Journal of *Cybersecurity and Privacy*

MDPI

Article

Steganalysis of Adaptive Multi-Rate Speech with Unknown Embedding Rates Using Multi-Scale Transformer and Multi-Task Learning Mechanism

Congcong Sun [1,*], Azizol Abdullah [1], Normalia Samian [1] and Nuur Alifah Roslan [2]

[1] Department of Communication Technology and Network, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang 43400, Selangor, Malaysia; azizol@upm.edu.my (A.A.); normalia@upm.edu.my (N.S.)
[2] Department of Multimedia, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang 43400, Selangor, Malaysia; nuuralifah@upm.edu.my
* Correspondence: gs68181@student.upm.edu.my

Abstract: As adaptive multi-rate (AMR) speech applications become increasingly widespread, AMR-based steganography presents growing security risks. Conventional steganalysis methods often assume known embedding rates, limiting their practicality in real-world scenarios where embedding rates are unknown. To overcome this limitation, we introduce a novel framework that integrates a multi-scale transformer architecture with multi-task learning for joint classification and regression. The classification task effectively distinguishes between cover and stego samples, while the regression task enhances feature representation by predicting continuous embedding values, providing deeper insights into embedding behaviors. This joint optimization strategy improves model adaptability to diverse embedding conditions and captures the underlying relationships between discrete embedding classes and their continuous distributions. The experimental results demonstrate that our approach achieves higher accuracy and robustness than existing steganalysis methods across varying embedding rates.

Keywords: steganography; steganalysis; unknown embedding rates; multi-task learning; multi-scale transformer

Academic Editors: Mario Antunes and Carlos Rabadão

Received: 14 April 2025
Revised: 24 May 2025
Accepted: 26 May 2025
Published: 3 June 2025

Citation: Sun, C.; Abdullah, A.; Samian, N.; Roslan, N.A. Steganalysis of Adaptive Multi-Rate Speech with Unknown Embedding Rates Using Multi-Scale Transformer and Multi-Task Learning Mechanism. *J. Cybersecur. Priv.* **2025**, *5*, 29. https://doi.org/10.3390/jcp5020029

1. Introduction

Steganography is a way of concealing sensitive information within multimedia carriers such as images [1,2], videos [3,4], text [5,6], and audio [7,8], allowing for clandestine communication. However, its dual-use nature poses major threats, including possible misuse in cybercrime [9,10]. To combat these dangers, the development of steganalysis techniques—which determine whether a given carrier contains hidden information—has become increasingly important. As a result, research on various steganalysis approaches has received significant interest in recent years [9,10].

Voice-over-IP (VoIP) technologies have experienced widespread global adoption in recent years, as evidenced by the continuous expansion of the global VoIP market. According to industry forecasts, the market size is projected to reach USD 108.5 billion by 2032, with a compound annual growth rate (CAGR) of 10.4%. This large-scale deployment and the growing volume of voice traffic over IP-based networks provide considerable covert bandwidth and payload capacity, making VoIP streams well-suited for steganographic communication. However, steganalysis in VoIP-based speech faces distinct challenges, primarily due to the need for real-time processing. Detection systems must analyze speech data frame

by frame with low latency, which limits the use of complex or heavy models. This strict timing constraint sets VoIP steganalysis apart from other modalities (images, videos, texts, and audio) and calls for lightweight, efficient methods tailored to streaming speech.

Among the various codecs employed in VoIP systems, adaptive multi-rate (AMR) speech coding—standardized by the 3rd Generation Partnership Project (3GPP) [11]—is a widely used low-bit-rate compression technique recognized for its coding efficiency and robustness in wireless transmission environments. AMR is extensively utilized not only in real-time voice communication but also as a speech file format for storing audio data. Its integration into mainstream messaging platforms such as WeChat and iMessage further underscores its relevance in modern voice-centric media. Owing to its prevalence in both real-time and stored speech applications, AMR-encoded speech streams have become important steganographic carriers and have attracted considerable attention in the domains of information hiding and steganalysis [12–18].

The cornerstone of these steganographic approaches is based on exploiting redundancies in the AMR encoding process. AMR, a common voice encoding system based on algebraic code-excited linear prediction (ACELP), provides three types of coding parameters for data embedding: pitch delay [12,13], linear prediction coefficients (LPCs) [14–16], and fixed-codebook (FCB) parameters [17,18]. FCB characteristics are particularly prominent, accounting for a significant chunk of the encoding process. For example, in the AMR-Narrowband (AMR-NB) codec running at 12.2 kbps, each 20 ms speech frame has 244 encoded bits, 140 of which (57.38%) correspond to FCB parameters [17,18], making them ideal for steganography. Furthermore, the depth-first search approach for determining pulse positions in FCB parameters frequently produces unsatisfactory results. This feature allows for the replacement of original pulse positions with other candidates, allowing hidden messages to be placed in the FCB domain without drastically affecting speech quality.

Steganographic techniques for AMR voice streams have mostly addressed the FCB domain [17,18]. Geiser and Vary [17] adapted the FCB parameter encoding procedure to embed secret information into AMR-based streams, obtaining a ten-bit embedding capacity per frame. Miao et al. [18] expanded on this notion by proposing an adaptive technique to improve the transparency of updated streams. Unlike previous methods, this system dynamically modified the embedding capacity with an embedding factor, allowing for a balance of transparency and capacity. Transparency is the imperceptibility of changes in the carrier after embedding, ensuring that changes are undetected via the human auditory or visual system. In contrast, embedding capacity measures the amount of secret information hidden inside the carrier. By optimizing these two parameters, the adaptive technique attempted to increase overall steganographic performance.

