SAFEGEN: Safeguarding AI-generated Content Through Enhanced Watermarking

Aditya Karthikeyan
Teacher/Mentor: Mr. Gotwals & Mr. Egger
Research in Computational Science
North Carolina School of Science and Math
8 November 2024

Abstract

The rapid advancement of generative AI has significantly enhanced our interactions with digital content, creating an urgent need for precise AI detection to address serious security implications. Current detection tools, like GPTZero, face limitations, with error margins as high as 15%, and are often model-specific, which restricts their broader applicability. These tools also rely on subjective criteria, introducing biases that may favor particular writing styles. SAFEGEN overcomes these challenges with an innovative approach that combines watermarking techniques and specialized context guiding to the Large Language Model. By classifying prompts into 15 distinct topic areas and embedding a unique Red-List of semantically rare words tailored to each category, SAFEGEN offers a robust methodology for the verification of AI-generated text. In trials within the "Technology and Science" domain, SAFEGEN achieved 93% accuracy, with an F1 score of 0.945, a true positive rate of 94%, and a false positive rate of 8%, demonstrating its precise detection capabilities. A threshold of six Red-List words per 150 words optimizes sensitivity and specificity, ensuring effective watermarking while preserving natural text flow. By objectively measuring AI-generated content, SAFEGEN surpasses the subjectivity of existing tools and establishes a scalable, reliable framework for transparent AI usage, strengthening accountability and the integrity of digital information across various applications.

Acknowledgments

I would like to thank Dr. Daniel Egger for his unwavering support throughout this year of research. I am also grateful to Mr. Bob Gotwals, my Research in Computational Science instructor, for his instruction in building fundamental research skills. I also would like to thank my family for their encouragement of my journey as a young scientist.

1. Introduction

The rapid growth of Large Language Models (LLMs) and AI-based text generation in the past three years has transformed how we interact with language (Doshi and Hauser, 2024). Tools like ChatGPT, Gemini, and Claude AI have expanded access to information and fundamentally altered our integration of AI into daily life. The rise of generative AI technologies has led to an unprecedented volume of content being produced automatically, resulting in millions of instances of LLM-generated content being created each year (Amrit and Singh, 2022; Kerry et al., 2024). With such widespread use, ensuring the accountability and traceability of generated content has become crucial, especially given the risks of misinformation, ethical breaches, and copyright concerns that can arise from the misuse of AI-generated content (Boenisch, 2021; Zhang et al., 2020).

One of the significant challenges in the current landscape of generative AI is detecting AI-generated text effectively. Tools such as GPTZero, which are designed to identify AI-generated content, face considerable challenges, including high error margins of up to 15% (Alavi and Westerman, 2023). These errors lead to unreliable detection, which can have unintended consequences, such as the misidentification of genuine human-authored content. This issue has particularly affected academic institutions, where innocent students' work has been inaccurately flagged as AI-generated, prompting policy changes and raising concerns about the fairness and accuracy of these detection mechanisms (Anderson et al., 2023). Furthermore, existing detection tools often rely on simple metrics like stylistic analysis, which are subjective and fail to capture the nuanced differences between human and AI-generated text (Boenisch, 2021).

The limitations of current detection methods underscore the need for a more sophisticated approach that can accurately identify AI-generated content while avoiding biases. SAFEGEN (Safeguarding AI-generated Content through Enhanced Watermarking) proposes a novel framework that addresses the deficiencies of existing tools by introducing model-level modifications through context guiding and traditional watermarking techniques (Amrit and Singh, 2022). By categorizing content into distinct topic areas and embedding certain domain-specific words into the model's output, SAFEGEN creates a subtle yet effective watermark within the generated text. Unlike traditional methods that attempt to replicate human writing, SAFEGEN offers an objective and quantifiable method for AI detection.

The concept of watermarking, as applied in SAFEGEN, is similar to digital watermarking in multimedia, where hidden information is embedded within the content to verify its authenticity and source (Boenisch, 2021; Mendonca et al., 2023). In the context of AI-generated text, this technique involves embedding specific, semantically rare words that are tailored to the content's domain. These Red-List words are selected based on the relative rarity and relevance of the particular word against a corpus specific

to its topic area, ensuring that the watermark remains unobtrusive while providing a reliable indicator of the text's origin. This method enhances the ability to detect AI-generated content while ensuring that the text retains coherence and quality, preserving its suitability for practical applications (Zhang et al., 2020; Li et al., 2023).

SAFEGEN leverages a BERT (Bidirectional Encoder Representations from Transformers) model to classify the text and determine the appropriate modifications for watermarking (Acheampong et al., 2021). BERT, a standard natural language processing model, is pre-trained on a diverse corpus of text, allowing it to understand language context and nuances effectively. By utilizing BERT for text classification, SAFEGEN can accurately categorize prompts into one of fifteen distinct topic areas, each with its own set of Red-List words, from which words are chosen to be embedded into the generated text. This domain-specific watermarking ensures that the detection process is both targeted and effective, significantly improving the reliability of AI-generated content detection.

In addition to enhancing the detection of AI-generated content, SAFEGEN also addresses the issue of intellectual property protection in the realm of generative AI. As more individuals and companies create fine-tuned versions of existing open-source models for domain-specific applications, the need for a mechanism to protect the outputs of these models becomes evident (Li et al., 2023). SAFEGEN provides a potential solution by embedding digital watermarks within the generated text, allowing creators to establish ownership and protect their intellectual property. This "digital copyright" for AI-generated content ensures that creators can maintain control over their work and prevents unauthorized use or modification of the content.

The implementation of SAFEGEN involves several key components, starting with the classification of prompts into distinct topic areas using the BERT model. Then a non-random increase in the presence of Red-List words from the topic area of the prompt is embedded into the generated output text. This is achieved through context guiding of the LLM before it provides an output to the prompt. Given the nuanced nature of the selected words, the watermarked output for a given prompt is nearly indistinguishable from that of an unwatermarked output. The simplified flowchart below lays the framework for SAFEGEN:

Figure 1: Flowchart representation of SAFEGEN

The goal of this project is to establish a dependable system for digitally watermarking AI-generated text, ensuring traceability and enhancing the authenticity of content across various domains. By combining standard natural language processing techniques with unique application watermarking methods, SAFEGEN offers a nominal-cost, scalable approach to enable more accountable and transparent use of generative AI.

