SN-Stego: Dataset for Social Networks Text Steganalysis via Local Group Discovery and Sample Distribution Regulation

Qiong Xu[a], Ru Zhang[†a], Jianyi Liu[a], Yongfeng Huang[b,c]

[a]School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China
[b]Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
[c]Zhongguancun Laboratory, Beijing 100094, China

Abstract

Social networks' rapid information dissemination, massive user bases, and diverse content make them vulnerable to text steganography—a covert technique embedding secret messages into texts undetected, threatening personal privacy and network security. While text steganalysis serves as a critical defense mechanism, existing datasets for this task suffer from critical limitations including missing social graphs, insufficient text attributes, mismatched sample distributions, and limited data scale, hindering research progress. To address these gaps, this paper proposes a novel methodology for constructing a social network text steganalysis dataset via meta path-constrained local group discovery and sample distribution dynamic regulation. It utilizes a local group discovery algorithm constrained by "user-tweet-hashtag" meta path to sample special user groups with potential covert communication intentions. In addition, a three-dimensional dynamic regulation strategy is designed to reshape the original tweets of the special users by adjusting the ratio, type, and distribution of steganographic texts, simulating complex and diverse covert communication patterns. Finally, a dataset is constructed with rich social graph information, namely SN-stego. It conforms to the characteristics of text fragmentation and steganography sparsity in real social networks, and simulates various social network text steganography analysis scenarios with complex and diverse sample distributions. Statistical analyses and empirical evaluations demonstrate that SN-stego exhibits substantial advancements in data scale, entity diversity, and scenario adaptability. The proposed method provides solid technical support for expanding and deepening the research on text steganalysis in social networks.

Keywords: social network; text steganalysis; dataset; local group discovery

1. Introduction

In the era of the information revolution, social networks have become an indispensable part of daily life. Not only does it change people's communication patterns, it also plays a crucial role in information dissemination, public opinion formation, commercial marketing, and other aspects. However, as social networks proliferate, information security challenges have surfaced. Text steganography[1, 2, 3, 4, 5], a technique that conceals secret messages within seemingly ordinary texts, has emerged as a critical tool for attackers to conduct covert communications and

[†]Corresponding author: Ru Zhang (Email: zhangru@bupt.edu.cn; ORCID:0000-0001-6641-3236)

disseminate malicious information. Accurate analysis and identification of steganographic text (which is usually termed "stego") in social networks to detect and mitigate potential threats are of paramount importance to safeguard cyberspace security and social stability. Consequently, text steganalysis[7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18] has emerged as a key research direction within the field of information security. It aims to detect hidden secret messages embedded within ordinary texts. However, the construction of an accurate and robust text steganalysis model largely depends on the quality and diversity of the training dataset. Although there are already some public text steganalysis datasets, such as T-Steg[9], TStego-THU[19], and Stego-Sandbox[12], etc., these datasets do not authentically reflect the complexity and diversity of real-world social network environments, thus constraining advances in text steganalysis technology.

Specifically, texts in the T-Steg[9] are generated by language models with fixed formats, ensuring data controllability and consistency to some extent but lacking the complexity and diversity of natural language in real social networks. Steganalysis algorithms trained on T-Steg often struggle to adapt to the real-world complexity, resulting in poor detection performance. The TStego-THU[19] significantly improves in scale and diversity compared to T-Steg, incorporating substantial real-world text data (e.g., Twitter[20], IMDB[21]). However, its isolated texts neglect contextual links (via retweets, replies, quotes) crucial for understanding meaning and propagation paths, limiting text steganalysis algorithms to isolated text semantic features. Stego-Sandbox[12] partially addresses TStego-THU's contextual gaps by considering tweet interactions, yet it still contains only text entities with limited inter text relationships, lacking exploration of diverse social network entities (e.g., users, hashtags). These entities and relationships play vital roles in covert communication. For example, users are the sender and receiver of covert communication, and hashtags tags can be used to establish stego propagation chains[**?**]. Therefore, these datasets ignore the rich entities and relationships in social networks are imperfect. Moreover, stegos in real social networks often spread within specific user groups with similar behaviors or social ties. Existing datasets lack targeted design for this, hindering text steganalysis techniques from leveraging group-specific steganographic traits for detection.

Given the limitations of current mainstream text steganalysis datasets, it is crucial to build a dataset that aligns with real social network patterns. This supports the development and evaluation of text steganalysis models in sparse steganographic information and fragmented text detection scenarios. Theoretically, it reflects fragmented text features and sparse steganographic information distributions in social networks, incorporating rich contextual and entity-relationship data. It provides a more comprehensive feature type for model training of text steganalysis in social networks, as well as a more accurate and reliable experimental verification platform, enhancing accuracy and robustness to advance text steganalysis. Practically, it supports cyberspace security regulators, protecting user privacy and information security, preventing malicious information spread, improving cyberspace governance, and maintaining social stability. Thus, this research holds both theoretical and practical significance.

However, constructing a dataset that can conform to the real schema of social networks and support text steganalysis research in scenarios where the steganography information is extremely sparse and the texts are extremely fragmented faces dual challenges: (1) capture of sparse steganographic signals. Stegos typically spread among a small number of special user groups, necessitating local group discovery algorithms for large-scale networks without explicit selection bias. (2) Simulation gaps of steganographic behavior. There is a lack of research on the behavior patterns of users posting stego. Therefore, a large-scale and diverse dataset is needed to cover different covert communication behavior patterns, thereby simulating multi-type text steganalysis scenarios. To address the above challenges, this paper proposes a dataset construction method for social network text steganalysis based on local group discovery and dynamic stego distribution regulation. Using Twitter as a case study, specifically, a meta path-constrained local group discovery algorithm is employed to sample user clusters with latent covert communication intentions. Subsequently, these users' original tweets are dynamically reshaped by adjusting the sparsity and fragmentation of stegos along three axes: ratio, type, and distribution. This enables the emulation of multifaceted steganalysis scenarios, thereby ensuring dataset realism and robustness to facilitate the development and evaluation of advanced text steganalysis models in social networks.

