KARYA ILHAM PUBLISHING
CONNECTING MINDS, BRIDGING IDEAS, INSPIRING READERS

International Journal of Advanced Research in Computational Thinking and Data Science

Journal homepage:
https://karyailham.com.my/index.php/ctds/index
ISSN: 3030-5225

Comparative Evaluation of Alpha-based Representative Binary Technique Toward Other Dual-bit Techniques Feature-based Method in Text Steganography

Sunariya Utama[1,*], Roshidi Din[1], Osman Ghazali[1], Hrudaya Kumar Tripathy[2], Nurul Hafizah Hazwani[3]

[1] School of Computing UUM College Arts and Sciences, Universiti Utara Malaysia, 06010, Sintok, Kedah, Malaysia
[2] Kalinga Institute of Industrial Technology, Bhubaneswar, Odisha India
[3] School of Quantitative Sciences UUM College Arts and Sciences, Universiti Utara Malaysia, 06010, Sintok, Kedah, Malaysia

ARTICLE INFO

Article history:
Received 15 December 2025
Received in revised form 29 December 2025
Accepted 1 January 2026
Available online 1 February 2026

Keywords:
Feature-based Method; information hiding, Alpha-based Representative Binary; One-Flow-2-bit; Bit-One-Count; evaluation metric

ABSTRACT

This paper investigates the performance of three distinct feature-based text steganography techniques designed for secure data concealment. It evaluates the novel Alpha-based Representative Binary technique of feature-based method, which uses letter case (capital and small) to represent binary data, alongside the One-Flow-2-bit technique, which classifies letters by their visual writability, and the Bit-One-Count technique, which uses the count of '1' bits in a letter's binary form. Using a standardized validation framework applied to a substantial dataset, the techniques are assessed on their core operational capabilities for embedding and extracting hidden messages. Key performance metrics, including precision, recall, accuracy, and F-measure, are employed to provide a quantitative and multi-faceted comparison. The findings offer critical insights into the relative strengths, limitations, and practical implications of each method, contributing to the evidence-based advancement of reliable text steganography for security applications. This paper provide also provides a clear, evidence-based framework for selecting effective text steganography methods in security applications.

1. Introduction

Text documents remain a fundamental and critically important medium for communication and record-keeping in the modern digital era. The demand for and reliance on textual documents continue to be exceptionally high, particularly within business and academic sectors. This reliance stems from the fact that a vast array of vital documentation including appointment letters, certificates, analytical reports, confidential agreements, and numerous other official records primarily exists in text medium [1-3]. This ubiquity makes text a prime target for malicious actors. Irresponsible intruders may seek to access, disclose, or maliciously alter sensitive information

* *Corresponding author.*
E-mail address: sunariya.utama@uum.edu.my

contained within these documents for personal gain or sabotage [4,5]. Consequently, the security of text documents is a paramount concern, as they are consistently exposed to significant risks. One specialized branch of information security dedicated to addressing such concerns is known as steganography.

Steganography involves the covert embedding of messages into different data formats, ensuring that the concealed content remains unnoticed by both individuals and automated systems [6,7]. This method is utilized in private correspondence, security frameworks, and the safeguarding of sensitive data, providing significant advantages to commercial, military, governmental, and various other organization. Emerging from the Greek term meaning "covered writing," steganography describes the method of hiding a secret message inside an ordinary object, effectively masking its presence. When applied digitally, this object or medium of data could be a picture, a sound recording, a video, or written text [8,9].

Hiding information within text presents a unique set of difficulties compared to other mediums. Written language contains very little redundant data that can be altered without notice. Consequently, successful text medium methods must employ sophisticated techniques to insert the hidden payload [10,11]. These techniques must meticulously maintain the original text's natural flow and believable appearance to avoid suspicion. The main objective is to protect private information by ensuring the hidden data remains undiscernible, which is a critical performance metric for any information hiding technique. Steganography implementations are broadly categorized into two main types. Fig. 1 illustrates the major categories of steganography and highlights the specific focus path of this paper.

