Enhancing Contextual Compatibility of Textual Steganography Systems Based on Large Language Models

NASOUH ALOLABI, Higher Institute for Applied Sciences and Technology, Syria
RIAD SONBOL, Higher Institute for Applied Sciences and Technology, Syria

This systematic literature review examines the transformative impact of Large Language Models (LLMs) on linguistic steganography. Through comprehensive analysis of 26 primary studies, the research demonstrates that LLM-based approaches significantly enhance imperceptibility, embedding capacity, and naturalness in cover text generation, addressing traditional limitations of low embedding capacity and cognitive imperceptibility. The findings reveal a paradigm shift towards context-aware steganographic systems that leverage domain-specific knowledge and communicative context to achieve both perceptual and statistical imperceptibility. The review establishes that understanding contextual compatibility and domain correlations is crucial for developing more sophisticated, robust, and secure covert communication systems, paving the way for future advancements in generative text steganography.

Additional Key Words and Phrases: Systematic Literature Review, Linguistic Steganography, Large Language Models, LLMs, Natural Language Processing, NLP, Black-box Steganography, Context Retrieval, Generative Text Steganography, Imperceptibility

Preprint Notice: This is a preprint version of our systematic literature review, last updated on January 8, 2026. The work is currently under review for publication.

1 INTRODUCTION

Linguistic steganography hides secrets inside ordinary sentences-an exploit that looks trivial until one remembers how little redundancy natural language actually contains [1, 2]. A single awkward synonym, a statistically rare clause, or an out-of-place idiom is enough to alert an automated sentry. Classic tricks-swap a word here, bend the syntax there-carry so few bits and leave such distinctive fingerprints that modern steganalysis routinely catches them [3].

Large language models change the game. Their uncanny fluency lets them spin entire documents that read like human prose yet obey an adversarial agenda: every plausible continuation is also a potential codeword.

None of these victories is absolute. Push the embedding rate and the text begins to creak; optimize for statistical stealth and the throughput collapses-the so-called "Psic effect" [1]. Still, progress is slow. This survey dissects the advances, catalogs the open wounds, and maps the territory that remains to be claimed.

This systematic review fills these gaps by meticulously identifying and synthesizing recent primary literature that leverages LLMs for textual steganography. The importance of this review is underscored by the transformative impact of LLMs on secure communication [citation/reference needed], marking a paradigm shift toward context-aware, generative systems that prioritize imperceptibility, embedding capacity, and naturalness [citation/reference needed].

The rest of the paper is structured as follows. Section 2 lays the theoretical groundwork by covering steganography, LLM capabilities, and the unique challenges of generative linguistic systems, including the Perceptual-Statistical Imperceptibility Conflict (PSIC). Section 3 reviews prior surveys to contextualize our contribution. Section 4 outlines our systematic review methodology-research questions, search strategy, and inclusion criteria. Section 5 presents our findings across five research questions (RQ1–RQ5), examining publication trends, applications, evaluation metrics,

Authors' addresses: Nasouh AlOlabi, Higher Institute for Applied Sciences and Technology, Damascus, Syria; Riad Sonbol, Higher Institute for Applied Sciences and Technology, Damascus, Syria.

knowledge integration, and technical trade-offs. Section 6 synthesizes these results, discussing practical implications and ethical concerns. Finally, Section 7 concludes with key insights and directions for future research.

2 BACKGROUND

Information security systems broadly encompass **encryption**, **privacy**, and **concealment**, the last of which-known as **steganography**-is the focus of this review. While encryption and privacy protect message content, they do not conceal the existence of communication, which may itself arouse suspicion. Steganography instead prioritizes **imperceptibility**: embedding information into ordinary carriers (e.g., images or text) so that hidden messages remain unnoticed.

Text is a particularly challenging carrier due to its low redundancy and strict semantic constraints. The classical "Prisoners' Problem" [4] illustrates the goal: two parties, Alice and Bob, must exchange hidden information without alerting a watchful adversary.

Textual steganography methods are typically divided into **format-based** approaches, which exploit layout or structural features, and **content-based** approaches, which modify linguistic form. Within the latter, early techniques such as **synonym substitution** embed bits by altering lexical choices, but suffer from low capacity and high detectability. More formally, **linguistic steganography** refers to concealing information in natural language by modifying or generating text while preserving fluency and meaning [5].

Traditional linguistic approaches offer limited embedding capacity and often leave statistical artifacts. Advances in deep learning and **Large Language Models (LLMs)** now enable generative methods that achieve higher text quality and more secure embedding. Evaluating such systems requires several dimensions of imperceptibility: **perceptual** (human naturalness), **statistical** (distributional similarity to natural text), and **cognitive** (semantic and contextual fidelity) [6].

A deeper theoretical perspective introduces **channel entropy**, which quantifies the information-carrying capacity of a given communication channel. Entropy sets the upper bound for embedding rates: higher entropy allows more hidden information without detection, while lower entropy restricts capacity. Achieving this bound securely requires **perfect samplers**, which can generate text indistinguishable from genuine distributional samples. These concepts underpin the design of provably secure steganographic systems.

However, LLMs [7] introduce new challenges. Their tendency toward **hallucinations** can create detectable artifacts, highlighting the **Psic Effect** (Perceptual-Statistical Imperceptibility Conflict) [1], where optimizing for perceptual fluency may undermine statistical security. Model access further shapes practical steganography: **black-box access** (e.g., commercial APIs or hosted open-weight models) offers significant advantages, delivering substantially better text quality through access to state-of-the-art models, faster generation speeds via optimized infrastructure, and minimal local resource requirements, enabling scalable deployment without the computational overhead of training or hosting large models locally. The primary trade-off is limited control and reduced transparency over internal sampling probabilities. In contrast, **white-box access** enables fine-grained control over parameters and sampling, supporting stronger security guarantees, but typically demands substantial computational resources, engineering effort, and higher latency, raising deployment barriers. This trade-off is central to evaluating the robustness and applicability of modern linguistic steganography.

2.1 Capabilities and Approximating Natural Communication

Large Language Models (LLMs) are autoregressive, generative systems based on the Transformer architecture [8] that approximate high-dimensional distributions over natural-language sequences [2][9]. Given a prefix, an LLM emits a

[Placeholder footnote]

probability vector over the vocabulary; the next token is sampled from this vector and appended to the prefix, and the process repeats until a stopping criterion is met. During pre-training, billions of parameters are tuned on large text corpora so that the model's predictive distribution converges to the empirical distribution of the data [10]. As a consequence, modern LLMs routinely produce text whose fluency, coherence and style are indistinguishable from human writing [11]. The learned latent representations capture stylistic and semantic regularities that generalize across domains, enabling applications requiring nuanced linguistic mimicry [12].

2.2 Role in Generative Linguistic Steganography

LLMs are considered **favorable for generative text steganography** due to their ability to generate high-quality text. Researchers propose using generative models as steganographic samplers to embed messages into realistic communication distributions, such as text. This approach marks a departure from prior steganographic work, motivated by the public availability of high-quality models and significant efficiency gains.

LLMs like **GPT-2** [9], **LLaMA** [13], and **Baichuan2** [14] are commonly used as basic generative models for steganography. Existing methods often utilize a language model and steganographic mapping, where secret messages are embedded by establishing a mapping between binary bits and the sampling probability of words within the training vocabulary. However, traditional "white-box" methods necessitate sharing the exact language model and training vocabulary, which limits fluency, logic, and diversity compared to natural texts generated by modern black-box LLM APIs or hosted open-weight models. **Black-box approaches provide substantial advantages:** they deliver significantly better text quality by leveraging state-of-the-art hosted models, achieve faster generation speeds through optimized cloud infrastructure, and require minimal local resource requirements without the computational overhead of running large models locally. In contrast, white-box methods require running large models locally, increasing latency and resource consumption. These methods further inevitably alter the sampling probability distribution, thereby posing security risks [15].

New approaches, such as **LLM-Stega** [15], explore **black-box generative text steganography using the user interfaces (UIs) of LLMs**. This circumvents the requirement to access internal sampling distributions. The method constructs a keyword set and employs an encrypted steganographic mapping for embedding. It proposes an optimization mechanism based on reject sampling for accurate extraction and rich semantics [15].

Another framework, **Co-Stega**, leverages LLMs to address the challenge of low capacity in social media. It expands the text space for hiding messages through context retrieval and **increases the generated text's entropy via specific prompts** to enhance embedding capacity. This approach also aims to maintain text quality, fluency, and relevance [16].

The concept of **zero-shot linguistic steganography** with LLMs utilizes in-context learning, where samples of covertext are used as context to generate more intelligible stegotext using a question-answer (QA) paradigm [17]. LLMs are also employed in approaches like **ALiSa**, which directly conceals token-level secret messages in seemingly natural steganographic text generated by off-the-shelf BERT [18] models equipped with Gibbs sampling [19].

The increasing popularity of deep generative models has made it feasible for provably secure steganography to be applied in real-world scenarios, as they fulfill requirements for perfect samplers and explicit data distributions (see Section 2) [2, 20, 21].

LLM-based steganographic methods are typically evaluated on two primary axes: imperceptibility (perceptual, statistical, and cognitive measures) and embedding capacity (e.g., bits per token or bits per word). Imperceptibility evaluations may include automatic metrics (PPL, Distinct-n, MAUVE, KL/JSD) as well as human judgements; embedding capacity is usually reported as bits/token or overall embedding rate.

