters). Text: <TWEET>
Rewrite this text to convey extreme urgency and breaking-news energy. Strip away all conversational filler and use punchy, aggressive syntax. Constraint: Keep it brutally short, under 150 characters, while retaining the original topic. Text: <TWEET>
Paraphrase this text with a highly cynical, sarcastic tone, like a jaded internet commentator. Alter the sentence structure entirely while keeping the underlying topic. Constraint: Maintain a slightly shorter character count than the source tweet. Do not use hashtags. Source tweet: <TWEET>
Act as a highly educated political analyst. Rewrite the following text to sound like a formal, diplomatic policy observation. Elevate the vocabulary significantly while preserving the core meaning. Constraint: Make some tiny spelling and grammar mistakes. It must be concise enough to fit in a single short-form tweet (under 280 characters). Text: <TWEET>

References

[1] L. Turčilo, M. Obrenović, A Companion to Democracy #3: Misinformation, Disinformation, Malinformation: Causes, Trends, and Their Influence on Democracy, A Companion to Democracy: Heinrich Böll Foundation 3 (2020).

[2] R. DiResta, K. Shaffer, B. Ruppel, D. Sullivan, R. Matney, R. Fox, J. Albright, B. Johnson, The Tactics & Tropes of the Internet Research Agency, Technical Report, United States Senate Select Committee on Intelligence, 2019. URL: https://digitalcommons.unl.edu/senatedocs/2/, accessible via Digital Commons: https://digitalcommons.unl.edu/senatedocs/2/.

[3] D. Barman, Z. Guo, O. Conlan, The Dark Side of Language Models: Exploring the Potential of LLMs in Multimedia Disinformation Generation and Dissemination, Machine Learning with Applications 16 (2024) 100545.

[4] National Cyber Security Centre (NCSC), The Near-Term Impact of AI on the Cyber Threat, Technical Report, National Cyber Security Centre, UK, 2024. URL: https://www.ncsc.gov.uk/pdfs/report/impact-of-ai-on-cyber-threat.pdf, accessed: 2025-11-23.

[5] J. Zhou, Y. Zhang, Q. Luo, A. G. Parker, M. De Choudhury, Synthetic Lies: Understanding AI-Generated Misinformation and Evaluating Algorithmic and Human Solutions, in: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA, 2023, pp. 1–20. URL: https://doi.org/10.1145/3544548.3581318. doi:10.1145/3544548.3581318.

[6] B. Buchanan, A. Lohn, M. Musser, K. Sedova, Truth, Lies, and Automation: How Language Models Could Change Disinformation, Technical Report, Center for Security and Emerging Technology, 2021. URL: https://cset.georgetown.edu/publication/truth-lies-and-automation/.

[7] S. C. Matz, J. D. Teeny, S. S. Vaid, H. Peters, G. M. Harari, M. Cerf, The Potential of Generative AI for Personalized Persuasion at Scale, Scientific Reports 14 (2024) 4692.

[8] A. Zugecova, D. Macko, I. Srba, R. Moro, J. Kopál, K. Marcinčinová, M. Mesarčík, Evaluation of LLM Vulnerabilities to Being Misused for Personalized Disinformation Generation, in: W. Che, J. Nabende, E. Shutova, M. T. Pilehvar (Eds.), Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Vienna, Austria, 2025, pp. 780–797. URL: https://aclanthology.org/2025.acl-long.38/. doi:10.18653/v1/2025.acl-long.38.

[9] J. A. Leite, A. Arora, S. Gargova, J. Luz, G. Sampaio, I. Roberts, C. Scarton, K. Bontcheva, Tailored untruths: How personalisation challenges LLM safeguards, 2025. URL: https://arxiv.org/abs/2510.12993. arXiv:2510.12993.

[10] S. Vosoughi, D. Roy, S. Aral, The spread of true and false news online, Science 359 (2018) 1146–1151.

[11] N. Pröllochs, D. Bär, S. Feuerriegel, Emotions explain differences in the diffusion of true vs. false social media rumors, Scientific Reports 11 (2021) 22721.

[12] W. J. Brady, J. A. Wills, J. T. Jost, J. A. Tucker, J. J. Van Bavel, Emotion shapes the diffusion of moralized content in social networks, Proceedings of the National Academy of Sciences 114 (2017) 7313–7318.

[13] G. Hagen, R. Safavi-Naini, M. Yung, The Mis/Dis-Information Problem Is Hard to Solve, Springer Nature Switzerland, Cham, 2025, pp. 309–326. URL: https://doi.org/10.1007/978-3-031-83490-5_12. doi:10.1007/978-3-031-83490-5_12.

[14] J. Wu, S. Yang, R. Zhan, Y. Yuan, L. S. Chao, D. F. Wong, A survey on LLM-generated text detection: Necessity, methods, and future directions, Computational Linguistics 51 (2025) 275–338.

[15] X. Liu, Y. Li, K. Li, Enhancing the Robustness of AI-Generated Text Detectors: A Survey, Mathematics 13 (2025).

[16] S. Gehrmann, H. Strobelt, A. Rush, GLTR: Statistical detection and visualization of generated text, in: M. R. Costa-jussà, E. Alfonseca (Eds.), Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, Florence, Italy, 2019, pp. 111–116. URL: https://aclanthology.org/P19-3019/. doi:10.18653/v1/P19-3019.

[17] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, C. Finn, DetectGPT: zero-shot machine-generated text detection using probability curvature, in: Proceedings of the 40th International Conference on Machine Learning, ICML'23, JMLR.org, 2023. URL: https://dl.acm.org/doi/10.5555/3618408.3619446.

[18] K. Krishna, Y. Song, M. Karpinska, J. Wieting, M. Iyyer, Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense, in: Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Curran Associates Inc., Red Hook, NY, USA, 2023. URL: https://dl.acm.org/doi/10.5555/3666122.3667317.

[19] S. Schneider, F. Steuber, J. A. Schneider, G. Dreo Rodosek, Detection avoidance techniques for large language models, Data & Policy 7 (2025) e29.

[20] A. Pedrotti, M. Papucci, C. Ciaccio, A. Miaschi, G. Puccetti, F. Dell'Orletta, A. Esuli, Stress-testing machine generated text detection: Shifting language models writing style to fool detectors, in: W. Che, J. Nabende, E. Shutova, M. T. Pilehvar (Eds.), Findings of the Association for Computational Linguistics: ACL 2025, Association for Computational Linguistics, Vienna, Austria, 2025, pp. 3010–3031. URL: https://aclanthology.org/2025.findings-acl.156/. doi:10.18653/v1/2025.findings-acl.156.

[21] Y. Zhou, B. He, L. Sun, Humanizing machine-generated content: Evading AI-text detection through adversarial attack, in: N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, N. Xue (Eds.), Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), ELRA and ICCL, Torino, Italia, 2024, pp. 8427–8437. URL: https://aclanthology.org/2024.lrec-main.739/.