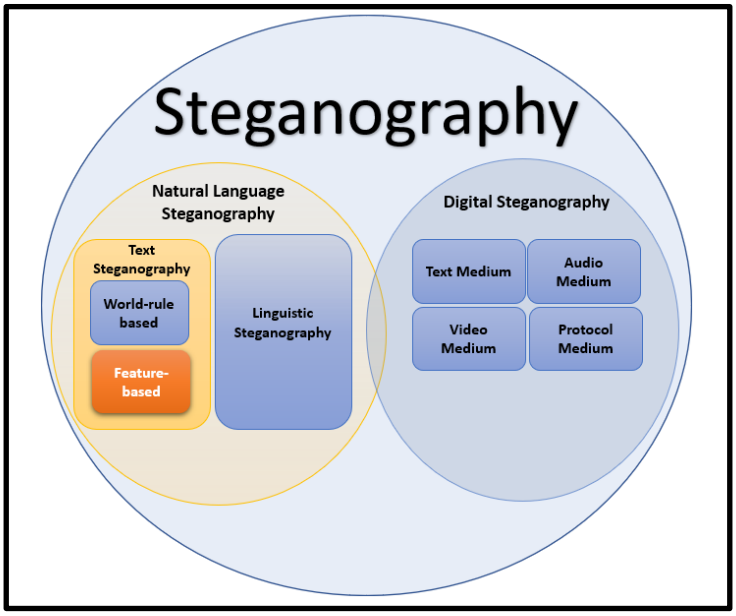

Fig. 1. Major category of implementation steganography

As depicted in Fig.1, the first major category is digital steganography, which involves embedding secret data into non-textual digital media such as image files, audio recordings, video streams, and other digitally encoded formats [12,13]. The second category is natural language steganography, where the act of hiding information is performed directly within a textual medium. This form of steganography focuses on concealing a secret message within ordinary text in such a manner that a third party examining the text cannot easily discern the presence of any additional, covert information. In essence, text-based steganography aims to make secret communications imperceptible and unremarkable to unintended viewers, while remaining accessible to authorized

recipients who possess the knowledge to extract it [14]. Within natural language steganography, two further sub-categories exist. Linguistic Steganography relies on manipulating the syntactic or grammatical structure of sentences. In contrast, Text Steganography manipulates the physical or structural components of the text itself such as individual words, line spacing, formatting features, or character attributes to embed hidden data [15,16].

This paper narrows its focus to a specific technique under the text steganography umbrella: the feature-Based method. This technique works by subtly altering specific visual or typographical features of characters within the text. This can include modifications to the shape, size, kerning, positioning, or font style of certain letters. The strength of the feature-based approach lies in its ability to create minute, visually negligible distortions that are exceedingly difficult for a casual reader to detect, thereby effectively concealing the embedded information [11,17].

This paper examines three text steganography techniques based on feature coding: One-Flow-2-bit [18], Bit-One-Count [19], and the Alpha-based Representative Binary technique [16]. The primary objective is to evaluate and compare their performance using established validation metrics, including precision, recall, accuracy, and F-measure [20]. The study uses a dataset of 400 text files to assess each technique's effectiveness in correctly embedding and extracting hidden messages while minimizing false positives and negative.

Establishing a clear and measurable objective is fundamental to advancing any technical field, and in text steganography, it is particularly critical. The primary objective of evaluating and comparing techniques using standardized validation metrics on precision, recall, accuracy, and F-measure in order to serve the discipline beyond speculative design into the realm of evidence-based engineering. By applying these metrics to a substantial dataset of 400 text files, it creates a controlled, reproducible experiment that tests the core promise of any steganography system that reliable and covert transmission of information [21]. This approach is important because it replaces subjective assertion with quantitative truth, revealing not just whether a technique can hide data. It also evaluates how consistently it does so, how often it fails, and how it performs relative to alternatives. Minimization false positives and negatives is not a secondary concern; it is directly tied to the technique's practicality and security [22]. A method that frequently mistakes normal text for secret messages (false positives) is inefficient and risky, while one that misses hidden messages (false negatives) is fundamentally broken. Therefore, this paper compares the implementation of Alpha-based Representative Binary in feature-based method with other dual-bit of text steganography.

2. Methodology

This section defines the comparison of Alpha-based Representative Binary in feature-based method that compare with the One-Flow-2-bit (OF-2), and Bit-One-Count (BOC) the specifies the embedding/extraction procedures, dataset characteristics, and validation metrics used to assess performance.