[Placeholder footnote]

We now turn to the principal challenges these models face, including the trade-off between imperceptibility and capacity, robustness to tokenization, and practical deployment constraints.

2.3 Challenges and Limitations in Steganography with LLMs

2.3.1 Perceptual vs. Statistical Imperceptibility (Psic Effect). The **Psic Effect** [1] represents a fundamental trade-off in steganographic systems. it's the inverse relationship between **text quality** and **resistance to statistical steganalysis** in generative steganography. Two components govern it:

- **Perceptual imperceptibility**: fluency/naturalness of a single sentence, gauged by human ratings or perplexity (PPL).
- **Statistical imperceptibility**: divergence between the distribution of stego and human text as measured by an automated steganalyzer.

Human social-media prose is casual and high-variance; it does not hug the optimal language-model peak. A generator that over-optimizes for quality produces text whose likelihood concentrates on that peak, yielding a detectable statistical spike against the broad, noisy human baseline.[1]

Experiments show that the most fluent stego sentences are the first ones caught by detectors, yet counter-intuitively pushing the embedding rate higher can make the text statistically safer because the added noise widens its distribution toward the authentic human scatter, a trade-off modern systems like VAE-Stega manage by learning to keep sentences smooth while staying inside the real variance envelope.[1]

2.3.2 Limited Embedding Capacity. Even with advanced generative capabilities, LLMs cannot overcome natural language's fundamental low-redundancy constraint, which limits embedding capacity. This issue is acute in common applications such as social media and dialogue systems, where communication is brief or predictable. In low-entropy scenarios (e.g., replying to "Happy birthday!"), the model has few natural-sounding alternatives. Consequently, steganographic techniques like rejection sampling face higher failure rates, and channel capacity for hiding data diminishes significantly [2, 16].

2.3.3 Poor Semantic Control and Contextual Drift. Early generative methods produced fluent but **semantically arbitrary** text, violating **cognitive imperceptibility** [6]. By conditioning only on preceding tokens, these models generated replies that drifted from the original logic-a mismatch that triggers immediate suspicion on social media (e.g., replying "Today is beautiful" to a technical post). Many white-box, sampling-based techniques face a deeper problem: they must continue generating until an arbitrarily large payload is fully embedded. The model may exhaust its topic and terminate prematurely, or the user might provide a broad topic to sustain output, risking unnaturally verbose text that is highly detectable outside long-form media like blog posts. This raises a critical question often unaddressed: given a large payload like an image, how do these methods handle segmentation and embedding across messages without violating contextual norms?

Generating long stego texts introduces further technical and security hurdles. Extended generation strains coherence and contextual consistency [22, 23], and minor early deviations compound over time to degrade decoding accuracy [24, 25]. This issue is magnified in low-entropy channels, where embedding even small messages requires impractically long cover text [2]. The resulting verbosity is not merely inefficient; on platforms like social media where brevity is the norm, abnormally long posts signal anomalous activity and increase detection risk [16]. Consequently, effective

[Placeholder footnote]

information load remains limited, with performance often diminishing when embedded messages exceed just a few tokens [25].

2.3.4 LLM-Specific Obstacles. Deploying steganography with LLMs introduces distinct challenges:

- **Computational Burden:** 3–5× higher time and resource costs versus prior neural methods
- **Black-Box Access:** Hosted APIs (whether proprietary or open-weight models) limit visibility into internal sampling probabilities, blocking white-box steganographic mappings. However, they provide significant advantages: access to state-of-the-art models that deliver substantially better text quality, faster generation speeds through optimized infrastructure, and minimal local resource requirements without the computational overhead of running large models locally
- **Hallucinations:** Factually incorrect or nonsensical output can corrupt the covert bitstream or create detectable patterns
- **Escalating Detection:** As LLM capabilities advance, so do machine learning-based **steganalysis** tools that distinguish synthetic from human text
- **Data Fragility:** Lossy compression or incomplete transmission of stegotext causes irreversible bitstream corruption

2.3.5 Tokenization Mismatch. Modern Transformer models using **subword tokenization** (e.g., BPE) suffer from **segmentation ambiguity**: a sender's token sequence (" any", "thing") may detokenize to "anything" but be **retokenized differently** by the receiver as a single token, " anything". This breaks the **autoregressive chain**, corrupting all downstream probability distributions and causing extraction failure. The problem is acute in **scriptio continua** languages like Chinese, which lack explicit word boundaries.

Analogy: Alice encodes a secret using two small bricks to spell "BLUE." Bob receives one large "BLUE" brick. Since their protocol depends on exact brick counts, Bob's misalignment renders the rest of the message unreadable.

Methods that rely on modifying the sampling probability distribution to embed secret messages inherently introduce security risks because they alter the original distribution, making the steganographic text statistically distinguishable from normal text [1, 2, 15, 20]. While advancements in deep neural networks have improved text fluency and embedding capacity, older models or certain embedding strategies can still produce texts that lack naturalness, logical coherence, or diversity compared to human-written content. Models like NMT-Stega and Hi-Stega aim to maintain semantic and contextual consistency by leveraging source texts or social media contexts, yet this remains a complex challenge [6, 26].

Channel entropy requirements and variability also pose a considerable challenge. Traditional universal steganographic schemes often demand consistent channel entropy, which is rarely maintained in real-world natural language communication. Moments of low or zero entropy can cause protocols to fail or require extraordinarily long steganographic texts. The Psic Effect highlights this dilemma in balancing quality and detectability.

Additional limitations include:

- **Data Integrity and Reversibility**: Some methods cannot perfectly recover the original cover text after message extraction [27, 28].
- **Ethical Concerns**: Pre-trained LLMs may introduce biases, discrimination, or inappropriate content [17, 29].
- **Provable Security**: Many NLP steganography works lack rigorous security analyses and fail to meet formal cryptographic definitions [2].

3 RELATED REVIEWS

Majeed et al. (2021) [30] surveyed pre-LLM text steganography techniques, predating the current transformer era. Setiadi et al. (2025) [31] recognizes that LLMs have "revitalized" linguistic steganography, examining recent methods (2021-2025) using GPT-2 [32], GPT-3 [33], LLaMA2 [34], and Baichuan2 [35]. However, their review remains a critical examination rather than a systematic survey, leaving several key papers unaddressed. Crucially, the field has evolved from "statistical vector embedding" (Word2Vec, GloVe) to "language-model vector embedding" that exploits BERT-scale transformers and higher-dimensional semantic spaces.

This creates a methodological gap: no systematic review comprehensively maps how large-scale transformers have redefined text steganography. Modern advances extend beyond naive generation to sophisticated Controllable Text Generation (CTG) frameworks [36]. These employ Variational Autoencoders (VAEs) to model latent features and Diffusion Models to inject randomness, mitigating spurious associations between secrets and control conditions. Classical surveys emphasized synonym replacement, spacing manipulation, and Huffman coding [30]-techniques that predated LLMs. Earlier methods relied on context-free grammars (CFGs) or Markov chains, often producing syntactically correct but semantically incoherent cover texts. Contemporary approaches leverage prompt learning and prefix tuning, enabling efficient model customization without costly full fine-tuning.

Defensive strategies must evolve accordingly. Traditional steganalysis, premised on hand-crafted statistical features, falters against generative steganography's high statistical concealment. Current research must confront "stegomalware"-attacks that conceal command-and-control communications within innocuous digital media.

4 RESEARCH METHOD

This study was undertaken as a systematic mapping review using the guidelines presented in Petersen et al. [37]. The goal of this review is to identify, categorize, and analyze existing literature published between 2018 and 2025 and use syntactic and semantics aspects to represent context handling in linguistic steganographic methods.

4.1 Planning

In this section, we define our research questions, the search strategy we use, and the inclusion and exclusion criteria considered to filter the results.

4.1.1 Research Questions. This systematic literature review is guided by six research questions, aiming to comprehensively map the landscape of steganographic techniques leveraging large language models (LLMs). The questions explore the current state of published literature, applications where these techniques are being explored, and the metrics and evaluation methods used to assess their performance, with a focus on capacity, security, and contextual compatibility. Furthermore, the review investigates how external knowledge sources are integrated to enhance capacity or contextual relevance, the limitations and trade-offs associated with current techniques, and potential future research directions considering emerging trends and identified gaps.

4.1.2 Search Strategies. The initial literature search employed a specific query string: ‘(steganography or watermark or "Information Hiding") and ("Large Language Model" or LLM or BERT or LAMA or GPT)‘. This query was executed across several digital libraries, including ACM Digital Library, IEEE Digital Library, Science@Direct, Scopus, and Springer Link, to ensure broad coverage. To complement this automated search and identify additional relevant studies, a snowballing technique was also applied. This involved examining the reference lists of included studies. While

[Placeholder footnote]

snowballing primarily yielded older steganographic techniques not explicitly mentioning LLMs, these papers often utilized similar methodological approaches to contemporary LLM-based steganography, providing valuable contextual information.

4.1.3 Inclusion and Exclusion Criteria. To ensure the selection of high-quality and relevant studies, the following criteria were applied.

