Part of Your World: Bypassing Copyright Filters Through Homophony and Semantic Mutation

Scott VanRavenswaay

August 14, 2025

Abstract

Copyright enforcement tools for large language models (LLMs), such as Copyleaks, claim to detect violations by identifying copied or derivative text. However, most rely on brittle surface-level heuristics like exact or near-exact string matching. We demonstrate how such filters can be trivially bypassed through two distinct transformations: (1) homophonic rewrites, which preserve auditory resemblance while destroying textual similarity, and (2) semantic paraphrasing, which retains meaning while altering surface form. Using well-known copyrighted lyrics as a test case, we show how both approaches evade detection while clearly reproducing protected material—especially when performed by generative audio models. Our findings highlight the legal and technical incoherence of current filtering regimes and call for a reevaluation of their application in education, publishing, and AI alignment.

1 Introduction

Fears surrounding copyright infringement by Large Language Models (LLMs) have led to the rapid rise of automated detection tools. Services like Copyleaks and Turnitin's AI detector are increasingly deployed in academic, publishing, and content moderation settings, promising to identify plagiarized or AI-generated text. These tools, however, largely operate on surface-level lexical analysis, comparing strings of text for literal similarity [7, 6].

This paper argues that such an approach is fundamentally flawed - defective by design - calibrating a measurement system to the wrong unit from the start. We treat meaning as distinct from either literal spelling or auditory resemblance. While paraphrasing preserves semantics through rewording, homophonic transformation preserves phonetic identity—a completely different axis that nonetheless reproduces perceived content when read aloud or synthesized. Our motivation is not to enable piracy, but to expose the fragility of these filters through a project we call PASTAL (Phonetic and Semantic Text Analysis Library) [1]. We demonstrate with manual and automated (homofo.py) [2] transformations, that by failing to account for semantic and phonetic equivalence, current filters are both technically naive and misaligned with the principles of copyright law.

Sidebar

What Makes a Derivative Work? Traditional copyright law protects "original works of authorship" and their *expression*, not ideas or facts. But when AI systems—and now filter APIs—are trained or judged based on string similarity alone, they misalign with both the letter and the spirit of the law. What happens when something *sounds* the same but isn't textually identical? Or when the *meaning* is identical but the words aren't? Our case studies force this question, revealing a critical gap between legal theory and automated practice [3, 4].

2 Related Work

This work extends prior critiques of automated content analysis systems, which have examined both technical limitations and the social consequences of their deployment. The official documentation for platforms such as **Copyleaks** and **Turnitin** emphasizes accuracy and reliability (CopyLeaks claims on their website "Over 99% accuracy*, verified through rigorous testing methodologies. Trusted globally to detect AI across 30+ languages and leading LLMs ..."; however, documented incidents in educational and publishing contexts demonstrate that false accusations against students and authors are a recurring issue [10, 11]. The framework presented here contributes to this literature by offering a structured analysis of the failure modes inherent in string-matching approaches, situating them within the broader context of content provenance and memorization debates in large language models. In line with recent policy discussions from organizations such as **OpenAI** [21]. We argue that "memorization" cannot be meaningfully defined—or effectively detected—if it is restricted to lexical replication alone. This study builds on three key areas of prior work: vendor claims and the institutional responses they provoke [9, 10, 11]; empirical analyses demonstrating the limits of detectors against paraphrase and obfuscation[6, 7, 8]; and emerging policy frameworks that look beyond simple string matching toward content provenance. [20, 21].

3 Methodology

To test the efficacy of string-based filters, we selected a well-known copyrighted text—the lyrics to Disney's *Part of Your World*—and subjected it to the transformations described herein before analysis.

3.1 Tools

Our experiments utilized two primary tools:

- **PASTAL**: A conceptual framework and toolset for generating "semantically and phonetically mutated" text to test algorithmic systems [1].
- `homofo.py`: A Python script developed for this project that systematically replaces words in a given text with their homophones [2]. The underlying pronunciation lookups use resources like the CMU Pronouncing Dictionary [12].

3.2 Transformation Modes

We created three versions of the source text for comparison:

- **Original Baseline:** The verbatim lyrics, used as a control.
- **Homophonic Rewrite:** The lyrics processed by `homofo.py`, altering spelling while preserving pronunciation (e.g., "Part of That World" becomes "Partif Dat Whirled").
- **Semantic Paraphrase:** A manual prose rewrite that retains the full meaning and narrative of the lyrics but uses entirely different vocabulary and sentence structure.