In recent years, various research [19–23] has concentrated on detecting steganographic techniques employed in AMR-based voice streams, with remarkable results. These steganalysis techniques generally employ a "feature plus classifier" structure, with support vector machines (SVMs) [19–21] serving as the primary classifier. The research focuses mostly on the creation of handmade characteristics that are customized to the specific aspects of AMR speech. For example, Miao et al. [19] suggested two statistical feature types for assessing pulse placements: Markov transition probabilities, which capture transition dynamics, and joint and conditional entropies, which reflect the probability distribution of pulses at certain places. However, these attributes ignore potential pulse position changes throughout the AMR encoding process, limiting their ability to adequately define AMR speech. Ren et al. [20] overcame this constraint by introducing features with similar pulse position probability. Despite improvements, these features failed to account for the distributions of two-track pulses at different places, limiting their usefulness. To address these

issues, Tian et al. [21] created a broader feature set that included long-term distribution features, short-term invariants, and track-to-track correlations. This set outperformed prior designs. However, its high dimensionality (2772 features) required a dimensionality reduction, which was accomplished using adaptive boosting (AdaBoost), lowering the feature set to 498 dimensions while maintaining effectiveness. Tian et al. [22] present a steganalysis approach with a multiple-classifier combination model at its foundation. There are three phases to the scheme. Initially, steganalysis features are fed into two different classifier sets, yielding the first and second types of prediction outputs. The second set of predictions is then handled as a unique form of detection feature and fed into an additional classifier set to produce the third type of prediction results. Finally, the three types of predictions are combined to generate the final detection result. Sun et al. [23] developed an adaptive steganalysis model for AMR-encoded speech. The method starts by creating a Markov transition matrix from the original speech signal to extract the statistical characteristics of pulse pairs (SCPPs) as the key feature. To capture global patterns, a convergence feature is used. These features are then combined and passed via an extreme gradient boosting (XGBoost) classifier. Model training optimizes the system for effective detection. The experimental findings showed that this method works effectively in detecting steganography in AMR-based speech streams.

Although previous studies have made significant progress in FCB-based steganalysis [18,20–23], they generally assume that the embedding rate—the proportion of hidden data in a speech sample—is known in advance. This assumption is often unrealistic in practice, emphasizing the importance of developing detection methods that can operate without prior knowledge of the embedding rate. To address this, Tian et al. [24] proposed a method based on Dempster–Shafer Theory (DST), which combines the outputs of multiple classifiers trained at different embedding rates to produce a final decision. However, this approach faces several issues: it performs poorly at low embedding rates due to data mismatch, struggles with high-dimensional features when using SVM classifiers, and lacks robustness to small deviations in the actual embedding rate. To improve adaptability, Sun et al. [25] introduced a method combining K-means clustering and ensemble learning. During training, speech samples are grouped based on feature and embedding rate similarities, and each cluster is handled by an XGBoost classifier to enhance detection performance under varying embedding conditions.

Despite the development of several steganalysis approaches [24,25] for recognizing AMR-based speech streams with unknown embedding rates, there are still several problems. Firstly, many existing steganalysis techniques [24,25] rely on handcrafted features, which are often limited in their ability to extract complex and intricate patterns from steganographic data. Secondly, the existing steganalysis approaches [24,25] treat AMR-based speech streams with different embedding rates independently, ignoring the inherent correlations between them. By neglecting the interdependencies among embedding rates, these methods fail to leverage the rich contextual information that could enhance classification accuracy.

Therefore, this paper proposes a multi-scale transformer-based framework integrated with multi-task learning [26,27], combining multi-rate regression with multi-rate classification to achieve enhanced steganalysis performance. By incorporating regression [28], the framework models embedding rates as continuous values, capturing fine-grained variations that complement the classification task and leveraging the inter-dependencies between discrete classes and continuous rates. This joint optimization approach improves the model's ability to generalize across diverse scenarios. Additionally, the hierarchical transformer [29,30]architecture enables unified multi-scale feature extraction, capturing local, pairwise, and global dependencies within speech streams. This design eliminates

reliance on handcrafted features, instead learning nuanced and robust representations directly from the speech streams. The main contributions of this paper are as follows:

- **Integrating multi-task learning:** Differing from the existing methods [24,25], our method utilizes the multi-task learning mechanism to enhance the detection performance by combining classification and regression, where classification distinguishes between cover and steganographic samples, and regression estimates the embedding rate. This joint learning framework improves feature extraction by capturing both discrete and continuous variations in embedding rates, enabling the model to learn more informative representations. The regression task provides additional supervision, reinforcing the classification process and improving the model's ability to detect steganographic samples with varying embedding rates.
- **Proposing a multi-scale transformer framework:** Unlike the existing steganalysis methods [24,25] that focus on the handcrafted features, our method proposes the multi-scale transformer-based framework, which can capture hierarchical dependencies within speech frames while preserving both local and global contextual information.

The remaining part of this paper is organized as follows. In Section 2, we provide a brief overview of related work on steganography and steganalysis in speech streams, focusing on AMR-based speech streams. Section 3 begins with an overview of the clustering and ensemble learning algorithms used. Section 4 explains the suggested detection mechanism. Section 5 compares the suggested method to past methods. Ultimately, the final section presents the conclusions.

2. Related Work

Steganography and steganalysis in VoIP-based speech streams have emerged as important research areas in information concealing. Because most speech codecs are ACELP-based, present steganography approaches focus on three-parameter domains: pitch delay [12,13], LPC [14,15,31], and FCB [17,18].

The unpredictability of the pitch period allows for information concealment. Huang et al. [12] suggested a steganography approach for G.723.1 encoded speech that involved changing the closed-loop adaptive codebook encoding procedure. Similarly, Yan et al. [13] proposed a triple-layer steganography solution for G.729 encoded speech that used pitch parameter direction adjustments and dual matrix encoding to improve embedding efficiency.

The LPC domain is another potential area for steganographic uses in speech frames. Steganographic approaches in this space usually include splitting LPC codebooks and embedding data during speech quantization process. For example, Xiao et al. [31] used a complementary neighbor–vertices technique for codebook partitioning and created the first QIM-based steganographic method for LPC coefficients, which was verified using iLBC and G.723.1 codecs. Liu et al. [14] proposed a QIM-based technique for G.723.1 streams, considering each quantization index set as a point in index space and using a genetic algorithm-based partitioning mechanism to reduce replacement distortion. Liu et al. [15] improved LPC-based steganography in G.723.1 streams by integrating matrix embedding, resulting in higher performance.

The pulse position search process involves considerable redundancy, as heuristic depth-first search methods often fail to achieve optimal results. This characteristic makes the Fixed Codebook (FCB) an effective medium for embedding additional information in speech signals [17,18]. Among the various coding parameters within each speech frame, FCB parameters are the most frequently utilized, particularly in adaptive multi-rate (AMR) speech coding. For instance, in the 12.2 kbps AMR-NB codec, FCB parameters account for 57.38% of the total frame data. As a result, the FCB domain has become a focal point for strategies that introduce auxiliary data within AMR-encoded speech, motivating research

into corresponding detection techniques. Early explorations in this direction include the work of Geiser and Vary [17], who refined the pulse position search by limiting the selection range to two out of eight possible positions. Building on this, Miao et al. [18] propose an approach that adjusts pulse combinations to accommodate additional signals within the FCB domain, providing a more flexible integration process through the application of an embedding factor.