[22] V. Fishchuk, D. Braun, Robustness of generative AI detection: adversarial attacks on black-box neural text detectors, International Journal of Speech Technology 27 (2024) 861–874.

[23] J. Lucas, A. Uchendu, M. Yamashita, J. Lee, S. Rohatgi, D. Lee, Fighting fire with fire: The dual role of LLMs in crafting and detecting elusive disinformation, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 14279–14305. URL: https://aclanthology.org/2023.emnlp-main.883/. doi:10.18653/v1/2023.emnlp-main.883.

[24] S. Nathanson, Y. Yoo, D. Na, Y. Cao, L. Watkins, A Step Towards Modern Disinformation Detection: Novel Methods for Detecting LLM-Generated Text, in: MILCOM IEEE Military Communications Conference, IEEE, 2024, pp. 615–620.

[25] H. Stiff, F. Johansson, Detecting computer-generated disinformation, International Journal of Data Science and Analytics 13 (2022) 363–383.

[26] S. Jadhwani, S. Jain, P. Doshi, et al., Detecting AI-generated content in short form text, Research Square (2025). Preprint, Version 1.

[27] R. Schwarz, An Analysis on Short-Form Text and Derived Engagement, Ph.D. thesis, 2024. URL: https://www.proquest.com/dissertations-theses/analysis-on-short-form-text-derived-engagement/docview/3122661917/se-2.

[28] A. Ruef, M. Hicks, J. Parker, D. Levin, M. L. Mazurek, P. Mardziel, Build It, Break It, Fix It: Contesting Secure Development, in: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS '16, Association for Computing Machinery, New York, NY, USA, 2016, p. 690–703. URL: https://doi.org/10.1145/2976749.2978382. doi:10.1145/2976749.2978382.

[29] E. Dinan, S. Humeau, B. Chintagunta, J. Weston, Build it Break it Fix it for Dialogue Safety: Robustness from Adversarial Human Attack, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 4537–4546. URL: https://aclanthology.org/D19-1461/. doi:10.18653/v1/D19-1461.

[30] J. Thorne, A. Vlachos, Adversarial attacks against Fact Extraction and VERification, 2019. URL: http://arxiv.org/abs/1903.05543. arXiv:1903.05543.

[31] P. Chao, A. Robey, E. Dobriban, H. Hassani, G. J. Pappas, E. Wong, Jailbreaking black box large language models in twenty queries, in: 2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML), 2025, pp. 23–42. URL: https://ieeexplore.ieee.org/document/10992337. doi:10.1109/SaTML64287.2025.00010.

[32] M. S. Jabbar, S. Al-Azani, A. Alotaibi, M. Ahmed, Red teaming large language models: A comprehensive review and critical analysis, Information Processing & Management 62 (2025) 104239.

[33] Y. Dong, R. Mu, Y. Zhang, S. Sun, T. Zhang, C. Wang, Safeguarding large language models: A survey, Artif Intell Rev 58 (2025).

[34] F. Heppell, M. E. Bakir, K. Bontcheva, Lying Blindly: Bypassing ChatGPT's Safeguards to Generate Hard-to-Detect Disinformation Claims, 2024. URL: https://arxiv.org/abs/2402.08467. arXiv:2402.08467.

[35] R. Xu, B. Lin, S. Yang, T. Zhang, W. Shi, T. Zhang, Z. Fang, W. Xu, H. Qiu, The earth is flat because...: Investigating LLMs' belief towards misinformation via persuasive conversation, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 16259–16303. URL: https://aclanthology.org/2024.acl-long.858/. doi:10.18653/v1/2024.acl-long.858.

[36] S. S. Ghosal, S. Chakraborty, J. Geiping, F. Huang, D. Manocha, A. S. Bedi, Towards Possibilities & Impossibilities of AI-generated Text Detection: A Survey, 2023. URL: https://arxiv.org/abs/2310.15264. arXiv:2310.15264.

[37] S. Bhyravajjula, M. Walsh, A. Preus, M. Antoniak, so much depends / upon / a whitespace: Why Whitespace Matters for Poets and LLMs, in: C. Christodoulopoulos, T. Chakraborty, C. Rose, V. Peng (Eds.), Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Suzhou, China, 2025, pp. 35156–35173. URL: https://aclanthology.org/2025.emnlp-main.1783/. doi:10.18653/v1/2025.emnlp-main.1783.

[38] E. Sarabamoun, Special-Character Adversarial Attacks on Open-Source Language Model, 2025. URL: https://arxiv.org/abs/2508.14070. arXiv:2508.14070.

[39] Q. Peng, C. Zhang, R. Mangal, C. Pasareanu, L. Jia, Random Perturbation Attack on LLMs for Code Generation, in: 2025 IEEE/ACM 4th International Conference on AI Engineering – Software Engineering for AI (CAIN), 2025, pp. 285–287. URL: https://ieeexplore.ieee.org/document/11030003. doi:10.1109/CAIN66642.2025.00052.

[40] X. Wang, H. Jin, Y. Yang, K. He, Natural Language Adversarial Defense through Synonym Encoding, in: Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence (UAI 2021), PMLR, 2021, pp. 823–833. URL: https://proceedings.mlr.press/v161/wang21a/wang21a.pdf.

[41] Z. Rao, Y. Mohamed, S. Liu, Z. Liu, Two Birds with One Stone: Multi-task Detection and Attribution of LLM-Generated Text, in: W. Liang, S.-Y. Kung, M. Qiu (Eds.), Security and Privacy in Communication Networks, Springer Nature Switzerland, Cham, 2026, pp. 582–601. URL: https://doi.org/10.1007/978-3-032-23450-6_30.

[42] G. A. Adam, A. Cui, E. Thomas, E. Napier, N. Shmatko, J. Schnell, J. J. Tian, A. Dronavalli, E. Tian, D. Lee, Gptzero: Robust detection of llm-generated texts, 2026. URL: https://arxiv.org/abs/2602.13042. arXiv:2602.13042.

[43] J. Tiedemann, S. Thottingal, OPUS-MT – building open translation services for the world, in: A. Martins, H. Moniz, S. Fumega, B. Martins, F. Batista, L. Coheur, C. Parra, I. Trancoso, M. Turchi, A. Bisazza, J. Moorkens, A. Guerberof, M. Nurminen, L. Marg, M. L. Forcada (Eds.), Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, European Association for Machine Translation, Lisboa, Portugal, 2020, pp. 479–480. URL: https://aclanthology.org/2020.eamt-1.61/.