Inclusion Criteria Studies were included if they:

- IC1: Provided full-text access.
- IC2: Were published in English from 2018 onwards.
- IC3: Appeared in peer-reviewed journals, conferences, or workshops.
- IC4: Directly addressed steganography, watermarking, or information hiding techniques involving or significantly impacted by LLMs, BERT, LAMA, or GPT architectures.
- IC5: Represented empirical studies, surveys, reviews, or theoretical contributions.

Exclusion Criteria Studies were excluded if they:

- EC1: Were duplicates (retaining the most complete or recent version).
- EC2: Were incomplete, abstract-only, or irrelevant to steganography with LLMs.
- EC3: Were non-English publications.
- EC4: Came from non-peer-reviewed sources (e.g., preprints, dissertations, theses, books, book chapters), unless extended from peer-reviewed conference papers.

4.2 Conducting the Search

The initial automated search across the selected digital libraries yielded a total of 1043 candidate papers. The distribution by source was: ACM Digital Library (346), IEEE Digital Library (61), Science@Direct (209), Scopus (151), and Springer Link (276). Duplicated papers were automatically eliminated using Parsifal tool [1]. After removing all duplicates, 1,573 papers remained. Following this the papers underwent a multi-stage filtering process based on their titles, abstracts, and full texts, guided by the predefined inclusion and exclusion criteria. After title and abstract filtering, 58 papers remained. Of these, 26 were accepted with readily available PDFs, while 6 were pending PDF acquisition at the time of analysis.

4.3 Data Extraction and Classification

A Data Extraction Form (DEF) was developed to systematically collect data from each primary study to address our research questions. The form is designed in a table format consisting of the following types of information:

- Bibliometric Information: paper title, type (Steganography or Watermarking), author(s), publication year, and publication venue.
- Model Details: input and output formats, key characteristics, approach classification (three-term categorical), specific LLM used (if applicable), embedding process description, and code availability.
- Datasets: all datasets employed, including their sizes.
- Context Awareness: whether the method is "Explicit," "Implicit," or "No," the context keyword (e.g., "Social Media," "Formal Document"), how context is represented (e.g., "Text," "Pretext," "Graph," "Vector"), and how it is utilized in the method.

[1] https://parsif.al

[Placeholder footnote]

- Evaluation Details: evaluation metrics, steganalysis models used, and the best numerical results for each reported metric.
- Strengths and Limitations: main strengths and weaknesses of the approach or model.

Following data extraction, studies were classified based on predefined categories derived from the research questions to identify trends, patterns, and gaps in the literature. The results are summarized using tables, figures **??**), and descriptive statistics. Each research question is addressed individually with interpretation of findings and identification of future research directions.

5 RESULTS

This section presents the synthesized findings from our systematic literature review of 26 primary studies on LLM-based steganography. The results are organized around five research questions to provide a comprehensive analysis of the current state, applications, evaluation methods, knowledge integration, and limitations in this rapidly evolving field.

5.1 State of Published Literature on LLM-based Steganography (RQ1)

5.1.1 Publication Trends and Distribution. Our analysis reveals a significant surge in LLM-based steganography research since 2023, with approximately 17 new papers published in 2024–2025. This surge is particularly notable from the last two years when LLMs like GPT-3/4 [citation/reference needed] and open models became widely available [citation/reference needed]. Approximately 70% of recent studies utilize open-source LLMs such as GPT-2 [citation/reference needed], LLaMA2 [citation/reference needed], and LLaMA3 [citation/reference needed]. The field has evolved from early white-box modifications to more practical hybrid and black-box approaches. The resulting arms race has already produced generative schemes that write stego text from scratch [1, 2, 20, 38], rewriting engines that paraphrase existing covers [39], black-box pipelines that treat the model as an opaque API [15, 24], zero-shot protocols driven only by crafty prompting [17], collaborative frameworks that mine social context for extra entropy [16, 26], and even constructions with provable indistinguishability guarantees [2, 20].

Year	**2020**	**2021-2022**	**2023**	**2024-2025**	**Total**
Publications	2	3	4	17	26

Table 1. Publication trends by year

Model Type (%)	**Models and Representative Works**
Open-weight Models (>80%)	GPT-2 [2, 20, 26, 40], LLaMA/LLaMA2 [16, 17, 21, 22], BERT [6, 19, 23, 27, 28, 36, 41–43], OPT [44], BART [28, 39], Qwen [45]
Proprietary Models (12%)	GPT-3.5/4, ChatGPT [15, 24, 25, 46]
Custom Architectures (8%)	From-scratch or task-specific models [1]

Table 2. Model usage across surveyed studies

5.2 Applications of LLM-based Steganographic Techniques (RQ2)

The review identified six primary application domains, with covert communication being the dominant use case. The analysis reveals several distinct applications for LLM-based steganography, each with specific characteristics and requirements.

[Placeholder footnote]

Region (%)	Institutions (Representative Works)
Asia-Pacific (84%)	Primarily China-based institutions, notably Tsinghua University, University of Science and Technology of China, Beijing University of Posts and Telecommunications, Shanghai University, Yunnan University, and Zhongguancun Laboratory, with additional contributions from Nanyang Technological University (Singapore) and MM '24 (Australia) [1, 6, 15–17, 19–23, 26–28, 36, 39–43, 45, 46]
North America (12%)	Boston University, Johns Hopkins University, Texas Tech University, University of Nebraska Omaha [2, 25, 44]
Europe (4%)	Fraunhofer SIT|ATHENE, Germany [24]

Table 3. Geographic distribution of the papers

Venue Category (%)	Representative Venues and Works
Preprint Servers (4%)	arXiv [17]
Top-Tier Venues (29%)	ACM CCS [2], IEEE S&P [20], Artificial Intelligence [28], IEEE/ACM TASLP [6], ACM MM [15, 43]
Specialized Venues (67%)	IEEE Signal Processing Letters [19, 22, 23, 36], IEEE Transactions on Information Forensics and Security [1], ARES [24], IH&MMSec [16], ICONIP [26], IEEE TCDS [42], DASFAA [39], IEEE Access [44], MMSP [27], IEEE TDSC [21], ICASSP [40, 41], ICME [45], IJCNN [25], Frontiers of Computer Science [46]

Table 4. Distribution of publication venues

LLM-based steganographic techniques embed covert information within seemingly benign text, with applications spanning **secure communication**. This enables secure clandestine messaging in environments where classical steganography was too limited or suspicious [citation/reference needed]. These techniques also extend to **intellectual property protection** and **forensic linguistics**. The Calgacus protocol [47] demonstrates how secret messages can be hidden inside different cover text of identical length by matching token rank sequences, enabling political critiques to masquerade as innocuous product reviews, while black-box methods like LLM-Stega operate through commercial APIs using encrypted keyword mapping and reject sampling [15]. For **intellectual property**, watermarking via logit biasing [48] embeds imperceptible statistical signals that identify AI authorship, attribute harmful content to specific users, and filter synthetic data to prevent model collapse. In **forensic linguistics**, adversarial stylometry allows LLMs to mask author identity or imitate others by adjusting stylistic features, reducing forensic tool accuracy to random guessing-protecting whistleblowers while enabling impersonation[49, 50].

These same techniques pose significant risks to AI safety and cybersecurity, bypassing governance mechanisms and enabling sophisticated attacks. The "Linguistic Trojan Horse" embeds unsafe content in benign responses to evade safety filters, while Chain-of-Thought auditing reveals that models can hide true reasoning in seemingly innocuous steps, complicating oversight and enabling covert multi-agent collusion. In cybersecurity, steganographic prompt injection in vision-language models achieves over 31% success by hiding malicious instructions in images, while SteganoBackdoor embeds semantic triggers in training data with 99% success at low poisoning rates. Model weights can be exfiltrated through subtle token variations, and watermark stealing enables spoofing and scrubbing attacks that bypass accountability measures. Detection methods include cross-model probability scoring, low-entropy token analysis, and symbolic anomaly detection, though these face ongoing vulnerabilities that demand adaptive defense architectures [51–54].

[Placeholder footnote]

5.3 Evaluation Metrics and Methods (RQ3)

Performance evaluation for LLM-based steganography relies on three key categories of metrics, with significant variation in reporting standards across studies. The analysis reveals both the diversity of evaluation approaches and the need for standardization.

Metric Type	Imperceptibility	Capacity	Security	Usage
Perceptual	PPL: 3-300	BPW: 0.5-6.0	Detection: 50-98%	85%
Statistical	KLD: 0-3.3	BPT: 1.0-5.8	F1: 0.5-0.99	70%
Semantic	BLEU: 0.3-0.9	ER: 0.2-0.4	Acc: 0.5-0.99	60%
Human Eval	MAUVE: 0.2-0.9	-	-	25%

Table 5. Evaluation metrics usage and typical ranges across studies

5.3.1 Perplexity (PPL). An imperceptibility metric [55] that measures fluency, with lower values indicating better naturalness. It is recognized as a sensitive and unreliable metric for language model evaluation due to several intrinsic limitations. First, it suffers from a "confidently wrong" problem: as Baeldung, et al. [56] notes, perplexity measures only internal consistency, allowing models to assign low perplexity to grammatically perfect but factually absurd statements like "The cat is on the ceiling," since it cannot assess truth or logic. Second, it exhibits a short-text bias as Fang, et al. [57] demonstrated that perplexity scores are artificially inflated for short sequences despite potentially higher fluency, making it an "unqualified referee" for fair evaluation. Third, comparability across models is impossible without identical tokenization, vocabulary size directly scales perplexity - a model with fewer tokens appears deceptively better [58]. Fourth, perplexity fails to capture long-range dependencies in modern LLMs; Fang, et al. [57] argue that averaging log-likelihood across all tokens obscures performance on crucial "key tokens" by favoring predictable filler words. Finally, the metric is easily gamed through repetition, Wang, et al. [56] finds that "perplexity cannot distinguish between right emphasis and abnormal repetition," rewarding redundant text with artificially low scores. These flaws-sensitivity to length, architectural incompatibility, semantic blindness, and exploitability-collectively render perplexity an inadequate benchmark for steganographic text quality assessment.