2. The Proposed Approach

In social networks, stego texts typically propagate within covert communication clusters (e.g., military operations or business acquisitions among decision makers). Members in covert communication groups establish communication chains through seemingly ordinary social behaviors such as like, retweet, or discuss hashtags (metadata tags starting with "#"), enabling the dissemination of secret information. Among them, excessive direct interactions such as likes and retweet are likely to catch the attention of regulators. Relatively speaking, the construction method of hashtags is flexible. They can be either regular words or combinations of text such as abbreviations and numerical symbols (such as "#YYDS", "#x4wl"). At the same time, a large number of hashtags can be carried in tweets, and some social networking platforms (such as Twitter) have no limit on the number of hashtags when searching for them. Their adaptability and ease of use enable users to establish indirect, stealthy interactions[22]. Thus, in constructing the dataset, we first sample local groups using the "user-tweet-hashtag" meta path. Then, we deploy stegos through a three-dimensional (3D) dynamic regulation strategy to simulate complex covert scenarios. Finally, we build a large-scale dataset for text steganalysis in social networks, named SN-Stego, which aligns with the characteristics of real social network schema and allows flexible control over the sparsity and fragmentation of stego.

2.1. Data Preparation

Research indicates that text content on the Twitter platform exhibits greater randomness and complexity[14]. Text steganography causes less interference to its feature distribution, making text steganalysis more challenging on Twitter. Therefore, in this paper, we use Twitter as a research case to illustrate the proposed dataset construction method. Before constructing SN-Stego, two preparatory tasks are required. The first involves collecting large-scale data from the Twitter network platform to build a heterogeneous information network (HIN). The second entails generating a stego library using text steganography algorithms and capacity as parameters.

2.1.1. Heterogeneous Information Network

Feng et al.[23] introduced the TwiBot-22 dataset in 2023. TwiBot-22 is a meticulously constructed, large-scale, high-quality Twitter bot detection dataset with complete graph structures. It contains extensive user and tweet data, ensuring accuracy and reliability through rigorous annotation and expert evaluation. Using TwiBot-22 as a data foundation enables leveraging its large-scale, authentic source data and HIN structure. This facilitates the construction of more representative and generalizable text steganalysis datasets. To clarify the HIN data basis, we briefly describe TwiBot-22's collection process. It primarily consists of two stages.

Phase 1: User Network Collection. This phase focuses on constructing the user network. Initially, a breadth-first search (BFS) approach is employed, starting from a selected "seed user"

Table 1: User metadata adopted in diversity-aware sampling[23].

Metadata Name	Description	Type
active days	days between user creation time and collected time	numerical
following count	number of user followings	numerical
followers count	number of user followers	numerical
tweet count	number of user tweets	numerical
listed count	number of user lists	numerical
verified	whether the user is verified or not	true-or-false
homepage URL	whether user has URLs in homepage or not	true-or-false

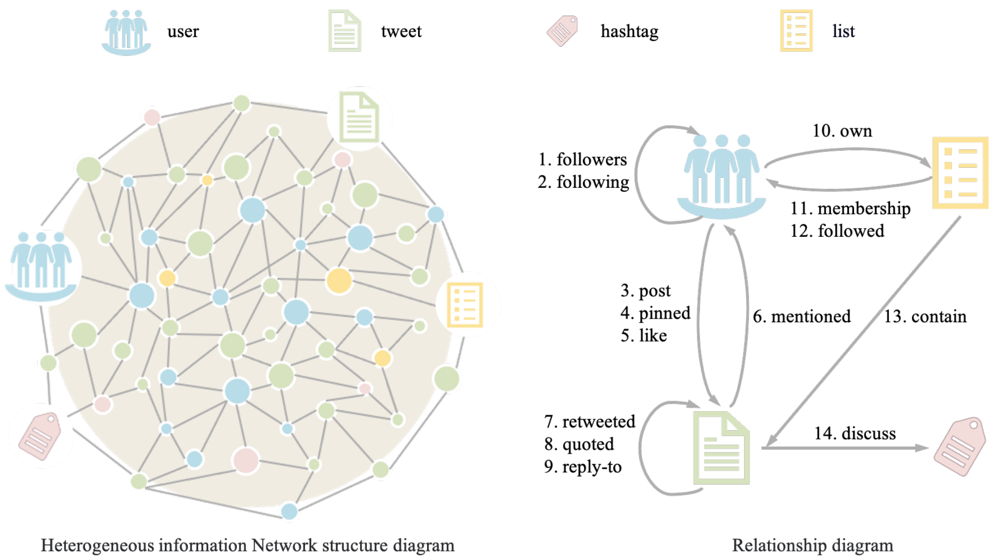

Figure 1: Heterogeneous graph of Twitter social network (left) and the HIN schema (right).

(@NeurIPSConf). Using the Twitter API to retrieve its 1,000 followers and 1,000 followees for BFS expansion. In addition, two diversity-aware strategies are applied, namely distribution diversity and value diversity, to optimize BFS-based user expansion. Thus, it ensures broader coverage of user types, making the collected users more representative. This phase constructs a homogeneous graph $G_U = (V_U, E_F)$, where V_U represents user nodes and E_F denotes follower relationships. Table 1 describes the user metadata used in diversity-aware sampling.

Distribution diversity: Given user metadata, different types of users fall into the metadata distribution differently. The goal of distribution diversity is to sample users from the top, middle, and bottom of the distribution. For numerical metadata, select k users with the highest values, k with the lowest, and k randomly from the rest. For true-or-false metadata, choose k users with "true" values and k with "false" values.

Value diversity: This sampling strategy prioritizes neighbors with metadata values differing significantly from the current user. For numerical metadata, the sampling probability of neighbor $v \in N(u)$ is defined as $p(v) \propto |u^{num} - v^{num}|$, where u^{num} is the user's metadata value. For true-or-

Table 2: Entities in the TwiBot-22 heterogeneous graph[23].

Entity Name	Description
User	Users are the most important entity on Twittersphere.
Tweet	Users post tweets to share their thoughts and interact with other users.
List	A list is curated feeds from selected users that allow you to listen to relevant discussions or influencers.
Hashtag	A hashtag is a metadata tag that is prefaced by “#”. It is used to link tweets with the same theme together.