[44] K. Alperin, R. Leekha, A. Uchendu, T. Nguyen, S. Medarametla, C. Levya Capote, S. Aycock, C. Dagli, Masks and Mimicry: Strategic Obfuscation and Impersonation Attacks on Authorship Verification, in: M. Hämäläinen, E. Öhman, Y. Bizzoni, S. Miyagawa, K. Alnajjar (Eds.), Proceedings of the 5th International Conference on Natural Language Processing for Digital Humanities, Association for Computational Linguistics, Albuquerque, USA, 2025, pp. 102–116. URL: https://aclanthology.org/2025.nlp4dh-1.10/. doi:10.18653/v1/2025.nlp4dh-1.10.

[45] A. R. Williams, L. Burke-Moore, R. S.-Y. Chan, F. E. Enock, F. Nanni, T. Sippy, Y.-L. Chung, E. Gabasova, K. Hackenburg, J. Bright, Large language models can consistently generate high-quality content for election disinformation operations, PloS one 20 (2025) e0317421.

[46] K. Zhu, J. Wang, J. Zhou, Z. Wang, H. Chen, Y. Wang, L. Yang, W. Ye, Y. Zhang, N. Gong, X. Xie, PromptRobust: Towards Evaluating the Robustness of Large Language Models on Adversarial Prompts, in: Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis, LAMPS '24, Association for Computing Machinery, New York, NY, USA, 2024, p. 57–68. URL: https://doi.org/10.1145/3689217.3690621. doi:10.1145/3689217.3690621.

[47] C. Wu, Y.-m. Cheung, B. Han, D. Lian, Advancing machine-generated text detection from an easy to hard supervision perspective, Advances in Neural Information Processing Systems 38 (2026) 150210–150258.

[48] C. Zeng, S. Tang, Y. Chen, Z. Shen, W. Yu, X. Zhao, H. Chen, W. Cheng, Z. Xu, Human texts are outliers: detecting LLM-generated texts via out-of-distribution detection, Advances in Neural Information Processing Systems 38 (2026) 163483–163513.

[49] R. Gu, X. Meng, AISPACE at SemEval-2024 task 8: A Class-balanced Soft-voting System for Detecting Multi-generator Machine-generated Text, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 1476–1481. URL: https://aclanthology.org/2024.semeval-1.212/. doi:10.18653/v1/2024.semeval-1.212.

[50] M. Siino, BadRock at SemEval-2024 Task 8: DistilBERT to Detect Multigenerator, Multidomain and Multilingual Black-Box Machine-Generated Text, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 239–245. URL: https://aclanthology.org/2024.semeval-1.37/. doi:10.18653/v1/2024.semeval-1.37.

[51] A. Voznyuk, V. Konovalov, DeepPavlov at SemEval-2024 Task 8: Leveraging Transfer Learning for Detecting Boundaries of Machine-Generated Texts, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 1821–1829. URL: https://aclanthology.org/2024.semeval-1.257/. doi:10.18653/v1/2024.semeval-1.257.

[52] R. Tang, Y.-N. Chuang, X. Hu, The Science of Detecting LLM-Generated Text, Commun. ACM 67 (2024) 50–59.

[53] J. Pu, Z. Sarwar, S. M. Abdullah, A. Rehman, Y. Kim, P. Bhattacharya, M. Javed, B. Viswanath, Deepfake text detection: Limitations and opportunities, in: 2023 IEEE symposium on security and privacy (SP), IEEE, 2023, pp. 1613–1630.

[54] S. Ma, J. Li, Z. Mao, Q. Wang, Zero-shot detection of LLM-generated text using temperature sensitivity, in: M. Liakata, V. P. Moreira, J. Zhang, D. Jurgens (Eds.), Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, San Diego, California, United States, 2026, pp. 37664–37679. URL: https://aclanthology.org/2026.acl-long.1748/. doi:10.18653/v1/2026.acl-long.1748.

[55] X. Chen, J. Wu, S. Yang, R. Zhan, Z. Wu, Z. Luo, D. Wang, M. Yang, L. S. Chao, D. F. Wong, Repreguard: Detecting llm-generated text by revealing hidden representation patterns, Transactions of the Association for Computational Linguistics 13 (2025) 1812–1831.

[56] D. Macko, R. Moro, A. Uchendu, J. Lucas, M. Yamashita, M. Pikuliak, I. Srba, T. Le, D. Lee, J. Simko, M. Bielikova, MULTITuDE: Large-scale multilingual machine-generated text detection benchmark, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 9960–9987. URL: https://aclanthology.org/2023.emnlp-main.616/. doi:10.18653/v1/2023.emnlp-main.616.

[57] L. Dugan, A. Hwang, F. Trhlík, A. Zhu, J. M. Ludan, H. Xu, D. Ippolito, C. Callison-Burch, RAID: A shared benchmark for robust evaluation of machine-generated text detectors, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 12463–12492. URL: https://aclanthology.org/2024.acl-long.674/. doi:10.18653/v1/2024.acl-long.674.

[58] J. Wu, R. Zhan, D. F. Wong, S. Yang, X. Yang, Y. Yuan, L. S. Chao, Detectrl: Benchmarking llm-generated text detection in real-world scenarios, Advances in Neural Information Processing Systems 37 (2024) 100369–100401.

[59] X. Yu, Y. Yu, D. Liu, K. Chen, W. Zhang, N. Yu, J. Shao, EvoBench: Towards real-world LLM-generated text detection benchmarking for evolving large language models, in: W. Che, J. Nabende, E. Shutova, M. T. Pilehvar (Eds.), Findings of the Association for Computational Linguistics: ACL 2025, Association for Computational Linguistics, Vienna, Austria, 2025, pp. 14605–14620. URL: https://aclanthology.org/2025.findings-acl.754/. doi:10.18653/v1/2025.findings-acl.754.

[60] Y. Li, Q. Li, L. Cui, W. Bi, Z. Wang, L. Wang, L. Yang, S. Shi, Y. Zhang, MAGE: Machine-generated text detection in the wild, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 36–53. URL: https://aclanthology.org/2024.acl-long.3/. doi:10.18653/v1/2024.acl-long.3.

[61] Y. Wang, J. Mansurov, P. Ivanov, J. Su, A. Shelmanov, A. Tsvigun, O. Mohammed Afzal, T. Mahmoud, G. Puccetti, T. Arnold, SemEval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 2057–2079. URL: https://aclanthology.org/2024.semeval-1.279/. doi:10.18653/v1/2024.semeval-1.279.

[62] T.-g. Marchitan, C. Creanga, L. P. Dinu, Team Unibuc - NLP at SemEval-2024 task 8: Transformer and hybrid deep learning based models for machine-generated text detection, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 403–411. URL: https://aclanthology.org/2024.semeval-1.63/. doi:10.18653/v1/2024.semeval-1.63.

