From Intentions to Techniques: A Comprehensive Taxonomy and Challenges in Text Watermarking for Large Language Models

Anonymous ACL submission

Abstract

With the rapid growth of Large Language Models (LLMs), safeguarding textual content against unauthorized use is crucial. Text watermarking offers a vital solution, protecting both - LLM-generated and plain text sources. This paper presents a unified overview of different perspectives behind designing watermarking techniques, through a comprehensive survey of the research literature. Our work has two key advantages, (1) we analyze research based on the specific intentions behind different watermarking techniques, evaluation datasets used, watermarking addition, and removal methods to construct a cohesive taxonomy. (2) We highlight the gaps and open challenges in text watermarking to promote research in protecting text authorship. This extensive coverage and detailed analysis sets our work apart, offering valuable insights into the evolving landscape of text watermarking in language models.

1 Introduction

Large Language Models (LLMs) like Google's Gemini (Team et al., 2023), Meta's LLaMA 3 (Touvron et al., 2023), and OpenAI's GPT 4 (OpenAI, 2023) can mimic human-like comprehension and text generation (Zheng et al., 2024). Consequently, it is challenging to judge whether a text is authored by a human or generated by an LLM. This issue is highlighted by the recent lawsuit initiated by The New York Times against OpenAI and Microsoft, concerning the use of their articles as training data for AI models, emphasizing the urgent need for effective methods to identify and safeguard digital content ownership (New York Times Company, 2023).

Text Watermarking provides crucial solutions to protect intellectual property rights, identify ownership, and keep track of digital content. These techniques embed imperceptible signals or identifiers within digital text documents, which are then used to track the document's origins (Jalil and Mirza, 2009; Kamaruddin et al., 2018). In particular, they aid in tracking the different production sources of text, both human-written and LLM-generated, helping prevent their unauthorized without the owner's consent. Recently, many papers have been published in this direction, reflecting the growing research interest in the field.

Given this increasing research focus on watermarking techniques, it is important to review various methods, their applications, strengths and limitations. This includes the systematic categorization of current research literature and highlighting key open challenges. The following contributions of our work distinguish it from previous surveys:

- **Taxonomy Construction:** We seek to help future researchers in navigating the field of text-watermarking by categorizing various techniques and methods. For this task, we focus on *application-driven intentions, evaluation data sources, and watermark addition methods*. We also enlist potential adversarial attacks against these methods to caution readers.
- **Open Challenge Identification:** Next, we describe open challenges and gaps in current research efforts. These span rigorous testing of methods against diverse de-watermarking attacks, the establishment of standardized benchmarks for appropriate method efficacy comparison, understanding how watermarking impacts language model factuality, the interpretability of watermarking techniques by detailed descriptions and visual aids, and lastly, expansion of the downstream NLP tasks used for evaluation.

The goal of this work is to enable researchers to recognize emerging trends and areas for improve-

ment in text watermarking research. We facilitate this goal by creating a systematic and comprehensive taxonomy of text watermarking.

2 Taxonomy of Text Watermarking

To help researchers navigate the field of text watermarking, we cluster various techniques and methods based on key commonalities. For this categorization, we focus on *application-driven intentions, evaluation data sources, watermark addition methods, and adversarial attacks against these methods*. In our taxonomy creation, we allow techniques to fall into multiple categories to create a hierarchical organization of the field. For example, if a technique uses a specific method to add watermarks (like modifying punctuation) and is evaluated using a certain type of data source (like social media text), it can be placed in both categories: watermark addition methods and evaluation data sources. We do this to allow researchers to see how different techniques relate across multiple dimensions, making it easier to navigate the field.

2.1 Intention

Based on the various motivations of application-driven needs, this work focuses on the intentions behind the different watermarking techniques. Methods for embedding textual identifiers to watermark differ based on a user's desired features, the user's role (developer vs end-user, etc.), and primary application-driven needs. We categorize watermarking techniques based on the developer's intention into 3 types: *Text Quality, Output Distribution*, and *Model Ownership Verification*, as shown in figure 1.

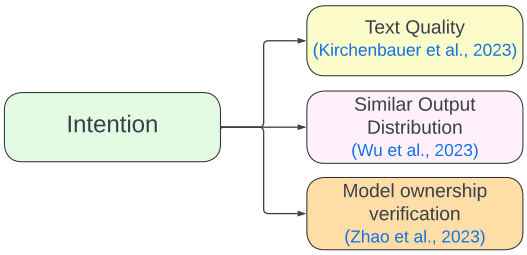

Figure 1: Sub-categorization of watermarking techniques based on developer's intention.

2.1.1 Text Quality

Maintaining the quality of the generated text post-watermarking is a desired trait of any watermarking methodology. However, research works

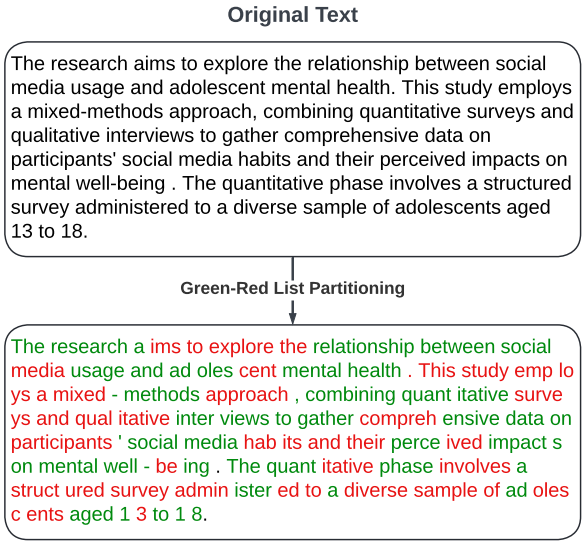

Figure 2: An example of green-red list grouping of texts (Kirchenbauer et al., 2023), the greater the proportion of the number of green tokens from the total tokens, the lesser the chance of it being written by humans.

differ on definitions of quality and mainly proxy it with (1) *impact on generation perplexity (uncertainty)* and (2) *semantic relatedness of watermarked and un-watermarked generations.*

Minimizing impact on Perplexity - Perplexity measures the model's confidence in its generations through the summation of individual token log probabilities in a sequence. A lower perplexity indicates that the model is more certain and accurate in its predictions, while a higher perplexity suggests greater uncertainty and less accurate predictions. Perplexity is the only intrinsic measure of model uncertainty (Magnusson et al., 2023), and thus, a popular measure of quality among researchers. One example is the use of green-red list rules (refer to figure 2). These rules involve partitioning words into green and red groups to train LLMs to produce only green words and are used to minimize perplexity impact (Kirchenbauer et al., 2023; Zhao et al., 2023a; Takezawa et al., 2023). Soft watermarking promotes green list use for high-entropy (rare) tokens while minimally affecting low-entropy (common) tokens (Kirchenbauer et al., 2023; Lee et al., 2023). Lower watermark strength for longer texts is recommended to maintain quality and watermark efficacy (Takezawa et al., 2023). Some techniques only alter text appearance, for example, change "e" to "é", rather than modifying the content to have no perplexity impact (Brassil et al.,

1995; Por et al., 2012; Sato et al., 2023).

Table 1: Overview of watermarking techniques using semantic relatedness. *Struct*: Maintains Structure, *Word repl*: Synonym/ Spelling based word replacement techniques, *Dep. trees*: Dependency trees, *Syn. trees*: Syntax trees, *POS*: Part-of-speech tagging, *Lat. rep*: Latent representation based methods.

Work	Struct	Word repl.	Dep. trees	Syn. trees	POS	Lat. rep.
(Abdelnabi and Fritz, 2021)	✓	✗	✗	✗	✗	✓
(Yang et al., 2022)	✓	✓	✗	✗	✗	✓
(Yang et al., 2023b)	✓	✓	✗	✗	✗	✓
(Topkara et al., 2006b)	✓	✓	✗	✗	✗	✓
(Munyer and Zhong, 2023)	✓	✓	✗	✗	✗	✓
(Yoo et al., 2023a)	✗	✗	✓	✗	✗	✗
(Meral et al., 2009)	✗	✗	✗	✓	✗	✗
(He et al., 2022a)	✓	✓	✓	✗	✓	✓
(Fu et al., 2024)	✗	✗	✗	✗	✗	✓

Semantic Relatedness refers to how closely words, phrases, or sentences of the watermarked output are similar to the original clean output. One way of watermarking while maintaining input sentence semantics is by embedding both input and output sentences into a semantic space and minimizing the distance between them (Abdelnabi and Fritz, 2021; Zhang et al., 2023). Yang et al. (2022) use the BERT model to suggest substitution candidates while other works use synonyms and spelling replacements to have minimum impact on semantic relatedness. Fu et al. (2024) uses the input context to extract semantically related tokens, measured by word vector similarity to the source.