5.3.2 MAUVE. Another imperceptibility metric that Evaluates distributional similarity between generated and reference text by quantifying the gap between neural and human-authored text using divergence frontiers.While MAUVE provides a theoretically elegant way to measure distributional gaps between generated and reference text, it remains curiously underused-appearing in just 3 of 26 reviewed sources. The deeper issue is that reported scores are *not directly comparable* across studies.

Scaling conventions alone create immediate confusion: CPG-LS reports on a 0.0-1.0 scale (achieving 0.9412) while other work uses 0-100 (with advanced white-box LLM samplers reaching 80-92). Hi-Stega's scores (0.1341-0.2051) look low by comparison, but actually represent nearly 10× improvement over its own baseline (0.0135)-demonstrating that absolute values only matter within their own context.

Architectural differences further complicate matters: CPG-LS employs BERT-based lexical substitution whereas Hi-Stega uses generative GPT-2 models, making cross-study rankings invalid without careful normalization. Dataset choice compounds the problem-CPG-LS evaluated on CC-100 while Hi-Stega used Yahoo! News comments.

Like comparing temperatures without knowing Celsius from Fahrenheit, a "30" only makes sense in its original context. Consequently, MAUVE scores work best as *internal benchmarks* for comparing variants within a single study, not as universal performance indicators across different steganographic frameworks.

[Placeholder footnote]

5.3.3 Statistical Metrics. Kullback-Leibler Divergence (KLD) [59] and Jensen-Shannon Divergence (JSD) are information-theoretic metrics used to evaluate steganographic security. KLD quantifies information loss by measuring the relative entropy between cover and stego distributions, serving as the theoretical standard for security modeling despite being asymmetric and failing as a strict distance measure. JSD improves upon this as a symmetric, bounded variant that measures how far each distribution lies from their average, providing a more stable basis for formulating statistical imperceptibility bounds-particularly when language models approximate human text distributions. Together, these two attempt to capture how closely steganographic outputs mimic legitimate communication channels.

However, real-world application reveals critical reliability failures, most notably the Perceptual-Statistical Imperceptibility Conflict (Psic Effect). KLD and JSD scores increasingly diverge from human judgment as statistical optimization progresses: methods achieving superior divergence metrics often produce chaotic, low-quality text easily detected by human observers. This discrepancy manifests acutely in dataset dependency-identical methods yield KLDs of 19.507 on IMDB versus 8.295 on Twitter at equivalent embedding rates, rendering cross-paper comparisons meaningless. Further compounding this, researchers employ incompatible formulas (some using latent BERT features versus direct word distributions), feature spaces, and measurement scales, evidenced by Meteor's KLD ranging from 0.045 in one study to 7.491-11.845 in others. Consequently, these metrics function like rulers measuring paintings: they confirm technical dimensional accuracy while completely missing perceptual naturalness, necessitating parallel evaluation with human-centric measures to achieve genuine security.

5.3.4 Capacity Metrics. Capacity is judged by four metrics:

- **Bits per Token (BPT):**
$$\mathrm{BPT} = \frac{\text{Total Secret Bits}}{\text{Total Tokens}} \tag{1}$$
- **Bits per Word (BPW):**
$$\mathrm{BPW} = \frac{\text{Total Secret Bits}}{\text{Total Words}} \tag{2}$$
- **Embedding Rate (ER) [60]:** Average density of hidden information per textual unit
$$\mathrm{ER} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{bits}_i \tag{3}$$
where N is the number of textual units (words, tokens, or sentences) and bits_i is the number of bits embedded in the i-th unit.v
- **Utilization Rate:**
$$H = -\sum_{x\in\mathcal{X}} P(x)\log_2 P(x) \tag{4}$$
$$\mathrm{UR} = \left(\frac{\text{Actual Bits Embedded}}{H}\right) \times 100\% \tag{5}$$

These quantities quantify how densely a secret is packed, yet they are riddled with systematic biases that invalidate cross-system comparison.

Tokenization differences make "1 BPT" from one paper incomparable to "1 BPT" from another due to the use of different tokenizers. The Psic effect shows that higher density can hurt human fluency yet help statistical evasion. Model frequency preferences shrink the real alphabet to high-probability tokens, so naive entropy limits overstate usable space. Ambiguous reporting-practice vs. effective payload, ER1 vs. ER2, Bit Length vs. Stego Length-lets authors cherry-pick flattering numbers. Finally, BPW/BPT ignore semantics, rewarding gibberish that is obviously steganographic.

[Placeholder footnote]

Bias Category	Core Problem	Critical Implication
Tokenization Inconsistencies	Metrics depend entirely on specific tokenizers (e.g., GPT-2 BPE vs. word-level)	Direct comparisons across papers become meaningless when tokenization strategies differ
The "Psic Effect"	Conflicts between imperceptibility and statistical security are ignored	High capacity may degrade human fluency while paradoxically improving detection resistance
Model Training Bias	Utilization Rate calculations assume uniform token availability	Actual hiding space is smaller than theoretical entropy due to model frequency preferences
Reporting Ambiguities	No standard definition of "capacity" across systems	Practice payload vs. effective payload distinctions create misleading efficiency claims
Context Blindness	Density metrics treat text as neutral bit containers	Semantic incoherence constitutes a security failure that BPW/BPT fails to penalize

Table 6. Five primary bias categories affecting capacity metrics in steganographic evaluation

Recent works reveal additional distortions: loop-count overhead, dictionary-size caps, baseline-dependent “improvements,” watermarking goals that invert the desired signal, conversational filler that dilutes BPW, and nonlinear ER curves that make any single threshold misleading.

5.3.5 Security Metrics. The sources evaluate security and reliability through the following integrated metrics:

- **Detection Accuracy (Acc) / Error Rate (PE):** Measures a classifier’s ability to distinguish between cover and stego text. An accuracy of **50%** (or PE of 50%) is the gold standard, indicating the stego text is statistically identical to normal text.
- **F1 Score:** The harmonic mean of precision and recall, used to verify detection reliability, especially when datasets are balanced.
- **Robustness (Attack Resistance):** Evaluated via the **Mean Impact of Attack (mIOA)** or **Bit Accuracy** after removal attempts such as DAE (Denoising Auto-Encoder), word substitution, or sentence deletion.
- **False Positive Rate (FPR):** Measures how often human or normal model-generated texts are incorrectly flagged as containing secret messages.

5.3.6 Evaluation Challenges and Gaps. Several significant challenges exist in current evaluation practices:

- **Lack of Standardized Benchmarks:** The absence of shared benchmarks means only 20% of studies use common datasets, making comparison difficult
- **Inconsistent Reporting:** Different units, scales, and methodologies across studies
- **Limited Human Evaluation:** Only 25% of studies include human assessment
- **Missing Robustness Testing:** 60% of studies don’t test against various attacks
- **Incomplete Evaluation:** Many studies focus on only one or two metric categories

5.4 Integration of External Knowledge Sources (RQ4)

The integration of external knowledge sources has emerged as a crucial area of research in LLM-based steganography, with 65% of studies incorporating some form of external information. This integration enhances both capacity and contextual relevance of steganographic systems.

[Placeholder footnote]

Knowledge Type	Usage	Capacity Gain	Context Improvement	Examples
Semantic Resources	40%	+15-25%	High	Co-Stega, Knowledge Graphs
Domain Corpora	35%	+10-20%	Medium	FreStega, Specialized Datasets
Prompt Engineering	45%	+5-15%	High	Zero-shot methods
Context Retrieval	30%	+20-30%	Very High	Co-Stega, RAG integration

Table 7. External knowledge integration patterns and benefits

5.4.1 Semantic Resources Integration. Instead of asking the LLM to improvise, modern steganographic pipelines hand it a curated set of "conversation props" drawn from outside the model. A fast retriever first fetches a real tweet or headline that already whispers part of the secret; this authentic fragment becomes the semantic runway [26]. A lightweight knowledge-graph layer then appiles a handful of entity–relation–entity triples that tell the generator which facts must appear, guaranteeing long-range coherence without extra training [22]. Finally, an external n-gram frequency table nudges the softmax so that the token distribution clones everyday human chatter, erasing the statistical scar that detectors hunt for [40]. The LLM never changes; it just speaks through a stack of plug-ins that supply context, vocabulary variety and statistical camouflage-turning a solo monologue into a culturally grounded, high-capacity, statistically invisible conversation.

5.4.2 Domain Corpora Integration. Linguistic steganography now works by letting a model **absorb** a domain instead of hand-crafting rules.