With the wide use of AMR coding in mobile communications, detecting steganography in AMR-encoded speech has attracted growing attention. Miao et al. [19] proposed two features: one based on entropy and another on Markov properties of pulse positions, but they did not handle pulse position swaps well. To address this limitation, Ren et al. [20] estimated the probability of repeated pulse positions to reduce the effect of position changes. However, their method mainly focused on pulses at the same position and lacked the ability to capture more complex patterns. Tian et al. [21] introduced the SCPP method, which includes three types of features: long-term distribution, short-term transitions, and track-to-track correlations. Due to the high feature dimension (2772), they used AdaBoost to reduce it to 498 and avoid overfitting in SVM classifiers. Building on this, Sun et al. [23] proposed an adaptive detection method. They used a Markov matrix and SCPP to extract features, added a convergence metric, and trained an XGBoost model for better accuracy.

Although the above steganalysis methods [19–21,23] are effective, they share a key limitation: they rely on prior knowledge of the embedding rate in the speech signal. In real-world applications, it is often unknown whether a signal contains hidden information, let alone the extent of embedding. Therefore, future research should aim to develop detection methods that do not depend on predefined embedding rates. To tackle this problem, Tian et al. [24] proposed a detection method based on Dempster–Shafer theory (DST), which fuses results from multiple classifiers trained on different embedding rates. These classifiers use the R-SCPP feature set to support detection under varying conditions. Similarly, Sun et al. [25] introduced a method that combines K-means clustering and ensemble learning. During training, samples are grouped by embedding intensity, and each cluster is analyzed with an XGBoost classifier, improving adaptability to unknown embedding rates. Despite these advancements, existing steganalysis techniques [24,25] still present notable challenges. Most approaches rely on manually designed features, which may not effectively capture intricate steganographic patterns. Additionally, many methods handle different embedding rates as independent cases, disregarding their underlying correlations and missing critical contextual relationships. This limitation reduces their ability to adapt to real-world scenarios where embedding characteristics can vary dynamically. Consequently, there remains a need for more robust detection frameworks capable of recognizing covert modifications without prior assumptions about embedding configurations.

3. Problem Statement

In the information-theoretic framework for steganalysis technology [9,10], the difference between the probability distributions of the original carrier and the steganographic carrier can be quantified using the Kullback–Leibler (KL) divergence, which is expressed as follows:

$$KL(P_C|P_S) = \sum_{x \in C} P_C(x) log \frac{P_C(x)}{P_S(x)}. \quad (1)$$

For steganography technology [7,13], the primary aim is to minimize the Kullback–Leibler (KL) divergence between the original carrier's and steganographic carrier's probability distributions, which enhances the imperceptibility of the hidden communication and makes detection more challenging. Conversely, steganalysis technology seeks to identify the presence of embedded secret information with maximum precision. This detection pro-

cess typically involves extracting distinctive features from both the original carrier and the steganographic carrier through a specific operation and then analyzing the KL divergence between these extracted features to determine the likelihood of covert communication.

Steganalysis technology performs feature extraction operation φ by searching for maximization of $\boldsymbol{KL}(\varphi_C|\varphi_S)$, thereby completing the corresponding detection task.

Most existing steganalysis methods [19–23] for speech streams operate under the assumption that the embedding rate is known, which can be formally described as follows:

$$\varphi_k = max \sum_{i=1}^{N} \boldsymbol{KL}(\boldsymbol{\varphi_{C_i}}|\boldsymbol{\varphi_{S_i}}(a_k)), a_k \in A. \tag{2}$$

where N represents the number of samples, and a_k represents a specific embedding rate. From the above formula, it can be seen that most existing covert communication methods search for a transformation operation, φ, that maximizes the Kullback–Leibler (KL) divergence between the original carrier and the steganographic carrier under the assumption that the embedding rate is known.

For the steganalysis of speech streams with unknown embedding rates [24,25], the formal description is as follows:

$$\varphi = max \sum_{i=1}^{N} \boldsymbol{KL}(\boldsymbol{\varphi_{C_i}}|\boldsymbol{\varphi_{S_i}}(\boldsymbol{A})). \tag{3}$$

In Equation (3), $\boldsymbol{A}$ represents the summation across all embedding rates. By combining Equations (2) and (3), it can be seen that steganalysis of speech streams with unknown embedding rates eliminates the assumption of known embedding rates.

However, existing steganalysis techniques [24,25] for speech streams with unknown embedding rates exhibit certain limitations. First, many of these methods rely on manually crafted feature representations, which often fail to comprehensively capture the complex and subtle characteristics of steganographic modifications. Second, current approaches [24,25] do not effectively account for the relationships between different embedding rates. Most detection models are trained separately for speech samples with varying embedding intensities, overlooking the intrinsic dependencies between different embedding levels.

4. Proposed Method

This section presents the Multi-Scale Transformer with Multi-Task Learning (MT-MTL) framework, designed for the steganalysis of speech streams without prior knowledge of embedding rates. The model leverages the hierarchical structure of speech signals and simultaneously addresses two interrelated objectives: classifying multiple embedding rates and estimating embedding intensity. The framework comprises four key components: an input processing module, a codeword-embedding module, a multi-scale transformer module, and a multi-task learning mechanism. The overall system architecture is illustrated in Figure 1.