Table 3: Relations in the TwiBot-22 heterogeneous graph[23].

Relation	Source Entity	Target Entity	Description
followers	user	user	source user follows target user
following	user	user	source user is followed by target user
post	user	tweet	user posts tweet
pinned	user	tweet	user pins tweet
like	user	tweet	user likes tweet
mentioned	tweet	user	tweet mentions user
retweeted	tweet	tweet	source tweet retweets target tweet
quoted	tweet	tweet	source tweet quotes target tweet with comments
reply	tweet	tweet	source tweet replies to target tweet
own	user	list	user is the creator of list
membership	list	user	user is a member of list
followed	list	user	user follows list
contain	list	tweet	list contains tweet
discuss	tweet	hashtag	tweet discussed hashtag

false metadata, k users are selected from the opposite class.

Phase 2: Heterogeneous graph construction. Upon the user network collected in Phase 1, Phase 2 primarily collects these users’ tweets, associated lists, hashtags, and 12 additional relations between users and these new entities. For user entities, their metadata, including tweets, lists, and follow relationships, are gathered. For tweet entities, detailed information is collected, encompassing retweets, quoted tweets, replies, and mentioned users. Additionally, all hashtags in listed tweets are extracted, and the Twitter API is used to search for more tweets related to these topics. As a result, TwiBot-22 forms a Twitter HIN comprising 4 types of entities (92,932,326 nodes) and 14 types of relations (170,185,937 edges). An instance of HIN for modeling Twitter social network is illustrated on the left side of Figure 1, while the right side presents the HIN schema, depicting node relationships. Detailed entities (nodes) and relations (edges) are shown in Table 2 and Table 3, respectively.

2.1.2. Stego Library

Text steganography methods based on automatic text generation can automatically generate a stego based on confidential information without requiring a carrier text (which is usually termed “cover”). These methods exhibit strong concealment and high embedding capacity, making them

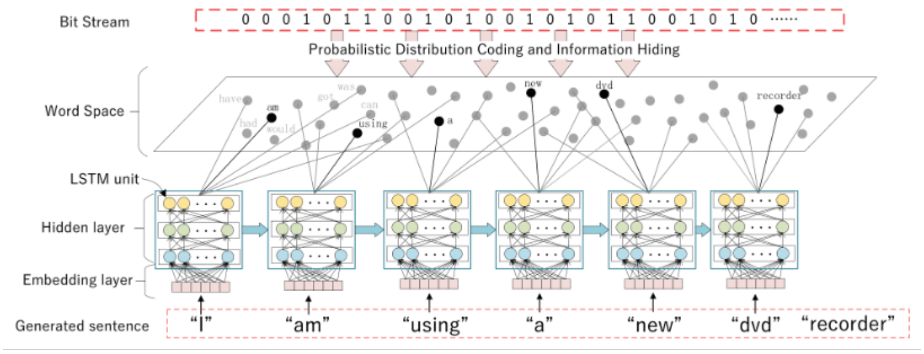

Figure 2: A detailed explanation of RNN-Stega[2].

the most widely used text steganography techniques. In this paper, to ensure the diversity of stegos, we employ three advanced generative text steganography algorithms and five types of embedding capacities as parameters. And RNN-Stega, a widely used generative text steganography model in the field of text steganalysis proposed by Yang et al. [2], is utilized to generate stegos. The detailed explanation of RNN-Stega is illustrated in Figure 2.

In the generation process, we first preprocess the Twitter texts collected in subsection 2.1.1 and use them as a corpus to train RNN-Stego. RNN-Stega employs Long Short-Term Memory (LSTM)[24] model to learn statistical features of covers. Then, we employ three widely used coding methods, namely Arithmetic Coding (AC)[1], Variable-Length Coding (VLC)[2], and Adaptive Dynamic Grouping (ADG)[3], to encode the probability distribution of words. Among them, AC[1] employs the reverse sequence of arithmetic coding, a data compression method used to encode strings of elements with known probability distributions. It first selects a (uniformly sampled) message and then maps the message to a sequence (of words), achieving information hiding while minimizing the difference in statistical characteristic distributions between the stegos and covers. VLC[2] employs huffman coding to map the secret message to conditional probabilities, reducing the discrepancy between the stegos and the covers. ADG[3] divides conditional probabilities into several buckets that are as equal and summed as possible, which has been mathematically proven to achieve the theoretical minimum difference. After encoding the probability distribution of words, RNN-Stega selects the corresponding word according to the secret bitstream, so as to achieve the purpose of hiding information. Additionally, we vary the embedding capacity by adjusting the embedded bits per word (bpw, which is set to 1, 2, 3, 4 and 5 respectively). This produces stegos with different lengths and secret information distributions. We generated 7,900 stegos respectively for each coding algorithm and embedding capacity. Ultimately, a stego library containing three steganographic algorithms and five embedding capacities is obtained:

$$D_{\text{stego}} = \bigcup_{\substack{a\in\{\text{AC,VLC,ADG}\}\\ c\in[1,5]}} S(a,c) \tag{1}$$

where $S(a,c)$ represents a stego generated by using the steganographic algorithm parameter $a \in \{AC, ADG, VLC\}$ and the embedding capacity parameter $c \in [1,5]$. Table 4 presents the average lengths of the stegos (SL) generated under different steganographic algorithms (SA) and embedding capacities (bpw) in the stego library D_{stego}.

Table 4: The average length of various types of stegos in D_{stego}.

SL \ bpw / SA	1	2	3	4	5	6.93
AC	6.81	8.91	11.25	12.88	14.36	/
VLC	5.88	7.55	10.34	12.75	13.98	/
ADG	/					12.33

2.2. *Local Group Discovery Based On Meta Path Constraint*

According to the phenomenon of community aggregation in social networks, covert communication groups often exhibit similar behavioral patterns or social relationships. As one of the essential tools in social network analysis, meta paths can reveal complex associations between different entities. Therefore, this subsection proposes a local group discovery method based on meta path constraint to sample special user groups with potential covert communication intent. First, we introduce several fundamental concepts.