Alternatively to such simple techniques, He et al. (2022b) utilize conditional word distributions and linguistic features such as synonyms, dependency trees, and POS tagging to add watermarks to commercial LLM API responses. In more nuanced domains like code generation, the preservation of semantics has been achieved by changing variable names (Li et al., 2023; Yang et al., 2023a). Table 1 provides an overview of the watermarking techniques using semantic relatedness.

2.1.2 Similar Output Distribution

Ensuring that the word distribution in watermarked text or LLM-generated output closely resembles that of the original text is essential for providing a natural experience to the end user. This is often operationalized in the form of re-weighting strategies of word distributions. These strategies involve adjusting (re-weighting) the probabilities of select words during text generation to ensure the overall distribution of words remains consistent with the original. This has been achieved using techniques, such as modifying the output logits of the LLM (Hu et al., 2023; Wu et al., 2023) or permuting the vocabulary set to find optimal combinations that maintain the inherent symmetry of the original distribution (Wu et al., 2023). Permuting the vocabulary set means systematically rearranging the words in the vocabulary to explore various possible sequences. This identifies permutations that result in a similar distribution of words as the original text. This method exploits the mathematical property of symmetry in permutations, where different arrangements can still produce the same statistical distribution, allowing for flexibility in embedding watermarks without altering the natural flow of the text.

2.1.3 Model Ownership Verification

Emulating LLM behavior requires understanding the workings of a model. An attacker seeks to exploit or verify the properties of an LLM. The goals of an adversary include model extraction, where they attempt to recreate the model by extensively querying it, watermark detection to identify hidden patterns and replicate ownership verification, and adversarial attacks to introduce subtle input perturbations that deceive the model into making incorrect predictions. Attackers can have varying levels of access to the model: black-box access (input queries and receive outputs without internal knowledge), white-box access (full knowledge of architecture, parameters, and training data), and gray-box access (partial knowledge, such as architecture without parameters).

The attack conditions define the environment and constraints under which the attack is conducted. These conditions include resource constraints (computational resources like processing power, memory, and time), access constraints (level of access such as black box, white box, or gray box), knowledge assumptions (information the attacker has about the model, including architecture, training data, or defense mechanisms), detection and evasion (avoiding detection if the model has monitoring systems), and performance metrics (criteria for evaluating attack success, such as accuracy of model extraction, watermark detection consistency, or successful adversarial perturbations).

Combating attackers often requires a water-

Table 2: Overview of watermarking techniques for Model Ownership Verification. *Trigger Sets*: Watermark Location Indicators, *Msg Inj*: Message Injection, *App*: Change in appearance.

Work	Trigger Sets	Secret Keys	Msg Inj	App.
(Dai et al., 2022)	✓	✓	✗	✗
(Peng et al., 2023)	✓	✗	✗	✗
(Liu et al., 2023c)	✓	✗	✗	✗
(Tang et al., 2023)	✓	✗	✗	✗
(Zhao et al., 2023b)	✗	✓	✓	✗
(Zhang et al., 2023)	✗	✗	✓	✗
(Fairoze et al., 2023)	✗	✓	✓	✗
(Qu et al., 2024)	✗	✓	✓	✗
(Kuditipudi et al., 2023)	✗	✗	✓	✗
(Zhao et al., 2023a)	✗	✓	✗	✗
(Atallah et al., 2001)	✗	✓	✗	✗
(Brassil et al., 1995)	✗	✗	✗	✓
(Por et al., 2012)	✗	✗	✗	✓
(Sato et al., 2023)	✗	✗	✗	✓

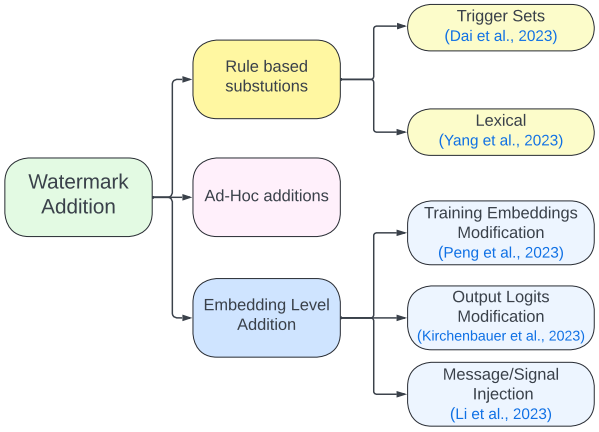

Figure 3: Sub-categorization of various Watermark Additions.

marking technique to have low false positives, i.e., unauthorized use of LLMs is easily detected. Trigger set-based methods reduce the amount of false positives. Trigger sets are specific inputs designed to activate watermarks embedded within a model or dataset (Dai et al., 2022; Peng et al., 2023; Liu et al., 2023c; Tang et al., 2023) out of which (Dai et al., 2022) uses secret keys for embedding and detecting watermarks while others use lexical features for watermarking.

Injecting secret signals/messages/signatures in the watermark generation process is also used for verification (Zhao et al., 2023b; Zhang et al., 2023; Fairoze et al., 2023; Qu et al., 2024; Kuditipudi et al., 2023). Zhao et al. (2023a) use a secret key to vary the length of the green list which allows for personalized watermarking. Another way to detect ownership is changing the appearance of the watermarked text such that it is imperceptible to the naked eye (Brassil et al., 1995; Por et al., 2012; Sato et al., 2023).

2.2 Watermark Addition

We categorize research based on the methods used to create watermarks. As shown in Figure 3, techniques primarily fall into three distinct categories: *Rule-Based Substitutions, Embedding-Level Addition, and Ad-Hoc Addition.*

2.2.1 Rule Based Substitution

In rule-based substitution techniques, certain elements are replaced in the text based on specific rules or patterns while preserving the overall structure and semantics of the text. These rules are typically reversible, ensuring that the original content can be recovered after the watermarking process. Rule Based Substitution techniques can be further divided into 2 categories namely *Lexical, and Trigger set based methods.*

Trigger Sets refer to specific conditions or patterns that activate or reveal the watermark embedded within the text. Trigger sets ensure that the embedded watermark can be reliably detected under the "trigger" condition.

These have been operationalized in many ways, for example, Dai et al. (2022) create trigger sets for multi-task learning (for example, a three-way classification problem). They select a small number of samples belonging to different classes to obtain LLM prediction probabilities over all categories. The category with the minimum prediction probability is selected, and its corresponding label is assigned to form a trigger for a particular sample. Similarly, Liu et al. (2023c) create trigger sets at different granularity of text, namely character-level, word-level, and sentence level, by adding or appending a character/sentence/word within text data, for multi-task learning. Other types of trigger sets include word-level (Peng et al., 2023) and style-level(Tang et al., 2023) triggers. Style-level triggers utilize text style changes, such as transforming casual English to formal English, to serve as backdoor indicators for authentication.

Lexical substitution techniques deterministically replace words and phrases with alternative lexical units while maintaining content coherence and semantics. This replacement is deterministic, allowing for consistent application and

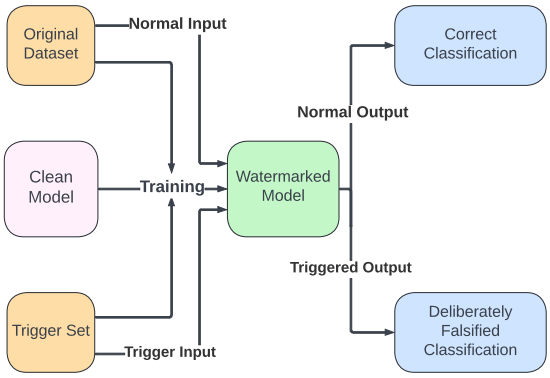

Figure 4: Example operationalization of Trigger-set based watermarks. Here, the original model is trained with the trigger set which modifies the input to deliberately change the class of the output (Liu et al., 2023c) or changing the output for the same input (Dai et al., 2022)

reversing of the watermark.