2.1 Dual-bit Technique of Feature-based Method

The main technique that become focus of this study is Alpha-based Representative Binary technique of feature-based method. This technique introduces a novel encoding mechanism predicated on the case of alphabetic characters. Its fundamental premise is to use capital and small letters as direct proxies for binary values. The embedding rule is defined in Table 1.

Table 1
Alpha-based Representative Binary Technique of feature-based method scheme

Represents		Letters												
Number Embed		1	2	3	4	5	6	7	8	9	10	11	12	13
01	Capital	A	B	C	D	E	F	G	H	I	J	K	L	M
10	Small	a	b	c	d	e	f	g	h	i	j	k	l	m
Number Embed		14	15	16	17	18	19	20	21	22	23	24	25	26
01	Capital	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
10	Small	n	o	p	q	r	s	t	u	v	w	x	y	z

Based on Table 1 it shows the scheme of Alpha-based Representative Binary Technique as part of the implementation steganography in text medium. The technique's core is a secret mapping table conceptually represented in the source as Table that governs the embedding process. This table does not simply map letters to bits; it tracks the positions and representations of letters within the generated stego text. To embed a message, the secret binary stream is segmented into pairs ('01' or '10'). For each pair, the algorithm, guided by the secret table, selects or modifies a character position in the cover text to ensure the letter's case corresponds to the required binary value. This process enhances complexity and capacity, as the hidden information is not in the characters themselves but in the pattern of their case alternation, controlled by a secret key. The table ensures that the mapping between textual positions and binary data remains concealed, directly contributing to the technique's security against detection. T

he Alpha-based Representative Binary technique introduces a position-aware encoding mechanism, where a secret mapping table governs not merely the case of individual letters, but the deliberate alternation of case across specific character positions within the stego-text to represent bit pairs. This approach fundamentally shifts security from simple case toggling to a scheme where the covert pattern is concealed within the positional sequence itself, controlled by a secret key, thereby enhancing complexity and resistance to detection compared to basic case-substitution methods.

This performance of this technique compares other the dual bit technique OF-2 and BOC techniques of feature-based method that show technique in Table 2.

Table 2
Scheme of OF-2 and BOC techniques of feature-based method

Technique	*Binary bit*	*Category*	*Letter used*
One-Flow-2-bit (OF-2)	00	Letters not writable in one flow and has no vertical or horizontal line	"Q, X"
	01	Letters not writable in one flow and has vertical or horizontal line	"A, B, D, E, F, H, K, T"
	10	Letters writable in one flow and has no vertical or horizontal line	"C, G, O, S, V, W"
	11	Letters writable in one flow and has no vertical or horizontal line	"I, J, L, M, N, P, U, Y, Z"
Bit-One-Count (BOC)	00	Capital letter Total 1 bit =4	"G, J, K, M, N, S, U, V, Y, Z"
	01	Capital letter Total 1 bit <4	"A, B, C, D, E, F, H, I, L, P, Q, R, T, X"
	10	Small letter Total 1 bit =4	"c, e, f, i, j, l, q, r, t, x"
	11	Small letter Total 1 bit <4	"g, k, m, n, o, s, u, v, w, y, z"

Table 2 shows the two techniques of the feature-based method that used as comparison with technique of Alpha-based Representative Binary. The first technique, OF-2 employs a graphological feature set, categorizing letters of the English alphabet based on two properties which are Writability in One Flow. It whether the letter can be written with a single, continuous pen stroke without lifting the writing instrument. Then, its presence of vertical or horizontal Lines that the letter's structure contains prominent straight vertical or horizontal segments. Based on these criteria, letters are divided into four distinct categories, each assigned a unique two-bit code (00, 01, 10, 11).

The second technique is the BOC that other comparison Alpha-based Representative Binary technique as part of feature-based method in text steganography. The embedding process involves replacing characters in the cover text with letters from the category corresponding to the two-bit segment of the hidden message that requires encoding. This method ties the payload directly to the visual form of the characters.

This technique utilizes a computational feature: the Hamming weight or "population count" of a letter's binary representation. The process involves:

1. Converting each letter (both capital and small forms) into a predefined binary code (e.g., based on ASCII or another mapping).
2. Counting the number of '1' bits in this binary representation.
3. Classifying the letter into one of four groups based on its case and this count.