Definition 1: HIN.[25] HIN is represented as a directed graph $G = (V, E, \phi, \psi)$, where V is the node set, E is the edge set, $\phi : V \rightarrow N$ maps nodes to types, and $\psi : E \rightarrow R$ maps edges to relation types, with $|N| + |R| > 2$. Each node $v \in V$ belongs to a type $\phi(v) \in N$, and each edge $e \in E$ belongs to a relation type $\psi(e) \in R$.

Definition 2: Network Schema.[25] Given the HIN $G = (V, E, \phi, \psi)$ with $\phi : V ß N$ and $\psi : E ß R$, its network schema is $S_G = (N, R)$, which describes how node types in N are connected via relation types in R. For example, the left of Figure 1 illustrates an HIN instance of the Twitter social network, while the right of Figure 1 depicts its schema with four node types and their relations.

Definition 3: Meta path.[25] Meta path P is a path defined on S_G, noted as $P = (N_1 \xrightarrow{R_1} N_2 \xrightarrow{R_2} \dots \xrightarrow{R_L} N_{L+1})$, where L is the length of meta path, $N_i \in N$ and $1 \leq i \leq L + 1$, $R_j \in R$ and $1 \leq j \leq L$. For brevity, P is usually denoted as a sequence of node types: $P = (N_1, N_2, \dots, N_{L+1})$. If there exists a path $p = (u_1, \dots, u_L)$ in S_G, and p satisfies $\phi(u_i) = N_i (1 \leq i \leq L)$, then p is an instance of the meta path P (denoted as $p \in P$). Different meta paths encode distinct semantics. For example, the meta path $P1 = (User, tweet, user)$ indicates that the user likes/retweets the same tweet. While the meta path $P2 = (user, tweet, hashtag, tweet, user)$ indicates that the user posts/retweets/likes tweets with the same topic.

Definition 4: P-Connected and P-Neighbors.[26] If node u_j is reachable from u_i via a path instance of meta path P, u_j is a P-connected node of u_i. All P-connected nodes of u_i are its P-neighbors.

Based on these definitions, we first construct a meta path to guide random walks. Since covert communication users prefer indirect interactions to evade detection, we model such links via shared hashtags or tweets. The proposed meta path is as follow:

$$P = (U \xrightarrow{\text{post/love}} T \xrightarrow{\text{discuss}} H \xleftarrow{\text{discuss}} T \xleftarrow{\text{post/love}} U) \tag{2}$$

where $U, T, H \in N$ denote respectively represent three node types of user, tweet and hashtag in the HIN, post,love,discuss $\in R$ respectively represent three edge types of post, like and discuss. This meta path P captures the characteristics of covert communication behaviors where users interact through tweets and topics.

Then, we define the correlation degree between nodes as the total number of path instances connecting them:

$$R_{u_i \sim u_j} = |\{p_{u_i \sim u_j} : p_{u_i \sim u_j} \in P\}| \tag{3}$$

where $p_{u_i \sim u_j}$ is a path instance with u_i as the starting node and u_j as the ending node, $|\cdot|$ indicates the number of elements in the set. This degree of correlation reflects the interaction intensity between nodes.

Based on the correlation degree $R_{u_i \sim u_j}$, the transition probability under the meta path P from nodeu_i to u_j is defined as follow:

$$P(u_j|u_i) = \frac{R_{u_i \sim u_j}}{\sum_{u_k \in N(u_i)} R_{u_i \sim u_k}} \tag{4}$$

where, $N(u_i)$ is the set of p-neighbors of node u_i. To avoid excessive deviation from the target area, a restart probability $\alpha = 0.15$ is set. That is, there is a 15% probability of returning to the initial user node in each random walk.

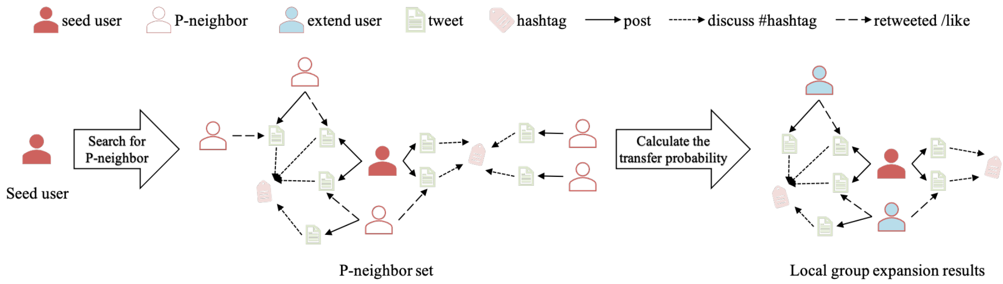

Figure 3: The schematic of "user-tweet-hashtag" meta-path-constrained local group discovery.

When sampling special user groups with potential covert communication intent, we we first randomly select multiple seed users that are not P-neighbors of each other to enhance the diversity of the users. Starting from each seed user, the random walk follows the guidance of the meta path to move to the top-k P-neighbor nodes with higher transition probabilities. When the number of recorded users during the walk reaches 5,000, it will be stopped. Next, we merge the users sampled from random walks initiated with different seeds, and remove duplicate users to ensure all elements in the final group set are unique. Figure 3 illustrates the meta path constrained group discovery process using one seed user as an example.

2.3. Tweet Reconstruction Based On 3D Dynamic Regulation Strategy

In this subsection, we simulate the covert communication behavior of users in social networks by replacing some users' tweets with stegos. To flexibly control the distribution and sparsity of stegos in the dataset while ensuring its authenticity and reliability, we designed a 3D dynamic regulation strategy, named S-RTD. S-RTD adjusts the stego ratio (SR), stego type (ST), and stego distribution (SD) to simulate covert communication scenarios with varying complexity and sparsity. Let the sampled set of local group be $U = \{u_1, u_2, ..., u_{5000}\}$, and the tweet set of user u be $T_u = \{t_1, t_2, \ldots, t_n\}$. It should be emphasized that T_u is an ordered sequence arranged in the order of publication time, where t_i represents the i-th tweet. Stego library $D_{\text{stego}} = \{S_{a,c}\}$, where

$a \in SA = \{AC, ADG, VLC\}$ and $c \in SC = [1, 5]$. The following is a detailed elaboration of S-RTD.