4.1. Input Module

For the speech streams $\boldsymbol{X}$, it contains multiple codeword sequences as follows:

$$\boldsymbol{X} = [\boldsymbol{X_1}, \boldsymbol{X_2}, \ldots, \boldsymbol{X_g}]. \tag{4}$$

where g indicates the number of the codeword sequences. Each codeword sequence $\boldsymbol{X_i}(1 \leq i \leq g)$ has multiple codewords, as follows:

$$\boldsymbol{X}_i = [\boldsymbol{X_{i,1}}, \boldsymbol{X_{i,2}}, \ldots, \boldsymbol{X_{i,m}}]. \tag{5}$$

where m is the number of speech frames in $\boldsymbol{X_i}$. For the j-th speech frame in the i-th codeword sequence, it contains four sub-frames and has four codewords as follows:

$$\boldsymbol{X}_{i,j} = [x_{i,j,1}, x_{i,j,2}, x_{i,j,3}, x_{i,j,4}]. \tag{6}$$

Therefore, $\boldsymbol{X_i}$ can also be described as follows:

$$\boldsymbol{X}_i = [x_{i,1,1}, \ldots, x_{i,1,4}, x_{i,2,1}, \ldots, x_{i,2,4}, \ldots, x_{i,m,1}, \ldots, x_{i,m,4}]. \tag{7}$$

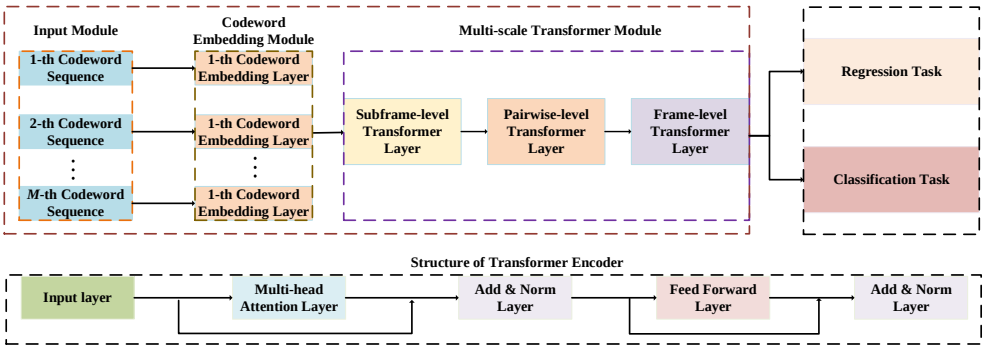


Figure 1. The framework of our method.

4.2. Codeword Embedding Module

To effectively represent semantic and structural relationships, discrete codewords are transformed into continuous vector representations through a codeword-embedding layer [32,33]. This embedding process generates dense, trainable feature representations that improve model learning efficiency. By mapping discrete codewords into a high-dimensional space, the embeddings enable subsequent layers to capture complex dependencies and structural patterns within the speech data [32,33].

For the $x_{i,j,k}(1 \leq k \leq 4)$ in the j-th speech frame of the i-th codeword sequence, the embedding process can be defined as follows:

$$\boldsymbol{r_{i,j,k}} = \boldsymbol{E_i F_{one\_hot}}(\boldsymbol{x_{i,j,k}}). \tag{8}$$

where $\boldsymbol{E_i}$ represents the embedding matrix for the i-th codeword sequence, and $\boldsymbol{F_{one\_hot}}$ denotes the one hot representation. This mapping enables the model to treat codewords as dense vectors, facilitating the extraction of semantic patterns in later stages. By repeating the same process to other codewords in the i-th codeword sequence, we can obtain the following result:

$$\boldsymbol{R_i} = [\boldsymbol{r_{i,1,1}}, \ldots, \boldsymbol{r_{i,1,4}}, \boldsymbol{r_{i,2,1}}, \ldots, \boldsymbol{r_{i,2,4}}, .., \boldsymbol{r_{i,m,1}}, \ldots, \boldsymbol{r_{i,m,4}}]. \tag{9}$$

Using the same operation in other codeword sequences, the mapped codeword sequences are as follows:

$$\boldsymbol{R} = [\boldsymbol{R_1}, \boldsymbol{R_2}, \ldots, \boldsymbol{R_g}]. \tag{10}$$

Then, to mitigate overfitting and reduce the dimensionality of the steganalysis features, the average operation is performed on the elements within $\boldsymbol{R}$.

$$\boldsymbol{R}_A = Average(\boldsymbol{R_1}, \boldsymbol{R_2}, \ldots, \boldsymbol{R_g}). \tag{11}$$

where $\boldsymbol{R_A}$ indicates the output of the average operation, and *Average* is the average operation.

4.3. Multi-Scale Transformer Module

The multi-scale transformer [29,30] is designed to effectively model the hierarchical structure of speech streams by capturing local subframe patterns, interactions between adjacent subframes, and global dependencies across entire frames. Unlike traditional methods [24,25] that rely on handcrafted features with limited adaptability, this approach automatically learns multi-scale representations, preserving structural information while maintaining computational efficiency. It adapts flexibly to different speech frame configurations and improves interpretability by revealing the contribution of each hierarchical level to the overall representation.

The multi-scale transformer captures hierarchical relationships in speech frames through three stages: (1) Subframe-Level Transformer, (2) Pairwise Subframe-Level Transformer, and (3) Frame-Level Transformer. Each stage incrementally integrates information, modeling both local and global dependencies.

4.3.1. Subframe-Level Transformer

Subframes, as fundamental units of a speech frame, contain four codewords and exhibit strong local dependencies. The subframe-level transformer focuses on capturing these relationships, extracting essential features while filtering out irrelevant information. This targeted processing preserves critical details for higher-level modeling and enables efficient parallel computation.

The subframe-level (SL) transformer extracts fine-grained features from individual subframes, each containing a single codeword. The first step is to apply the multi-head attention mechanism to model dependencies between subframes, following these steps:

$$\boldsymbol{C_{mha-sl}} = Contact(H_{sl,1}, H_{sl,2}, \ldots, H_{sl,h})\boldsymbol{W_{sl}^O}. \tag{12}$$

where h indicates the number of the heads, and $\boldsymbol{C_{sl}}$ indicates the output of the multi-head attention mechanism. And $\boldsymbol{H_{sl,i}}(1 \leq i \leq h)$ computes as follows:

$$\boldsymbol{H_{sl,i}} = Softmax(\frac{\boldsymbol{Q_{sl,i}}\boldsymbol{K_{sl,i}^{(T)}}}{\sqrt{d_k}})\boldsymbol{V_{sl,i}}. \tag{13}$$

where d_k indicates the dimension of the i-th head, (T) represents the transition operation and the definition of $\boldsymbol{Q_{sl,i}}$, $\boldsymbol{K_{sl,i}}$ and $\boldsymbol{V_{sl,i}}$ are as follows:

$$\boldsymbol{Q_{sl,i}} = \boldsymbol{R_A}\boldsymbol{W_{sl,i}^Q}, \boldsymbol{K_{sl,i}} = \boldsymbol{R_A}\boldsymbol{W_{sl,i}^K}, \boldsymbol{V_{sl,i}} = \boldsymbol{R_A}\boldsymbol{W_{sl,i}^V}. \tag{14}$$