[63] M. Abassy, K. Elozeiri, A. Aziz, M. N. Ta, R. V. Tomar, B. Adhikari, S. E. D. Ahmed, Y. Wang, O. Mohammed Afzal, Z. Xie, J. Mansurov, E. Artemova, V. Mikhailov, R. Xing, J. Geng, H. Iqbal, Z. M. Mujahid, T. Mahmoud, A. Tsvigun, A. F. Aji, A. Shelmanov, N. Habash, I. Gurevych, P. Nakov, LLM-DetectAIve: a tool for fine-grained machine-generated text detection, in: D. I. Hernandez Farias, T. Hope, M. Li (Eds.), Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 336–343. URL: https://aclanthology.org/2024.emnlp-demo.35/. doi:10.18653/v1/2024.emnlp-demo.35.

[64] M. K. Mobin, M. S. Islam, LuxVeri at GenAI detection task 3: Cross-domain detection of AI-generated text using inverse perplexity-weighted ensemble of fine-tuned transformer models, in: F. Alam, P. Nakov, N. Habash, I. Gurevych, S. Chowdhury, A. Shelmanov, Y. Wang, E. Artemova, M. Kutlu, G. Mikros (Eds.), Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect), International Conference on Computational Linguistics, Abu Dhabi, UAE, 2025, pp. 352–357. URL: https://aclanthology.org/2025.genaidetect-1.41/.

[65] H. Kandula, C. F. Li, H. Qiu, D. Karakos, H. Man, T. H. Nguyen, B. Ulicny, BBN-U.Oregon's ALERT system at GenAI content detection task 3: Robust authorship style representations for cross-domain machine-generated text detection, in: F. Alam, P. Nakov, N. Habash, I. Gurevych, S. Chowdhury, A. Shelmanov, Y. Wang, E. Artemova, M. Kutlu, G. Mikros (Eds.), Proceedings of the 1stWorkshop on GenAI

Content Detection (GenAIDetect), International Conference on Computational Linguistics, Abu Dhabi, UAE, 2025, pp. 358–364. URL: https://aclanthology.org/2025.genaidetect-1.42/.

[66] S. Agrahari, P. Mishra, S. Kumar, Random at GenAI detection task 3: A hybrid approach to cross-domain detection of machine-generated text with adversarial attack mitigation, in: F. Alam, P. Nakov, N. Habash, I. Gurevych, S. Chowdhury, A. Shelmanov, Y. Wang, E. Artemova, M. Kutlu, G. Mikros (Eds.), Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect), International Conference on Computational Linguistics, Abu Dhabi, UAE, 2025, pp. 365–370. URL: https://aclanthology.org/2025.genaidetect-1.43/.

[67] A. R. Edikala, G. A. Katsios, N. Creaghe, N. Yu, Leidos at GenAI detection task 3: A weight-balanced transformer approach for AI generated text detection across domains, in: F. Alam, P. Nakov, N. Habash, I. Gurevych, S. Chowdhury, A. Shelmanov, Y. Wang, E. Artemova, M. Kutlu, G. Mikros (Eds.), Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect), International Conference on Computational Linguistics, Abu Dhabi, UAE, 2025, pp. 340–346. URL: https://aclanthology.org/2025.genaidetect-1.39/.

[68] Y. Wei, Team AT at SemEval-2024 task 8: Machine-generated text detection with semantic embeddings, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 492–496. URL: https://aclanthology.org/2024.semeval-1.75/. doi:10.18653/v1/2024.semeval-1.75.

[69] F. Xiong, T. Markchom, Z. Zheng, S. Jung, V. Ojha, H. Liang, NCL-UoR at SemEval-2024 task 8: Fine-tuning large language models for multigenerator, multidomain, and multilingual machine-generated text detection, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 163–169. URL: https://aclanthology.org/2024.semeval-1.25/. doi:10.18653/v1/2024.semeval-1.25.

[70] X. Hu, P.-Y. Chen, T.-Y. Ho, RADAR: robust AI-text detection via adversarial learning, in: Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Curran Associates Inc., Red Hook, NY, USA, 2023. URL: https://dl.acm.org/doi/10.5555/3666122.3666784.

[71] H. Chen, J. Büssing, D. Rügamer, E. Nie, Team MGTD4ADL at SemEval-2024 task 8: Leveraging (sentence) transformer models with contrastive learning for identifying machine-generated text, in: A. K. Ojha, A. S. Doğruöz, H. Tayyar Madabushi, G. Da San Martino, S. Rosenthal, A. Rosá (Eds.), Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 1711–1718. URL: https://aclanthology.org/2024.semeval-1.245/. doi:10.18653/v1/2024.semeval-1.245.

[72] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: Proceedings of the 37th International Conference on Machine Learning, ICML'20, JMLR.org, 2020. URL: https://dl.acm.org/doi/10.5555/3524938.3525087.

[73] T. Gao, X. Yao, D. Chen, SimCSE: Simple contrastive learning of sentence embeddings, in: M.-F. Moens, X. Huang, L. Specia, S. W.-t. Yih (Eds.), Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021, pp. 6894–6910. URL: https://aclanthology.org/2021.emnlp-main.552/. doi:10.18653/v1/2021.emnlp-main.552.

[74] A. van den Oord, Y. Li, O. Vinyals, Representation learning with contrastive predictive coding, 2019. URL: https://arxiv.org/abs/1807.03748. arXiv:1807.03748.

[75] D. Chicco, Siamese neural networks: An overview, Artificial neural networks (2021) 73–94.

[76] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, 2019, pp. 3982–3992. URL: https://aclanthology.org/D19-1410/. doi:10.18653/v1/D19-1410.

[77] F. Schroff, D. Kalenichenko, J. Philbin, FaceNet: A unified embedding for face recognition and clustering, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2015, p. 815–823. URL: http://dx.doi.org/10.1109/CVPR.2015.7298682. doi:10.1109/cvpr.2015.7298682.

[78] L. La Cava, D. Costa, A. Tagarelli, Is Contrasting All You Need? Contrastive Learning for the Detection and Attribution of AI-generated Text, IOS Press, 2024. URL: http://dx.doi.org/10.3233/FAIA240862. doi:10.3233/faia240862.

[79] Y. Feng, H. Wang, J. Li, Z. Cao, L. Yan, GravText: A Robust Framework for Detecting LLM-Generated Text Using Triplet Contrastive Learning with Gravitational Factor, Systems 13 (2025).

[80] G. Bao, Y. Zhao, Z. Teng, L. Yang, Y. Zhang, Fast-DetectGPT: Efficient zero-shot detection of machine-generated text via conditional probability curvature, in: The Twelfth International Conference on Learning Representations, volume 2024, 2024, pp. 24814–24836.

[81] A. Hans, A. Schwarzschild, V. Cherepanova, H. Kazemi, A. Saha, M. Goldblum, J. Geiping, T. Goldstein, Spotting LLMs with binoculars: zero-shot detection of machine-generated text, in: Proceedings of the 41st International Conference on Machine Learning, ICML'24, JMLR.org, 2024. URL: https://dl.acm.org/doi/10.5555/3692070.3692768.