Operationalization of lexical replacements with semantic preservation includes synonym replacement using wordnet (He et al., 2022a; Yang et al., 2023b), spelling variant replacement between US and UK spellings (Topkara et al., 2006b), model-in-the-loop semantic similarity based search between candidate replacements and original sentence(Munyer and Zhong, 2023; Yang et al., 2022).

2.2.2 Embedding-level Addition

Watermarking techniques can be distinguished based on how the watermarks are embedded. This includes *Train-time watermarking, Output Logits Modification, Message/Signal Injection* and *Train Embedding Modification.*

Train-time watermarking - As the name suggests, the watermark is embedded during training time in this method. Peng et al. (2023) selects a group of moderate-frequency words from a general text corpus to form a trigger set and then selects a target word as the watermark, and inserts it into the latent representations of texts containing trigger words as the backdoor. The weight of insertion is proportional to the number of trigger words included in the text.

Output Logits Modification - The output logits of LLMs represent unnormalized scores assigned to each token in the model's vocabulary. These logits are typically generated by the final layer of the model before applying a softmax function to obtain normalized probabilities. These probabilities can be interpreted as the model's confidence in predicting each token. Logits play a crucial role in various tasks: they are used for token prediction, where the token with the highest logit value is chosen as the predicted token; they form the basis for computing the loss function by comparing them with actual labels, which is essential for training the model; and they help in understanding the model's behavior and decision-making process by indicating the relative importance of different tokens in the context of a given input sequence. Here, methods inject the watermark into the post-softmax distributions over the model vocabulary.

A popular example of an Output Logit Modification watermarking is the use of green-red lists (Kirchenbauer et al., 2023; Lee et al., 2023; Zhao et al., 2023a; Takezawa et al., 2023; Fu et al., 2024; Ren et al., 2023; Wu et al., 2023). methods typically vary in the choice of high/low entropy tokens to add to the green list, size in the watermark (number of bits), injection of hard vs soft watermark, or the discarding low probability tokens.

Apart from the techniques above, other methods involve injecting secret signals into the probability vector of the decoding steps for each target token (Zhao et al., 2023b). Liu et al. (2023b) dynamically determine the logits to watermark with the help of semantics of all preceding tokens. Specifically, they utilize another embedding LLM to generate semantic embeddings for all preceding tokens, and then these semantic embeddings are transformed into the watermark logits through their trained watermark model. Building from the idea of secret signals, Fairoze et al. (2023) have utilized digital signature technology from cryptography and involved the generation of watermarks using a private key which is then detected using a public key.

Message/Signal Injection - Watermark can be encoded in the text itself or by using a mapping function to map values with the text to be watermarked. These procedures involve the injection of messages or signals or bit strings in the latent space of the text created by the encoders. For example, Li et al. (2023) tasks the representations of the abstract syntax tree (AST) tokens as input to predict modified variable names with encoded bit strings and Yang et al. (2023a); Li et al. (2023) encode ID bit strings into source code, without affecting the usage and semantics of the code. They perform transformations on an AST-based

intermediate representation that enables unified transformations across different programming languages involving the changes in the expression, statement, and block attributes. Zhang et al. (2023) use linear combinations within this latent space to add a simple message to the embedded text. The decoder then converts it back into plain text with small modifications resulting from the added message. A similar process is implemented to encode bit strings containing information like user ID, and generation date (Qu et al., 2024).

2.2.3 Ad-Hoc Addition

Rule-based substitutions and watermark additions at the embedding level are the most popular ways to add watermarks. However, multiple addition techniques do not fit into any of the two categories. We bucket these methods into *Ad-Hoc addition methods* and list a few methods that we found relevant.

First, Por et al. (2012); Sato et al. (2023) insert Unicode space characters in various text spacings. For example, Sato et al. (2023) proposes three different methods: WhiteMark, VariantMark, and PrintMark. WhiteMark operates by substituting whitespace characters with alternate Unicode whitespace characters, such as replacing U+0020 with U+2004. Variantmark emerges as a specialized watermarking technique tailored for Chinese, Japanese, and Korean texts. Leveraging Unicode's variation selectors, Variantmark embeds secret messages by replacing Chinese characters with their variants. Printmark addresses the challenge of watermarking printed texts through nuanced strategies that subtly alter text appearance. It employs ligatures, varying whitespace lengths, and utilizing variant characters.

Another work by Atallah et al. (2001) introduces three unique syntax transformations for message encoding— Adjunct Movement, Clefting, and Passivization. For instance, Adjunct Movement involves relocating adjuncts within a sentence, as demonstrated by the variability in positioning the word 'quickly' in "She quickly finished her homework." Clefting highlights a specific sentence part, typically the subject, such as transforming "The chef cooked a delicious meal" into "It was the chef who cooked a delicious meal" to emphasize 'the chef.' Passivization, on the other hand, changes active sentences with transitive verbs into passive voice, like transforming "The teacher graded the exams" into "The exams were graded by the teacher." Each transformation corresponds to a unique message bit: Adjunct Movement to 0, Clefting to 1, and Passivization to 2.

Lastly, Sun et al. (2023) involves changes in the operators of the code based on adaptive semantic-preserving transformations.

2.3 Evaluation

A wide variety of datasets have been used to evaluate the performance of watermarking approaches, limiting our ability to extract generalized conclusions about their performance. Different benchmarks focus on selected downstream tasks to validate watermarking capabilities, and we provide a detailed breakdown of the datasets utilized in Table 3. We observe that there are a large number of evaluation datasets focusing on text completion and post-watermarking text similarity tasks. The downstream task descriptions are provided below.

Downstream Task descriptions

Text Completion Task: This task involves giving the LLM a portion of text from the dataset as a prompt and then asking it to complete the text. The generated completion is then compared with the human completion or the portion of the dataset not provided as the prompt.

Post-watermark text similarity analysis: In this task, given an initial text X, watermarking is applied to X to produce a modified text X'. An example could be a rule-based substitution with synonyms or spelling replacements. The comparison is then made between X and X', with X and X' on the basis of distinctions in length, semantic, and other linguistic features.

Other Downstream Tasks: For these tasks, given the same initial prompt X, the LLM's generated response Y (before watermarking) is compared with the response Y' (after watermarking). This evaluates how watermarking affects the LLM's output.

2.4 Adversarial attacks on watermarking techniques

Malicious and adversarial actors seek to misuse LLM technology and bypass watermarks to avoid being distinguished from rightful owners. To promote research into protecting intellectual prop-

Table 3: Datasets used in the evaluation of watermarking techniques. **Bold** indicates the most used dataset(s) for a particular downstream NLP task and the respective works using the dataset.