2. Computational Methodolgy

To enhance the safety and accountability of AI-generated text, SAFEGEN implements an objective method that involves introducing a subtle modification—or implant—into the output of the language model. The detection of this implant allows for accurate identification of AI-generated content. This implant must remain as inconspicuous as possible to ensure that the language model's responses are of high quality and align naturally with the given prompts. This approach provides an objective means of detection, contrasting with existing tools that rely on subjective comparisons to human language.

SAFEGEN achieves this modification by developing a defined Red-List, for each of the fifteen topic areas, which consists of words that are semantically associated with that topic area but are relatively uncommon in everyday language. A statistically significant increase, or above a preset threshold, presence of these words in the output of a model indicates that the text was generated by AI.

2.1 **Bert Classification Model**

A BERT model, or a Bidirectional Encoder Representation from Transformers, was developed by Google to help improve upon Natural Language Processing (NLP) tasks. Its core innovation lies in its ability to process words in relation to all the other words in a sentence, rather than one at a time (Devlin et al., 2019). This helps it understand the larger context of the sentence and thereby make accurate predictions. BERT operates based on the transformer architecture, which uses mechanisms called attention heads to focus on different parts of a sentence simultaneously. Therefore, instead of reading input text sequentially (left to right or right to left), BERT reads the entire sequence of words at once. Moreover, BERT is pre-trained on a vast corpus of text from the internet, which helps it develop a general understanding of language. This ensures the model is familiar with common human language syntax and semantics.

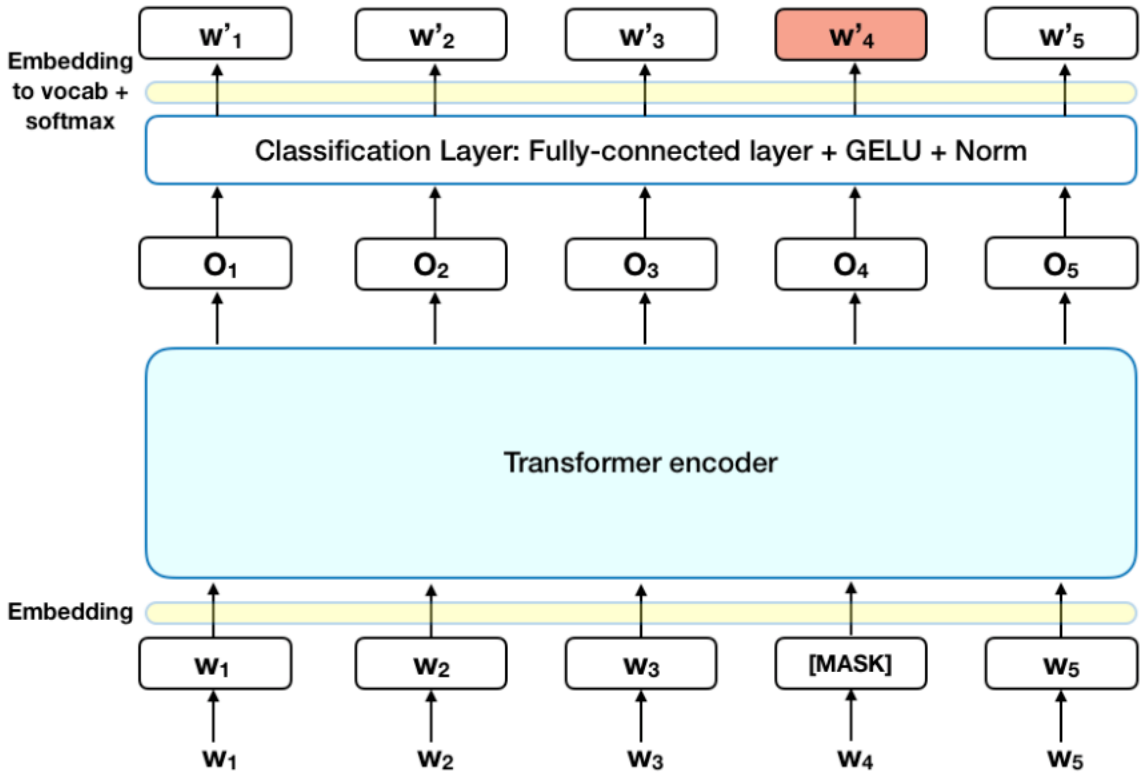

Figure 2: Simplified Schematic of BERT Architecture. Figure obtained from (Horev, 2018).

In Figure 2, BERT's input comprises token embeddings, segment embeddings, and positional embeddings, with a maximum input length of 512 tokens. Each sentence input is processed into a series of token embeddings, with each token represented as either a word or subword. During training, BERT acquires contextualized embeddings by predicting masked tokens and establishing relationships between sentences. The model's output consists of contextual embeddings for each token, which are utilized in downstream tasks such as classification or named entity recognition through task-specific layers added to the architecture.

The BERT model assists with the first stage of implementing an effective watermark - classifying the given prompt into the most appropriate topic area given all topics areas. SAFEGEN uses a cased BERT model; where "cased" simply means that it can make distinctions between upper and lower case letters. This particular model had 24 layers and over 336 million parameters; enabling it to achieve high levels of accuracy when making predictions. Having accurate predictions is particularly important in the case of SAFEGEN, as the topic-area classification determines which Red-List of words are to be invoked by the LLM model before it provides an output. This project chose to classify prompts into one of **15 distinct topic-areas**. The use of fifteen topic areas allows for the encapsulation of a wide range

of prompts. Some categorization is essential because the words used to modify the output must be semantically appropriate to the prompt's overall topic area; otherwise, the manipulations could become conspicuous, revealing the watermarking. The exact number of topic areas can of course be adjusted without affecting the basic system architecture, but it appeared to maintain good performance while keeping the watermark undetectable.