(1) Stego Ratio (SR). Covert communication users may post both steganographic and normal tweets to conceal the presence of hidden messages. We design different stego ratios $\rho \in [0.1, 0.3, 0.5, 0.7, 0.9, 1.0]$ to replace tweets in T_u. The number of tweets to be replaced for user u is:

$$C_T = \lceil \rho \cdot |T_u| \rceil \tag{5}$$

where $\lceil \cdot \rceil$ represents rounding up, and $|T_u|$ denotes the cardinality (number of elements) of the set T_u. By adjusting SR, we can flexibly control the sparsity of stegos in the dataset. Lower SR results in higher sparsity.

(2) Stego Type (ST). Covert communication users can adopt different strategies to generate and disseminate stegos. For example, when dealing with fixed secret information, they may employ a single steganographic algorithm to enhance information transmission efficiency, or use multiple steganographic algorithms to increase detection difficulty. Additionally, the secret information can be concentrated in a small number of stegos to avoid frequent posting of steganographic content that might reveal their identity. Alternatively, the secret information can be dispersed across multiple short texts to reduce the embedding capacity per stego, thereby improving imperceptibility. Therefore, we use the steganographic algorithm and embedding capacity as parameters to dynamically adjust the types of stegos in the dataset, denoted as S_T. Both the steganographic algorithm and embedding capacity can be configured as either "single" or "multiple", leading to the following four subsets of stegos:

- Single algorithm and single capacity. The subset $S(a, c)$ where a is a specific algorithm and c is a fixed capacity:
$$S_T = S(a, c) \in \{S_{a,c} \mid a \times c\} \tag{6}$$
where $a \times c$ denotes all possible combinations of (a, c) from sets SA and SC.

- Multiple algorithms and single capacity. The subset $S(\sim, c)$, where $\sim$ indicates the diversity of steganographic algorithms:
$$S_T = S(\sim, c) = \{S_{a,c} \mid a \in \mathrm{SA}\} \tag{7}$$

- Single algorithm and multiple capacities. The subset $S(a, \sim)$, where $\sim$ indicates the diversity of steganographic capacity:
$$S_T = S(a, \sim) = \{S_{a,c} \mid c \in \mathrm{SC}\} \tag{8}$$

- Multiple algorithm and multiple capacities. The subset $S(\sim, \sim)$:
$$S_T = S(\sim, \sim) \in \{S_{a,c} \mid a \in \mathrm{SA}, c \in \mathrm{SC}\} \tag{9}$$

By controlling ST, we simulate text steganalysis environments with varying fragmentation and complexity. Lower embedding capacities result in shorter stegos with more pronounced fragmentation, while more steganographic algorithms and capacities increase scenario complexity.

(3) Stego Distribution (SD). When urgently needing to publish large amounts of secret information, covert communication users must consecutively post multiple stegos to complete their covert communication tasks. If time permits, they may instead distribute stegos in smaller batches interspersed with covers on social platforms. Consequently, stegos may appear densely clustered within certain time periods, while remaining relatively sparse at other times. We model the stego distribution $R(T_u, \rho, m)$ by posting stegos in four patterns:

- Front: stegos are concentrated at the beginning of the tweet sequence.

$$R(T_u, \rho, m) = S_T \cup \{t_{C_T+1}, \cdots, t_n\}, \quad m = \text{Front} \tag{10}$$

  where C_T is the number of stegos obtained from formula 5, and S_T is the subset of stegos obtained from formula 6 to formula 9.

- Middle: stegos are centered in the middle.

$$R(T_u, \rho, m) = \{t_1, \cdots, t_d\} \cup S_T \cup \{t_{n-d-C_T}, \cdots, t_n\}, \quad m = \text{Middle} \tag{11}$$

  where $d = \left\lfloor \frac{n-C_T}{2} \right\rfloor$.

- Latter: stegos are concentrated at the end.

$$R(T_u, \rho, m) = \{t_1, \cdots, t_{n-C_T}\} \cup S_T, \quad m = \text{Latter} \tag{12}$$

- Random: stegos are scattered randomly.

$$R(T_u, \rho, m) = S_T \cup (T_u \setminus Index(n, C_T)), \quad m = \text{Random} \tag{13}$$

  where function $Index(n, C_T)$ randomly selects C_T indices from n positions.

Adjusting SD allows simulation of different steganographic distribution patterns, reflecting the complexity text steganalysis environment in real social networks.

2.4. SN-Stego Construction

Based on the above-mentioned local group discovery method and the S-RTD strategy, we construct a dataset named SN-Stego that authentically reflects social network patterns, and supports dynamic control over stego sparsity and fragmentation. Figure 4 illustrates the detailed workflow of SN-Stego construction.

The input consists of the large-scale Twitter heterogeneous information network collected in subsection 2.1.1 and the stego library D_{stego} generated in subsection 2.1.2. First, we employ the local group discovery algorithm proposed in subsection 2.2 to identify small-scale user groups with potential covert communication intent from the large-scale Twitter heterogeneous information network. Next, we apply the S-RTD strategy introduced in subsection 2.3 to reconstruct the tweets of sampled users by adjusting the stego ratio (SR), stego type (ST), and stego distribution (SD). This simulates various types of covert communication users with different behavioral patterns. Subsequently, we remove the association relationships of the replaced tweets in the heterogeneous information network while preserving all other unmodified entities and relationships. Finally, SN-Stego is constructed that simulates complex text steganalysis environments with varying fragmentation and sparsity of stegos.