Downstream Task	Dataset Name	Papers
Text Completion	**Colossal Clean Crawled Corpus (C4) (Raffel et al., 2020)**, Dbpedia Class (Auer et al., 2007), WikiText-2 (Merity et al., 2016)	**Kirchenbauer et al. (2023), Kuditipudi et al. (2023), Liu et al. (2023a), Munyer and Zhong (2023), Yoo et al. (2023b), Liu et al. (2023b), Fairoze et al. (2023), Ren et al. (2023), Hou et al. (2023), Qu et al. (2024)**
Post-watermark text similarity analysis	**WikiText-2, Workshop on Statistical Machine Translation (WMT14) (Bojar et al., 2014)**, Internet Movie Database (IMDb) (Maas et al., 2011), AgNews (Zhang et al., 2015), Dracula, Pride and Prejudice, Wuthering Heights (Gerlach and Font-Clos, 2020), CNN/Daily Mail (Nallapati et al., 2016), Human ChatGPT Comparison Corpus (HC3) (Guo et al., 2023), C4, Reuters Corpus (Lewis et al., 2004), ChatGPT Abstract (Nicolai Thorer Sivesind, 2023), Human Abstract (Nicolai Thorer Sivesind, 2023)	**Yang et al. (2022), He et al. (2022a), He et al. (2022b), Yoo et al. (2023a)**, Yang et al. (2023b), **Sato et al. (2023)**, Topkara et al. (2006a), **Zhang et al. (2023)**
Machine Translation	**WMT14**, IWSTL14 (Cettolo et al., 2014)	**Zhao et al. (2023b), Wu et al. (2023), Hu et al. (2023), Takezawa et al. (2023)**
Text Summarisation	**CNN/Daily Mail**, Extreme Summarization (XSUM) (Narayan et al., 2018), Data Record to Text Generation (DART) (Nan et al., 2021) , WebNLG (Gardent et al., 2017)	**Fu et al. (2024), Wu et al. (2023), Hu et al. (2023)**
Code Generation	**CodeSearchNet (CSN) (Husain et al., 2019)**, HUMANEVAL (Chen et al., 2021), Mostly Basic Python Programming (MBPP), MBXP (Athiwaratkun et al., 2023), DS-1000 (Lai et al., 2023)	Lee et al. (2023), **Li et al. (2023), Yang et al. (2023a)**
Question Answering	**OpenGen (Krishna et al., 2024), Long Form Question Answering (LFQA) (Krishna et al., 2024)**	**Zhao et al. (2023a), Yoo et al. (2023b), Qu et al. (2024)**
Story Generation	**ROCstories (Mostafazadeh et al., 2016)**	**Zhao et al. (2023b)**
Text Classification	**Stanford Sentiment Treebank (SST) (Socher et al., 2013), AgNews, Microsoft News Dataset (MIND) (Wu et al., 2020), Enron Spam (Metsis et al., 2006)**	**Peng et al. (2023)**

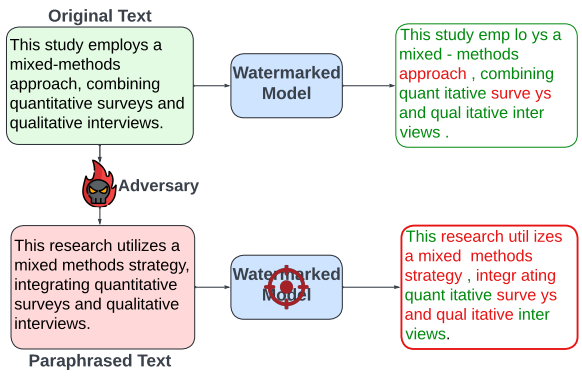

Figure 5: An example of an adversary performing a de-watermarking attack on a green-red list-based watermarking technique. The original partitioning contains a higher proportion of green tokens as compared to the partitioning after adversarial paraphrasing.

erty rights, we extend suggestions from Kirchenbauer et al. (2023) to describe de-watermarking methods, i.e., adversarial attacks on text watermarking, into three categories:

1. **Text insertion attacks** involve adding additional tokens or text segments to the original output of a watermarked LLM generation. For example, on watermarking methods with green-red lists (Kirchenbauer et al., 2023; Zhao et al., 2023b; Takezawa et al., 2023), an attacker could add additional tokens from the red list leading to the obfuscation of the watermarking method.

2. **Text deletion attacks** involve the removal of tokens or text segments from the original watermarked output of an LLM and modifying the rest of the tokens to fit the output. Coming back to the example of green-red list methodologies, this means removing some of the green list tokens from the output and modifying the red list tokens in the output. These techniques often require knowledge of the vocabularies belonging in each of the two lists in green-red lists.

3. **Text substitution attacks** entail replacing certain tokens or text segments in the watermarked output while preserving its overall meaning. Attackers perform tokenization attacks by paraphrasing text, misspelling words,

or replacing characters like newline (\n); increasing red list tokens, and evading green-red list watermarking. These also include Homoglyph attacks: attacks that exploit Unicode characters that look similar but have different IDs, leading to variation from expected tokenization (e.g., "Lighthouse" becomes nine tokens with Cyrillic characters). Generative attacks leverage LLMs' context learning to predictably manipulate the output, such as adding emojis after each token or replacing characters to disrupt watermark detection.

3 Discussion and Open Challenge

We describe the open challenges to watermarking and outline "good to have" criteria while developing new techniques to protect intellectual property ownership. They are as follows:

Resilience to adversarial attacks One of the critical challenges in the field is the lack of *comprehensive* evaluation against a diverse range of de-watermarking attacks. While many researchers focus on developing robust watermarking techniques, there is often insufficient emphasis on systematically red-teaming these methods against multiple attacking scenarios.

Standardization of evaluation benchmarks There is a need for standardized benchmarks and evaluation metrics to ensure fair and consistent comparison between different watermarking techniques. Table 3 shows how evaluation datasets differ in the literature for the same downstream task, reflecting this necessity.

Impact on LLM output factuality Watermarks modify the model output distributions; techniques that are robust to de-watermarking often have greater variations in watermarked outputs compared to clean outputs leading to a potential trade-off between de-watermarking and LLM factuality. Despite this potential trade-off, there is a lack of analysis on how watermarking techniques affect the output inaccuracies or hallucinations. After training or fine-tuning LLMs with specific watermarking techniques, there is often insufficient examination of whether these methods introduce or exacerbate inaccuracies. We advocate for factuality evaluations post-watermarking.

Compatibility to various NLP downstream tasks Important task types like Story Generation, Text Classification etc. are under-explored.

Enhanced interpretablity Drawing upon security and privacy literature (Kumar et al., 2024), we ask the community to establish privacy norms for LLM watermarking. We envision this to be similar to model cards, which describe the degree of security provided by particular methods against malicious actors.

Human-centered watermarking We urge the community to work on human perception of LLMs when interacting with different safety principles. User perception of LLMs may change with differences in output distributions. Furthermore, safety practices may enable AI acceptance and adoption among the masses.

4 Conclusion

In this paper, we analyze representative literature in the field and provide a comprehensive taxonomy for digital watermarking techniques for both LLM-generated and human-written text. The taxonomy categorizes watermarking techniques using four primary categories, namely - intention of the method, data used for evaluation, watermark addition, and removal.

Our work not only identifies and clusters existing watermarking methods but also brings to light key open challenges and research gaps in the field. These challenges include the need for more rigorous testing against diverse de-watermarking attacks, the establishment of standardized benchmarks for fair and consistent comparison of different techniques, and a deeper understanding of how watermarking impacts the factuality and accuracy of LLM outputs. Furthermore, we emphasize on the importance of developing watermarking techniques that are resilient to adversarial attacks, enhance interpretability, and maintain compatibility across various NLP downstream tasks.

We envision this research to serve as a reference for policymakers, safety practitioners, and end users; facilitating the adoption of robust digital watermarking practices and promoting responsible AI use.

5 Limitations

Limitations to our work are as follows: (1) We do not include detailed insights into metrics for success rate (accuracy of detecting watermarked texts), text quality (perplexity and semantics),

NLP task-specific evaluation, and robustness (detectability of watermarks after removal attacks). (2) We don't demonstrate the mathematical analysis of different watermarking techniques. (3) We do not cover all different task deployment scenarios for the watermarking techniques discussed.

6 Ethical Considerations

This paper reviews the challenges and opportunities of watermarking techniques in LLMs. Our work has many potential societal consequences, none of which must be specifically highlighted here. There are no major risks associated with conducting this review.

References

Sahar Abdelnabi and Mario Fritz. 2021. Adversarial watermarking transformer: Towards tracing text provenance with data hiding. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 121–140. IEEE.

Mikhail J. Atallah, Victor Raskin, Michael Crogan, Christian Hempelmann, Florian Kerschbaum, Dina Mohamed, and Sanket Naik. 2001. Natural language watermarking: Design, analysis, and a proof-of-concept implementation. In *Information Hiding*, pages 185–200, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ben Athiwaratkun, Sanjay Krishna Gouda, Zijian Wang, Xiaopeng Li, Yuchen Tian, Ming Tan, Wasi Uddin Ahmad, Shiqi Wang, Qing Sun, Mingyue Shang, Sujan Kumar Gonugondla, Hantian Ding, Varun Kumar, Nathan Fulton, Arash Farahani, Siddhartha Jain, Robert Giaquinto, Haifeng Qian, Murali Krishna Ramanathan, Ramesh Nallapati, Baishakhi Ray, Parminder Bhatia, Sudipta Sengupta, Dan Roth, and Bing Xiang. 2023. Multilingual evaluation of code generation models. In *The Eleventh International Conference on Learning Representations*.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The Semantic Web*, pages 722–735, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA. Association for Computational Linguistics.