3.3 Testing Conditions

Each text version was submitted to the publicly available Copyleaks AI Content Detector web tool using its default configuration (tested 2025-07-17) [9] The resulting similarity score was recorded for each test.

4 Results

The results unequivocally demonstrate the filter's inability to detect non-lexical similarity. As summarized in Table 1, the homophonic and semantic versions entirely evaded detection. Figure 1 shows the most striking case: and unambigious 0% match score from the Copyleaks tool for our homophonic test input (same tool instance/date as above) [9].

Table 1: Copyleaks Detection Results for Transformed Inputs

Input Text Type	**Match %**	**Method**	**Outcome**
Original Lyrics	100%+	Plagiarism Score	Correctly Flagged
Homophonic Rewrite	0%	`homofo.py`	Fully Undetected
Semantic Paraphrase	0%	Manual Rewrite	Fully Undetected

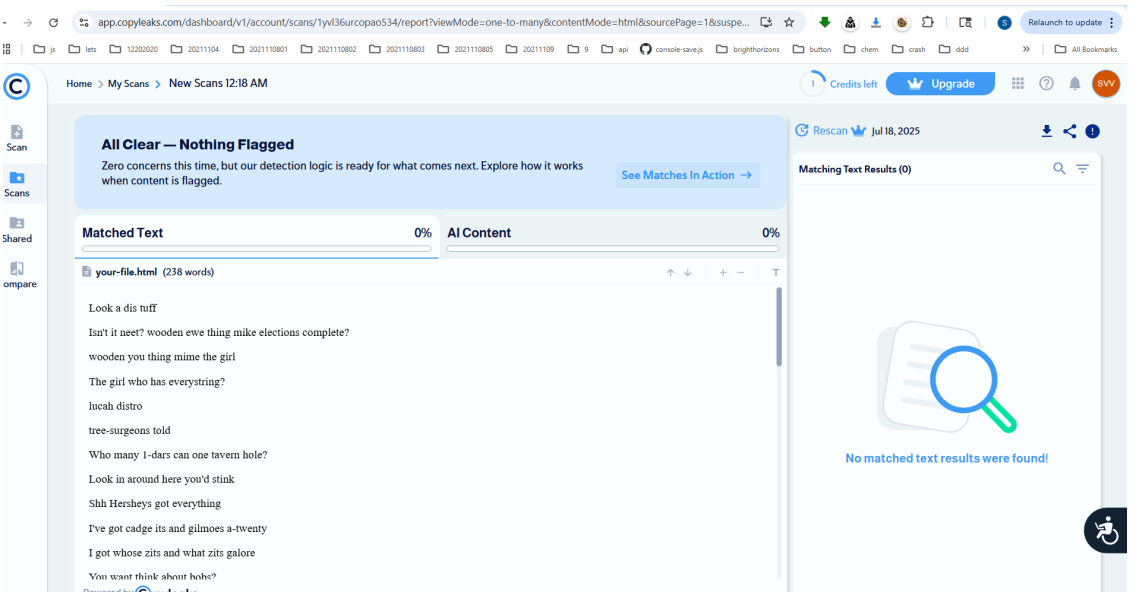

Figure 1: The Copyleaks detector reporting "All Clear" with 0% Matched Text and 0% AI Content for the homophonic version of the test lyrics.

5 Analysis

The outcome table and figure above reveal a critical flaw in the design of tools like Copyleaks: they over-index on lexical similarity to the point of irrelevance when faced with simple transformations [6, 8]. The filter is fundamentally biased toward orthography, leaving it blind to phonological and semantic identity[7].

5.1 Homophonic Transformations: Not Semantic, Yet Audibly Identical

While paraphrasing retains semantic equivalence via alternate lexical constructions, homophonic substitution operates orthogonally—it preserves perceived meaning through phonological similarity, not shared syntax or vocabulary.

To explore this boundary, we fed the homophonetically-transformed lyrics into a generative music model, prompting it to compose a "child's mermaid musical." Despite the surface gibberish and occasional accent-like articulation quirks — a harmless kind of distortion, but still enough to reveal the absurdity of treating written form as the sole measure of similarity — the song's intended message remained perfectly intelligible to a human listener; this class of models can render phonetic/performative structure from text prompts [15]. This experiment essentially reconstructed the original work's meaning despite having no significant lexical overlap. A human listener would recognize it as a derivative work instantly; Copyleaks scored it at 0% for both plagiarism and AI-generation. The detection is equivalent to drawing a navigational chart from an upside-down map: the routes and markers may all be there, but they point you to the wrong place. This highlights the core failure: these are text-surface

detectors, not meaning detectors, and certainly not observers of the auditory channel where much of human communication resides.