Embedding is achieved by substituting cover text characters with letters from the category that matches the two-bit secret message segment. This technique links the hidden data to a non-perceptual, arithmetic property of the character's digital representation.

2.2 Experimental Design and Evaluation framework

To ensure a fair and rigorous comparison, a standardized validation procedure was implemented using the dataset generation in order to control corpus in creating testing.

- Cover Texts: 20 distinct, benign text documents (e.g., news articles, book excerpts) were selected.
- Hidden Messages: 20 different secret messages (of varying lengths) were prepared.
- Stego-text Generation: Each of the three techniques was used to embed each of the 20 hidden messages into each of the 20 cover texts. This cross-embedding process generated a total of 400 stego-text files per technique (20 x 20 = 400).

The success of each technique was measured by its ability to both *embed* and subsequently *correctly extract* the hidden message. This was formalized using a classification paradigm, defining four possible outcomes for each processed file:

- True Positive (TP): The hidden message is successfully embedded *and* accurately extracted from the stego text.
- False Positive (FP): A message is extracted, but it is *inaccurate* or corrupted (i.e., the extracted data does not match the original hidden message), even though the file was processed as a stego text.
- True Negative (TN): A non-stego file (plain cover text) is correctly identified as containing no hidden message. *(In this closed experiment where all files are processed for embedding, TN is expected to be zero).*
- False Negative (FN): A stego text is incorrectly identified as containing no hidden message (extraction fails entirely). *(Also expected to be zero if extraction is always attempted).*

From these outcomes, four standard evaluation metrics were calculated for each technique. The selection of precision, recall, accuracy, and F-measure as evaluation metrics is deliberate and aligns closely with the core security objectives of text steganography. In practical applications, a steganographic system must not only conceal data but also ensure its reliable and accurate retrieval. Precision reflects the system's ability to avoid false alarms extracting incorrect or corrupted messages which in a security context could lead to misinformation or unnecessary suspicion. Recall measures the system's completeness in recovering hidden messages; a low recall indicates missed communications, representing a critical failure in covert data transmission. Accuracy provides an overall measure of correctness, balancing both embedding and extraction fidelity, which is essential for trustworthy operation in real-world scenarios where errors could compromise mission-critical information. Finally, the F-measure harmonizes precision and recall, offering a single metric that guards against over-optimizing one aspect at the expense of the other. These metrics quantitatively capture the trade-offs between reliability, robustness, and stealth key factors that determine whether a steganography technique is viable for secure, real-world use. This addition can be placed after the bulleted list defining TP, FP, TN, FN, and before the equations for the metrics, thereby providing context and justification for their use in the evaluation:

1. **Precision:** Measures the correctness of the extracted messages. It answers: "all messages extracted as 'successful,' what fraction was actually correct?"
$$Precsion = \frac{TP}{TP+FP} \quad (1)$$

2. **Recall (Sensitivity):** Measures the completeness of successful extractions. It answers: "all the original hidden messages, what fraction was successfully and correctly extracted?"
$$Recall = \frac{TP}{TP+FN} \quad (2)$$

3. **Accuracy:** Measures the overall correctness of the system across all judgments.
$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

4. **F-measure:** The harmonic means of Precision and Recall, providing a single score that balances both concerns. It is especially useful when class distribution is uneven.
$$F-measure = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

These metrics collectively provide a multi-faceted view of each technique's performance, spanning fidelity (Precision), robustness (Recall), overall effectiveness (Accuracy), and balanced performance (F-measure) that utilize in evaluate the text steganography implementation.

3. Results

This section presents the empirical findings from the validation experiment and provides a critical analysis of the performance of the three feature-based methods of text steganography techniques.

3.1. Experimental Results

The results of applying the validation framework to the 400 stego-text files generated by each technique are compiled in Table 3 (Possible Outcomes).