The second step is the residual connection and layer normalization operation, which are applied to stabilize learning as follows:

$$\boldsymbol{C_{sl}^1} = LayerNorm(\boldsymbol{C} + \boldsymbol{C_{mha-sl}}). \tag{15}$$

The third step is to utilize a two-layer FFN to refine the features as follows:

$$\boldsymbol{C}_{\boldsymbol{sl}}^{\boldsymbol{2}} = ReLU(\boldsymbol{C}_{\boldsymbol{sl}}^{\boldsymbol{1}}\boldsymbol{W}_{\boldsymbol{sl,1}}^{\boldsymbol{ffn}} + \boldsymbol{b}_{\boldsymbol{sl,1}}^{\boldsymbol{ffn}})\boldsymbol{W}_{\boldsymbol{sl,2}}^{\boldsymbol{ffn}} + \boldsymbol{b}_{\boldsymbol{sl,2}}^{\boldsymbol{ffn}}, \tag{16}$$

where $ReLU$ represents the rectified linear unit operation. Then, another residual connection and normalization performs on the $\boldsymbol{C}_{\boldsymbol{sl}}^{\boldsymbol{2}}$ as described in the following equation:

$$\boldsymbol{C}_{\boldsymbol{sl}}^{\boldsymbol{3}} = LayerNorm(\boldsymbol{C}_{\boldsymbol{sl}}^{\boldsymbol{1}} + \boldsymbol{C}_{\boldsymbol{sl}}^{\boldsymbol{2}}). \tag{17}$$

The $\boldsymbol{C}_{\boldsymbol{sl}}^{\boldsymbol{3}}$ is the output of the subframe-level transformer, which encodes local dependencies for each subframe.

4.3.2. Pairwise Subframe-Level Transformer

While the subframe-level transformer captures local features, adjacent subframes share contextual information crucial for detecting embedding rates. The pairwise subframe-level transformer models these dependencies, integrating subframe features into higher-level representations. By merging subframes early, it reduces feature space and computational complexity while preserving essential information for frame-level modeling.

This stage captures dependencies between adjacent subframes while reducing the feature space. The process begins with pairwise grouping, a key step in the pairwise subframe-level (PSL) transformer that facilitates subframe interaction modeling. Specifically, each frame, consisting of four subframes, is divided into two adjacent subframe pairs to enhance contextual representation. The process is as follows:

$$\begin{aligned} \boldsymbol{C}_{\boldsymbol{psl}}[t,1] &= Contact(\boldsymbol{C}_{\boldsymbol{sl,t,1}}^{\boldsymbol{3}}, \boldsymbol{C}_{\boldsymbol{sl,t,2}}^{\boldsymbol{3}}), \\ \boldsymbol{C}_{\boldsymbol{psl}}[t,2] &= Contact(\boldsymbol{C}_{\boldsymbol{sl,t,3}}^{\boldsymbol{3}}, \boldsymbol{C}_{\boldsymbol{sl,t,4}}^{\boldsymbol{3}}), \end{aligned} \tag{18}$$

Therefore, we can obtain the following result:

$$\boldsymbol{C}_{\boldsymbol{psl}} = [\boldsymbol{C}_{\boldsymbol{psl,1,1}}, \boldsymbol{C}_{\boldsymbol{psl,1,2}}, \ldots, \boldsymbol{C}_{\boldsymbol{psl,m,2}}]. \tag{19}$$

Then, to normalize the pairwise representation for subsequent transformer operations, a linear transformation is applied to each sub-pair:

$$\boldsymbol{C}_{\boldsymbol{psl}}^{\boldsymbol{proj}} = \boldsymbol{C}_{\boldsymbol{psl}}\boldsymbol{W}_{\boldsymbol{psl}} + \boldsymbol{b}_{\boldsymbol{psl}}. \tag{20}$$

This reduces the dimensionality of each pair from $2d$ to d.

The second step is multi-head attention (MHA) within pairs, which captures relationships between subframes within each pair.

$$\boldsymbol{C}_{\boldsymbol{mha-psl}} = \textbf{Concat}(H_{psl,1}, H_{psl,2}, \ldots, H_{psl,h})\boldsymbol{W}_{\boldsymbol{psl}}^{\boldsymbol{O}}. \tag{21}$$

The third step is the residual connections and normalization operations as follows:

$$\boldsymbol{C}_{\boldsymbol{psl}}^{\boldsymbol{1}} = LayerNorm(\boldsymbol{C}_{\boldsymbol{psl}}^{\boldsymbol{proj}} + \boldsymbol{C}_{\boldsymbol{mha-psl}}). \tag{22}$$

The fourth step is to use the FFN further to process the pairwise representations by the following equation:

$$\boldsymbol{C}_{\boldsymbol{psl}}^{\boldsymbol{2}} = LayerNorm(\boldsymbol{C}_{\boldsymbol{psl}}^{\boldsymbol{1}} + \textbf{FFN}(\boldsymbol{C}_{\boldsymbol{psl}}^{\boldsymbol{1}})). \tag{23}$$

The output $\boldsymbol{C}_{\boldsymbol{psl}}^{\boldsymbol{2}}$ aggregates dependencies across adjacent subframes.

4.3.3. Frame-Level Transformer

The frame-level transformer aggregates subframe features to capture global dependencies across the entire frame. It models long-range relationships to generate a unified representation, enhancing the differentiation between original and steganographic samples at various embedding rates. By leveraging full-frame context, it strengthens classification performance and improves model robustness.