SAFEGEN uses the following fifteen topic areas:

Table 1: SAFEGEN Classification Categories

No.	Category Name	No.	Category Name
1	Business and Finance	8	History and Culture
2	Education and Academics	9	Legal and Ethics
3	Entertainment and Arts	10	News and Current Events
4	Environment and Geography	11	Politics and Government
5	Fashion and Lifestyle	12	Social Issues and Advocacy
6	Food and Cooking	13	Sports
7	Health and Wellness	14	Technology and Science
		15	Travel and Leisure

We achieved significant improvement in performance while fine-tuning the pre-trained model using 5000 distinct text snippets for each classification category. The dataset was generated using existing open-source LLMs (ChatGPT 4, Gemini, and Claude). Each system was prompted to generate a paragraph with a mean length of about 100 words in each of the 15 topic areas. This allows for the creation of a wide array of text within each domain. This data was then combined into a single DataFrame object and each category was converted into a numerical code to facilitate the model's training process. Finally, the model's output labels were set to match the unique categories in our dataset. The distribution of the data was as follows:

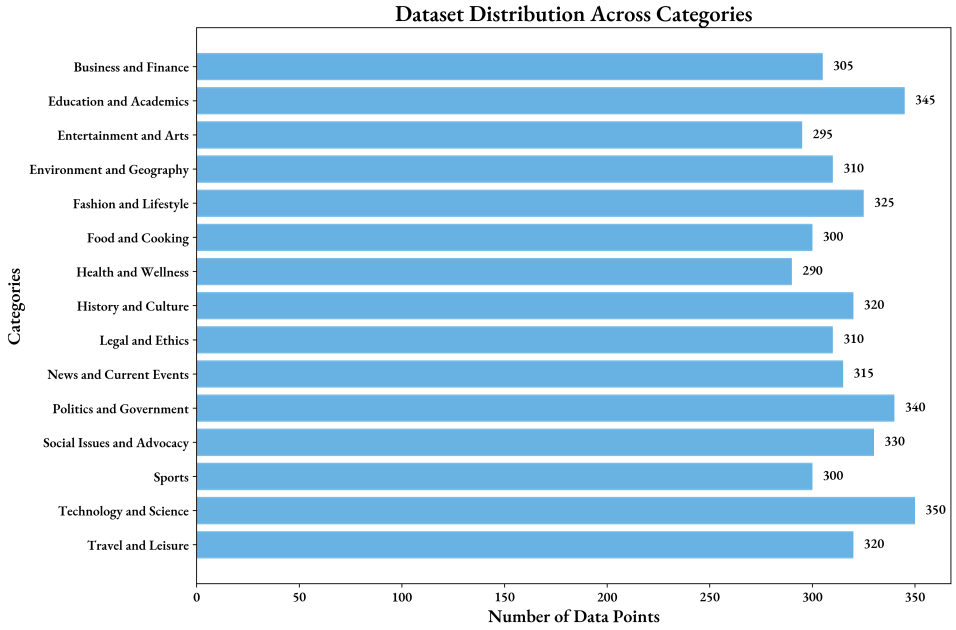

Figure 3: Distribution of BERT fine-tuning Data Set. Figure created by the author.

This fine-turned BERT model can then help make accurate classifications of a new piece of text into one of the 15 defined categories above. Given a prompt, it first makes a probabilistic prediction for each of the 15 categories and applies a built-in argmax function on TensorFlow. This enables the model to normalize these probabilities and pick the one that corresponds to the model's most confident classification.

2.2 Text Corpus Definition

In constructing corpora to generate fifteen different Red-list, one for each topic area previously defined, a formalized selection process was implemented to ensure that the text fits the requirements. For the purpose of demonstrating a proof of concept, the rest of the steps to create a successfully watermarked text were done downstream for one of the classifications: **Technology and Science**. The corpus comprised 500 research articles sourced primarily from Science Direct, a leading academic repository. These articles were chosen based on several criteria to maintain its core focus on "Technology and Science." First, all selected articles were published within the past five years, ensuring the content's contemporary relevance. Second, articles were only included if they were tagged as Physical Sciences and Engineering, Life Sciences, or Health Sciences — fields defined by Science Direct. Additionally, each article needed to be classified explicitly as a research article, thereby guaranteeing the inclusion of rigorous, peer-reviewed scientific studies. These steps ensured that the articles picked included the right vocabulary to aid in the construction of the Red-List in a manner that enables the inclusion of words from the list to modify the

LLM output in a manner that is not detectable to the end-user.

The subsequent step involves extracting the word frequencies from these articles to perform a TF-IDF analysis (further explained in section 2.3). This process is performed using a Python script that uses the `PyPDF2` library to read and extract text from PDF files. The extracted text is then subjected to a cleaning process where all non-dictionary words and non-alphabetic characters are filtered out. This cleaning ensures that the analysis is performed on linguistically valid and relevant terms only. The `nltk` library, specifically the `words.words()` function, providing a comprehensive set of English words that served as a filter to retain meaningful content.

Each cleaned text is stored as an individual text file, ensuring that the original structure and content are preserved for subsequent analysis. This approach allows for a streamlined analysis process, where each text file represents a single document in the corpus.

2.3 **TF-IDF Analysis**

Once the corpus is built, to statistically define the red list for each classification, this project uses a modified version of the Term Frequency-Inverse Document Frequency (TF-IDF) analysis. Traditionally, TF-IDF is a statistical measure used to evaluate the relevance of a word in a document relative to a collection (or corpus) of documents. It works by assigning a score to each word based on how frequently it appears in a document (Term Frequency, TF) and how rare that word is across the entire corpus (Inverse Document Frequency, IDF).

The term frequency (TF) is the ratio of the number of times a word appears in a document to the total number of words in that document:

$$TF(t, d) = \frac{\text{Number of times term } t \text{ appears in document } d}{\text{Total number of terms in document } d} \tag{1}$$

Inverse Document Frequency (IDF) measures the rarity of a word across all documents in the corpus:

$$IDF(t, D) = \log\left(\frac{N}{|\{d \in D : t \in d\}|}\right) \tag{2}$$

Where:

- N is the total number of documents in the corpus,
- $|\{d \in D : t \in d\}|$ is the number of documents in which the term t appears.

The final TF-IDF score for each term t in document d is calculated by multiplying the TF and IDF values:

$$TF - IDF(t, d, D) = TF(t, d) \times IDF(t, D) \tag{3}$$

However, following this approach, we get a TF-IDF value per word per paragraph, and while this provides valuable information regarding how "relevant" a word is within a document in the context of the corpus, it does not fully address what is needed to create the Red-List. This project needs one value for every word in the corpus to statically make an inference on what words best fit our definition of the Red-List. For this purpose, a modified TF-IDF calculation was performed, resulting in adjustments in Term Frequency (TF). The calculations for the modified TF-IDF are as follows:

Modified Term Frequency (TF_{mod}) measures the occurrence of a term across the entire corpus rather than in a single document, thereby giving a broader view of the term's relevance throughout all texts.

$$TF_{mod}(t, D) = \frac{\text{Number of times term } t \text{ appears in the entire corpus } D}{\text{Total number of terms in the entire corpus } D} \tag{4}$$

Inverse Document Frequency remains unchanged and is used to offset the TF by diminishing the weight of terms that occur very frequently across many documents, thereby balancing out extremely common terms.

$$IDF(t, D) = \log\left(\frac{N}{|\{d \in D : t \in d\}|}\right) \tag{5}$$

Where:

- N is the total number of documents in the corpus,
- $|\{d \in D : t \in d\}|$ is the number of documents in which the term t appears.