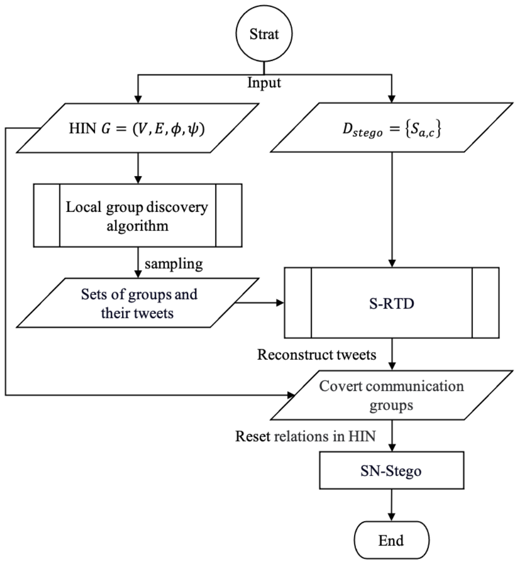

Figure 4: The workflow of SN-Stego construction.

3. Dataset Evaluation

3.1. Statistical Analysis

We compared SN-Stego with three existing mainstream text steganalysis datasets[9, 19, 12]. The statistical results are presented in Table 5. It shows that SN-Stego boasts a significantly larger data scale, being hundreds of times larger than TStego-THU[19]. Notably, SN-Stego contains abundant entities and relational connections. In contrast to other datasets that only include isolated text data or simple reply relationships, SN-Stego features a more extensive data volume, richer data types, and broader application scenarios. Its heterogeneous information network structure can reveal more deep-level and potential steganographic features, providing researchers with a more comprehensive and reliable platform for study and testing.

Table 5: The statistical comparison results of SN-Stego with three mainstream text steganography analysis datasets.

Dataset	Entity	Relation	Graph structure	Text scale
T-Steg[9]	1	×	×	30,000
TStego-THU[19]	1	×	×	40,000
Stego-Sandbox[12]	1	2	✓	15,639
SN-Stego	4	14	✓	6,580,000

3.2. Experimental Analysis

Through experiments, we aim to reveal the limitations of existing text steganalysis methods when applied to real-world social network scenarios characterized by text fragmentation and sparse steganographic information. This highlights the urgency and necessity of developing new social network-oriented text steganalysis approaches, thereby demonstrating the significant value of our proposed dataset construction method and the SN-Stego dataset in supporting related research.

3.2.1. Experimental Setup

Table 6: Related parameters settings.

Parameter	Values
Epoch	40
Batch size	100
Optimizer	Adam[27]
Learning rate	1e-5
Criterion	Cross entropy loss
Dropout rate	0.5
Class number	128

Benchmark Models. We selected five mainstream deep learning-based text steganalysis models as benchmark models for SN-Stego. FCN[7], based on a single-layer fully connected network, identifies semantic correlations between words in text and performs steganalysis by exploiting the disruption of statistical correlations between words caused by the embedding of secret information. RNN[8] utilizes a bidirectional recurrent neural network (BiRNN) to extract conditional distribution features for each word in the text. CSW[9] refines word correlations in text into continuous word correlation, cross-word correlation, and cross-sentence correlation, and employs convolutional sliding windows (CSW) of various sizes to extract these correlation features for LS. ATT[10] adopts an attention mechanism that strategically focuses on salient parts of the input, enhancing the model's ability to extract meaningful insights from the data. EILGF[13] simultaneously extracts and fuses local and global features of the text, and introduces a group-wise enhancement mechanism to improve the quality of features. All of these models utilize BERT for text feature extraction. Other parameters are consistent with those described in their respective papers.

Sample Distribution. We randomly selected 4,000 covers and the same number of stegos from the constructed SN-Stego dataset, and divided them into the training set, the validation set and the test set in a ratio of 3:1:1. During the training stage, the same amount of covers and stegos is adopted to enable the model to fully learn the text features. During the testing phase, in order to evaluate the generalization of the benchmark models in text steganography analysis environments with different steganography sparsity, we designed five test sets, among which the stego ratio (SR) was 10%, 20%, 30%, 40% and 50% respectively.

Evaluation Metrics. Since we used imbalanced test sets, we employed the F1 score (F1) as the evaluation indicator. The F1 score is a metric in statistics used to measure the accuracy of binary classification models, taking into account both precision and recall. It is sensitive to changes in data distribution and thus more useful when dealing with class imbalance issues. The formulas is described as follows:

$$F1 = \frac{2 \cdot TP}{2 \cdot TP + FN + FP} \tag{14}$$

Where TP (True Positive) represents the number of stegos that are predicted correctly by the model. FP (False Positive) indicates the number of covers predicted to be stegos. FN (False Negative) illustrates the number of stegos predicted to be covers. And TN (True Negative) represents the number of covers predicted correctly.

Experimental Environment and Parameters: All experimental codes in this paper are written based on PyTorch and executed on a GeForce RTX 3080 GPU with 10 Gb of graphics memory. Other parameters related to the experiments are shown in Table. 6, and the selection of some hyper-parameters will be discussed and explained in subsequent experiments.

3.2.2. Results and Discussion

Table 7: F1 result of benchmark models in different text steganalysis scenarios.