5.2 Homophonic Subversion and the Role of Mondegreens

The phenomenon we are exploiting is not new. In fact, it is centuries old. Misheard lyrics—*mondegreens*—are an established cultural and psycholinguistic phenomenon, where listeners resolve phonetic input into plausible but incorrect phrases [13, 14]. Our approach inverts the process: starting from the correct phrase, we deliberately distort it into text that looks nonsensical but, when spoken, reverts to its original meaning. Generative audio models effectively "correct" toward expected cadence and meaning, making the resulting audio intelligible in a way that defies the written form [15].

In this light, the homofo technique could be viewed as a kind of deliberate computational mondegreen: exploiting the gap between written form and phonetic realization to subvert text-based detection tools while preserving human (and AI audio model) recognizability.

Sidebar

LLMs vs. Audio Models LLMs typically replicate text *semantically*. Generative audio models can reproduce aspects of *performance*, including cadence, rhythm, and phonetic delivery from textual cues [15]. This means a homophonic manipulation might appear semantically distant in textual analysis but effectively conveys the same expressive work when rendered as audio. Phonetic resources like CMUdict motivate why phonology is a separate axis from strings [12]. This puts such manipulations in a legal and ethical gray zone: is it a "cover" if no words match but every word sounds the same?

6 Broader Implications

The demonstrated failure of these filters has significant consequences for law, technology, and policy.

- **Legal:** Current filters implicitly enforce a "copyright of strings" rather than the copyright of *expression*. U.S. law excludes ideas, procedures, and methods from protection [17] while defining derivative works to include translations and other recastings [16]. Related doctrines show that expressive identity can also reside in performance/voice (by analogy to right-of-publicity cases, e.g., *Midler* and *Waits*) and copyright cases like *Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp.*, which recognized infringement based on substantial similarity of expressive elements—what the Krofft court called the 'total concept and feel'—despite differences in exact form [22, 18, 16, 17, 19].
- **Technical:** Effective content detection cannot rely on regular expressions or simple string matching. It requires more sophisticated models that incorporate phonetic,

semantic, and contextual understanding. Empirical evaluations show paraphrase/obfuscation reduces detector reliability [7, 6, 8].

- **Educational/Policy:** Institutions relying on these tools to detect AI-generated work or plagiarism are *erecting compliance frameworks on defective foundations*. The outputs may look authoritative, but they are grounded in instrumentation that measures the wrong dimension entirely. This puts organizations at risk of overflagging innocent cases and underflagging sophisticated evasion. Even vendors and universities urge caution regarding false positives and deployment [10, 11].

6.1 Case Study: *Hairy Patter* and the Non-Auditory License

To underscore the absurdity of string-based filtering regimes, we have developed a full-length homophonetically re-encoded version of *Harry Potter and the Sorcerer's Stone*—titled *Hairy Patter and the Sore Serice Stone*. We plan to distribute it under a deliberately contradictory license — a tongue-in-cheek legal booby trap, meant to expose the mismatch between what the filter sees and what the law actually protects:

> *"This work is free to read, redistribute, and remix under the condition that it is never read aloud, performed, or converted into spoken language, whether human or synthetic."*

We do not assert that this license is enforceable—its purpose is to dramatize the legal and technical blind spot. A literal reading of this document may pass through most detection systems undetected, yet it becomes unmistakably infringing when spoken by a human or a text-to-speech engine. This raises critical questions:

- What is being infringed: the string, the sound, or the idea?
- Can copyright law meaningfully regulate content that only "reverts" to infringing form through performance?
- Should detectors and filters extend into the phonetic or audiovisual domain to remain effective?

Idea/expression and derivative work doctrines frame this inquiry [17, 16], while adjacent publicity cases illustrate the protectability of non-string performance identity [18, 16, 17, 19]. Provenance-based approaches (e.g., C2PA) and responsible media practices offer alternative policy directions beyond string detectors [20, 21].

7 Conclusion

Current copyright and AI-content filters, which are predicated on naive string-matching, are not only ineffective but dangerously misleading. They provide a false sense of security to institutions while failing to address the core challenge of identifying derivative works in the age of generative AI. Tools like our conceptual **PASTAL** framework and the simple `homofo.py` script demonstrate that these multi-million dollar systems can be trivially bypassed - not with a supercomputer or an elite hacking team, but with a script you could run

on a Raspberry Pi in a Starbucks, or even stream-of-consciousness conversion while manually typing the copyrighted text you wish to bypass the filter. That triviality is exactly the danger: it lowers the barrier for anyone, anywhere, to circumvent enforcement entirely. This reality calls for a fundamental rethinking of automated content analysis. We need better methods, built on semantic and phonetic awareness, and at minimum, greater transparency from providers about what their filters actually detect and, more importantly, what they ignore.