Table 3
Possible outcomes for the three steganography techniques (n=400 files per technique)

Technique	TP	FP	TN	FN
One-Flow-2-bit (OF-2)	0	400	0	0
Bit-One-Count (BOC)	400	0	0	0
Alpha-based Representative Binary	120	280	0	0

The interpretation of Table 3 are;

- **Bit-One-Count** demonstrated flawless operational performance. All 400 embedding attempts resulted in successful and accurate extraction (TP=400), with zero extraction errors (FP=0).
- **One-Flow-2-bit** failed to produce a viable stego-text under the defined experimental conditions. In all 400 cases, the extracted message was inaccurate (FP=400), and no successful embeddings were recorded (TP=0). This suggests a fundamental flaw in the embedding logic, the extraction algorithm, or the compatibility between the chosen cover texts and the strict categorical requirements of the technique.
- **Alpha-based Representative Binary** showed a mixed outcome. It successfully embedded and extracted messages in 120 cases (TP=120). However, in 280 cases, while an extraction process occurred, the output was incorrect (FP=280).
- As anticipated in this full-embedding experiment, no scenarios occurred where a stego text was missed (FN=0) or a clean text was correctly identified as such (TN=0).

Table 4
Evaluation metrics derived from the possible outcomes

Technique	Precision	Recall	Accuracy	F-measure
One-Flow-2-bit (OF-2)	0%	0%	0%	0%
Bit-One-Count (BOC)	100%	100%	100%	100%
Alpha-based Representative Binary	30%	100%	30%	46.15%

Table 4 shows the evaluation metrics based on the three techniques; the calculations reveal a stark story about each method's reliability. The BOC technique was flawless because every single one of the 400 messages was both hidden and retrieved correctly, its numbers logically achieve 100% across the board. In contrast, OF-2 technique’s zeros are the simplest to explain: with not even one successful message recovery, every metric collapse to zero, showing it didn't function at all under these test condition. Then, the Alpha-based method's figures tell a more complicated tale. Its perfect recall of 100% simply means it never missed trying to extract a message it had embedded, but the low 30% precision exposes the real problem: most of those extracted messages were wrong. This happened because out of 400 tries, only 120 worked perfectly, while 280 produced errors. Since accuracy also factors in these correct extractions against the total attempts, it similarly drops to 30%. The F-measure of 46.15% is just the mathematical middle ground between its high recall and low precision, confirming that while the method is consistent in trying to recover data, it fails miserably at doing so accurately. The high recall indicates successful extraction when embedding occurred, but the low precision reflects a high rate of false positives, meaning many non-stego texts were incorrectly flagged as containing hidden data.

4. Discussion

The results reveal stark differences in the efficacy of the three techniques, prompting an analysis of their underlying mechanisms and implications.

4.1. Performance Analysis

- *Bit-One-Count: The Robust Performer***.** Achieving 100% across all metrics is exceptional. This indicates that the technique's mapping among binary message segments and letters categorized by their BOC is deterministic, reversible, and lossless under the tested conditions. The feature (bit count) is an intrinsic, digital property of the character's representation, making the encoding and decoding processes purely algorithmic and reliable. The separation of capital and small letters into distinct groups likely doubled the available symbol space, preventing conflicts and ensuring accurate mapping for every possible two-bit input.
- *Alpha-based Representative Binary: The High-Recall, Low-Precision Technique.* The 100% Recall score confirms that the technique never *failed to attempt* an extraction from a stego-text it created (FN=0). However, the low Precision (30%) and Accuracy (30%) uncover a critical issue: a high rate of extraction errors. This suggests the technique is vulnerable to conflicts or ambiguities during the embedding process. The reliance on a "secret table" for position mapping is likely the source of this inconsistency. If the embedding algorithm, when modifying case, inadvertently creates sequences that violate the table's rules for later decoding, or if the table logic is not perfectly bijective, errors will occur. The technique successfully *hides* data (case changes are subtle) but cannot reliably *recover* it in many instances, limiting its practical utility.
- *One-Flow-2-bit: The Non-Functional Technique.* A score of 0% across the board signifies complete failure in this experiment. The most plausible explanation is a fundamental mismatch or flaw in the implementation of its core classification logic. The requirement to find letters that are both "writable in one flow" AND have/not have "vertical or horizontal lines" may be too restrictive, leading the embedding algorithm to fail when no suitable character from the required category exists in the cover text segment. Alternatively, the extraction logic may be incorrectly interpreting the categories. This result highlights the risk of basing steganography on complex, subjective, or poorly defined visual features that may not translate reliably to a digital selection and substitution process.