SR	Model	AC					VLC					ADG
		1	2	3	4	5	1	2	3	4	5	6.93
	FCN	59.06	60.63	62.42	49.87	41.30	60.57	60.77	59.98	62.94	60.95	0.21
	CSW	64.49	64.87	67.82	61.11	63.47	69.97	67.82	70.81	68.34	70.65	13.30
10%	RNN	64.39	65.38	67.21	63.39	64.49	69.24	68.51	71.17	69.82	70.28	9.62
	ATT	63.58	63.31	66.43	61.82	62.92	68.39	66.87	69.91	69.11	70.77	7.85
	EILGF	61.21	62.05	64.33	56.46	57.12	63.66	62.73	64.96	64.43	66.79	9.88
	FCN	63.07	63.71	64.24	65.57	58.47	67.29	69.17	68.42	72.16	70.12	4.73
	CSW	81.38	79.39	75.05	75.70	73.47	77.86	75.36	76.15	76.89	76.48	28.95
20%	RNN	79.86	79.76	76.90	75.84	74.30	79.84	74.52	76.49	78.19	77.67	21.25
	ATT	79.68	78.89	75.29	75.38	72.97	79.17	76.98	76.85	77.29	77.15	16.58
	EILGF	78.37	76.87	73.51	71.71	69.92	74.64	73.65	73.16	74.89	73.81	25.06
	FCN	73.16	70.71	73.43	70.09	68.22	72.41	75.26	73.82	75.76	74.87	7.95
	CSW	87.94	86.71	86.11	82.55	82.76	87.35	85.78	84.88	83.01	81.27	45.42
30%	RNN	86.39	85.79	85.36	82.33	80.18	86.23	85.47	84.75	81.41	82.22	40.78
	ATT	86.04	85.18	84.95	81.87	79.84	86.19	84.25	83.68	83.58	81.46	27.81
	EILGF	85.12	84.38	83.73	79.11	77.02	84.42	82.53	82.12	80.88	78.63	31.35
	FCN	84.37	81.92	79.50	74.44	71.80	79.18	78.89	75.57	75.86	74.65	30.47
	CSW	91.94	91.06	89.79	87.97	85.62	91.80	89.89	89.08	87.79	87.42	55.76
40%	RNN	90.53	89.28	88.98	87.28	86.30	90.65	89.17	88.21	88.53	87.11	49.87
	ATT	90.25	89.30	88.96	87.51	86.33	90.37	89.11	87.94	87.69	86.99	41.31
	EILGF	89.11	87.90	87.75	86.04	83.15	88.87	87.31	87.26	85.21	84.53	42.49
	FCN	89.20	87.32	85.69	77.30	75.89	87.99	86.79	85.69	80.03	78.44	43.31
	CSW	94.53	93.62	92.44	91.02	89.74	94.12	92.86	91.96	90.99	89.92	64.22
50%	RNN	93.32	92.16	91.34	90.41	89.31	92.81	91.88	91.66	90.73	89.36	56.99
	ATT	93.31	92.09	91.37	90.25	89.17	92.98	91.79	91.64	90.52	89.41	50.85
	EILGF	92.28	91.08	90.23	88.61	87.01	91.69	90.55	89.99	88.98	88.27	62.41

Table 7 presents the detection F1 scores of various benchmark models across different text steganalysis scenarios. Here, AC, VLC, and ADG represent stegos generated using corresponding encoding methods. The numbers below them indicate the embedding capacity in bits per word (bpw). By analyzing the experimental data in Table 7, we can draw the following conclusions:

First, as the embedding capacity (bpw) increases, the detection accuracy of the benchmark models generally shows a declining trend. This is attributed to the Psic effect[4] in generative stegos, where higher bpw values blur the statistical distribution boundaries between stegos and covers, making them harder to distinguish. Furthermore, even at lower bpw values, while the detection performance of the benchmark models improves slightly, it remains unsatisfactory. This is due to the fragmented nature of social network texts in SN-Stego, which poses significant challenges to existing text steganalysis that rely solely on textual semantic features.

Second, the F1 scores in the table decrease from bottom to top, indicating that as the sparsity of stegos increases, the detection performance of the models declines. When the sparsity is high (e.g., SR=10%), the F1 scores of the benchmark models rarely exceed 70%. For stegos generated using the ADG steganographic algorithm, the F1 scores even drop below 10%. Only when the SR approaches 50%, where the ratio of positive to negative samples is relatively balanced, do these benchmark models achieve relatively better detection performance. It demonstrates that while these benchmark models perform well under ideal experimental conditions, they struggle in real-world scenarios where steganographic information is extremely sparse.

Third, stegos generated by the ADG algorithm are more challenging to detect compared to those generated by AC and VLC. This is because the ADG-generated stegos in our dataset have a higher embedding capacity, and according to the Psic effect[4], their statistical concealment is superior. Since the benchmark methods detect stegos based on statistical distribution differences before and after embedding, the F1 scores for ADG are significantly lower than those for AC and VLC.

In summary, existing text steganalysis methods exhibit considerable limitations when applied to highly fragmented and extremely sparse stegos in social networks. Therefore, it is necessary to broaden research perspectives and develop new algorithms and models to counter the continuously evolving text steganography techniques in social network. The proposed dataset construction method serves as a foundational and critical step to support such advancements, holding substantial significance for future research.

4. Conclusion and Future Work

Addressing the limitations of existing steganalysis datasets—such as the lack of social graphs, inadequate text attributes, mismatched sample distributions, and limited data scales—which severely constrain text steganalysis research in social networks, this study uses Twitter as a case study to propose a dataset construction method based on local group discovery and sample distribution regulation. Specifically, we first collect HIN by aggregating multi-type entities and their relationships from Twitter, then generate diverse stegos using advanced text steganography model parameterized by steganography algorithms and embedding capacities. Subsequently, a local group discovery algorithm constrained by "user-tweet-hashtag" meta path is introduced to sample special user groups with latent covert communication intentions. Next, we apply the S-RTD strategy to reconstruct user tweet sequences across stego ratio, type, and distribution, enabling dynamic control over the fragmentation and sparsity of stegos. Finally, we construct SN-stego, a large-scale dataset rich in social graph information and diverse sample distributions. Statistical analyses confirm SN-stego's advantages in data scale, content diversity, and scenario adaptability, aligning with the fragmented text and sparse steganography observed in real-world social networks. Benchmarking existing mainstream text steganalysis models on SN-stego reveals their significant limitations in real-world scenarios, further validating the effectiveness of SN-Stego.

Yet, since SN-Stego is Twitter-based, differences in user behavior and text style across platforms may limit its generalization and stego text diversity. In the future, enrich the dataset with more platform types, language styles, and steganography algorithms to better support social network text steganalysis. Our work provides high-quality data support for text steganalysis in social networks, contributing scientific value and practical significance to advancing text steganalysis technologies and safeguarding cyberspace security and social stability.

Author Contributions

Qiong Xu: Conceptualization, Data curation, Software, Visualization, Writing – original draft. **Ru Zhang:** Supervision, Investigation, Writing – review & editing. **Jianyi Liu:** Methodology, Validation, Resources, Writing – review & editing. **Yongfeng Huang:** Investigation, Writing – review & editing.