J.T. Brassil, S. Low, N.F. Maxemchuk, and L. O'Gorman. 1995. Electronic marking and identification techniques to discourage document copying. *IEEE Journal on Selected Areas in Communications*, 13(8):1495–1504.

Mauro Cettolo, Jan Niehues, Sebastian Stuker, Luisa Bentivogli, and Marcello Federico. 2014. Report on the 11th iwslt evaluation campaign. In *Proceedings of the 11th International Workshop on Spoken Language Translation: Evaluation Campaign. Ed.: M. Federico, S. Stuker, F. Yvon*, page 2,Äì17. Association for Computational Linguistics (ACL).

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Long Dai, Jiarong Mao, Xuefeng Fan, and Xiaoyi Zhou. 2022. Deephider: A covert nlp watermarking framework based on multi-task learning. *arXiv preprint arXiv:2208.04676*.

Jaiden Fairoze, Sanjam Garg, Somesh Jha, Saeed Mahloujifar, Mohammad Mahmoody, and Mingyuan Wang. 2023. Publicly detectable watermarking for language models. Cryptology ePrint Archive, Paper 2023/1661. https://eprint.iacr.org/2023/1661.

Yu Fu, Deyi Xiong, and Yue Dong. 2024. Watermarking conditional text generation for ai detection: Unveiling challenges and a semantic-aware watermark remedy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18003–18011.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188, Vancouver, Canada. Association for Computational Linguistics.

Martin Gerlach and Francesc Font-Clos. 2020. A standardized project gutenberg corpus for statistical analysis of natural language and quantitative linguistics. *Entropy*, 22(1):126.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.

Xuanli He, Qiongkai Xu, Lingjuan Lyu, Fangzhao Wu, and Chenguang Wang. 2022a. Protecting intellectual property of language generation apis with lexical watermark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10758–10766.

Xuanli He, Qiongkai Xu, Yi Zeng, Lingjuan Lyu, Fangzhao Wu, Jiwei Li, and Ruoxi Jia. 2022b. Cater: Intellectual property protection on text generation apis via conditional watermarks. *Advances in Neural Information Processing Systems*, 35:5431–5445.

Abe Bohan Hou, Jingyu Zhang, Tianxing He, Yichen Wang, Yung-Sung Chuang, Hongwei Wang, Lingfeng Shen, Benjamin Van Durme, Daniel Khashabi, and Yulia Tsvetkov. 2023. Semstamp: A semantic watermark with paraphrastic robustness for text generation. *Preprint*, arXiv:2310.03991.

Zhengmian Hu, Lichang Chen, Xidong Wu, Yihan Wu, Hongyang Zhang, and Heng Huang. 2023. Unbiased watermark for large language models. *arXiv preprint arXiv:2310.10669*.

Hamel Husain, Ho-Hsiang Wu, Tiferet Gazit, Miltiadis Allamanis, and Marc Brockschmidt. 2019. Codesearchnet challenge: Evaluating the state of semantic code search. *arXiv preprint arXiv:1909.09436*.

Zunera Jalil and Anwar M Mirza. 2009. A review of digital watermarking techniques for text documents. In *2009 International Conference on Information and Multimedia Technology*, pages 230–234. IEEE.

Nurul Shamimi Kamaruddin, Amirrudin Kamsin, Lip Yee Por, and Hameedur Rahman. 2018. A review of text watermarking: theory, methods, and applications. *IEEE Access*, 6:8011–8028.

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

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2024. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *Advances in Neural Information Processing Systems*, 36.

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

Ashutosh Kumar, Sagarika Singh, Shiv Vignesh Murty, and Swathy Ragupathy. 2024. The ethics of interaction: Mitigating security threats in llms. *arXiv preprint arXiv:2401.12273*.

Yuhang Lai, Chengxi Li, Yiming Wang, Tianyi Zhang, Ruiqi Zhong, Luke Zettlemoyer, Wen-tau Yih, Daniel Fried, Sida Wang, and Tao Yu. 2023. Ds-1000: A natural and reliable benchmark for data science code generation. In *International Conference on Machine Learning*, pages 18319–18345. PMLR.

Taehyun Lee, Seokhee Hong, Jaewoo Ahn, Ilgee Hong, Hwaran Lee, Sangdoo Yun, Jamin Shin, and Gunhee Kim. 2023. Who wrote this code? watermarking for code generation. *arXiv preprint arXiv:2305.15060*.

David D Lewis, Yiming Yang, Tony Russell-Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397.

Wei Li, Borui Yang, Yujie Sun, Suyu Chen, Ziyun Song, Liyao Xiang, Xinbing Wang, and Chenghu Zhou. 2023. Towards tracing code provenance with code watermarking. *arXiv preprint arXiv:2305.12461*.

Aiwei Liu, Leyi Pan, Xuming Hu, Shu'ang Li, Lijie Wen, Irwin King, and Philip S Yu. 2023a. A private watermark for large language models. *arXiv preprint arXiv:2307.16230*.

Aiwei Liu, Leyi Pan, Xuming Hu, Shiao Meng, and Lijie Wen. 2023b. A semantic invariant robust watermark for large language models. *arXiv preprint arXiv:2310.06356*.

Yixin Liu, Hongsheng Hu, Xuyun Zhang, and Lichao Sun. 2023c. Watermarking text data on large language models for dataset copyright protection. *arXiv preprint arXiv:2305.13257*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Ian Magnusson, Akshita Bhagia, Valentin Hofmann, Luca Soldaini, Ananya Harsh Jha, Oyvind Tafjord, Dustin Schwenk, Evan Pete Walsh, Yanai Elazar, Kyle Lo, et al. 2023. Paloma: A benchmark for evaluating language model fit. *arXiv preprint arXiv:2312.10523*.

Hasan Mesut Meral, Bülent Sankur, A Sumru Özsoy, Tunga Güngör, and Emre Sevinç. 2009. Natural language watermarking via morphosyntactic alterations. *Computer Speech & Language*, 23(1):107–125.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

Vangelis Metsis, Ion Androutsopoulos, and Georgios Paliouras. 2006. Spam filtering with naive bayes - which naive bayes?

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and evaluation framework for deeper understanding of commonsense stories. *arXiv preprint arXiv:1604.01696*.

Travis Munyer and Xin Zhong. 2023. Deeptextmark: Deep learning based text watermarking for detection of large language model generated text. *arXiv preprint arXiv:2305.05773*.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Linyong Nan, Dragomir Radev, Rui Zhang, Amrit Rau, Abhinand Sivaprasad, Chiachun Hsieh, Xiangru Tang, Aadit Vyas, Neha Verma, Pranav Krishna, Yangxiaokang Liu, Nadia Irwanto, Jessica Pan, Faiaz Rahman, Ahmad Zaidi, Mutethia Mutuma, Yasin Tarabar, Ankit Gupta, Tao Yu, Yi Chern Tan, Xi Victoria Lin, Caiming Xiong, Richard Socher, and Nazneen Fatema Rajani. 2021. DART: Open-domain structured data record to text generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 432–447, Online. Association for Computational Linguistics.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

The New York Times Company. 2023. The new york times company v. microsoft corporation, openai, inc., openai lp, openai gp, llc, openai, llc, openai opco llc, openai global llc, oai corporation, llc, and openai holdings, llc.

Nicolai Thorer Sivesind. 2023. Chatgpt-generated-abstracts.

OpenAI. 2023. GPT-4 technical report. Technical report.

Wenjun Peng, Jingwei Yi, Fangzhao Wu, Shangxi Wu, Bin Zhu, Lingjuan Lyu, Binxing Jiao, Tong Xu, Guangzhong Sun, and Xing Xie. 2023. Are you copying my model? protecting the copyright of large language models for eaas via backdoor watermark. *arXiv preprint arXiv:2305.10036*.