Therefore, these combine to form the modified definition of the TF-IDF that is used to select the Red-List for each classification. The final modified TF-IDF score is calculated as follows:

$$TF - IDF_{mod}(t, D) = TF_{mod}(t, D) \times IDF(t, D) \tag{6}$$

Using this formula we compute a TF-IDF score for every work in the corpus, that tells us the relative importance of that work in the corpus defined. This valuable information can then be translated into our Red-List by setting thresholds to pick words that are optimal for watermarking.

Performing the above-defined analysis on the corpora created for the topic area of Technology and Science, the following data table is produced.

Table 2: Selected TF-IDF Scores (Rarest to Most Frequent)

Word	TF-IDF
technology	0.008079
energy	0.005204
digital	0.003853
research	0.003656
⋮	
subsoil	0.000002
supportable	0.000002
substantiality	0.000002
subsistence	0.000002

2.4 Red-List definition

A systematic approach was used to select words for the Red-List that effectively balanced semantic value and rarity. To identify suitable words for the list, the entire distribution of TF-IDF scores was examined, and after experimentation, final thresholds were chosen to select words between the 94th to 97th percentiles in terms of rarity. The decision to use this specific percentile range was driven by two key considerations:

1. **Balancing Rarity and Presence**: The strategy of the Red-List is to introduce into the prompt response words that are semantically associated with the topic area of the prompt but are not overly common. Words below the 94th percentile were deemed too frequent within the corpus, reducing their effectiveness as statistical indicators of AI watermarking. Conversely, words above the 97th percentile were often too rare, leading to potential semantic awkwardness or incoherence when integrated into the generated text, such words are potentially detectable as a modification introduced for the purpose of watermarking. By focusing on the 94th to 97th percentile range, we ensured that selected words maintained a balance: they were distinct enough to serve as reliable markers while still being common enough to blend naturally into the text.

2. **Optimal Semantic Value**: The Red-List's effectiveness depends not only on the rarity of words but also on their semantic alignment with the generated text. Words in the 94th to 97th percentile are typically associated with higher semantic value, as they are uncommon enough to be noticeable but still contextually appropriate within the target topic area. This characteristic makes them ideal for watermarking, as they integrate smoothly into the text without causing disruptions to coherence or

readability.

The selection process for the Red-List involved two steps. First, words within the 94th to 97th percentile range were filtered from the overall corpus. Next, a random sampling method was applied to select 100 words from this subset. The use of random sampling was critical for several reasons:

- **Prevents Predictability**: Random selection of words within the defined percentile range reduces the predictability of the Red-List, enhancing the watermark's robustness. If the selection were purely deterministic, it could increase the risk of identifying and removing the watermark.
- **Time-based Adaptability**: The implementation of random sampling also enables the periodic updating of the Red-List at fixed intervals, such as daily or weekly. This adaptation ensures that the watermark remains unpredictable. Additionally, by modifying the Red-List at predetermined intervals, SAFEGEN introduces an additional dimension of temporal traceability. This enables the detection mechanism to not only determine whether a text was AI-generated but also to infer the specific time period of its generation. Such temporal markers enhance accountability by linking generated content to defined timeframes, thereby increasing the overall efficacy and precision of the watermark.

Overall the Red-List selection criteria can be summarized as follows:

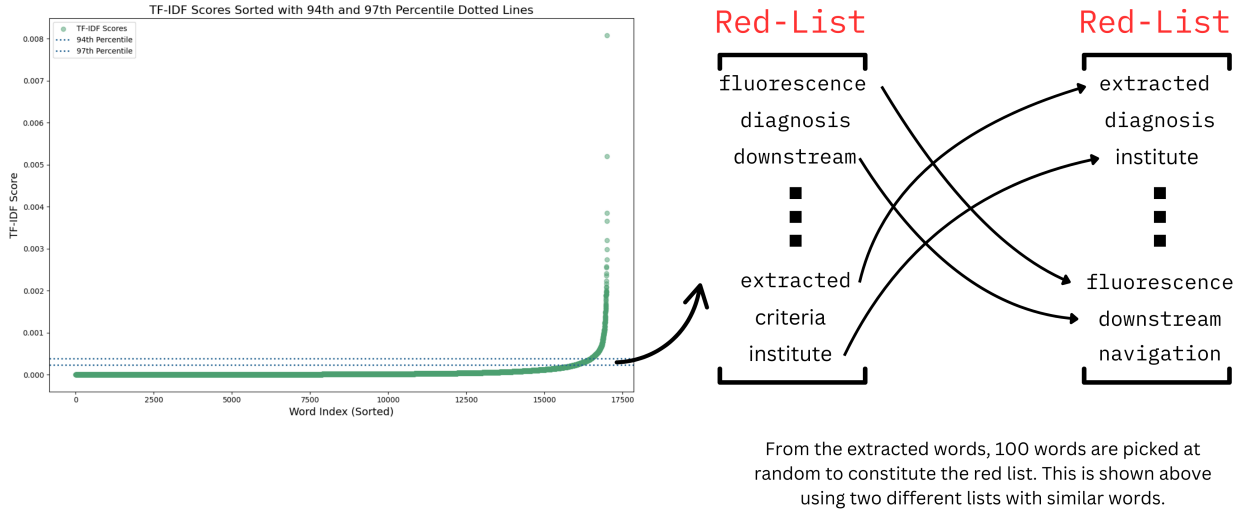

Figure 4: Schematic representation of Red-List Creation. Figure created by the author.

The approach outlined here ensures that the Red-List remains subtle yet effective, integrating seamlessly into AI-generated text while providing a reliable means of detection. By selecting words based on a combination of statistical filtering and random sampling, this method achieves a robust watermark that maintains coherence, preserves semantic value, and can be adapted to provide temporal information about the content.

2.5 **Context Guiding the Large Language Model**

The final stage of implementing the watermarking framework involves context guiding the Large Language Model (LLM) to introduce the Red-List words effectively into the generated text. Context guiding refers to a process in which the LLM is prompted to adjust its output based on predefined constraints — here, the Red-List words — without compromising the naturalness of the response. This process involves two primary elements: prompt engineering and semantic coherence.

The prompt engineering component aims to influence the LLM's response patterns by incorporating contextual hints that increase the likelihood of generating outputs containing Red-List words. This is achieved through a modified prompt, not visible to the end user, that forces the model to use the words from the Red-List more often. Through the prompt, the model's weights on the tokens corresponding to those of the Red-List words are artificially raised, which ultimately causes the shift in output pattern.