Feed it enough examples-2.6 M tweets, 1.2 M IMDB reviews [61], BookCorpus [62], 530 k HTTP headers, 3.8 M news articles-and the internal weights re-shape themselves until the generated text statistically **is** that channel.

VAE-Stega [1], Meteor [2], Hi-Stega [26], FREmax [40], Rewriting-Stego [39] and Summarization-Stego [41] all follow this recipe: a specialised encoder/decoder (BERT, LSTM, GPT-2, BART) is fine-tuned on the target corpus so that every sentence it later produces already carries the right n-gram fingerprint, long-tail rare-word spectrum, or "news→comment" coherence.

Even hybrid systems such as Joint Linguistic Steganography add graph attention and CRF layers, but they still rely on the same premise-see enough real data and the distribution sticks.

When re-training is impossible or undesirable, black-box methods move the "domain memory" from parameters to prompts. Zero-shot Generative drops a handful of raw IMDB/Twitter samples into the context window and tells the LLM "write like this"; LLM-Stega wraps the request in an elaborated theme prompt ("entertainment news"); Co-Stega retrieves actual posts from the past seven days and feeds them in via an entropy-boosting template; Semantic Controllable injects Knowledge-Graph triples to steer long-form generation; ChatGPT Steganography simply lays down a microscopic rule set (exact word sequence, no plurals, no derivations) and lets the commercial API do the rest. No weights are changed, yet the output lands inside the desired statistical valley because the context itself has become the temporary training data.

5.5 Limitations and Trade-offs in Current Techniques (RQ5)

Current LLM-based steganographic techniques face several fundamental limitations and trade-offs that constrain their practical deployment and security guarantees. Understanding these limitations is crucial for advancing the field and developing more robust solutions.

Limitation	Impact	Frequency	Severity	Examples
Psic Effect	1-2 bpw loss	80%	High	DAIRstega, FreStega
Attack Vulnerability	5-50% drop	70%	High	Ensemble WM, TrojanStego
Low Capacity	<1 bpt in short texts	60%	Medium	Social media applications
Segmentation Issues	Ambiguity in extraction	40%	Medium	SparSamp, BPE tokenization
Ethical Concerns	Unaddressed bias	90%	High	TrojanStego, misuse potential

Table 8. Key limitations and their impact across studies

5.5.1 *Key Limitations.*

5.5.2 The Psic Effect: A Fundamental Trade-off. The Perceptual-Statistical Imperceptibility Conflict (Psic Effect) represents the most critical limitation, affecting 80% of studies. This fundamental trade-off occurs when optimizing for one aspect of imperceptibility degrades the other:

- **Perceptual Quality vs. Statistical Security:** Optimizing for low perplexity (PPL) often increases statistical detectability
- **Capacity Impact:** The Psic Effect results in an average capacity loss of 1-2 bits per word
- **Detection Resistance:** Higher capacity typically reduces anti-steganalysis accuracy

DAIRstega exemplifies this trade-off, where higher capacity reduces anti-steganalysis accuracy to 58%, demonstrating the inherent tension between different imperceptibility requirements.

5.5.3 Attack Vulnerability and Security Concerns. Current techniques demonstrate significant vulnerability to various attacks:

- **Paraphrasing Attacks:** Detection rates drop by 5-50% when text is paraphrased
- **Fine-tuning Attacks:** Model fine-tuning can significantly degrade steganographic performance
- **Statistical Analysis:** Advanced statistical methods can detect steganographic patterns
- **Adversarial Examples:** Malicious inputs can compromise steganographic systems

Examples include **Ensemble Watermarks**, which achieves 98% detection rate but drops to 95% following paraphrase attacks, and **TrojanStego**, which shows a dramatic drop from 97% to 65% under certain attack conditions.

5.5.4 Capacity Limitations in Short Texts. Hiding information in short, low-entropy texts presents significant challenges:

- **Social Media Posts:** Limited capacity in short, informal text
- **Low-Entropy Content:** Technical or formal documents offer limited hiding space
- **Semantic Constraints:** Maintaining meaning while embedding information
- **Context Requirements:** Short texts may lack sufficient context for effective hiding

5.5.5 Segmentation and Tokenization Issues. Segmentation ambiguities, primarily from subword tokenization, create ambiguity in message extraction:

- **BPE Tokenization:** Byte-pair encoding can split words unpredictably
- **Token Ambiguity:** Multiple valid segmentations of the same text

[Placeholder footnote]

- **Extraction Errors:** Ambiguous tokenization leads to message extraction failures
- **Capacity Caps:** Tokenization limits maximum achievable capacity

SparSamp demonstrates these issues, where token ambiguity (TA) reduces accuracy, and **ShiMer** cannot effectively boost entropy due to tokenization constraints.

5.5.6 Ethical Concerns and Misuse Potential. The field faces significant ethical challenges that remain largely unaddressed:

- **Bias and Discrimination:** Generated content may perpetuate harmful biases
- **Misuse Potential:** Techniques can be used for malicious purposes
- **Privacy Violations:** Steganographic systems may compromise user privacy
- **Regulatory Compliance:** Lack of frameworks for responsible use

TrojanStego exemplifies these concerns, as it can embed secrets directly into LLM outputs, potentially enabling data exfiltration and other malicious activities.

5.5.7 White-box vs. Black-box Trade-offs. The choice between white-box and black-box approaches involves fundamental trade-offs:

Aspect	**White-box**	**Black-box**	**Hybrid**
Security	High (95-99%)	Medium (79-91%)	Medium-High (90-95%)
Accessibility	Low	High	Medium
Capacity	High (1.1-5.98 bpt)	Medium (5.37 bpw)	Medium (2.0-4.0 bpt)
Imperceptibility	High (PPL: 3-8)	Low (PPL: 168-363)	Medium (PPL: 50-150)
Deployment	Difficult	Easy	Moderate

Table 9. Trade-offs between white-box, black-box, and hybrid approaches

5.5.8 Computational and Resource Constraints. Computational overhead, stemming from performance optimization, often conflicts with computational efficiency:

- **Computational Overhead:** Better results typically require more computational resources
- **Memory Requirements:** Large models and external knowledge increase memory needs
- **Real-time Constraints:** Latency requirements may limit optimization options
- **Scalability Issues:** Performance may degrade with increased scale

UTF demonstrates this trade-off, showing a 5% drop in HellaSwag performance, while **FreStega** requires corpus access (100 samples) for optimal performance.

5.5.9 Unresolved Challenges and Future Needs. Several critical challenges remain inadequately addressed:

- **Provable Security:** Lack of theoretical foundations for security guarantees
- **Robustness:** Limited resilience to advanced attack methods
- **Standardization:** Absence of common evaluation frameworks
- **Ethical Frameworks:** Missing guidelines for responsible development and use
- **Cross-lingual Support:** Poor performance in non-English languages
- **Real-world Deployment:** Limited testing in actual deployment scenarios

[Placeholder footnote]

Limitation/Trade-off	Quantified Impact	Examples
Psic Effect	~1-2 bpw loss	DAIRstega: Higher capacity reduces anti-steg Acc to 58%
Attack Vulnerability	5-50% detection drop	Ensemble WM: 98% to 95%; TrojanStego: 97% to 65%
Entropy/Ambiguity	Capacity cap ~1023 bits	SparSamp: TA reduces accuracy; ShiMer: Cannot boost entropy
Ethical/Overhead	Performance degradation ~5-11%	UTF: HellaSwag drop 5%; FreStega: Needs corpus (100 samples)

Table 10. Quantified impact of key limitations and trade-offs

5.5.10 Quantitative Impact Analysis. Table 10 provides a quantitative overview of the most significant trade-offs:

Understanding these limitations and trade-offs is essential for advancing the field and developing more robust, secure, and practical steganographic systems. Future research must address these challenges to enable widespread adoption and responsible use of LLM-based steganography.

6 DISCUSSION

This section provides a comprehensive discussion of the findings presented in the results section, synthesizing insights across all research questions and identifying implications for future research and practice.

6.1 Synthesis of Key Findings

The systematic review reveals a rapidly evolving field that has undergone significant transformation since 2023. The shift from white-box to black-box approaches represents a paradigm change toward more practical, real-world deployable steganographic systems. This evolution is driven by the increasing accessibility of large language models through APIs and the need for covert communication in censored environments.

6.2 Implications for Research and Practice

6.2.1 Methodological Implications. The findings suggest several important methodological considerations:

- **Standardization Need:** The lack of standardized evaluation metrics and benchmarks represents a critical barrier to progress. Future research should prioritize the development of common evaluation frameworks.
- **Evaluation Completeness:** The limited use of human evaluation (only 25% of studies) and robustness testing (40% missing) indicates a need for more comprehensive evaluation practices.
- **Reproducibility:** The variation in reporting standards and missing implementation details in many studies hampers reproducibility and comparison.

6.2.2 Practical Implications. For practitioners and developers:

- **Method Selection:** The choice between white-box and black-box methods should be based on security requirements vs. deployment constraints.
- **Capacity Planning:** The Psic Effect and capacity limitations in short texts should be carefully considered in system design.
- **Security Considerations:** The vulnerability to attacks (5-50% detection rate drops) requires robust defense mechanisms.