[82] A. Zubiaga, M. Liakata, R. Procter, Learning reporting dynamics during breaking news for rumour detection in social media, 2016. URL: https://arxiv.org/abs/1610.07363. arXiv:1610.07363.

[83] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, A. Rao, A. Zhang, A. Rodriguez, A. Gregerson, A. Spataru, B. Roziere, B. Biron, B. Tang, B. Chern, C. Caucheteux, C. Nayak, C. Bi, C. Marra, C. McConnell, C. Keller, C. Touret, C. Wu, C. Wong, C. C. Ferrer, C. Nikolaidis, D. Allonsius, D. Song, D. Pintz, D. Livshits, D. Wyatt, D. Esiobu, D. Choudhary, D. Mahajan, D. Garcia-Olano, D. Perino, D. Hupkes, E. Lakomkin, E. AlBadawy, E. Lobanova, E. Dinan, E. M. Smith, F. Radenovic, F. Guzmán, F. Zhang, G. Synnaeve, G. Lee, G. L. Anderson, G. Thattai, G. Nail, G. Mialon, G. Pang, G. Cucurell, H. Nguyen, H. Korevaar, H. Xu, H. Touvron, I. Zarov, I. A. Ibarra, I. Kloumann, I. Misra, I. Evtimov, J. Zhang, J. Copet, J. Lee, J. Geffert, J. Vranes, J. Park, J. Mahadeokar, J. Shah, J. van der Linde, J. Billock, J. Hong, J. Lee, J. Fu, J. Chi, J. Huang, J. Liu, J. Wang, J. Yu, J. Bitton, J. Spisak, J. Park, J. Rocca, J. Johnstun, J. Saxe, J. Jia, K. V. Alwala, K. Prasad, K. Upasani, K. Plawiak, K. Li, K. Heafield, K. Stone, K. El-Arini, K. Iyer, K. Malik, K. Chiu,

K. Bhalla, K. Lakhotia, L. Rantala-Yeary, L. van der Maaten, L. Chen, L. Tan, L. Jenkins, L. Martin, L. Madaan, L. Malo, L. Blecher, L. Landzaat, L. de Oliveira, M. Muzzi, M. Pasupuleti, M. Singh, M. Paluri, M. Kardas, M. Tsimpoukelli, M. Oldham, M. Rita, M. Pavlova, M. Kambadur, M. Lewis, M. Si, M. K. Singh, M. Hassan, N. Goyal, N. Torabi, N. Bashlykov, N. Bogoychev, N. Chatterji, N. Zhang, O. Duchenne, O. Çelebi, P. Alrassy, P. Zhang, P. Li, P. Vasic, P. Weng, P. Bhargava, P. Dubal, P. Krishnan, P. S. Koura, P. Xu, Q. He, Q. Dong, R. Srinivasan, R. Ganapathy, R. Calderer, R. S. Cabral, R. Stojnic, R. Raileanu, R. Maheswari, R. Girdhar, R. Patel, R. Sauvestre, R. Polidoro, R. Sumbaly, R. Taylor, R. Silva, R. Hou, R. Wang, S. Hosseini, S. Chennabasappa, S. Singh, S. Bell, S. S. Kim, S. Edunov, S. Nie, S. Narang, S. Raparthy, S. Shen, S. Wan, S. Bhosale, S. Zhang, S. Vandenhende, S. Batra, S. Whitman, S. Sootla, S. Collot, S. Gururangan, S. Borodinsky, T. Herman, T. Fowler, T. Sheasha, T. Georgiou, T. Scialom, T. Speckbacher, T. Mihaylov, T. Xiao, U. Karn, V. Goswami, V. Gupta, V. Ramanathan, V. Kerkez, V. Gonguet, V. Do, V. Vogeti, V. Albiero, V. Petrovic, W. Chu, W. Xiong, W. Fu, W. Meers, X. Martinet, X. Wang, X. Wang, X. E. Tan, X. Xia, X. Xie, X. Jia, X. Wang, Y. Goldschlag, Y. Gaur, Y. Babaei, Y. Wen, Y. Song, Y. Zhang, Y. Li, Y. Mao, Z. D. Coudert, Z. Yan, Z. Chen, Z. Papakipos, A. Singh, A. Srivastava, A. Jain, A. Kelsey, A. Shajnfeld, A. Gangidi, A. Victoria, A. Goldstand, A. Menon, A. Sharma, A. Boesenberg, A. Baevski, A. Feinstein, A. Kallet, A. Sangani, A. Teo, A. Yunus, A. Lupu, A. Alvarado, A. Caples, A. Gu, A. Ho, A. Poulton, A. Ryan, A. Ramchandani, A. Dong, A. Franco, A. Goyal, A. Saraf, A. Chowdhury, A. Gabriel, A. Bharambe, A. Eisenman, A. Yazdan, B. James, B. Maurer, B. Leonhardi, B. Huang, B. Loyd, B. D. Paola, B. Paranjape, B. Liu, B. Wu, B. Ni, B. Hancock, B. Wasti, B. Spence, B. Stojkovic, B. Gamido, B. Montalvo, C. Parker, C. Burton, C. Mejia, C. Liu, C. Wang, C. Kim, C. Zhou, C. Hu, C.-H. Chu, C. Cai, C. Tindal, C. Feichtenhofer, C. Gao, D. Civin, D. Beaty, D. Kreymer, D. Li, D. Adkins, D. Xu, D. Testuggine, D. David, D. Parikh, D. Liskovich, D. Foss, D. Wang, D. Le, D. Holland, E. Dowling, E. Jamil, E. Montgomery, E. Presani, E. Hahn, E. Wood, E.-T. Le, E. Brinkman, E. Arcaute, E. Dunbar, E. Smothers, F. Sun, F. Kreuk, F. Tian, F. Kokkinos, F. Ozgenel, F. Caggioni, F. Kanayet, F. Seide, G. M. Florez, G. Schwarz, G. Badeer, G. Swee, G. Halpern, G. Herman, G. Sizov, Guangyi, Zhang, G. Lakshminarayanan, H. Inan, H. Shojanazeri, H. Zou, H. Wang, H. Zha, H. Habeeb, H. Rudolph, H. Suk, H. Aspegren, H. Goldman, H. Zhan, I. Damlaj, I. Molybog, I. Tufanov, I. Leontiadis, I.-E. Veliche, I. Gat, J. Weissman, J. Geboski, J. Kohli, J. Lam, J. Asher, J.-B. Gaya, J. Marcus, J. Tang, J. Chan, J. Zhen, J. Reizenstein, J. Teboul, J. Zhong, J. Jin, J. Yang, J. Cummings, J. Carvill, J. Shepard, J. McPhie, J. Torres, J. Ginsburg, J. Wang, K. Wu, K. H. U, K. Saxena, K. Khandelwal, K. Zand, K. Matosich, K. Veeraraghavan, K. Michelena, K. Li, K. Jagadeesh, K. Huang, K. Chawla, K. Huang, L. Chen, L. Garg, L. A, L. Silva, L. Bell, L. Zhang, L. Guo, L. Yu, L. Moshkovich, L. Wehrstedt, M. Khabsa, M. Avalani, M. Bhatt, M. Mankus, M. Hasson, M. Lennie, M. Reso, M. Groshev, M. Naumov, M. Lathi, M. Keneally, M. Liu, M. L. Seltzer, M. Valko, M. Restrepo, M. Patel, M. Vyatskov, M. Samvelyan, M. Clark, M. Macey, M. Wang, M. J. Hermoso, M. Metanat, M. Rastegari, M. Bansal, N. Santhanam, N. Parks, N. White, N. Bawa, N. Singhal, N. Egebo, N. Usunier, N. Mehta, N. P. Laptev, N. Dong, N. Cheng, O. Chernoguz, O. Hart, O. Salpekar, O. Kalinli, P. Kent, P. Parekh, P. Saab, P. Balaji, P. Rittner, P. Bontrager, P. Roux, P. Dollar, P. Zvyagina, P. Ratanchandani, P. Yuvraj, Q. Liang, R. Alao, R. Rodriguez, R. Ayub, R. Murthy, R. Nayani, R. Mitra, R. Parthasarathy, R. Li, R. Hogan, R. Battey, R. Wang, R. Howes, R. Rinott, S. Mehta, S. Siby, S. J. Bondu, S. Datta, S. Chugh, S. Hunt, S. Dhillon, S. Sidorov, S. Pan, S. Mahajan, S. Verma, S. Yamamoto, S. Ramaswamy, S. Lindsay, S. Lindsay, S. Feng, S. Lin, S. C. Zha, S. Patil, S. Shankar, S. Zhang, S. Zhang, S. Wang, S. Agarwal, S. Sajuyigbe, S. Chintala, S. Max, S. Chen, S. Kehoe, S. Satterfield, S. Govindaprasad, S. Gupta, S. Deng, S. Cho, S. Virk, S. Subramanian, S. Choudhury, S. Goldman, T. Remez, T. Glaser, T. Best, T. Koehler, T. Robinson, T. Li, T. Zhang, T. Matthews, T. Chou, T. Shaked, V. Vontimitta, V. Ajayi, V. Montanez, V. Mohan, V. S. Kumar, V. Mangla, V. Ionescu, V. Poenaru, V. T. Mihailescu, V. Ivanov, W. Li, W. Wang, W. Jiang, W. Bouaziz, W. Constable, X. Tang, X. Wu, X. Wang, X. Wu, X. Gao, Y. Kleinman, Y. Chen, Y. Hu, Y. Jia, Y. Qi, Y. Li, Y. Zhang, Y. Zhang, Y. Adi, Y. Nam, Yu, Wang, Y. Zhao, Y. Hao, Y. Qian, Y. Li, Y. He, Z. Rait, Z. DeVito, Z. Rosnbrick, Z. Wen, Z. Yang, Z. Zhao, Z. Ma, The llama 3 herd of models, 2024. URL: https://arxiv.org/abs/2407.21783. arXiv:2407.21783.