The semantic coherence, is essential for maintaining the natural flow of the generated text while embedding watermarking words. This involves using prompt strategies that align with the model's training distribution, ensuring that the selected Red-List words are used in a way that reflects common usage patterns. By preserving semantic coherence, the method retains the model's original intent while still integrating the desired watermark. A sample prompt structure to achieve this is shown below:

```
[Red-List....]

Answer this question in about 150 words (answer in a simple level of complexity and in one paragraph). I
    want you to do so by using as many of the above "Red-List" words as possible. Make sure the paragraph
    still makes sense, while also maximizing the amount of "Red-List" words.

(Question)
```

The context guiding the LLM for watermarking through prompt engineering and semantic coherence is essential for achieving robust AI detection. By drawing inspiration from established prompt engineering techniques and adapting them to the watermarking domain, the SAFEGEN framework enhances both the traceability and the authenticity of AI-generated content.

3. Results and Discussion

The implementation of the SAFEGEN watermarking framework demonstrated promising results, showcasing its potential to enhance the traceability and accountability of AI-generated content. The findings from this study highlight the efficacy of using a modified watermarking technique that combines a BERT-based classification model and context-guided of the large language model, to achieve a robust paradigm shift in AI-detection.

3.1 **BERT Model Training and Evaluation**

To ensure robust classification accuracy in watermarking AI-generated content, the BERT-based model was trained with a focus on optimizing both training and validation performance. The training process was monitored across multiple steps, as illustrated in Figure 5, which displays the progression of training and validation loss alongside accuracy improvements over time.

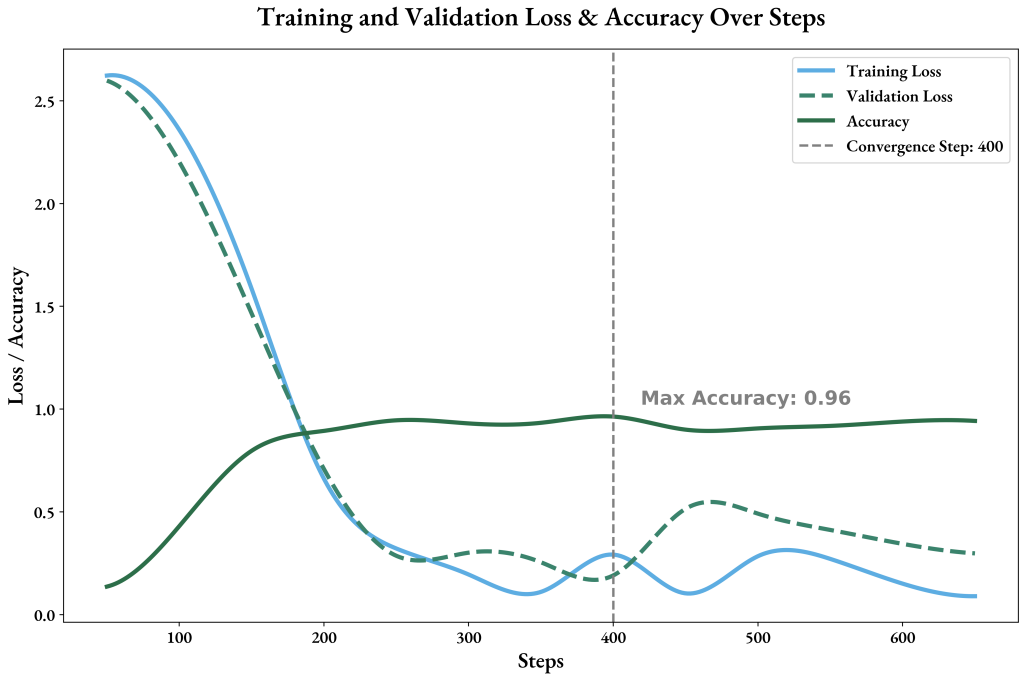

Figure 5: Training and Validation Loss & Accuracy Over Steps. This plot illustrates the convergence of training and validation losses along with accuracy improvements, showcasing the model's learning curve. Figure created by the author.

The plot in Figure 5 reveals critical insights into the model's learning dynamics and convergence behavior. The solid blue line represents the **training loss**, while the dashed green line denotes the **validation loss**. These loss metrics provide a quantitative measure of error during the training process, calculated by comparing the model's predictions with the true labels. Initially, both training and validation

losses are high, reflecting the model's early-stage struggle to capture meaningful patterns in the data.

As training progresses, a steady decline in **training loss** is observed, indicating that the model is effectively learning from the training data and adjusting its parameters to minimize errors. By approximately step 200, the training loss has dropped significantly, signaling that the model is beginning to fit the training data well. Notably, the **validation loss** follows a similar decreasing trend initially, which suggests that the model's learning is generalizing effectively to unseen data and avoiding overfitting at this stage.

The **convergence point** occurs around **step 400**, marked by a dashed vertical line in the plot. This convergence step represents the optimal training stage, where the model achieves a balance between minimizing training loss and maintaining low validation loss. Beyond this point, both training and validation losses stabilize, with minor fluctuations. This stability implies that the model has reached an equilibrium state, where further training does not significantly improve performance and may even risk overfitting if extended excessively.

Accompanying the loss curves, the green solid line in Figure 5 represents the **accuracy** metric over time, providing an assessment of the model's predictive performance. Accuracy measures the proportion of correctly classified instances relative to the total, offering an intuitive metric for evaluating model efficacy. The accuracy curve shows a sharp rise in the initial stages, aligning with the decreasing loss, as the model rapidly gains competence in identifying patterns in the data. As training progresses, accuracy plateaus near the maximum observed value of 0.96, reflecting the model's peak performance capability. This high level of accuracy indicates that the model can reliably classify inputs within its defined task, reinforcing the robustness of the SAFEGEN watermarking framework.

The **final accuracy value of 0.96**, achieved near the convergence step, highlights the effectiveness of the model's learning process, balancing high classification performance with controlled generalization. The alignment between training and validation loss trends suggests that the model avoids significant overfitting, as evidenced by the similar trajectories of both losses after convergence. This trend underscores the model's robustness, ensuring that it performs well not only on the training data but also on unseen validation data.

In summary, the training and validation loss convergence, combined with the high accuracy score, affirms the model's suitability for the classification tasks required by SAFEGEN. The convergence at step 400 and the high accuracy achieved provide confidence in the model's ability to input prompts into topic areas consistently, supporting the overall objectives of the SAFEGEN framework.