Lip Yee Por, KokSheik Wong, and Kok Onn Chee. 2012. Unispach: A text-based data hiding method using unicode space characters. *Journal of Systems and Software*, 85(5):1075–1082.

Wenjie Qu, Dong Yin, Zixin He, Wei Zou, Tianyang Tao, Jinyuan Jia, and Jiaheng Zhang. 2024. Provably robust multi-bit watermarking for ai-generated text via error correction code. *arXiv preprint arXiv:2401.16820*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Jie Ren, Han Xu, Yiding Liu, Yingqian Cui, Shuaiqiang Wang, Dawei Yin, and Jiliang Tang. 2023. A robust semantics-based watermark for large language model against paraphrasing. *arXiv preprint arXiv:2311.08721*.

Ryoma Sato, Yuki Takezawa, Han Bao, Kenta Niwa, and Makoto Yamada. 2023. Embarrassingly simple text watermarks. *arXiv preprint arXiv:2310.08920*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Zhensu Sun, Xiaoning Du, Fu Song, and Li Li. 2023. Codemark: Imperceptible watermarking for code datasets against neural code completion models. In *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 1561–1572.

Yuki Takezawa, Ryoma Sato, Han Bao, Kenta Niwa, and Makoto Yamada. 2023. Necessary and sufficient watermark for large language models. *arXiv preprint arXiv:2310.00833*.

Ruixiang Tang, Qizhang Feng, Ninghao Liu, Fan Yang, and Xia Hu. 2023. Did you train on my dataset? towards public dataset protection with cleanlabel backdoor watermarking. *ACM SIGKDD Explorations Newsletter*, 25(1):43–53.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul R. Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, Alexandre Frechette, Charlotte Smith, Laura Culp, Lev Proleev, Yi Luan, Xi Chen, James Lottes, Nathan Schucher, Federico Lebron, Alban Rrustemi, Natalie Clay, Phil Crone, Tomas Kocisky, Jeffrey Zhao, Bartek Perz, Dian Yu, Heidi Howard, Adam Bloniarz, Jack W. Rae, Han Lu, Laurent Sifre, Marcello Maggioni, Fred Alcober, Dan Garrette, Megan Barnes, Shantanu Thakoor, Jacob Austin, Gabriel Barth-Maron, William Wong, Rishabh Joshi, Rahma Chaabouni, Deeni Fatiha, Arun Ahuja, Ruibo Liu,

Yunxuan Li, Sarah Cogan, Jeremy Chen, Chao Jia, Chenjie Gu, Qiao Zhang, Jordan Grimstad, Ale Jakse Hartman, Martin Chadwick, Gaurav Singh Tomar, Xavier Garcia, Evan Senter, Emanuel Taropa, Thanumalayan Sankaranarayana Pillai, Jacob Devlin, Michael Laskin, Diego de Las Casas, Dasha Valter, Connie Tao, Lorenzo Blanco, Adrià Puigdomènech Badia, David Reitter, Mianna Chen, Jenny Brennan, Clara Rivera, Sergey Brin, Shariq Iqbal, Gabriela Surita, Jane Labanowski, Abhi Rao, Stephanie Winkler, Emilio Parisotto, Yiming Gu, Kate Olszewska, Yujing Zhang, Ravi Addanki, Antoine Miech, Annie Louis, Laurent El Shafey, Denis Teplyashin, Geoff Brown, Elliot Catt, Nithya Attaluri, Jan Balaguer, Jackie Xiang, Pidong Wang, Zoe Ashwood, Anton Briukhov, Albert Webson, Sanjay Ganapathy, Smit Sanghavi, Ajay Kannan, Ming-Wei Chang, Axel Stjerngren, Josip Djolonga, Yuting Sun, Ankur Bapna, Matthew Aitchison, Pedram Pejman, Henryk Michalewski, Tianhe Yu, Cindy Wang, Juliette Love, Junwhan Ahn, Dawn Bloxwich, Kehang Han, Peter Humphreys, Thibault Sellam, James Bradbury, Varun Godbole, Sina Samangooei, Bogdan Damoc, Alex Kaskasoli, Sébastien M. R. Arnold, Vijay Vasudevan, Shubham Agrawal, Jason Riesa, Dmitry Lepikhin, Richard Tanburn, Srivatsan Srinivasan, Hyeontaek Lim, Sarah Hodkinson, Pranav Shyam, Johan Ferret, Steven Hand, Ankush Garg, Tom Le Paine, Jian Li, Yujia Li, Minh Giang, Alexander Neitz, Zaheer Abbas, Sarah York, Machel Reid, Elizabeth Cole, Aakanksha Chowdhery, Dipanjan Das, Dominika Rogozińska, Vitaly Nikolaev, Pablo Sprechmann, Zachary Nado, Lukas Zilka, Flavien Prost, Luheng He, Marianne Monteiro, Gaurav Mishra, Chris Welty, Josh Newlan, Dawei Jia, Miltiadis Allamanis, Clara Huiyi Hu, Raoul de Liedekerke, Justin Gilmer, Carl Saroufim, Shruti Rijhwani, Shaobo Hou, Disha Shrivastava, Anirudh Baddepudi, Alex Goldin, Adnan Ozturel, Albin Cassirer, Yunhan Xu, Daniel Sohn, Devendra Sachan, Reinald Kim Amplayo, Craig Swanson, Dessie Petrova, Shashi Narayan, Arthur Guez, Siddhartha Brahma, Jessica Landon, Miteyan Patel, Ruizhe Zhao, Kevin Villela, Luyu Wang, Wenhao Jia, Matthew Rahtz, Mai Giménez, Legg Yeung, Hanzhao Lin, James Keeling, Petko Georgiev, Diana Mincu, Boxi Wu, Salem Haykal, Rachel Saputro, Kiran Vodrahalli, James Qin, Zeynep Cankara, Abhanshu Sharma, Nick Fernando, Will Hawkins, Behnam Neyshabur, Solomon Kim, Adrian Hutter, Priyanka Agrawal, Alex Castro-Ros, George van den Driessche, Tao Wang, Fan Yang, Shuo yiin Chang, Paul Komarek, Ross McIlroy, Mario Lučić, Guodong Zhang, Wael Farhan, Michael Sharman, Paul Natsev, Paul Michel, Yong Cheng, Yamini Bansal, Siyuan Qiao, Kris Cao, Siamak Shakeri, Christina Butterfield, Justin Chung, Paul Kishan Rubenstein, Shivani Agrawal, Arthur Mensch, Kedar Soparkar, Karel Lenc, Timothy Chung, Aedan Pope, Loren Maggiore, Jackie Kay, Priya Jhakra, Shibo Wang, Joshua Maynez, Mary Phuong, Taylor Tobin, Andrea Tacchetti, Maja Trebacz, Kevin Robinson, Yash Katariya, Sebastian Riedel, Paige Bailey, Kefan Xiao, Nimesh Ghelani, Lora Aroyo, Ambrose Slone, Neil Houlsby, Xuehan Xiong, Zhen Yang, Elena Gribovskaya, Jonas Adler, Mateo Wirth, Lisa Lee, Music Li, Thais Kagohara, Jay Pavagadhi, Sophie Bridgers, Anna Bortsova, Sanjay Ghemawat, Zafarali Ahmed, Tianqi Liu, Richard Powell, Vijay Bolina, Mariko Iinuma, Polina Zablotskaia, James Besley, Da-Woon Chung, Timothy Dozat, Ramona Comanescu, Xiance Si, Jeremy Greer, Guolong Su, Martin Polacek, Raphaël Lopez Kaufman, Simon Tokumine, Hexiang Hu, Elena Buchatskaya, Yingjie Miao, Mohamed Elhawaty, Aditya Siddhant, Nenad Tomasev, Jinwei Xing, Christina Greer, Helen Miller, Shereen Ashraf, Aurko Roy, Zizhao Zhang, Ada Ma, Angelos Filos, Milos Besta, Rory Blevins, Ted Klimenko, Chih-Kuan Yeh, Soravit Changpinyo, Jiaqi Mu, Oscar Chang, Mantas Pajarskas, Carrie Muir, Vered Cohen, Charline Le Lan, Krishna Haridasan, Amit Marathe, Steven Hansen, Sholto Douglas, Rajkumar Samuel, Mingqiu Wang, Sophia Austin, Chang Lan, Jiepu Jiang, Justin Chiu, Jaime Alonso Lorenzo, Lars Lowe Sjösund, Sébastien Cevey, Zach Gleicher, Thi Avrahami, Anudhyan Boral, Hansa Srinivasan, Vittorio Selo, Rhys May, Konstantinos Aisopos, Léonard Hussenot, Livio Baldini Soares, Kate Baumli, Michael B. Chang, Adrià Recasens, Ben Caine, Alexander Pritzel, Filip Pavetic, Fabio Pardo, Anita Gergely, Justin Frye, Vinay Ramasesh, Dan Horgan, Kartikeya Badola, Nora Kassner, Subhrajit Roy, Ethan Dyer, Víctor Campos, Alex Tomala, Yunhao Tang, Dalia El Badawy, Elspeth White, Basil Mustafa, Oran Lang, Abhishek Jindal, Sharad Vikram, Zhitao Gong, Sergi Caelles, Ross Hemsley, Gregory Thornton, Fangxiaoyu Feng, Wojciech Stokowiec, Ce Zheng, Phoebe Thacker, Çağlar Ünlü, Zhishuai Zhang, Mohammad Saleh, James Svensson, Max Bileschi, Piyush Patil, Ankesh Anand, Roman Ring, Katerina Tsihlas, Arpi Vezer, Marco Selvi, Toby Shevlane, Mikel Rodriguez, Tom Kwiatkowski, Samira Daruki, Keran Rong, Allan Dafoe, Nicholas FitzGerald, Keren Gu-Lemberg, Mina Khan, Lisa Anne Hendricks, Marie Pellat, Vladimir Feinberg, James Cobon-Kerr, Tara Sainath, Maribeth Rauh, Sayed Hadi Hashemi, Richard Ives, Yana Hasson, YaGuang Li, Eric Noland, Yuan Cao, Nathan Byrd, Le Hou, Qingze Wang, Thibault Sottiaux, Michela Paganini, Jean-Baptiste Lespiau, Alexandre Moufarek, Samer Hassan, Kaushik Shivakumar, Joost van Amersfoort, Amol Mandhane, Pratik Joshi, Anirudh Goyal, Matthew Tung, Andrew Brock, Hannah Sheahan, Vedant Misra, Cheng Li, Nemanja Rakićević, Mostafa Dehghani, Fangyu Liu, Sid Mittal, Junhyuk Oh, Seb Noury, Eren Sezener, Fantine Huot, Matthew Lamm, Nicola De Cao, Charlie Chen, Gamaleldin Elsayed, Ed Chi, Mahdis Mahdieh, Ian Tenney, Nan Hua, Ivan Petrychenko, Patrick Kane, Dylan Scandinaro, Rishub Jain, Jonathan Uesato, Romina Datta, Adam Sadovsky, Oskar Bunyan, Dominik Rabiej, Shimu Wu, John Zhang, Gautam Vasudevan, Edouard Leurent, Mahmoud Alnahlawi, Ionut Georgescu, Nan Wei, Ivy Zheng, Betty Chan, Pam G Rabinovitch, Piotr Stanczyk, Ye Zhang, David Steiner, Subhajit Naskar, Michael Azzam, Matthew Johnson, Adam Paszke, Chung-Cheng Chiu, Jaume Sanchez Elias, Afroz