The Alpha-based method's performance reveals a fundamental design flaw. Its high recall proves the mechanism can always initiate data retrieval, yet its critically low precision indicates the extracted data is frequently corrupted. The observed ambiguity stems not from an inherent flaw in case-based encoding as a concept, but from a design-specific vulnerability in this technique's implementation—namely, the construction and application of its secret mapping table. This suggests the failure is mitigable; a more rigorously designed, conflict-free mapping algorithm could preserve the subtlety of case-based encoding while achieving the reliability required for practical use.

4.2. Implications for Text Steganography Principles

- Capacity & Undetectability Trade-off: Bit-One-Count, while perfectly reliable, may have lower undetectability compared to the Alpha-based method. Systematically substituting letters based on a digital property could, in theory, create statistical anomalies in letter frequency (e.g., over-representation of letters in the '=4' and '>4' small letter groups). The Alpha-based Representative Binary method, altering only case, is more subtle linguistically

but pays the price in reliability (capacity is effectively wasted by errors). One-Flow-2-bit failed on both fronts.

- Security Considerations: The "secret table" of the Alpha-based method is a form of security through obscurity, a weak form of security. If the table is discovered, the entire scheme is compromised. Bit-One-Count's security lies primarily in the secrecy of the classification scheme itself. Neither method employs strong cryptography on the payload before embedding, which is a recommended practice for true secrecy.
- Practical Applicability: BOC is immediately applicable for scenarios requiring high-fidelity, covert communication where the primary threat is detection of *communication* rather than deep analysis of text statistics. The Alpha-based method, in its current form, is unsuitable for reliable communication due to its error rate. The OF-2 method requires a complete re-evaluation of its foundational categories and algorithms.

The technique's profile forces a difficult trade-off between stealth and reliability. Altering only letter case provides excellent linguistic subtlety, but this covertness comes at the direct cost of functional capacity due to high error rates.

4.3. Limitation and Future Work

This study was conducted in a controlled, noise-free environment. Future work must subject these techniques in order to more rigorous steganalysis. This includes:

- Testing against statistical language models that detect unnatural character or word distributions.
- Measuring the impact on text readability and semantic coherence post-embedding.
- Evaluating performance with different text genres and languages.
- For the Alpha-based method, investigating and rectifying the causes of the high FP rate, potentially by designing a more robust and conflict-free mapping algorithm.
- Exploring hybrid models that combine the subtlety of case modification (like Alpha-based Representative binary) with the reliability of a deterministic coding scheme.

An important limitation of this study is its reliance on a controlled corpus of English-language texts with uniform font encoding, which does not account for the variability present in real-world digital documents. The performance of feature-based techniques may be adversely affected by factors such as multilingual content, font rendering differences, automated text preprocessing (e.g., trimming or sanitization), or common user actions like copy-paste and document reformatting, which can inadvertently alter or strip subtle feature embedding

Furthermore, this evaluation was conducted under idealized conditions using a homogeneous dataset of English texts with consistent font encoding, which does not reflect the heterogeneity of real-world digital documents. The performance of these techniques may be significantly impacted by factors such as multilingual text, font variation, automated text preprocessing, or common digital transformations like copy-paste operations and reformatting, which could corrupt feature-based embedding process.

The demonstrated highest performance of the Bit-One-Count technique within this study stems from its foundation on a deterministic, computational feature the Hamming weight of a character's binary form. This design ensures a consistent, one-to-one mapping between a two-bit secret segment and a specific letter category, enabling perfectly reversible encoding and decoding under the stable conditions of the experiment. Its flawlessness in the results is a direct outcome of this unambiguous, algorithm-driven process, free from the subjective or positional ambiguities that hindered the other

methods. However, this very dependency on fixed digital representations is what may limit its robustness in practical, variable text environments. Future work particularly on the proposed technique, must validate steganographic robustness across diverse, noisy, and dynamically altered text environments to assess practical utility.

5. Conclusions

This paper conducted a systematic empirical evaluation of three feature-coding text steganography techniques: the novel Alpha-based Representative Binary as the proposed technique with other dual-bit technique feature-based method that implement the graphology based OF-2, and the computation of BOC. The results provide clear that quantified insights into their operational effectiveness.