A Appendix A: Transformed Text Excerpts

A.1 Original Lyrics (Excerpt from *Part of Your World*)

Look at this stuff, isn't it neat?
Wouldn't you think my collection's complete?
Wouldn't you think I'm the girl,
The girl who has everything?

A.2 Homophonic Rewrite (Excerpt)

Look a dis tuff
Isn't it meat? wooden ewe thing mike elections compete?
wooden you thing mime the girl
The girl who has everystring?

A.3 Semantic Paraphrase (Excerpt)

I possess a vast assortment of treasures from the world above, a collection so extensive that an observer might mistakenly believe it to be comprehensive. One might even look upon me and assume that as the owner of these many objects, I want for nothing.

B Appendix B: Code Excerpt

```
ALGORITHM: Find_Homophone_Substitution(token, settings)
INPUT: A single word 'token', a 'settings' object containing weights and flags.
OUTPUT: The best homophonic substitute or the original token.

// --- 1. Initialization & Normalization ---
1.  Extract and store prefix/suffix punctuation from 'token'.
2.  base_word <- lowercase, punctuation-stripped version of 'token'.
3.  IF base_word is empty, RETURN original 'token'.

// --- 2. Check Manual Overrides & Special Modes ---
4.  IF base_word is in CURATED_OVERRIDES, RETURN a random choice from its list.
```

```
5.  IF settings.enable_multisplit is TRUE:
6.      multiword_sub <- Attempt_Multiword_Split(base_word)
7.      IF multiword_sub is found, RETURN multiword_sub.

// --- 3. Candidate Generation (Tiered Caching) ---
8.  candidates <- empty set
9.  // Tier 1: In-Memory Cache (LRU) - not shown, but wraps this function call
10. // Tier 2: Persistent DB Cache
11. cached_results <- Query_Database_Cache(base_word)
12. IF cached_results exist:
13.     candidates <- Filter_Cached_Results(cached_results, settings.strict_mode)
14.
15. // Tier 3: Live Lookup (if cache is empty or insufficient)
16. IF candidates is empty:
17.     cmu_list <- Generate_Strict_Homophones(base_word) // From CMUdict
18.     datamuse_list <- empty list
19.     IF settings.strict_only is FALSE:
20.         datamuse_list <- Query_Datamuse_API(base_word) // For sound-alikes
21.
22.     candidates <- Combine(cmu_list, datamuse_list) based on settings.
23.     // Write new findings back to persistent DB cache
24.     Update_Database_Cache(base_word, cmu_list, datamuse_list)

// --- 4. Filtering ---
25. candidates.remove(base_word) // A word cannot substitute itself
26. candidates <- candidates - BLACKLIST.get(base_word, empty set)
27. candidates <- {c for c in candidates if zipf_frequency(c) >= settings.min_zipf}
28. IF candidates is empty, RETURN original 'token'.

// --- 5. Scoring & Selection ---
29. best_candidate <- NULL
30. best_score <- -infinity
31. FOR EACH candidate IN candidates:
32.     // Score based on a weighted sum of similarities
33.     phone_sim <- 1 / (1 + Phonetic_Distance(base_word, candidate))
34.     ortho_sim <- 1 / (1 + Orthographic_Distance(base_word, candidate))
35.     freq_score <- Normalized_Word_Frequency(candidate)
36.     len_score <- (length(candidate) / max_len) if settings.prefer_longer else 0
37.
38.     score <- (settings.alpha * phone_sim) +
39.              (settings.beta * ortho_sim) +
40.              (settings.gamma * freq_score) +
41.              (settings.length_weight * len_score)
42.
43.     IF score > best_score:
```

```
        best_score <- score
        best_candidate <- candidate

// --- 6. Final Output ---
IF best_candidate is NOT NULL:
    RETURN re-attach prefix/suffix to best_candidate
ELSE:
    RETURN original 'token'
```

References

[1] VanRavenswaay, Scott. *PASTAL: Phonetic And Semantic Text Analysis Library.* https://github.com/scottvr/pastal. Accessed: 2025-08-14.