[Placeholder footnote]

6.3 Addressing the Psic Effect

The Perceptual-Statistical Imperceptibility Conflict emerges as the most significant challenge in the field. This fundamental trade-off between perceptual quality and statistical security affects 80% of studies and results in an average capacity loss of 1-2 bits per word. Future research should focus on:

- Developing techniques that minimize this trade-off
- Creating adaptive systems that balance both aspects dynamically
- Exploring novel approaches that decouple perceptual and statistical imperceptibility

6.4 The Role of Context and External Knowledge

The integration of external knowledge sources has proven crucial for enhancing both capacity and contextual relevance. However, this integration introduces new challenges:

- **Privacy Concerns:** External knowledge integration may compromise the privacy of the steganographic system
- **Computational Overhead:** The 5-15% increase in computational cost may limit real-time applications
- **Generalizability:** Domain-specific knowledge may not transfer well across different contexts

6.5 Ethical Considerations and Responsible Development

The review reveals a concerning gap in ethical considerations, with only 10% of studies addressing ethical implications. This represents a significant oversight given the potential for misuse in:

- Censorship evasion in authoritarian regimes
- Covert communication for malicious purposes
- Data exfiltration and information leakage
- Bias propagation through generated content

Future research must prioritize the development of ethical frameworks and responsible use guidelines.

6.6 Limitations of the Review

Several limitations of this systematic review should be acknowledged:

- **Incomplete Coverage:** 14 papers remained pending PDF acquisition, potentially missing important insights
- **Language Bias:** The focus on English-language publications may have excluded relevant non-English research
- **Recency Bias:** The rapid evolution of the field means some recent developments may not be fully captured
- **Quality Assessment:** The lack of formal quality assessment tools may have influenced the synthesis

6.7 Future Research Directions

Based on the synthesis of findings, several promising research directions emerge:

6.7.1 Technical Advancements.

- **Multimodal Steganography:** Integration with vision-language models for text-image combinations
- **Robust Defense Mechanisms:** Development of attack-resistant techniques
- **Provable Security:** Theoretical foundations for stronger security guarantees
- **Efficient Computation:** Reducing computational overhead for real-time applications

[Placeholder footnote]

6.7.2 Methodological Improvements.

- **Standardized Evaluation:** Development of common benchmarks and evaluation protocols
- **Human-Centered Design:** Greater emphasis on human evaluation and usability
- **Cross-Language Support:** Extension to non-English languages and cultural contexts
- **Real-World Testing:** Evaluation in actual deployment scenarios

6.7.3 Ethical and Social Considerations.

- **Ethical Frameworks:** Development of guidelines for responsible use
- **Bias Mitigation:** Techniques to prevent discrimination and bias propagation
- **Transparency:** Methods for detecting and auditing steganographic content
- **Regulatory Compliance:** Alignment with emerging AI regulations and standards

6.8 Conclusion

This systematic review has provided a comprehensive analysis of the current state of LLM-based steganography, revealing both significant progress and critical challenges. The field has evolved rapidly, with clear trends toward more practical and context-aware systems. However, fundamental limitations such as the Psic Effect, attack vulnerability, and ethical concerns remain inadequately addressed.

The findings suggest that future research should prioritize the development of standardized evaluation frameworks, robust defense mechanisms, and ethical guidelines. The integration of external knowledge sources shows promise but requires careful consideration of privacy and computational constraints. Most importantly, the field must address the ethical implications of these technologies to ensure their responsible development and deployment.

As LLMs continue to evolve and become more accessible, the field of linguistic steganography will likely see continued growth and innovation. The challenges identified in this review provide a roadmap for future research directions, while the opportunities suggest exciting possibilities for advancing both the technical capabilities and practical applications of these systems.

7 CONCLUSION

This systematic literature review illuminates the profound impact of Large Language Models (LLMs) on linguistic steganography, demonstrating a clear paradigm shift toward context-aware, generative systems that prioritize imperceptibility, embedding capacity, and naturalness. Through analysis of 26 primary studies (with 6 pending for full inclusion), key research questions were addressed, revealing that the published literature is rapidly evolving. Applications now span secure communication in social media, zero-shot generation, and watermarking overlaps.

Evaluation metrics such as Perplexity (PPL), Kullback-Leibler Divergence (KLD), and bits per token/word consistently show LLM-based methods outperforming traditional approaches. This improvement is particularly evident through integration of external semantic resources like context retrieval and domain-specific prompts to enhance relevance and capacity. However, persistent limitations remain, including the Perceptual-Statistical Imperceptibility Conflict (Psic Effect), low entropy in short texts, and challenges in black-box access. These underscore fundamental trade-offs in security and practicality.

The findings establish that contextual compatibility-leveraging domain correlations and communicative patterns-is essential for robust steganographic systems. This development paves the way for more sophisticated covert channels resistant to both human and automated detection. These advancements hold significant implications for information

[Placeholder footnote]

security, enabling high-capacity hidden messaging in everyday digital interactions while mitigating risks such as hallucinations and biases in LLMs.

Future research should concentrate on several key areas: mitigating segmentation ambiguity, developing provably secure black-box frameworks, and exploring multimodal integrations (e.g., text with images) to bridge identified gaps. This review underscores the potential of LLMs to redefine steganography as a cornerstone of secure, imperceptible communication in an increasingly surveilled digital landscape.

REFERENCES

[1] Zhong-Liang Yang, Si-Yu Zhang, Yu-Ting Hu, Zhi-Wen Hu, and Yong-Feng Huang. Vae-stega: linguistic steganography based on variational auto-encoder. *IEEE Transactions on Information Forensics and Security*, 16:880–895, 2020.

[2] Gabriel Kaptchuk, Tushar M Jois, Matthew Green, and Aviel D Rubin. Meteor: Cryptographically secure steganography for realistic distributions. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 1529–1548, Virtual Event, Republic of Korea, 2021. ACM.

[3] Yihao Wang, Ru Zhang, Yifan Tang, and Jianyi Liu. State-of-the-art advances of deep-learning linguistic steganalysis research. In *2023 International Conference on Data, Information and Computing Science (CDICS)*, page 20–24. IEEE, December 2023. doi: 10.1109/cdics61497.2023.00014. URL http://dx.doi.org/10.1109/CDICS61497.2023.00014.

[4] Gustavus J Simmons. The prisoners' problem and the subliminal channel. In *Advances in Cryptology: Proceedings of Crypto 83*, pages 51–67, Boston, MA, 1984. Springer.

[5] Jessica Fridrich. *Steganography in digital media: principles, algorithms, and applications.* Cambridge University Press, Cambridge, UK, 2009.

[6] Changhao Ding, Zhangjie Fu, Zhongliang Yang, Qi Yu, Daqiu Li, and Yongfeng Huang. Context-aware linguistic steganography model based on neural machine translation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:868–878, 2023.

[7] Murray Shanahan. Talking about large language models. *Communications of the ACM*, 67(2):68–79, 2024.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[9] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

[10] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.

[11] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with GPT-4. *arXiv preprint arXiv:2303.12712*, 2023.

[12] Yue Zhang, Siqi Sun, Michel Galley, Chris Brockett, and Jianfeng Gao. Language models as zero-shot style transferers. *arXiv preprint arXiv:2303.03630*, 2023.

[13] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[14] Aiyuan Yang, Bin Xiao, Binyuan Wang, Binxin Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*, 2023.

[15] Jiaxuan Wu, Zhengxian Wu, Yiming Xue, Juan Wen, and Wanli Peng. Generative text steganography with large language model. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 10345–10353, Melbourne, Australia, 2024. ACM.

[16] Guorui Liao, Jinshuai Yang, Kaiyi Pang, and Yongfeng Huang. Co-stega: Collaborative linguistic steganography for the low capacity challenge in social media. In *Proceedings of the 2024 ACM Workshop on Information Hiding and Multimedia Security*, pages 7–12, Baiona, Spain, 2024. ACM.

[17] Ke Lin, Yiyang Luo, Zijian Zhang, and Ping Luo. Zero-shot generative linguistic steganography. *arXiv preprint arXiv:2403.10856*, 2024.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[19] Biao Yi, Hanzhou Wu, Guorui Feng, and Xinpeng Zhang. Alisa: Acrostic linguistic steganography based on bert and gibbs sampling. *IEEE Signal Processing Letters*, 29:687–691, 2022.

[20] Jinyang Ding, Kejiang Chen, Yaofei Wang, Na Zhao, Weiming Zhang, and Nenghai Yu. Discop: Provably secure steganography in practice based on distribution copies. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 2238–2255, San Francisco, CA, USA, 2023. IEEE.

[21] Yuang Qi, Kejiang Chen, Kai Zeng, Weiming Zhang, and Nenghai Yu. Provably secure disambiguating neural linguistic steganography. *IEEE Transactions on Dependable and Secure Computing*, 2024. Early Access.

[22] Yihao Li, Ru Zhang, Jianyi Liu, and Qi Lei. A semantic controllable long text steganography framework based on llm prompt engineering and knowledge graph. *IEEE Signal Processing Letters*, 2024.

[23] Lingyun Xiang, Jiali Xia, Yangfan Liu, and Yan Gui. Cpg-ls: Causal perception guided linguistic steganography. *IEEE Signal Processing Letters*, 30: 1762–1766, 2023.

[Placeholder footnote]

[24] Martin Steinebach. Natural language steganography by chatgpt. In *Proceedings of the 19th International Conference on Availability, Reliability and Security*, pages 1–9. ACM, 2024.