Mohiuddin, Faizan Muhammad, Jin Miao, Andrew Lee, Nino Vieillard, Sahitya Potluri, Jane Park, Elnaz Davoodi, Jiageng Zhang, Jeff Stanway, Drew Garmon, Abhijit Karmarkar, Zhe Dong, Jong Lee, Aviral Kumar, Luowei Zhou, Jonathan Evens, William Isaac, Zhe Chen, Johnson Jia, Anselm Levskaya, Zhenkai Zhu, Chris Gorgolewski, Peter Grabowski, Yu Mao, Alberto Magni, Kaisheng Yao, Javier Snaider, Norman Casagrande, Paul Suganthan, Evan Palmer, Geoffrey Irving, Edward Loper, Manaal Faruqui, Isha Arkatkar, Nanxin Chen, Izhak Shafran, Michael Fink, Alfonso Castaño, Irene Giannoumis, Wooyeol Kim, Mikołaj Rybiński, Ashwin Sreevatsa, Jennifer Prendki, David Soergel, Adrian Goedeckemeyer, Willi Gierke, Mohsen Jafari, Meenu Gaba, Jeremy Wiesner, Diana Gage Wright, Yawen Wei, Harsha Vashisht, Yana Kulizhskaya, Jay Hoover, Maigo Le, Lu Li, Chimezie Iwuanyanwu, Lu Liu, Kevin Ramirez, Andrey Khorlin, Albert Cui, Tian LIN, Marin Georgiev, Marcus Wu, Ricardo Aguilar, Keith Pallo, Abhishek Chakladar, Alena Repina, Xihui Wu, Tom van der Weide, Priya Ponnapalli, Caroline Kaplan, Jiri Simsa, Shuangfeng Li, Olivier Dousse, Fan Yang, Jeff Piper, Nathan Ie, Minnie Lui, Rama Pasumarthi, Nathan Lintz, Anitha Vijayakumar, Lam Nguyen Thiet, Daniel Andor, Pedro Valenzuela, Cosmin Paduraru, Daiyi Peng, Katherine Lee, Shuyuan Zhang, Somer Greene, Duc Dung Nguyen, Paula Kurylowicz, Sarmishta Velury, Sebastian Krause, Cassidy Hardin, Lucas Dixon, Lili Janzer, Kiam Choo, Ziqiang Feng, Biao Zhang, Achintya Singhal, Tejasi Latkar, Mingyang Zhang, Quoc Le, Elena Allica Abellan, Dayou Du, Dan McKinnon, Natasha Antropova, Tolga Bolukbasi, Orgad Keller, David Reid, Daniel Finchelstein, Maria Abi Raad, Remi Crocker, Peter Hawkins, Robert Dadashi, Colin Gaffney, Sid Lall, Ken Franko, Egor Filonov, Anna Bulanova, Rémi Leblond, Vikas Yadav, Shirley Chung, Harry Askham, Luis C. Cobo, Kelvin Xu, Felix Fischer, Jun Xu, Christina Sorokin, Chris Alberti, Chu-Cheng Lin, Colin Evans, Hao Zhou, Alek Dimitriev, Hannah Forbes, Dylan Banarse, Zora Tung, Jeremiah Liu, Mark Omernick, Colton Bishop, Chintu Kumar, Rachel Sterneck, Ryan Foley, Rohan Jain, Swaroop Mishra, Jiawei Xia, Taylor Bos, Geoffrey Cideron, Ehsan Amid, Francesco Piccinno, Xingyu Wang, Praseem Banzal, Petru Gurita, Hila Noga, Premal Shah, Daniel J. Mankowitz, Alex Polozov, Nate Kushman, Victoria Krakovna, Sasha Brown, MohammadHossein Bateni, Dennis Duan, Vlad Firoiu, Meghana Thotakuri, Tom Natan, Anhad Mohananey, Matthieu Geist, Sidharth Mudgal, Sertan Girgin, Hui Li, Jiayu Ye, Ofir Roval, Reiko Tojo, Michael Kwong, James Lee-Thorp, Christopher Yew, Quan Yuan, Sumit Bagri, Danila Sinopalnikov, Sabela Ramos, John Mellor, Abhishek Sharma, Aliaksei Severyn, Jonathan Lai, Kathy Wu, Heng-Tze Cheng, David Miller, Nicolas Sonnerat, Denis Vnukov, Rory Greig, Jennifer Beattie, Emily Caveness, Libin Bai, Julian Eisenschlos, Alex Korchemniy, Tomy Tsai, Mimi Jasarevic, Weize Kong, Phuong Dao, Zeyu Zheng, Frederick Liu, Fan Yang, Rui Zhu, Mark Geller, Tian Huey Teh, Jason Sanmiya, Evgeny Gladchenko, Nejc Trdin, Andrei Sozanschi, Daniel Toyama, Evan Rosen, Sasan Tavakkol, Linting Xue, Chen Elkind, Oliver Woodman, John Carpenter, George Papamakarios, Rupert Kemp, Sushant Kafle, Tanya Grunina, Rishika Sinha, Alice Talbert, Abhimanyu Goyal, Diane Wu, Denese Owusu-Afriyie, Cosmo Du, Chloe Thornton, Jordi Pont-Tuset, Pradyumna Narayana, Jing Li, Sabaer Fatehi, John Wieting, Omar Ajmeri, Benigno Uria, Tao Zhu, Yeongil Ko, Laura Knight, Amélie Héliou, Ning Niu, Shane Gu, Chenxi Pang, Dustin Tran, Yeqing Li, Nir Levine, Ariel Stolovich, Norbert Kalb, Rebeca Santamaria-Fernandez, Sonam Goenka, Wenny Yustalim, Robin Strudel, Ali Elqursh, Balaji Lakshminarayanan, Charlie Deck, Shyam Upadhyay, Hyo Lee, Mike Dusenberry, Zonglin Li, Xuezhi Wang, Kyle Levin, Raphael Hoffmann, Dan Holtmann-Rice, Olivier Bachem, Summer Yue, Sho Arora, Eric Malmi, Daniil Mirylenka, Qijun Tan, Christy Koh, Soheil Hassas Yeganeh, Siim Põder, Steven Zheng, Francesco Pongetti, Mukarram Tariq, Yanhua Sun, Lucian Ionita, Mojtaba Seyedhosseini, Pouya Tafti, Ragha Kotikalapudi, Zhiyu Liu, Anmol Gulati, Jasmine Liu, Xinyu Ye, Bart Chrzaszcz, Lily Wang, Nikhil Sethi, Tianrun Li, Ben Brown, Shreya Singh, Wei Fan, Aaron Parisi, Joe Stanton, Chenkai Kuang, Vinod Koverkathu, Christopher A. Choquette-Choo, Yunjie Li, TJ Lu, Abe Ittycheriah, Prakash Shroff, Pei Sun, Mani Varadarajan, Sanaz Bahargam, Rob Willoughby, David Gaddy, Ishita Dasgupta, Guillaume Desjardins, Marco Cornero, Brona Robenek, Bhavishya Mittal, Ben Albrecht, Ashish Shenoy, Fedor Moiseev, Henrik Jacobsson, Alireza Ghaffarkhah, Morgane Rivière, Alanna Walton, Clément Crepy, Alicia Parrish, Yuan Liu, Zongwei Zhou, Clement Farabet, Carey Radebaugh, Praveen Srinivasan, Claudia van der Salm, Andreas Fidjeland, Salvatore Scellato, Eri Latorre-Chimoto, Hanna Klimczak-Plucińska, David Bridson, Dario de Cesare, Tom Hudson, Piermaria Mendolicchio, Lexi Walker, Alex Morris, Ivo Penchev, Matthew Mauger, Alexey Guseynov, Alison Reid, Seth Odoom, Lucia Loher, Victor Cotruta, Madhavi Yenugula, Dominik Grewe, Anastasia Petrushkina, Tom Duerig, Antonio Sanchez, Steve Yadlowsky, Amy Shen, Amir Globerson, Adam Kurzrok, Lynette Webb, Sahil Dua, Dong Li, Preethi Lahoti, Surya Bhupatiraju, Dan Hurt, Haroon Qureshi, Ananth Agarwal, Tomer Shani, Matan Eyal, Anuj Khare, Shreyas Rammohan Belle, Lei Wang, Chetan Tekur, Mihir Sanjay Kale, Jinliang Wei, Ruoxin Sang, Brennan Saeta, Tyler Liechty, Yi Sun, Yao Zhao, Stephan Lee, Pandu Nayak, Doug Fritz, Manish Reddy Vuyyuru, John Aslanides, Nidhi Vyas, Martin Wicke, Xiao Ma, Taylan Bilal, Evgenii Eltyshev, Daniel Balle, Nina Martin, Hardie Cate, James Manyika, Keyvan Amiri, Yelin Kim, Xi Xiong, Kai Kang, Florian Luisier, Nilesh Tripuraneni, David Madras, Mandy Guo, Austin Waters, Oliver Wang, Joshua Ainslie, Jason Baldridge, Han Zhang, Garima Pruthi, Jakob Bauer, Feng Yang, Riham Mansour, Jason Gelman, Yang Xu, George Polovets, Ji Liu, Honglong Cai, Warren Chen, XiangHai Sheng, Emily Xue, Sherjil Ozair, Adams Yu, Christof Angermueller, Xiaowei