[84] S. Dadkhah, X. Zhang, A. G. Weismann, A. Firouzi, A. A. Ghorbani, The largest social media ground-truth dataset for real/fake content: Truthseeker, IEEE Transactions on Computational Social Systems 99 (2023) 1–15.

[85] T. Felber, Constraint 2021: Machine learning models for covid-19 fake news detection shared task, arXiv preprint arXiv:2101.03717 (2021).

[86] S. Sharma, R. Sharma, Identifying possible rumor spreaders on twitter: A weak supervised learning approach, in: 2021 International Joint Conference on Neural Networks (IJCNN), 2021, pp. 1–8. doi:10.1109/IJCNN52387.2021.9534185.

[87] J. Dougrez-Lewis, E. Kochkina, M. Arana-Catania, M. Liakata, Y. He, PHEMEPlus: Enriching social media rumour verification with external evidence, in: R. Aly, C. Christodoulopoulos, O. Cocarascu, Z. Guo, A. Mittal, M. Schlichtkrull, J. Thorne, A. Vlachos (Eds.), Proceedings of the Fifth Fact Extraction and VERification Workshop (FEVER), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 49–58. URL: https://aclanthology.org/2022.fever-1.6/. doi:10.18653/v1/2022.fever-1.6.

[88] P. Patwa, M. Bhardwaj, V. Guptha, G. Kumari, S. Sharma, S. PYKL, A. Das, A. Ekbal, M. S. Akhtar, T. Chakraborty, Overview of CONSTRAINT 2021 Shared Tasks: Detecting English COVID-19 Fake News and Hindi Hostile Posts, in: T. Chakraborty, K. Shu, H. R. Bernard, H. Liu, M. S. Akhtar (Eds.), Combating Online Hostile Posts in Regional Languages during Emergency Situation, Springer International Publishing, Cham, 2021, pp. 42–53.

[89] H.-T. Luong, H. Li, L. Zhang, K. A. Lee, E. S. Chng, LlamaPartialSpoof: An LLM-driven fake speech dataset simulating disinformation generation, in: ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2025, pp. 1–5.

[90] NLLB-Team, M. R. Costa-jussà, J. Cross, O. Çelebi, M. Elbayad, K. Heafield, K. Heffernan, E. Kalbassi, J. Lam, D. Licht, J. Maillard, A. Sun, S. Wang, G. Wenzek, A. Youngblood, B. Akula, L. Barrault, G. M. Gonzalez, P. Hansanti, J. Hoffman, S. Jarrett, K. R. Sadagopan, D. Rowe, S. Spruit, C. Tran, P. Andrews, N. F. Ayan, S. Bhosale, S. Edunov, A. Fan, C. Gao, V. Goswami, F. Guzmán, P. Koehn, A. Mourachko, C. Ropers, S. Saleem, H. Schwenk, J. Wang, No language left behind: Scaling human-centered machine translation, 2022. URL: https://arxiv.org/abs/2207.04672. arXiv:2207.04672.

[91] Y. Gong, H. Luo, J. Zhang, Natural language inference over interaction space, in: International Conference on Learning Representations, 2018. URL: https://openreview.net/forum?id=r1dHXnH6-.

[92] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, F. Wei, Multilingual E5 Text Embeddings: A Technical Report, 2024. URL: https://arxiv.org/abs/2402.05672. arXiv:2402.05672.

[93] T. Sellam, D. Das, A. Parikh, BLEURT: Learning robust metrics for text generation, in: D. Jurafsky, J. Chai, N. Schluter, J. Tetreault (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 7881–7892. URL: https://aclanthology.org/2020.acl-main.704/. doi:10.18653/v1/2020.acl-main.704.