Acknowledgements

Authors are deeply indebted to anonymous reviewers for their constructive suggestions and helpful comments. This work was supported by the National Natural Science Foundation of China under Grant U21B2020.

References

[1] Ziegler Z., Deng Y., Rush A.: Neural Linguistic Steganography. In: the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong, China: Association for Computational Linguistics, (2019).

[2] Yang Z., Guo X., Chen Z., Huang Y., and Zhang Y.: RNN-Stega: linguistic steganography based on recurrent neural networks. IEEE Transactions on Information Forensics and Security. 14 (5), 1280-1295 (2019).

[3] Zhang S., Yang Z., Yang J., and Huang Y.: Provably Secure Generative Linguistic Steganography. In: the Findings of the Association for Computational Linguistics. [s.l.]: Association for Computational Linguistics (ACL-IJCNLP), pp. 3046-3055 (2021).

[4] Yang Z., Zhang S., Hu Y., Hu Z., and Huang Y.: VAE-Stega: Linguistic steganography based on variational auto-encoder. IEEE Transactions on Information Forensics and Security. 16, 880–895 (2021).

[5] Wang R., Xiang L., Liu Y., and Yang C.: PNG-Stega: Progressive Non-Autoregressive Generative Linguistic Steganography. IEEE Signal Processing Letters. 30, 528-532 (2023).

[6] Ding C., Fu Z., Yang Z., Yu Q., Li D., and Huang Y.: Context-Aware Linguistic Steganography Model Based on Neural Machine Translation. IEEE/ACM Transactions on Audio, Speech, and Language Processing. 32, 868-878 (2024).

[7] Yang Z., Huang Y., and Zhang Y.: A fast and efficient text steganalysis method. IEEE Signal Processing Letters. 26(4), 627-631 (2019).

[8] Yang Z., Wang K., Li J., Huang Y., and Zhang Y.: Ts-rnn: Text steganalysis based on recurrent neural networks. IEEE Signal Processing Letters. 26(12) 1743-1747 (2019).

[9] Yang Z., Huang Y., and Zhang Y.: TS-CSW: Text steganalysis and hidden capacity estimation based on convolutional sliding windows. Multimedia Tools and Applications. 79 (25), 18293–18316 (2020).

[10] Zou J., Yang Z., Zhang S., Rehman S., and Huang Y.: High-performance linguistic steganalysis, capacity estimation and steganographic positioning. In: International Workshop on Digital Watermarking (IWDW), pp. 80-93 (2020).

[11] Yi B., Wu H., Feng G., and Zhang X.: Exploiting Language Model for Efficient Linguistic Steganalysis. In: the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3074-3078 (2022).

[12] Yang J., Yang Z., Zou J., Tu H., and Huang Y.: Linguistic steganalysis toward social network. IEEE Transactions on Information Forensics and Security. 18, 859–871 (2022).

[13] Xu Q., Zhang R., and Liu J.: Linguistic steganalysis by enhancing and integrating local and global features. IEEE Signal Processing Letters. 30, 16-20 (2023).

[14] Wang H., Yang Z., Yang J., Chen C., and Huang Y.: Linguistic Steganalysis in Few-Shot Scenario. IEEE Transactions on Information Forensics and Security. 13, 4870-4882 (2023).

[15] Wang Y., Zhang R., and Liu J.: RLS-DTS: Reinforcement-Learning Linguistic Steganalysis in Distribution-Transformed Scenario. IEEE Signal Processing Letters. 30, 1232-1236 (2023).

[16] Xue Y., Wu J., Ji R., Zhong P., Wen J., and Peng W.: Adaptive Domain-Invariant Feature Extraction for Cross-Domain Linguistic Steganalysis. IEEE Transactions on Information Forensics and Security. 19, 920-933 (2024).

[17] Li S., Du H., and Wang J.: General Steganalysis of Generative Linguistic Steganography Based on Dynamic Segment-Level Lexical Association Extraction. IEEE Signal Processing Letters, 32, 191-195 (2025).

[18] Yang Z., Luo Y., Yang J., Xu X., Zhang R., and Huang Y.: Class-Aware Adversarial Unsupervised Domain Adaptation for Linguistic Steganalysis. IEEE Transactions on Information Forensics and Security, 20, 5181-5194 (2025).

[19] Yang Z., He J., Zhang S., Yang J., and Huang Y.: Tstego-thu: Large-scale text steganalysis dataset. In: International Conference on Artificial Intelligence and Security (ICAIS), Cham, Switzerland, pp. 335-344 (2021).

[20] Go A., Bhayani R., and Huang L.: Twitter sentiment classification using distant supervision. In: CS224N project report, 1(12), (2009).

[21] Maas A., Daly R., Pham P., Huang D., Ng A., and Potts C.: Learning word vectors for sentiment analysis. In: the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 142–150 (2011).

[22] Szczypiorski K.: StegHash: New Method for Information Hiding in Open Social Networks. International Journal of Electronics and Telecommunications. 62(4), 347-352 (2016).

[23] Feng S., Tan Z., Wan H., Wang N., Chen Z., Zhang B., Zheng Q., Zhang W., Lei Z., Yang S., Feng X., Zhang Q., Wang H., Liu Y., Bai Y., Wang H., Cai Z., Wang Y., Zheng L., Ma Z., Li J., and Luo M.: TwiBot-22: Towards Graph-Based Twitter Bot Detection. arXiv preprint arXiv: 2206.04564 (2023).

[24] Hochreiter S., and Schmidhuber J.: Long Short-Term Memory. Neural Computation. 9(8), 1735-1780 (1997).

[25] SHI C., WANG R., WANG X.: A Survey of Heterogeneous Information Networks Analysis and Applications. Journal of Software, 33(2), 598-621 (2022).

[26] Wang J., Zhou L., Wang X., Wang L., and Li S.: Attribute-sensitive community search over attributed heterogeneous information networks. Expert Systems with Applications. 235, 121153 (2024).

[27] Loshchilov I. and Hutter F.: Decoupled Weight Decay Regularization. In: the 7th International Conference on Learning Representations (ICLR), pp. 6-9 (2019).