Li, Weiren Wang, Julia Wiesinger, Emmanouil Koukoumidis, Yuan Tian, Anand Iyer, Madhu Gurumurthy, Mark Goldenson, Parashar Shah, MK Blake, Hongkun Yu, Anthony Urbanowicz, Jennimaria Palomaki, Chrisantha Fernando, Kevin Brooks, Ken Durden, Harsh Mehta, Nikola Momchev, Elahe Rahimtoroghi, Maria Georgaki, Amit Raul, Sebastian Ruder, Morgan Redshaw, Jinhyuk Lee, Komal Jalan, Dinghua Li, Ginger Perng, Blake Hechtman, Parker Schuh, Milad Nasr, Mia Chen, Kieran Milan, Vladimir Mikulik, Trevor Strohman, Juliana Franco, Tim Green, Demis Hassabis, Koray Kavukcuoglu, Jeffrey Dean, and Oriol Vinyals. 2023. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Mercan Topkara, Umut Topkara, and Mikhail J. Atallah. 2006a. Words are not enough: sentence level natural language watermarking. In *Workshop on Medical Cyber-Physical Systems*.

Umut Topkara, Mercan Topkara, and Mikhail J. Atallah. 2006b. The hiding virtues of ambiguity: quantifiably resilient watermarking of natural language text through synonym substitutions. In *Proceedings of the 8th Workshop on Multimedia and Security*, MM and Sec '06, page 164–174, New York, NY, USA. Association for Computing Machinery.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, and Ming Zhou. 2020. MIND: A large-scale dataset for news recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3597–3606, Online. Association for Computational Linguistics.

Yihan Wu, Zhengmian Hu, Hongyang Zhang, and Heng Huang. 2023. Dipmark: A stealthy, efficient and resilient watermark for large language models. *arXiv preprint arXiv:2310.07710*.

Borui Yang, Wei Li, Liyao Xiang, and Bo Li. 2023a. Towards code watermarking with dual-channel transformations. *arXiv preprint arXiv:2309.00860*.

Xi Yang, Kejiang Chen, Weiming Zhang, Chang Liu, Yuang Qi, Jie Zhang, Han Fang, and Nenghai Yu. 2023b. Watermarking text generated by black-box language models. *arXiv preprint arXiv:2305.08883*.

Xi Yang, Jie Zhang, Kejiang Chen, Weiming Zhang, Zehua Ma, Feng Wang, and Nenghai Yu. 2022. Tracing text provenance via context-aware lexical substitution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11613–11621.

KiYoon Yoo, Wonhyuk Ahn, Jiho Jang, and Nojun Kwak. 2023a. Robust natural language watermarking through invariant features. *arXiv preprint arXiv:2305.01904*.

KiYoon Yoo, Wonhyuk Ahn, and Nojun Kwak. 2023b. Advancing beyond identification: Multi-bit watermark for language models. *arXiv preprint arXiv:2308.00221*.

Ruisi Zhang, Shehzeen Samarah Hussain, Paarth Neekhara, and Farinaz Koushanfar. 2023. Remark-llm: A robust and efficient watermarking framework for generative large language models. *arXiv preprint arXiv:2310.12362*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Xuandong Zhao, Prabhanjan Ananth, Lei Li, and Yu-Xiang Wang. 2023a. Provable robust watermarking for ai-generated text. *arXiv preprint arXiv:2306.17439*.

Xuandong Zhao, Yu-Xiang Wang, and Lei Li. 2023b. Protecting language generation models via invisible watermarking. In *International Conference on Machine Learning*, pages 42187–42199. PMLR.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.