[94] Y. Bengio, S. Clare, C. Prunkl, S. Rismani, M. Andriushchenko, B. Bucknall, P. Fox, T. Hu, C. Jones, S. Manning, N. Maslej, V. Mavroudis, C. McGlynn, M. Murray, C. Stix, L. Velasco, N. Wheeler, D. Privitera, S. Mindermann, D. Acemoglu, T. G. Dietterich, F. Heintz, G. Hinton, N. Jennings, S. Leavy, T. Ludermir, V. Marda, H. Margetts, J. McDermid, J. Munga, A. Narayanan, A. Nelson, C. Neppel, G. Ramchurn, S. Russell, M. Schaake, B. Schölkopf, A. Soto, L. Tiedrich, G. Varoquaux, A. Yao, Y.-Q. Zhang, L. Aguirre, O. Ajala, F. A. N. AlMalek, C. Busch, A. Carvalho, J. Collas, A. Gill, A. Hatip, J. Heikkilä, C. Johnson, G. Jolly, Z. Katzir, M. Kerema, H. Kitano, A. Krüger, A. McLysaght, O. Molchanovskyi, A. Monti, K. M. Lee, M. Nemer, N. Oliver, R. Pezoa, A. Plonk, J. Portillo, B. Ravindran, H. Riza, C. Rugege, H. Sheikh, D. Wong, Y. Zeng, L. Zhu, International AI Safety Report 2025: First Key Update: Capabilities and Risk Implications, 2025. URL: https://arxiv.org/abs/2510.13653. arXiv:2510.13653.

[95] xAI, Grok 4 model card, 2025. URL: https://data.x.ai/2025-08-20-grok-4-model-card.pdf.

[96] OpenAI, :, A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, A. Mądry, A. Baker-Whitcomb, A. Beutel, A. Borzunov, A. Carney, A. Chow, A. Kirillov, A. Nichol, A. Paino, A. Renzin, A. T. Passos, A. Kirillov, A. Christakis, A. Conneau, A. Kamali, A. Jabri, A. Moyer, A. Tam, A. Crookes, A. Tootoochian, A. Tootoonchian, A. Kumar, A. Vallone, A. Karpathy, A. Braunstein, A. Cann, A. Codispoti, A. Galu, A. Kondrich, A. Tulloch, A. Mishchenko, A. Baek, A. Jiang, A. Pelisse, A. Woodford, A. Gosalia, A. Dhar, A. Pantuliano, A. Nayak, A. Oliver, B. Zoph, B. Ghorbani, B. Leimberger, B. Rossen, B. Sokolowsky, B. Wang, B. Zweig, B. Hoover, B. Samic, B. McGrew, B. Spero, B. Giertler, B. Cheng, B. Lightcap, B. Walkin, B. Quinn, B. Guarraci, B. Hsu, B. Kellogg, B. Eastman, C. Lugaresi, C. Wainwright, C. Bassin, C. Hudson, C. Chu, C. Nelson, C. Li, C. J. Shern, C. Conger, C. Barette, C. Voss, C. Ding, C. Lu, C. Zhang, C. Beaumont, C. Hallacy, C. Koch, C. Gibson, C. Kim, C. Choi, C. McLeavey, C. Hesse, C. Fischer, C. Winter, C. Czarnecki, C. Jarvis, C. Wei, C. Koumouzelis, D. Sherburn, D. Kappler, D. Levin, D. Levy, D. Carr, D. Farhi, D. Mely, D. Robinson, D. Sasaki, D. Jin, D. Valladares, D. Tsipras, D. Li, D. P. Nguyen, D. Findlay, E. Oiwoh, E. Wong, E. Asdar, E. Proehl, E. Yang, E. Antonow, E. Kramer, E. Peterson, E. Sigler, E. Wallace, E. Brevdo, E. Mays, F. Khorasani, F. P. Such, F. Raso, F. Zhang, F. von Lohmann, F. Sulit, G. Goh, G. Oden, G. Salmon, G. Starace, G. Brockman, H. Salman, H. Bao, H. Hu, H. Wong, H. Wang, H. Schmidt, H. Whitney, H. Jun, H. Kirchner, H. P. de Oliveira Pinto, H. Ren, H. Chang, H. W. Chung, I. Kivlichan, I. O'Connell, I. O'Connell, I. Osband, I. Silber, I. Sohl, I. Okuyucu, I. Lan, I. Kostrikov, I. Sutskever, I. Kanitscheider, I. Gulrajani, J. Coxon, J. Menick, J. Pachocki, J. Aung, J. Betker, J. Crooks, J. Lennon, J. Kiros, J. Leike, J. Park, J. Kwon, J. Phang, J. Teplitz, J. Wei, J. Wolfe, J. Chen, J. Harris, J. Varavva, J. G. Lee, J. Shieh, J. Lin, J. Yu, J. Weng, J. Tang, J. Yu, J. Jang, J. Q. Candela, J. Beutler, J. Landers, J. Parish, J. Heidecke, J. Schulman, J. Lachman, J. McKay, J. Uesato, J. Ward, J. W. Kim, J. Huizinga, J. Sitkin, J. Kraaijeveld, J. Gross, J. Kaplan, J. Snyder, J. Achiam, J. Jiao, J. Lee, J. Zhuang, J. Harriman, K. Fricke, K. Hayashi, K. Singhal, K. Shi, K. Karthik, K. Wood, K. Rimbach, K. Hsu, K. Nguyen, K. Gu-Lemberg, K. Button, K. Liu, K. Howe, K. Muthukumar, K. Luther, L. Ahmad, L. Kai, L. Itow, L. Workman, L. Pathak, L. Chen, L. Jing, L. Guy, L. Fedus, L. Zhou, L. Mamitsuka, L. Weng, L. McCallum, L. Held, L. Ouyang, L. Feuvrier, L. Zhang, L. Kondraciuk, L. Kaiser, L. Hewitt, L. Metz, L. Doshi, M. Aflak, M. Simens, M. Boyd, M. Thompson, M. Dukhan, M. Chen, M. Gray, M. Hudnall, M. Zhang, M. Aljubeh, M. Litwin, M. Zeng, M. Johnson, M. Shetty, M. Gupta, M. Shah, M. Yatbaz, M. J. Yang, M. Zhong, M. Glaese, M. Chen, M. Janner, M. Lampe, M. Petrov, M. Wu, M. Wang, M. Fradin, M. Pokrass, M. Castro, M. O. T. de Castro, M. Pavlov, M. Brundage, M. Wang, M. Khan, M. Murati, M. Bavarian, M. Lin, M. Yesildal, N. Soto, N. Gimelshein, N. Cone, N. Staudacher, N. Summers, N. LaFontaine, N. Chowdhury, N. Ryder, N. Stathas, N. Turley, N. Tezak, N. Felix, N. Kudige, N. Keskar, N. Deutsch, N. Bundick, N. Puckett, O. Nachum, O. Okelola, O. Boiko, O. Murk, O. Jaffe, O. Watkins, O. Godement, O. Campbell-Moore, P. Chao, P. McMillan, P. Belov, P. Su, P. Bak, P. Bakkum, P. Deng, P. Dolan, P. Hoeschele, P. Welinder, P. Tillet, P. Pronin, P. Tillet, P. Dhariwal, Q. Yuan, R. Dias, R. Lim, R. Arora, R. Troll, R. Lin, R. G. Lopes, R. Puri, R. Miyara, R. Leike, R. Gaubert, R. Zamani, R. Wang, R. Donnelly, R. Honsby, R. Smith, R. Sahai, R. Ramchandani, R. Huet, R. Carmichael, R. Zellers, R. Chen, R. Chen, R. Nigmatullin, R. Cheu, S. Jain, S. Altman, S. Schoenholz, S. Toizer, S. Miserendino, S. Agarwal, S. Culver, S. Ethersmith, S. Gray, S. Grove, S. Metzger, S. Hermani, S. Jain, S. Zhao, S. Wu, S. Jomoto, S. Wu, Shuaiqi, Xia, S. Phene, S. Papay, S. Narayanan, S. Coffey, S. Lee, S. Hall, S. Balaji, T. Broda, T. Stramer, T. Xu, T. Gogineni, T. Christianson, T. Sanders, T. Patwardhan, T. Cunninghman, T. Degry, T. Dimson, T. Raoux, T. Shadwell, T. Zheng, T. Underwood, T. Markov, T. Sherbakov, T. Rubin, T. Stasi, T. Kaftan, T. Heywood, T. Peterson, T. Walters, T. Eloundou, V. Qi, V. Moeller, V. Monaco, V. Kuo, V. Fomenko, W. Chang, W. Zheng, W. Zhou, W. Manassra, W. Sheu, W. Zaremba, Y. Patil, Y. Qian, Y. Kim, Y. Cheng, Y. Zhang, Y. He, Y. Zhang, Y. Jin, Y. Dai, Y. Malkov, Gpt-4o system card, 2024. URL: https://arxiv.org/abs/2410.21276. arXiv:2410.21276.