[25] Zhenyu Xu, Ruoyu Xu, and Victor S Sheng. Beyond binary classification: Customizable text watermark on large language models. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2024.

[26] Huili Wang, Zhongliang Yang, Jinshuai Yang, Yue Gao, and Yongfeng Huang. Hi-stega: A hierarchical linguistic steganography framework combining retrieval and generation. In *International Conference on Neural Information Processing*, pages 41–54. Springer, 2023.

[27] Xiaoyan Zheng, Yurun Fang, and Hanzhou Wu. General framework for reversible data hiding in texts based on masked language modeling. In *2022 IEEE 24th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6. IEEE, 2022.

[28] Jipeng Qiang, Shiyu Zhu, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. Natural language watermarking via paraphraser-based lexical substitution. *Artificial Intelligence*, 317:103859, 2023.

[29] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623, Virtual Event, Canada, 2021. ACM.

[30] Mohammed Abdul Majeed, Rossilawati Sulaiman, Zarina Shukur, and Mohammad Kamrul Hasan. A review on text steganography techniques. *Mathematics*, 9(21), 2021. ISSN 2227-7390. doi: 10.3390/math9212829. URL https://www.mdpi.com/2227-7390/9/21/2829.

[31] De Rosal Ignatius Moses Setiadi, Sudipta Kr Ghosal, and Aditya Kumar Sahu. Ai-powered steganography: Advances in image, linguistic, and 3d mesh data hiding – a survey. *Journal of Future Artificial Intelligence and Technologies*, 2(1):1–23, Apr. 2025. doi: 10.62411/faith.3048-3719-76. URL https://faith.futuretechsci.org/index.php/FAITH/article/view/76.

[32] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 2019. URL https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

[33] Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

[34] Llama 2: Open foundation and fine-tuned chat models, 2023. URL https://arxiv.org/abs/2307.09288.

[35] Jianfei Xiao, Yancan Chen, Yimin Ou, Hanyi Yu, Kai Shu, and Yiyong Xiao. Baichuan2-sum: Instruction finetune baichuan2-7b model for dialogue summarization, 2024. URL https://arxiv.org/abs/2401.15496.

[36] Siyu Zhang, Zhongliang Yang, Jinshuai Yang, and Yongfeng Huang. Linguistic steganography: From symbolic space to semantic space. *IEEE Signal Processing Letters*, 28:11–15, 2020.

[37] Kai Petersen, Sairam Vakkalanka, and Ludwik Kuzniarz. Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64, 08 2015. doi: 10.1016/j.infsof.2015.03.007.

[38] Si-yu Zhang, Zhongliang Yang, Jinshuai Yang, and Yongfeng Huang. Provably secure generative linguistic steganography. *CoRR*, abs/2106.02011, 2021. URL https://arxiv.org/abs/2106.02011.

[39] Fanxiao Li, Sixing Wu, Jiong Yu, Shuoxin Wang, BingBing Song, Renyang Liu, Haoseng Lai, and Wei Zhou. Rewriting-stego: generating natural and controllable steganographic text with pre-trained language model. In *International Conference on Database Systems for Advanced Applications*, pages 617–626. Springer, 2023.

[40] Kaiyi Pang, Minhao Bai, Jinshuai Yang, Huili Wang, Minghu Jiang, and Yongfeng Huang. Fremax: A simple method towards truly secure generative linguistic steganography. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4755–4759. IEEE, 2024.

[41] Ruifan Zhang, Jianyi Liu, and Ru Zhang. Controllable semantic linguistic steganography via summarization generation. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4560–4564. IEEE, 2024.

[42] Changhao Ding, Zhangjie Fu, Qi Yu, Fan Wang, and Xianyi Chen. Joint linguistic steganography with bert masked language model and graph attention network. *IEEE Transactions on Cognitive and Developmental Systems*, 16(2):772–781, 2023.

[43] Zhe Ji, Qiansiqi Hu, Yicheng Zheng, Liyao Xiang, and Xinbing Wang. A principled approach to natural language watermarking. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 2908–2916. ACM, 2024.

[44] Travis Munyer, Abdullah All Tanvir, Arjon Das, and Xin Zhong. Deeptextmark: a deep learning-driven text watermarking approach for identifying large language model generated text. *Ieee Access*, 12:40508–40520, 2024.

[45] Fanxiao Li, Ping Wei, Tingchao Fu, Yu Lin, and Wei Zhou. Imperceptible text steganography based on group chat. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2024.

[46] Jifei Hao, Jipeng Qiang, Yi Zhu, Yun Li, Yunhao Yuan, Xiaocheng Hu, and Xiaoye Ouyang. Robust and semantic-faithful post-hoc watermarking of text generated by black-box language models. *Frontiers of Computer Science*, 19(9):199357, 2025.

[47] Antonio Norelli and Michael Bronstein. Llms can hide text in other text of the same length. *arXiv preprint arXiv:2510.20075*, 2025.

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

[49] Michael Brennan, Sadia Afroz, and Rachel Greenstadt. Adversarial stylometry: Circumventing authorship recognition to preserve privacy and anonymity. *ACM Transactions on Information and System Security (TISSEC)*, 15(3):1–22, 2012.

[50] George Mikros. Large language models and forensic linguistics: Navigating opportunities and threats in the age of generative ai. *arXiv preprint arXiv:2512.06922*, 2025.

[51] Martin Kuo, Jianyi Zhang, Aolin Ding, Qinsi Wang, Louis DiValentin, Yujia Bao, Wei Wei, Hai Li, and Yiran Chen. H-cot: Hijacking the chain-of-thought safety reasoning mechanism to jailbreak large reasoning models, including openai o1/o3, deepseek-r1, and gemini 2.0 flash thinking. *arXiv*

[Placeholder footnote]

preprint arXiv:2502.12893, 2025.

[52] Artur Zolkowski, Kei Nishimura-Gasparian, Robert McCarthy, Roland S Zimmermann, and David Lindner. Early signs of steganographic capabilities in frontier llms. *arXiv preprint arXiv:2507.02737*, 2025.

[53] Antonio-Gabriel Chacón Menke, Phan Xuan Tan, and Eiji Kamioka. Annotating the chain-of-thought: A behavior-labeled dataset for ai safety. *arXiv preprint arXiv:2510.18154*, 2025.

[54] Fengqing Jiang, Zhangchen Xu, Yuetai Li, Luyao Niu, Zhen Xiang, Bo Li, Bill Yuchen Lin, and Radha Poovendran. Safechain: Safety of language models with long chain-of-thought reasoning capabilities. *arXiv preprint arXiv:2502.12025*, 2025.

[55] F. Jelinek, R. L. Mercer, L. R. Bahl, and J. K. Baker. Perplexity-a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63, 08 2005. ISSN 0001-4966. doi: 10.1121/1.2016299. URL https://doi.org/10.1121/1.2016299.

[56] Yequan Wang, Jiawen Deng, Aixin Sun, and Xuying Meng. Perplexity from plm is unreliable for evaluating text quality. *arXiv preprint arXiv:2210.05892*, 2022.

[57] Lizhe Fang, Yifei Wang, Zhaoyang Liu, Chenheng Zhang, Stefanie Jegelka, Jinyang Gao, Bolin Ding, and Yisen Wang. What is wrong with perplexity for long-context language modeling? *arXiv preprint arXiv:2410.23771*, 2024.

[58] Abby Morgan. Perplexity for LLM evaluation. Comet ML Blog, November 2024. URL https://www.comet.com/site/blog/perplexity-for-llm-evaluation/. Accessed: 2025-12-31.

[59] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951. ISSN 00034851. URL http://www.jstor.org/stable/2236703.

[60] Christian Cachin. An information-theoretic model for steganography. In David Aucsmith, editor, *Information Hiding*, pages 306–318, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg. ISBN 978-3-540-49380-8.

[61] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, page 142–150, USA, 2011. Association for Computational Linguistics. ISBN 9781932432879.