3.2 Impact of Context-Guided Watermarking on Red-List Word Integration

We notice the effectiveness of the context-guided large language approach by comparing the presence of red-list words in watermarked and unwatermarked paragraphs. This method aims to increase the integration of predefined Red-list words into the generated text, allowing for an objective measure for detecting AI-generated content. By guiding the model's output to incorporate a specific set of domain-relevant terms, we aim to achieve a robust watermark that is detectable yet unobtrusive, ensuring the generated text remains natural and coherent.

To assess the impact of watermarking, we compared the frequency distribution of unique Red-list words across paragraphs generated by the model before and after the watermarking process. Figures 6 and 7 show histograms representing the number of **unique** Red-List words in each paragraph for both pre-watermarking and post-watermarking cases. These distributions allow us to observe the effect of context guidance on embedding Red-List words.

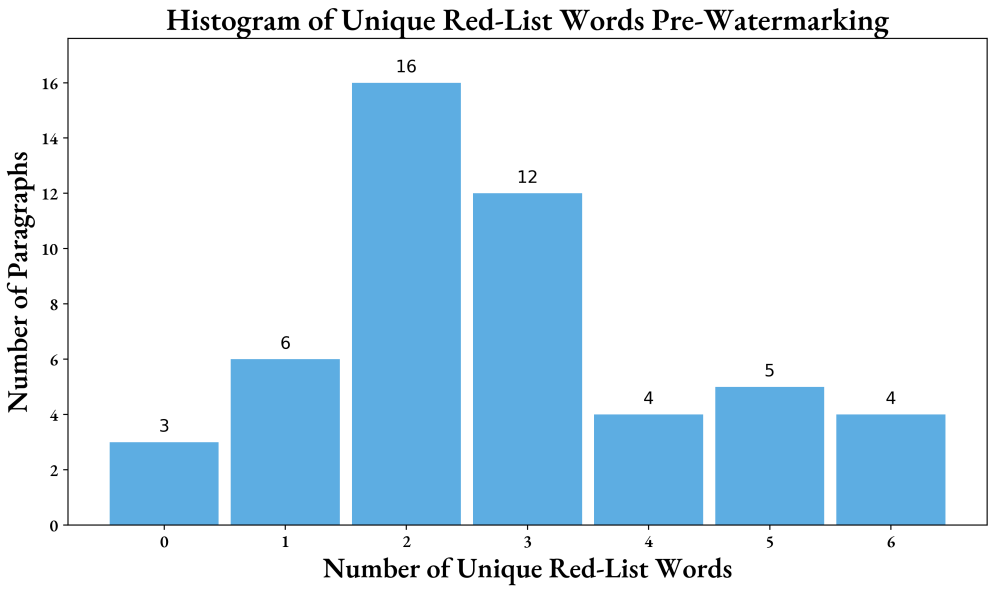

Figure 6: **Histogram of Unique Red-List Words Pre-Watermarking.** This histogram shows the distribution of unique Red-List words across paragraphs before the watermarking process was applied. Figure created by the author.

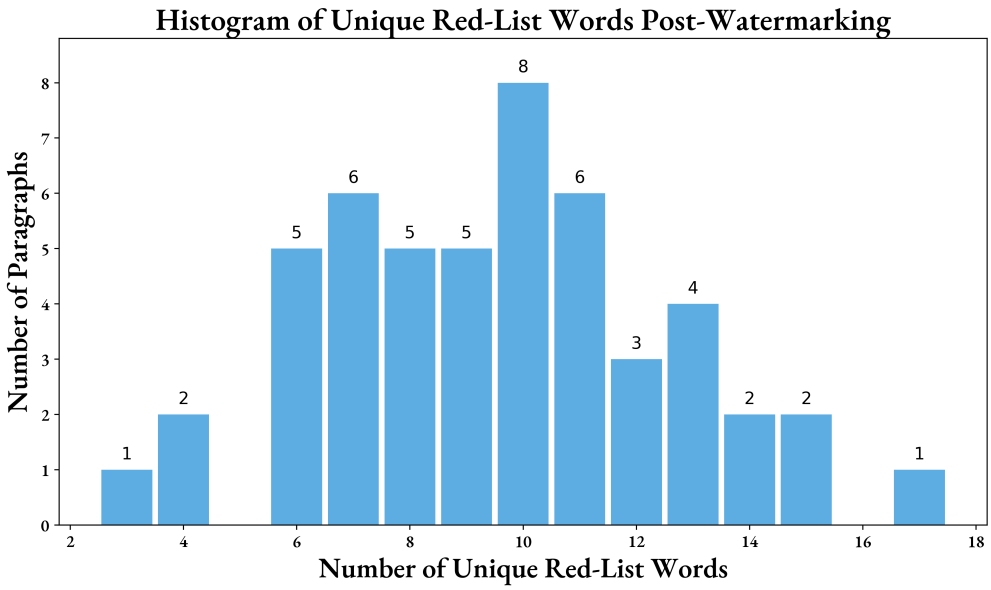

Figure 7: **Histogram of Unique Red-List Words Post-Watermarking.** This histogram shows the distribution of unique Red-List words across paragraphs after the watermarking process was applied, indicating a marked increase in the presence of Red-List words. Figure created by the author.

In Figure 6, the distribution of unique Red-List words before watermarking reveals a relatively low occurrence of these words across paragraphs. The majority of paragraphs contain between **0 and 3 unique Red-List words**, with very few paragraphs exceeding this range. This pattern reflects the natural distribution of these domain-specific terms without any external guidance. The low density of Red-List words suggests that, without intentional embedding, the model does not frequently use these words in its standard output, resulting in a sparse presence across generated paragraphs.

In contrast, Figure 7 illustrates the distribution of unique Red-List words following the application of context-guided watermarking. Here, a significant shift in the distribution is observed, with most paragraphs containing between **8 and 12 unique Red-List words**. This marked increase in Red-List word presence demonstrates the effectiveness of context-guided prompting, as the model has been successfully influenced to incorporate a greater number of predefined terms within its output. The central range of 8 to 12 unique words per paragraph suggests a well-distributed watermark, with sufficient density to ensure detectability while maintaining the semantic integrity of the text.

Comparing these two histograms highlights the effectiveness of context guidance in embedding Red-List words. The pre-watermarking distribution was heavily skewed toward lower counts of unique Red-List words, whereas the post-watermarking distribution shows a more balanced integration, with the peak frequency shifted toward higher counts. This shift is evidence that context-guided watermarking

enables the controlled addition of specific terms without disrupting the text's readability or flow.