The frame-level (FL) transformer follows four steps, beginning with subframe aggregation. Each frame, previously divided into two pairwise representations, is reconstructed into a unified frame token to facilitate global modeling. This process involves the following:

$$\boldsymbol{C_{fl}}[t] = Contact(C^2_{psl,t,1}, C^2_{psl,t,2}), \tag{24}$$

where $t \in [1, 100]$ is the frame index. Therefore, we can obtain the following result:

$$\boldsymbol{C_{fl}} = [\boldsymbol{C_{fl,1}}, \boldsymbol{C_{fl,2}}, \ldots, \boldsymbol{C_{fl,m}}]. \tag{25}$$

Then, a linear layer reduces the frame representation back to the original embedding dimension (d) for computational efficiency:

$$\boldsymbol{C^{proj}_{fl}} = \boldsymbol{C_{fl}}\boldsymbol{W_{fl}} + \boldsymbol{b_{fl}}. \tag{26}$$

The second step is to use MHA to capture global dependencies across subframes within a frame:

$$\boldsymbol{mha - fl} = \textbf{Concat}(H_{fl,1}, H_{fl,2}, \ldots, H_{fl,h})\boldsymbol{W^O_{fl}}. \tag{27}$$

The third step is to apply the residual connections and layer normalization as follows:

$$\boldsymbol{C^1_{fl}} = LayerNorm(\boldsymbol{C^{proj}_{fl}} + \boldsymbol{C_{mha-fl}}). \tag{28}$$

The fourth step is to use a two-layer FFN to refine the global features as follows:

$$\boldsymbol{U} = LayerNorm(\boldsymbol{C^1_{fl}} + \textbf{FFN}(\boldsymbol{C^1_{fl}})). \tag{29}$$

4.4. Multi-Task Learning Mechanism

The proposed method adopts a multi-task learning framework to enhance steganalysis performance. The primary task focuses on classifying cover and steganographic samples, which is essential for detecting hidden information. Meanwhile, the regression task captures continuous variations in embedding rates, helping to model the underlying distribution of steganographic modifications. By jointly optimizing both tasks, the model learns more expressive representations, improving classification accuracy and robustness against diverse embedding strategies.

4.4.1. Classification Task

The classification task distinguishes original samples from those with different embedding rates by predicting discrete embedding categories. The final transformer output, which encodes contextualized features across all codewords, is fed into a fully connected layer to produce classification logits. Formally, this process operates as follows:

$$y_{class} = Softmax(\boldsymbol{W_{class}}\boldsymbol{U} + b_{class}) \tag{30}$$

where $\boldsymbol{W_{class}}$ and b_{class} are learnable parameters, and $\boldsymbol{h}$ is the aggregated output from the last transformer layer. The classification loss $\boldsymbol{\zeta_{class}}$, is computed using cross-entropy.

4.4.2. Regression Task

In addition to the classification task, the model performs regression to estimate the exact embedding rate. The regression head consists of a fully connected layer that outputs a continuous value representing the embedding rate. It is formulated as follows:

$$y_{reg} = \boldsymbol{W_{reg}} \boldsymbol{U} + b_{reg} \tag{31}$$

where $\boldsymbol{W_{reg}}$ and b_{reg} are learnable parameters. We use the mean squared error (MSE) as the loss function for regression, denoted as $\boldsymbol{\zeta_{reg}}$.

4.4.3. Joint Loss Function

To train the model in a multi-task setting, we combine the classification and regression losses into a joint objective. The total loss, $\boldsymbol{\zeta}$, is given via the following:

$$\boldsymbol{\zeta} = \alpha \boldsymbol{\zeta class} + (1 - \alpha) \boldsymbol{\zeta_{reg}} \tag{32}$$

where α and β are weighting factors that control the trade-off between classification and regression performance. This combined objective allows the model to simultaneously improve performance on both tasks, leveraging shared features between them.

5. Experimental Result and Analysis

Section 5.1 describes the AMR-based steganalysis dataset and the evaluation metrics employed to validate the effectiveness of the proposed framework. Section 5.2 investigates the influence of embedding dimension on detection performance, followed by an analysis of the impact of transformer-layer depth in Section 5.3. Section 5.4 focuses on optimizing the joint loss coefficient for multi-task learning. In Section 5.5, the regression task is evaluated to assess its contribution to modeling continuous embedding rates. Section 5.6 presents a comparative study between the proposed method and existing state-of-the-art approaches. Finally, Section 5.7 conducts an ablation study to examine the contribution of each module within the proposed architecture.

5.1. Experimental Setup and Metrics

Dataset: The speech dataset used in this study was published by the research group at Tsinghua University [34]. It consists of PCM-format voice samples, which serve as the source data. These samples are monophonic, sampled at 8000 Hz, with a quantization depth of 16 bits. Following prior studies [24,25], each speech sample was truncated to a duration of 10 s. The dataset includes 1000 utterances each from Chinese male, Chinese female, English male, and English female speakers, resulting in a total of 4000 speech samples. To construct the cover speech dataset, all samples were encoded using the AMR codec at a bit rate of 12.2 kbps.

Steganography methods: In addition, we constructed four steganographic voice datasets by inserting hidden messages simulated as random binary bit streams into each sample using various steganographic techniques. There are four steganography methods, including Geiser's method [17], Miao's method $\eta = 1$ [18], Miao's method $\eta = 2$ [18], and Miao's method $\eta = 4$ [18].

Training set and testing set: The training set included 3000 cover samples chosen at random from the cover speech dataset. 30,000 steganographic samples were created using a specific steganographic approach [17,18] with different embedding rates. The embedding rates varied from 10% to 100% in 10% increments, with 3000 steganographic samples generated for each rate. As a result, the training set for each steganographic dataset included 3000 cover samples and 30,000 steganographic samples. The testing set was

created using the same method, using 1000 cover samples and 10,000 steganographic samples that had comparable embedding rate distributions to the training set. In both sets, steganographic samples were labeled as positive, whereas cover samples were designated as negative.

Comparison methods: As described in Section 2, Tian et al. [24] proposed a DST-based approach, known as DST, for recognizing AMR-based voice streams with unknown embedding rates. Sun et al. [25] have presented three steganalysis methods. The first method is called SVM-MSDA, which trains the SVM classifier based on the mixed sample data augmentation. The second method is called XGBoost-MSDA, which trains XGBoost based on the mixed sample data augmentation. The third method is called Kmeans-XGBoost, which combines clustering and ensemble learning.

Training Setup: The training and evaluation are performed on a GeForce GTX 3080 GPU equipped with 11 GB of memory. The model and algorithms were implemented using the PyTorch v2.2.2 framework. The Adam optimizer was adopted with a learning rate of 0.001, and cross-entropy loss was utilized for the classification task, while the mean squared error is used for the regression task during training. The number of training epochs is set to 50, and the batch size is fixed at 64. A summary of the hyperparameters used in our method is presented in Table 1.