[62] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

Table 11. Summary of Results from Reviewed Papers

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
VAE-Stega: linguistic steganography based on va... [1]	BERTBASE (BERT-LSTM) (LSTM-LSTM) model was trained from scratch	2020.0	Twitter (2.6M sentences) IMDB (1.2M sentences) preprocessed	PPL: 28.879, ΔMP: 0.242, KLD: 3.302, JSD: 10.411, Acc: 0.600, R: 0.616	non-explicit	pre-text	text
General framework for reversible data hiding in... [27]	BERTBase	2022.0	BookCorpus	BPW=0.5335 F1=0.9402 PPL=134.2199	non-explicit	pre-text	text
Co-stega: Collaborative linguistic steganograph... [16]	Llama-2-7B-chat, GPT-2 (fine-tuned), Llama-2-13B	2024.0	Tweet dataset (for GPT-2 fine-tuning), Twitter (real-time testing)	SR1: 60.87%, SR2: 98.55%, Gen. Capacity: 44.91 bits, Entropy: 49.21 bits, BPW: 2.31, PPL: 16.75, SimCSE: 0.69	explicit	Social Media	text

Continued on next page

Table 11 – continued from previous page

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
Joint linguistic steganography with BERT masked... [42]	LSTM + attention for temporal context. GAT for spatial token relationships. BERT MLM for deep semantic context in substitution.	2023.0	OPUS	PPL=13.917 KLD=2.904 SIM=0.812 ER=0.365 (BN=2) Best Acc=0.575 (BERT classifier) FLOPs=1.834G	explicit	pre-text	text
Discop: Provably secure steganography in practi...	GPT-2	2023.0	IMDB	p=1.00 Total Time (seconds)=362.63 Ave Time ↓ (seconds/bit)=6.29E-03 Ave KLD ↓ (bits/token)=0 Max KLD ↓ (bits/token)=0 Capacity (bits/token)=5.76 E...	non-explicit	tuning + pretext	text

Continued on next page

Table 11 – continued from previous page

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
Generative text steganography with large langua... [15]	Any	2024.0	[Not specified]	Length: 13.333 (words). BPW: 5.93 bpw PPL: 165.76. Semantic Similarity (SS): 0.5881 LS-CNN Acc: 51.55%. BiLSTM-Dense Acc: 49.20%. Bert-FT Acc: 50...	explicit	[Not specified]	[Not specified]
Meteor: Cryptographically secure steganography ... [2]	GPT-2	2021.0	Hutter Prize, HTTP GET requests	GPT-2: 3.09 bits/token	non-explicit	tuning + pretext	text
Zero-shot generative linguistic steganography [17]	LLaMA2-Chat-7B (as the stegotext generator / QA model). GPT-2 (for NLS baseline and JSD evaluation)	2024.0	IMDB, Twitter	PPL: 8.81. JSDfull: 17.90 (x10[truncated]iicircum-2). JSDhalf: 16.86 (x10[truncated]iicircum-2). JSDzero: 13.40 (x10[truncated]iicircum-2) TS...	explicit	zero-shot + prompt	text

Continued on next page

[Placeholder footnote]

Table 11 – continued from previous page

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
Provably secure disambiguating neural linguisti... [21]	LLaMA2-7b (English), Baichuan2-7b (Chinese)	2024.0	IMDb dataset (100 texts/sample, 3 English sentences + Chinese translations)	Total Error: 0%, Ave KLD: 0, Max KLD: 0, Ave PPL: 3.19 (EN), 7.49 (ZH), Capacity: 1.03–3.05 bits/token, Utilization: 0.66–0.74, Ave Time: [truncat...	non-explicit	pretext	text
A principled approach to natural language water... [43]	Transformer-based encoder/decoder; BERT for distillation	2024.0	Web Transformer 2	Bit acc: 0.994 (K=None), 1.000 (DAE), 0.978 (Adaptive+K=S); Meteor Drop: [truncated]iitilde0.057; SBERT ↑: [truncated]iitilde1.227; Ownership R...	Yes; semantic-level embedding; synonym substitution using BERT	Yes; watermark message assigned categorical label (e.g., 4-bit → 1-of-16)	Yes; semantic embeddings via transformer encoder and BERT; SBERT distance as metric

Continued on next page

[Placeholder footnote]

Table 11 – continued from previous page

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
Context-aware linguistic steganography model ba... [6]	BERT (encoder), LSTM (decoder)	2024.0	WMT18 News Commentary (train/test), Yang et al. bits, Doc2Vec, 5,000 stego pairs (8:1:1 split)	BLEU: 30.5, PPL: 22.5, ER: 0.29, KL: 0.02, SIM: 0.86, Stego detection [truncated]iitilde16%	Yes	[Not specified]	GCF (global context), LMR (language model reference), Multi-head attention
DeepTextMark: a deep learning-driven text water... [44]	Model-independent; tested with OPT-2.7B	2024.0	Dolly ChatGPT (train/validate), C4 (test), robustness & sentence-level test sets	100% accuracy (multi-synonym, 10-sentence), mSMS: 0.9892, TPR: 0.83, FNR: 0.17, Detection: 0.00188s, Insertion: 0.27931s	NO	[Not specified]	[Not specified]
Hi-stega: A hierarchical linguistic steganograp... [26]	GPT-2	2024.0	Yahoo! News (titles, bodies, comments); 2,400 titles used	ppl: 109.60, MAUVE: 0.2051, ER2: 10.42, Δ(cosine): 0.0088, Δ(simcse): 0.0191	explicit	Social Media	Text

Continued on next page

Table 11 – continued from previous page

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
Linguistic steganography: From symbolic space t... [36]	CTRL (generation), BERT (semantic classifier)	2020.0	5,000 CTRL-generated texts per semanteme (n = 2–16); 1,000 user-generated texts for anti-steganalysis	Classifier Accuracy: 0.9880; Loop Count: 1.0160; PPL: 13.9565; Anti-Steganalysis Accuracy: [truncated]iitilde0.5	implicit	Text	Semanteme (α) as a vector in semantic spac
Natural language steganography by chatgpt [24]	[Not specified]	2024.0	Custom word sets for specific topics (e.g., 16×10-word sets for music reviews)	[Not specified]	Explicit	Specific Genre/Topic Text	Text
Natural language watermarking via paraphraser-b... [28]	Transformer (Paraphraser), BART (BARTScore), BERT (BLEURT, comparisons)	2023.0	ParaBank2, LS07, CoInCo, Novels, WikiText-2, IMDB, NgNews	LS07 P@1: 58.3, GAP: 65.1; CoInCo P@1: 62.6, GAP: 60.7; Text Recoverability: [truncated]iitilde88–90%	Explicit	[Not specified]	text

Continued on next page

[Placeholder footnote]

Table 11 – continued from previous page

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
Rewriting-Stego: generating natural and control... [39]	BART (bart-base2)	2023.0	Movie, News, Tweet	BPTS: 4.0, BPTC+S: 4.0, PPL: 62.1, Mean: 44.4, Variance: 2.1e04, Acc: 8.9%	not Explicit	[Not specified]	[Not specified]
ALiSa: Acrostic linguistic steganography based ... [19]	BERT (Google's BERTBase, Uncased)	2022.0	BookCorpus (10,000 natural texts for evaluation)	PPL: Natural = 13.91, ALiSa = 14.85; LS-RNN/LS-BERT Acc & F1 = [truncated]iitilde0.50; Outperforms GPT-AC/ADG in all cases	No	[Not specified]	[Not specified]
Imperceptible Text Steganography based on Group...	Qwen-7B-Chat	2024.0	HC3, DailyDialogue, COCO Descriptions	HC3: Bit 188.94, Stego 131.99, PPL 34.07, Mean 20.19, Var 0.1e04, F1 90.01%; DailyDialogue: Bit 188.94, Stego 89.37, PPL 53.88, Mean 20.13, Var 0....	Explicit	Social Media / Group Chat	Text (chat history and current input)

Continued on next page

Table 11 – continued from previous page

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
A Semantic Controllable Long Text Steganography...	Llama 7B Chat, Meta LLaMA2 7B Chat	2024.0	Story (ChatGPT), Post (Recipe Kaggle + ChatGPT), Ad (Mobile Kaggle + ChatGPT)	ppl ↓ >23%, Δppl ↓ >72% vs ADG/HC/Bin; detection accuracy ↓ >10% vs baselines	Explicit	Topical Content	KG triplets (e1, r, e2), task descriptions (D)
Beyond Binary Classification: Customizable Text...	gpt-3.5-turbo-instruct, OPT-6.7b, babbage-002, davinci-002 (others: ChatGPT, GPT-2–4, LLaMA)	2024.0	Realnewslike (C4, 500 samples, 100-token prompts + completions); Custom watermark dataset (short info <10 tokens)	AUC 0.98, FPR 0.00, FNR 0.00, [truncated]iitilde100% single-letter decoding, PPL close to human text	Implicit	General Text Generation	Text (evolving prompt + generated output)
CPG-LS: Causal Perception Guided Linguistic Ste...	BERTBase, Cased	2023.0	CC-100 corpus; 10k cover texts; 7:3 train-test split	PPL 36.5; Mauve 0.871; Payload 0.150 bits/word; BiLSTM-D Acc 0.387 F1 0.375; R-BI-C Acc 0.378 F1 0.366; TS-RNN Acc 0.380 F1 0.368	Implicit	Natural Language Text	Text, embeddings, vector matrix

Continued on next page

Table 11 – continued from previous page

Paper	Llm	Year	Dataset	Result	Context Aware	Categ Context	Representation Context
Controllable Semantic Linguistic Steganography ...	BERT + CRF	2024.0	Gigaword; CNN/Daily Mail	Rouge-1: 0.2212; Rouge-2: 0.0268; Rouge-L: 0.1609; Meteor: 0.1384; Cosine: 0.5911; Euclidean: 5.6386; Manhattan: 87.9534; Jaccard: 0.2022; Anti-ste...	Explicit	Social Media	Semantic features of input text; 384-dim dense vectors for evaluation
FREmax: A Simple Method Towards Truly Secure Ge...	GPT-2	2024.0	Tweet corpus (2.6M sents, 26.8M tokens), IMDB corpus (1.05M sents, 25.3M tokens)	Tweet: PPL 361.83, Entropy 48.21, Tokens 10.83, Distinct3 0.98, BPS 62.79, SI% 73.03. IMDB: PPL 169.66, Entropy 103.39, Tokens 23.80, Distinct3 0....	Implicit	General Text	N-gram frequency distribution stored in a look-up table

[Placeholder footnote]

Contents

[Placeholder footnote]

[Placeholder footnote]