This increase in the density of Red-List words has multiple implications. First, it enhances the watermark's detectability, making it easier to identify AI-generated content. The clear shift in Red-List word presence from the pre- to post-watermarking stages demonstrates that SAFEGEN's context-guided approach effectively embeds traceable markers within the generated text. Secondly, the structured distribution of Red-List words across paragraphs indicates that the watermark remains subtle, preserving the text's natural quality, which is essential for maintaining reader trust and usability in real-world applications.

Overall, these findings affirm that the context-guided watermarking approach meets the goals of SAFEGEN by embedding a detectable yet unobtrusive watermark into AI-generated text. This approach offers a promising solution for tracking and verifying AI-generated content, with potential applications across a variety of domains where accountability and transparency in content generation are paramount.

3.3 Confusion Matrix Evaluation of Watermark Detection Performance

To evaluate the performance of watermark detection in the SAFEGEN framework, we analyzed the model's output using a confusion matrix (Figure 8) with a threshold of six Red-List words per 150 words. This threshold was set to distinguish between watermarked and non-watermarked paragraphs accurately, aiming to minimize misclassifications. The confusion matrix provides a detailed breakdown of the model's predictive performance, capturing true positives, true negatives, false positives, and false negatives. Additionally, we present key evaluation metrics in Table 3, formatted to highlight the effectiveness of our watermarking approach.

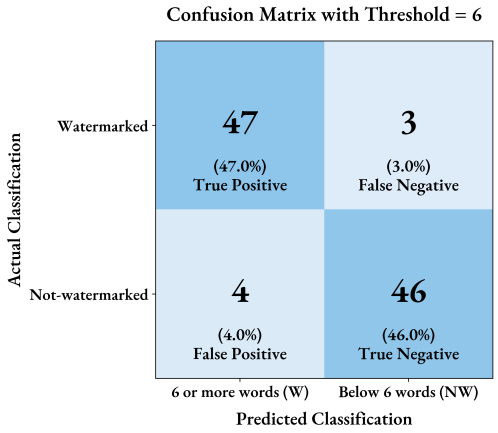

Figure 8: **Confusion Matrix with Threshold = 6.** This matrix shows the classification results for watermarked (W) and non-watermarked (NW) paragraphs based on the defined threshold. Figure created by the author.

Metrics	SAFEGEN Model
Type I Error (False Positive Rate)	0.04
Type II Error (False Negative Rate)	0.03
False Discovery Rate (FDR)	0.078
Accuracy	0.93
F1 Score	0.945
True Positive Rate (Sensitivity)	0.94

Table 3: **Evaluation Metrics for Watermark Detection**

Table 3 details several performance metrics, each of which quantifies a specific aspect of watermark detection effectiveness:

- **F1 Score**: At 0.945, the F1 score reflects a strong balance between precision and recall, indicating effective watermark detection.
- **Type I Error (False Positive Rate)**: The false positive rate is 4%, showing a low tendency to misclassify non-watermarked content as watermarked.
- **Type II Error (False Negative Rate)**: The false negative rate is 3%, indicating a low rate of missed watermark detection.
- **False Discovery Rate (FDR)**: With an FDR of 7.8%, this metric represents the proportion of incorrect watermark classifications among the predicted watermarked content.
- **True Positive Rate (Sensitivity)**: At 94%, this high sensitivity value demonstrates the model's accuracy in identifying watermarked text.
- **Overall Accuracy**: Achieving an accuracy of 93%, the model demonstrates reliable classification of watermarked and non-watermarked content.

The threshold of six Red-List words per 150 words was determined through iterative experimentation and analysis. Our primary goal in selecting this threshold was to balance sensitivity (true positive rate) and specificity (true negative rate) in the watermark detection process. We conducted a series of trials where the threshold was incrementally adjusted, observing the effects on Type I and Type II error rates. Lower thresholds resulted in higher Type I errors, where non-watermarked paragraphs were falsely classified as watermarked. Conversely, higher thresholds led to an increase in Type II errors, where watermarked paragraphs were incorrectly classified as non-watermarked.

After thorough analysis, we identified that six Red-List words per 150 words offered an optimal balance, minimizing both false positives and false negatives while maintaining a high true positive rate.

This threshold ensures that the watermark remains subtle yet detectable, preserving the natural quality of the text while embedding a traceable marker. This careful calibration of the threshold supports SAFEGEN's goal of creating a robust, accountable watermark that is resistant to detection errors.

Using this information we are able to formalize the Threshold Definition to be: **Six Red-List words per 150 words in the generated output is the defined threshold for determining watermark presence in the SAFEGEN framework.**

3.4 **Discussion**

The implementation of the SAFEGEN watermarking framework yielded significant results, providing insights into the efficacy of watermark detection for AI-generated text using a structured, context-guided approach. This section will discuss the model's performance as indicated by various evaluation metrics, compare pre- and post-watermarking text outputs, and address the implications of these findings for practical applications in AI-generated content monitoring.

The SAFEGEN model achieved a high accuracy rate, with an F1 score of 0.945 and an overall accuracy of 0.93, indicating that the model reliably distinguishes between watermarked and non-watermarked text. This is further supported by a low Type I error (False Positive Rate) of 0.04 and a Type II error (False Negative Rate) of 0.03, which reveal the model's precision in identifying true positives and true negatives. Such accuracy levels suggest that the model effectively mitigates false positives and negatives, which is critical for maintaining the credibility of watermark detection in practical use cases.

The confusion matrix in Figure (Figure 8) shows the classification distribution across watermarked and non-watermarked paragraphs based on a threshold of six Red-List words. The matrix reveals that 47 instances were correctly classified as watermarked (True Positives), while only 3 instances were misclassified as non-watermarked (False Negatives). This low false negative rate confirms that SAFEGEN can reliably detect watermarked content, a core requirement for its intended applications. Similarly, the model demonstrated strong performance in correctly classifying non-watermarked text, with 46 true negatives and only 4 false positives. These results highlight the model's robustness and consistency.

A further examination of the Red-List distribution across the model's outputs before and after watermarking implementation (as shown in Figures 6 and 7) suggests that the context-guided approach successfully increased the presence of Red-List words in generated content. The histogram of unique Red-List words pre-watermarking shows a limited spread, with most paragraphs containing fewer than three Red-List words. After implementing the watermarking protocol, however, the distribution shifted significantly, with many paragraphs containing six or more Red-List words, consistent with the applied threshold. This increase in Red-List word frequency demonstrates the effectiveness of context-guiding

as a method for embedding watermarks without compromising textual coherence.