Table 1. The hyperparameter setting.

Hyper-Parameter	Value
The embedding dimension	20
The number of attention heads in the transformer layer	5
The number of transformer layers	2
The dropout rate in the transformer layer	0.2
Hidden size in transformer	20
Dropout rate	0.3

Metrics: Furthermore, we use three criteria to assess the effectiveness of the suggested method. The first statistic is accuracy (ACC), which indicates the ratio of correctly classified samples to the total number of samples, defined as follows:

$$ACC = \frac{N_{TN} + N_{TP}}{N_{TN} + N_{TP} + N_{FN} + N_{FP}} \tag{33}$$

where N_{TN} represents the number of cover samples correctly classified, N_{TP} denotes the number of steganographic samples correctly classified, N_{FP} refers to the number of steganographic samples misclassified as cover samples, and N_{FN} is the number of cover samples misclassified as steganographic samples. The second metric is the error rate, which quantifies the proportion of incorrect predictions made by the model over the total number of predictions. It is formally defined as follows:

$$ErrorRate = 1 - Accuracy \tag{34}$$

The third metric is the false positive rate (FPR), defined as follows:

$$FPR = \frac{N_{FP}}{N_{TN} + N_{FP}} \tag{35}$$

The fourth metric is the false negative rate (FNR), namely

$$FNR = \frac{N_{FN}}{N_{FN} + N_{TP}} \tag{36}$$

5.2. Effect of Embedding Dimension

To evaluate the influence of embedding dimension on model performance, experiments were conducted on four steganographic datasets with varying dimension configurations. The embedding dimension is critical for representing informative features and capturing intricate dependencies within the data. By assessing classification accuracy under different dimension settings, the goal is to identify an optimal configuration that achieves a balance between computational efficiency and detection effectiveness. The experimental results are illustrated in Figure 2.

As illustrated in Figure 2, increasing the embedding dimension generally leads to improved classification accuracy across all evaluated datasets; however, the performance gains tend to plateau beyond a dimension of 20. Specifically, for the Geiser dataset [17], the peak accuracy of 92.46% is observed at dimension 20. For the Miao ($\eta = 1$) dataset, the highest accuracy (86.68%) is achieved at dimension 35. In the case of Miao ($\eta = 2$) [18], accuracy continues to increase with larger dimensions, reaching 92.45% at 35. The Miao ($\eta = 4$) dataset [18] demonstrates the most significant benefit from higher dimensions, achieving 94.79% at 35. Nonetheless, the limited performance improvement beyond dimension 20 indicates that, for less complex datasets, lower-dimensional embeddings can achieve comparable detection performance with reduced computational cost, suggesting that a dimension of 20 offers a practical trade-off.

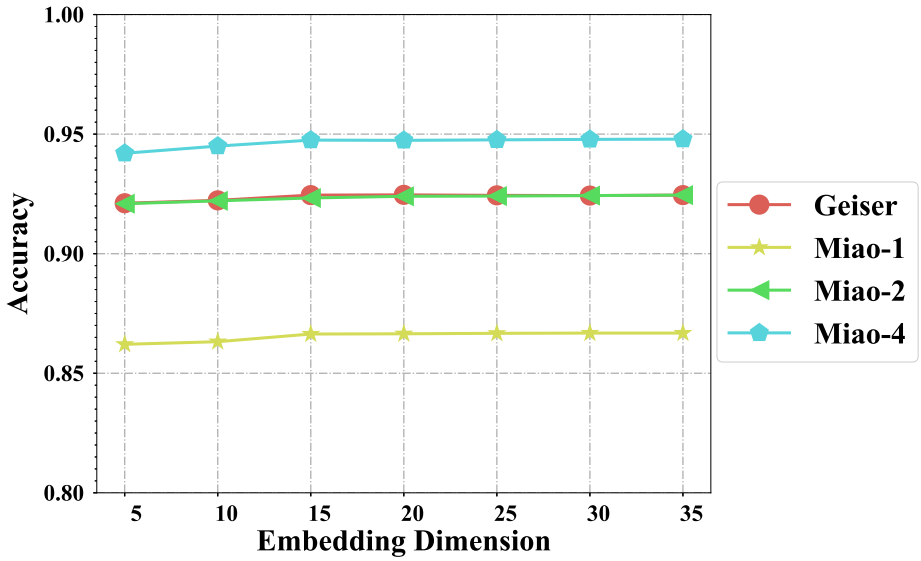


Figure 2. Impact of embedding dimension on detection accuracy across multiple datasets.

5.3. Influence of the Transformer Layer

To investigate the effect of transformer layer depth on classification performance, we conducted a series of experiments across four steganographic datasets. Transformer depth is critical for capturing hierarchical structures and modeling long-range dependencies, which are essential for extracting subtle steganographic patterns. While deeper networks can enhance representational capacity, they also introduce additional computational costs. This analysis aims to identify an optimal depth that balances accuracy and efficiency. The experimental results are summarized in Figure 3.

As shown in Figure 3, increasing the number of transformer layers generally leads to improved classification accuracy across all datasets, although the performance gains diminish beyond a certain depth. On the Geiser dataset [17], accuracy reaches a maximum of 92.53% with five layers, with the most significant improvement observed between one and two layers. For the Miao ($\eta = 1$) dataset [18], performance peaks at 86.71% with five layers,

stabilizing after three layers. Similarly, the Miao ($\eta = 2$) dataset [18] achieves 92.42% accuracy, with marginal improvements beyond three layers. In contrast, the Miao ($\eta = 4$) dataset [18], characterized by higher embedding complexity, benefits more from deeper architectures, achieving 94.81% accuracy with five layers. These findings suggest that, while increased transformer depth can enhance detection performance, simpler datasets can achieve near-optimal accuracy with two to three layers, whereas more complex cases necessitate deeper models for effective feature extraction.