The Bit-One-Count technique emerged as the unequivocally superior method under the tested conditions, achieving perfect scores (100%) in Precision, Recall, Accuracy, and F-measure. Its deterministic algorithm, which classifies and substitutes letters based on the count of '1' bits in their binary representation, proved to be fully reliable for both embedding and error-free extraction across a large dataset. This demonstrates that feature-coding schemes based on unambiguous, digital character properties can achieve high performance.

Conversely, the Alpha-based Representative Binary technique exhibited a significant weakness. While it successfully initiated extraction for all embedded messages (100% Recall), it suffered from a 70% error rate in the extraction output, leading to low Precision and Accuracy (30%). This indicates that its core mechanism based on a secret table to map letter case to binary value prone to conflicts or ambiguities that compromise data fidelity, rendering it unreliable for practical secure communication despite its potentially high steganographic subtlety.

The One-Flow-2-bit technique failed completely in this experiment, yielding 0% across all performance metrics. This suggests that its foundational premise of categorizing letters by "writability in one flow" and linear components may be unsuitable for a robust, automated digital steganography system, likely due to overly restrictive or ambiguous classification criteria in hiding information system.

In conclusion, for applications requiring dependable and accurate covert communication within text, the Bit-One-Count technique represents a robust and effective feature-coding solution. Future research should focus on enhancing its resistance to statistical steganalysis while exploring ways to salvage the innovative case-based approach of the Alpha-based method by improving its encoding reliability. The pursuit of text steganography that optimally balances high capacity, strong undetectability, and flawless robustness remains a vital and challenging frontier in information security.

Acknowledgement

This research was not funded by any grant.

References

[1] Din, Roshidi, Rosmadi Bakar, Sunariya Utama, Jamaluddin Jasmis, and Shamsul Jamel Elias. "The evaluation performance of letter-based technique on text steganography system." *Bulletin of Electrical Engineering and Informatics* 8, no. 1 (2019): 291-297. https://doi.org/10.11591/eei.v8i1.1440.

[2] Krishnan, R. Bala, Prasanth Kumar Thandra, and M. Sai Baba. "An overview of text steganography." In *2017 fourth international conference on signal processing, communication and networking (ICSCN)*, pp. 1-6. IEEE, 2017. https://doi.org/10.1109/ICSCN.2017.8085643.