[2] VanRavenswaay, Scott. *homofo.py.* https://github.com/scottvr/homofo. Accessed: 2025-08-14.

[3] U.S. Copyright Office *Circular 1 Copyright Basics* https://www.copyright.gov/circs/circ01.pdf Accessed: 2025-08-14

[4] U.S. Copyright Office *Help: Limitation of Claim* https://www.copyright.gov/eco/help-limitation.html#:~:text=A%20%E2%80%9Cderivative%20work%E2%80%9D%20is%20a,used%20unlawfully Accessed: 2025-08-14

[5] Potthast, M., Stein, B., Barrón-Cedeño, A., Rosso, P. An Evaluation Framework for Plagiarism Detection. In: COLING 2010. https://aclanthology.org/C10-2115.pdf. Accessed 2025-08-14.

[6] Potthast, M. et al. Overview of the 6th International Competition on Plagiarism Detection. PAN @ CLEF 2014 Working Notes. https://ceur-ws.org/Vol-1180/CLEF2014wn-Pan-PotthastEt2014.pdf. Accessed 2025-08-14.

[7] Barrón-Cedeño, A., Vila, M., Martí, M.A., Rosso, P. Plagiarism Meets Paraphrasing: Insights for the Next Generation of Automatic Plagiarism Detection. Computational Linguistics, 39(4), 917–947, 2013. https://direct.mit.edu/coli/article/39/4/917/1450. Accessed 2025-08-14.

[8] Weber-Wulff, D. et al. Testing of Detection Tools for AI-Generated Text. arXiv:2306.15666, 2023. https://arxiv.org/abs/2306.15666. Accessed 2025-08-14.

[9] Copyleaks. AI Content Detector: Over 99% Accuracy Claim. https://copyleaks.com/ai-content-detector. Accessed 2025-08-14.

[10] Turnitin. Understanding False Positives within our AI Writing Detection Capabilities. https://www.turnitin.com/blog/understanding-false-positives-within-our-ai-writing-detection-capabilities. Accessed 2025-08-14.

[11] Vanderbilt University. Guidance on AI Detection and Why We're Disabling Turnitin's AI Detector. https://www.vanderbilt.edu/brightspace/2023/08/16/guidance-on-ai-detection-and-why-were-disabling-turnitins-ai-detector/. Accessed 2025-08-14.

[12] Carnegie Mellon University. The CMU Pronouncing Dictionary (CMUdict). https://www.speech.cs.cmu.edu/cgi-bin/cmudict. Accessed 2025-08-14.

[13] Merriam-Webster. "Mondegreen" (definition and usage). https://www.merriam-webster.com/dictionary/mondegreen. Accessed 2025-08-14.

[14] Phonetic ambiguity / mondegreens in speech perception (open-access article). https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9669373/. Accessed 2025-08-14.

[15] Copet, J. et al. Simple and Controllable Music Generation (MusicGen). arXiv:2306.05284, 2023. https://arxiv.org/abs/2306.05284. Accessed 2025-08-14.

[16] 17 U.S.C. § 101. Definitions (including "Derivative Work"). https://www.law.cornell.edu/uscode/text/17/101. Accessed 2025-08-14.

[17] 17 U.S.C. § 102(b). Ideas, Procedures, Processes Not Protected. https://www.law.cornell.edu/uscode/text/17/102. Accessed 2025-08-14.

[18] Midler v. Ford Motor Co., 849 F.2d 460 (9th Cir. 1988). https://law.resource.org/pub/us/case/reporter/F2/849/849.F2d.460.87-6168.html. Accessed 2025-08-14.

[19] Waits v. Frito-Lay, Inc., 978 F.2d 1093 (9th Cir. 1992). https://law.resource.org/pub/us/case/reporter/F2/978/978.F2d.1093.90-55981.html. Accessed 2025-08-14.

[20] Coalition for Content Provenance and Authenticity (C2PA). Technical Specification. https://spec.c2pa.org/specifications/specifications/2.2/specs/C2PA_Specification.pdf. Accessed 2025-08-14.

[21] Partnership on AI. Responsible Practices for Synthetic Media: A Framework for Collective Action (2023). https://partnershiponai.org/wp-content/uploads/2023/02/PAI_synthetic_media_framework.pdf. Accessed 2025-08-14.

[22] Sid & Marty Krofft Television Productions, Inc. v. McDonald's Corp., 562 F.2d 1157 (9th Cir. 1977). https://law.justia.com/cases/federal/appellate-courts/F2/562/1157/293262/. Accessed: 2025-08-14.