The decision to set the watermark detection threshold at six Red-List words per 150-word segment was based on a balance between sensitivity and specificity. Higher thresholds, while increasing specificity, would risk under-detecting watermarked content, particularly in cases where the generated text naturally incorporates fewer Red-List words. Conversely, lower thresholds would likely lead to over-detection, potentially flagging non-watermarked content. The chosen threshold of six words aligns with the model's performance metrics, as it minimizes Type I and Type II errors, ensuring that watermarked content is neither underrepresented nor over-identified.

4. Conclusion

The SAFEGEN framework represents a significant advancement in the field of AI-generated content detection, addressing the growing need for accountability and traceability in generative models. Through the innovative use of context-guided watermarking, SAFEGEN enhances the reliability of AI detection by embedding certain words, from a pre-defined Red-List in a manner that is both detectable and unobtrusive. By leveraging a BERT-based classification model to categorize content into specific domains, the framework ensures that watermarking is tailored to the semantic context, thereby preserving the natural flow and quality of the generated text.

Our evaluation metrics highlight the framework's effectiveness in watermark detection. The SAFEGEN model achieved an F1 score of 0.945 and an overall accuracy of 93%, with low Type I and Type II error rates of 4% and 3%, respectively. These results underscore the model's robustness in correctly identifying watermarked and non-watermarked content, minimizing the risk of false positives and negatives that could compromise trust in the detection process. The optimized threshold of six Red-List words per 150 words represents a carefully calibrated balance between sensitivity and specificity, ensuring that the watermark is detectable without incorrectly flagging non-watermarked content.

The histogram analysis of unique Red-List words across paragraphs pre- and post-watermarking demonstrates the efficacy of the context-guided approach in embedding Red-List words at the desired frequency. The shift in word distribution after watermarking implementation confirms that the framework effectively integrates domain-specific terminology into AI outputs, achieving a reliable watermark. This subtle yet effective watermark offers a powerful tool for institutions, organizations, and content creators seeking to identify and verify AI-generated material, thus addressing the critical issue of AI accountability.

SAFEGEN's open-source nature opens up possibilities for broader adoption and collaboration. By providing a customizable and adaptable framework, SAFEGEN can be integrated into various AI

systems to enhance transparency, protect intellectual property, and support ethical AI usage. As AI continues to evolve and proliferate, frameworks like SAFEGEN are essential in fostering responsible usage that maximizes the benefits of generative AI without compromising on trust.

In conclusion, SAFEGEN demonstrates that with carefully designed watermarking techniques, AI-generated content can be both creative and accountable, bridging the gap between innovation and responsibility. SAFEGEN is not just a watermarking framework—it's an opportunity to establish a new paradigm for AI detection in an increasingly digital world.

References

Acheampong, Francisca Adoma, et al. "Transformer models for text-based emotion detection: a review of BERT-based approaches." *Artificial Intelligence Review*, vol. 54, no. 8, 2021, pp. 5789–829.

Adami, Marina. How ai-generated disinformation might impact this year's elections and how journalists should report on it. *Reuters Institute for the Study of Journalism*, Mar. 2024. reutersinstitute.politics.ox.ac.uk/news/how-ai-generated-disinformation-might-impact-years-elections-and-how-journalists-should-report.

Alavi, Maryam, and George Westerman. How generative Ai will transform knowledge work. *Harvard Business Review*, Nov. 2023.

Amrit, Preetam, and Amit Kumar Singh. "Survey on watermarking methods in the artificial intelligence domain and beyond." *Computer Communications*, vol. 188, 2022, pp. 52–65.

Anderson, Nash, et al. "AI did not write this manuscript, or did it? Can we trick the AI text detector into generated texts? The potential future of ChatGPT and AI in Sports & Exercise Medicine manuscript generation." *BMJ Open Sport & Exercise Medicine*, vol. 9, no. 1, 2023. https://doi.org/10.1136/bmjsem-2023-001568.

Boenisch, Franziska. "A systematic review on model watermarking for neural networks." *Frontiers in big Data*, vol. 4, 2021, p. 729663.

Devlin, Jacob, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2019. *arXiv*, arxiv.org/abs/1810.04805.

Doshi, Anil R., and Oliver P. Hauser. "Generative AI enhances individual creativity but reduces the collective diversity of novel content." *Science Advances*, vol. 10, no. 28, 2024, eadn5290. https://doi.org/10.1126/sciadv.adn5290.

Horev, Rani. Bert explained: State of the art language model for NLP. *Medium*, Nov. 2018. towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270.

Kamaruddin, Nurul Shamimi, et al. "A Review of Text Watermarking: Theory, Methods, and Applications." *IEEE Access*, vol. 6, 2018, pp. 8011–28. https://doi.org/10.1109/ACCESS.2018.2796585.

Kerry, Cameron F., et al. Detecting AI fingerprints: A guide to watermarking and beyond. *Brookings*, Mar. 2024. www.brookings.edu/articles/detecting-ai-fingerprints-a-guide-to-watermarking-and-beyond/.

Kirchenbauer, John, et al. "A Watermark for Large Language Models." *Proceedings of the 40th International Conference on Machine Learning*. Edited by Andreas Krause et al., Proceedings of Machine Learning Research, PMLR, 23–29 Jul 2023, pp. 17061–84, proceedings.mlr.press/v202/kirchenbauer23a.html.

———. A Watermark for Large Language Models. 2024. *arXiv*, arxiv.org/abs/2301.10226.

Li, Peixuan, et al. "Plmmark: a secure and robust black-box watermarking framework for pre-trained language models." *Proceedings of the AAAI Conference on Artificial Intelligence*. 2023, pp. 14991–99.

Mendonca, John, et al. Simple LLM Prompting is State-of-the-Art for Robust and Multilingual Dialogue Evaluation. 2023. *arXiv*, arxiv.org/abs/2308.16797.

Raman, Raghu, et al. "Fake news research trends, linkages to generative artificial intelligence and sustainable development goals." *Heliyon*, vol. 10, no. 3, 2024, e24727. https://doi.org/https://doi.org/10.1016/j.heliyon.2024.e24727.

Zaiane, Osmar, et al. "Digital Watermarking: Status, Limitations and Prospects." 2002.

Zhang, Jie, et al. “Model watermarking for image processing networks.” *Proceedings of the AAAI conference on artificial intelligence*. 2020, pp. 12805–12.

Zhao, Wayne Xin, et al. “A Survey of Large Language Models.” 2023. *arXiv*, arxiv.org/abs/2303.18223.