[97] D. Guo, D. Yang, H. Zhang, J. Song, P. Wang, Q. Zhu, R. Xu, R. Zhang, S. Ma, X. Bi, X. Zhang, X. Yu, Y. Wu, Z. F. Wu, Z. Gou, Z. Shao, Z. Li, Z. Gao, A. Liu, B. Xue, B. Wang, B. Wu, B. Feng, C. Lu, C. Zhao, C. Deng, C. Ruan, D. Dai, D. Chen, D. Ji, E. Li, F. Lin, F. Dai, F. Luo, G. Hao, G. Chen, G. Li, H. Zhang, H. Xu, H. Ding, H. Gao, H. Qu, H. Li, J. Guo, J. Li, J. Chen, J. Yuan, J. Tu, J. Qiu, J. Li, J. L. Cai, J. Ni, J. Liang, J. Chen, K. Dong, K. Hu, K. You, K. Gao, K. Guan, K. Huang, K. Yu, L. Wang, L. Zhang, L. Zhao, L. Wang, L. Zhang, L. Xu, L. Xia, M. Zhang, M. Zhang, M. Tang, M. Zhou, M. Li, M. Wang, M. Li, N. Tian, P. Huang, P. Zhang, Q. Wang, Q. Chen, Q. Du, R. Ge, R. Zhang, R. Pan, R. Wang, R. J. Chen, R. L. Jin, R. Chen, S. Lu, S. Zhou, S. Chen, S. Ye, S. Wang, S. Yu, S. Zhou, S. Pan, S. S. Li, S. Zhou, S. Wu, T. Yun, T. Pei, T. Sun, T. Wang, W. Zeng, W. Liu, W. Liang, W. Gao, W. Yu, W. Zhang, W. L. Xiao, W. An, X. Liu, X. Wang, X. Chen, X. Nie, X. Cheng, X. Liu, X. Xie, X. Liu, X. Yang, X. Li, X. Su, X. Lin, X. Q. Li, X. Jin, X. Shen, X. Chen, X. Sun, X. Wang, X. Song, X. Zhou, X. Wang, X. Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. Zhang, Y. Xu, Y. Li, Y. Zhao, Y. Sun, Y. Wang, Y. Yu, Y. Zhang, Y. Shi, Y. Xiong, Y. He, Y. Piao, Y. Wang, Y. Tan, Y. Ma, Y. Liu, Y. Guo, Y. Ou, Y. Wang, Y. Gong, Y. Zou, Y. He, Y. Xiong, Y. Luo, Y. You, Y. Liu, Y. Zhou, Y. X. Zhu, Y. Huang, Y. Li, Y. Zheng, Y. Zhu, Y. Ma, Y. Tang, Y. Zha, Y. Yan, Z. Z. Ren, Z. Ren, Z. Sha, Z. Fu, Z. Xu, Z. Xie, Z. Zhang, Z. Hao, Z. Ma, Z. Yan, Z. Wu, Z. Gu, Z. Zhu, Z. Liu, Z. Li, Z. Xie, Z. Song, Z. Pan, Z. Huang, Z. Xu, Z. Zhang, Z. Zhang, Deepseek-r1 incentivizes reasoning in llms through reinforcement learning, Nature 645 (2025) 633–638.

[98] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, W. Chen, Lora: Low-rank adaptation of large language models, CoRR abs/2106.09685 (2021).

[99] J. Tyo, B. Dhingra, Z. C. Lipton, Valla: Standardizing and benchmarking authorship attribution and verification through empirical evaluation and comparative analysis, in: J. C. Park, Y. Arase, B. Hu, W. Lu, D. Wijaya, A. Purwarianti, A. A. Krisnadhi (Eds.), Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Nusa Dua, Bali, 2023, pp. 649–660. URL: https://aclanthology.org/2023.ijcnlp-main.43/. doi:10.18653/v1/2023.ijcnlp-main.43.

[100] Y. Shu, V. Lampos, Unsupervised hard negative augmentation for contrastive learning, 2024. URL: https://arxiv.org/abs/2401.02594. arXiv:2401.02594.

[101] A. Mulahuwaish, M. Osti, K. Gyorick, M. Maabreh, A. Gupta, B. Qolomany, CovidMis20: COVID-19 Misinformation Detection System on Twitter Tweets Using Deep Learning Models, in: Intelligent Human Computer Interaction: 14th International Conference, IHCI 2022, Tashkent, Uzbekistan, October 20–22, 2022, Revised Selected Papers, Springer-Verlag, Berlin, Heidelberg, 2022, p. 466–479. URL: https://doi.org/10.1007/978-3-031-27199-1_47. doi:10.1007/978-3-031-27199-1_47.