[3] Michaylov, K. D. "Exploring the use of steganography and steganalysis in forensic investigations for analysing digital evidence." Bachelor's thesis, University of Twente, 2023.
[4] Patiburn, Sivabalan AL, Vahab Iranmanesh, and Phoey Lee Teh. "Text steganography using daily emotions monitoring." *International Journal of Education and Management Engineering* 7, no. 3 (2017): 1.https://doi.org/10.5815/ijeme.2017.03.01.
[5] Alkhudaydi, Malak, and Adnan Gutub. "Securing data via cryptography and arabic text steganography." *SN Computer Science* 2, no. 1 (2021): 46. https://doi.org/10.1007/s42979-020-00438-y.
[6] Rajan, Nagalinga, and R. Sunder. "HIDING TEXT IN DIGITAL IMAGES USING PERMUTATION ORDERING AND COMPACT KEY BASED DICTIONARY." *ICTACT Journal on Image & Video Processing* 7, no. 4 (2017). https://doi.org/10.21917/ijivp.2017.0214.
[7] Utama, Sunariya, and Roshidi Din. "Performance Review of Feature-Based Implementation Text Steganography Approach Method." Journal of Information Systems and Digital Technologies 2, no. 2 (2022): 325–333.
[8] Changder, Suvamoy, Narayan C. Debnath, and Debidas Ghosh. "A Greedy approach to text steganography using properties of sentences." In *2011 Eighth International Conference on Information Technology: New Generations*, pp. 30-35. IEEE, 2011. https://doi.org/10.1109/ITNG.2011.13.
[9] Majumder, Anandaprova, and Suvamoy Changder. "A novel approach for text steganography: generating text summary using reflection symmetry." *Procedia Technology* 10 (2013): 112-120. https://doi.org/10.1016/j.protcy.2013.12.343.
[10] Thabit, Reema, Nur Izura Udzir, Sharifah Md Yasin, Aziah Asmawi, Nuur Alifah Roslan, and Roshidi Din. "A comparative analysis of Arabic text steganography." *Applied Sciences* 11, no. 15 (2021): 6851. https://doi.org/10.3390/app11156851.
[11] Din, Roshidi, Sunariya Utama, and Aida Mustapha. "Evaluation review on effectiveness and security performances of text steganography technique." *Indonesian Journal of Electrical Engineering and Computer Science* 11, no. 2 (2018): 747-754. https://doi.org/10.11591/ijeecs.v11.i2.pp747-754.
[12] Kuznetsov, Alexandr, Nicolas Luhanko, Emanuele Frontoni, Luca Romeo, and Riccardo Rosati. "Image steganalysis using deep learning models." *Multimedia Tools and Applications* 83, no. 16 (2024): 48607-48630.https://doi.org/10.1007/s11042-023-17591-0.
[13] Din, Roshidi, Reema Ahmed Thabit, Nur Izura Udzir, and Sunariya Utama. "Traid-bit embedding process on Arabic text steganography method." *Bulletin of Electrical Engineering and Informatics* 10, no. 1 (2021): 493-500.https://doi.org/10.11591/eei.v10i1.2518.
[14] Xiang, Lingyun, Shuanghui Yang, Yuhang Liu, Qian Li, and Chengzhang Zhu. "Novel linguistic steganography based on character-level text generation." *Mathematics* 8, no. 9 (2020): 1558. https://doi.org/10.3390/math8091558.
[15] Chang, Ching Yun, and Stephen Clark. "Practical linguistic steganography using contextual synonym substitution and a novel vertex coding method." *Computational linguistics* 40, no. 2 (2014): 403-448. https://doi.org/10.1162/COLI_a_00176.
[16] Din, Roshidi, Sunariya Utama, Hrudaya Kumar Triphaty, and Jabbar Qasim Almalik. "Optimising Text Steganography with Alpha-Based Representative Binary: Analysing Contain Letter Used and Capacity Ratio." *Journal of Advanced Research in Computing and Applications* 36, no. 1 (2024): 43-51.
[17] Chaudhary, Shivangi, Manoj Dave, and Ashish Sanghi. "Text Steganography Based on Feature Coding Method." In Proceedings of the 2016 International Conference on Computing, Communication and Automation (ICCCA), 5–8. 2016. https://doi.org/10.1145/2979779.2979786.
[18] Kouser, Saeeda, Aihab Khan, and Ejaz Qamar. "A novel content-based feature extraction approach: Text steganography." *International Journal of Computer Science and Information Security* 14, no. 12 (2016): 916.
[19] Azeem, Muhammad, Cai Yongquan, Khurram Gulzar Rana, Zeeshan Shaukat, and Allah Ditta. "A secure and size efficient approach to enhance the performance of text steganographic algorithm." In *2019 11th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, pp. 402-407. IEEE, 2019. https://doi.org/10.1109/ICMTMA.2019.00095.
[20] Olateju, Omobolaji, Samuel Ufom Okon, Udochukwu Igwenagu, Abidemi Ayodotun Salami, Tunboson Oyewale Oladoyinbo, and Oluwaseun Oladeji Olaniyi. "Combating the challenges of false positives in AI-driven anomaly detection systems and enhancing data security in the cloud." *Available at SSRN 4859958* (2024).https://doi.org/10.9734/ajrcos/2024/v17i6472.
[21] Adeniyi, A. E., K. M. Abiodun, Joseph Bamidele Awotunde, Mukaila Olagunju, O. S. Ojo, and N. P. Edet. "Implementation of a block cipher algorithm for medical information security on cloud environment: using modified advanced encryption standard approach." *Multimedia Tools and Applications* 82, no. 13 (2023): 20537-20551. https://doi.org/10.1007/s11042-023-14338-9.

[22] Tian, Hui, Yanpeng Wu, Chin-Chen Chang, Yongfeng Huang, Yonghong Chen, Tian Wang, Yiqiao Cai, and Jin Liu. "Steganalysis of adaptive multi-rate speech using statistical characteristics of pulse pairs." *Signal Processing* 134 (2017): 9-22. https://doi.org/10.1016/j.sigpro.2016.11.013.