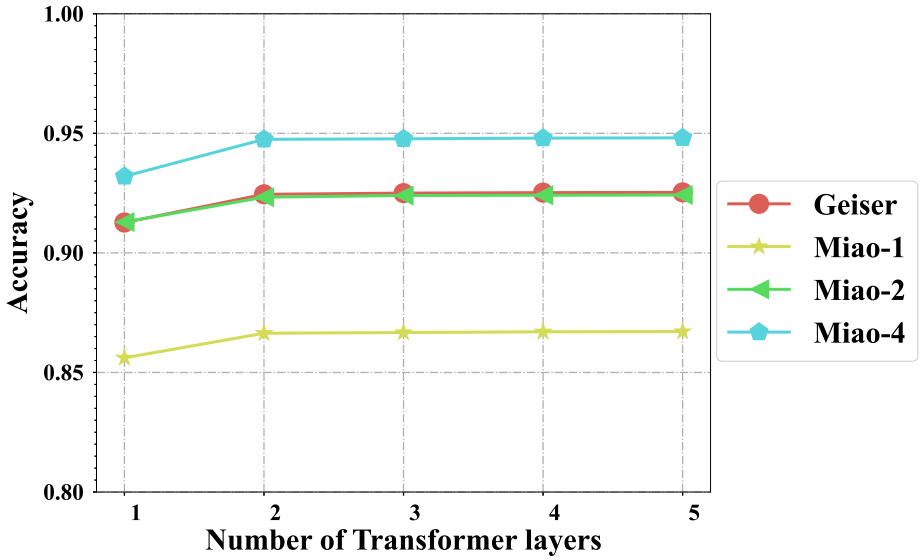


Figure 3. Effect of transformer layers on detection accuracy across multiple steganography methods.

5.4. Effect of Joint Optimization Coefficient

In the joint optimization framework, the total loss is a weighted combination of the classification and regression losses:

$$\zeta = \alpha \zeta class + (1 - \alpha)\zeta_{reg} \tag{37}$$

where $\alpha \in [0, 1]$ controls the trade-off between the two tasks. Proper selection of α is crucial for balancing the contributions of both tasks, ensuring neither dominates the optimization process. This experiment investigates the effect of different α values on multi-task performance.

To assess the influence of the joint optimization coefficient (α) on the performance of the proposed multi-task learning framework, we conduct experiments with α values ranging from 0.1 to 1.0. This coefficient determines the relative weighting of the classification and regression losses during training. Appropriate calibration of α enables the model to jointly learn discrete classification and continuous regression tasks, thereby enhancing the quality of the learned feature representations. The goal is to identify an optimal α that achieves the best overall performance across various datasets. The experimental results are presented in Figure 4.

As shown in Figure 4, the relationship between α and classification accuracy exhibits a non-linear trend. Accuracy improves significantly as α increases from 0.1 to 0.6, highlighting the benefits of incorporating regression for improved generalization and feature learning. The Geiser dataset [17] achieves its highest accuracy of 92.454% at $\alpha = 0.6$, while the Miao ($\eta = 1$) dataset [18] reaches 86.64% under the same configuration. Similarly, the Miao ($\eta = 2$) and Miao ($\eta = 4$) datasets [18] attain peak accuracies of 92.33% and 94.75%, respectively, at $\alpha = 0.6$. However, further increasing α leads to a slight degradation in performance, suggesting that excessive emphasis on regression may undermine the

classification objective. These findings emphasize the importance of balancing both tasks, with $\alpha = 0.6$ identified as the optimal setting for maximizing detection performance.

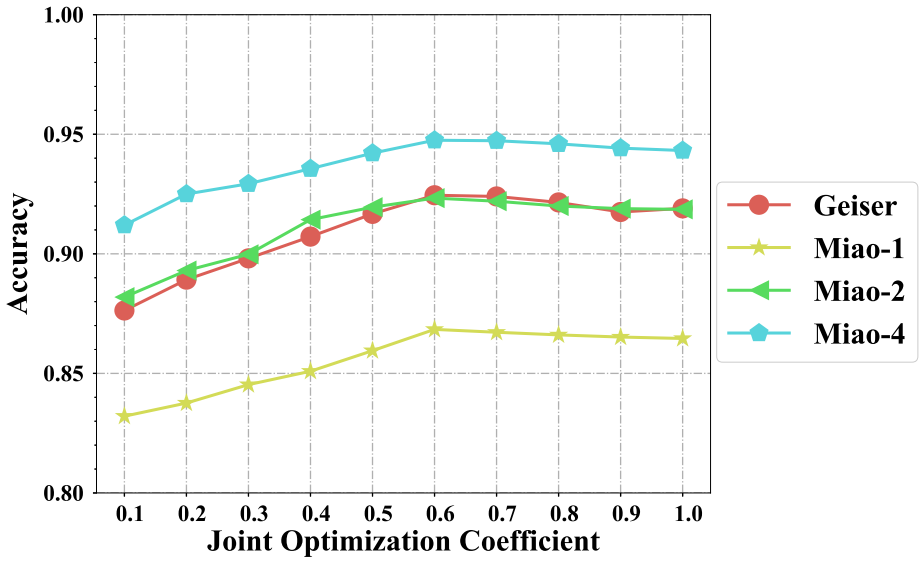


Figure 4. Impact of joint optimization coefficient on detection accuracy across multiple steganography methods.

5.5. Analysis of the Regression Task

To validate the effectiveness of the regression branch in modeling continuous embedding rates, we conducted a mean squared error (MSE) evaluation. This metric quantifies the model's ability to capture fine-grained embedding levels, which is essential for supporting multi-task learning and improving the discriminative representation shared with the classification task. The results are shown in Table 2.

Table 2. The average MSE result for multiple steganography methods.

Steganography Method	Average MSE Value
Geiser's method [17]	0.0206
Miao's ($\eta = 1$) method [18]	0.0341
Miao's ($\eta = 2$) method [18]	0.0224
Miao's ($\eta = 4$) method [18]	0.0101

From Table 2, it can be seen that the regression task consistently achieves low mean squared error (MSE) values across multiple steganography methods, demonstrating the effectiveness of the proposed framework in estimating embedding rates. Notably, the model achieves an MSE of 0.0101 on the Miao ($\eta = 4$) dataset, indicating high estimation accuracy under higher embedding rates. Even under lower embedding rates, such as Miao ($\eta = 1$), the MSE remains relatively low at 0.0341, suggesting strong robustness. These results confirm that the regression branch successfully captures continuous variations in embedding rates, thereby enhancing the overall performance and generalization capability of the multi-task learning framework.

5.6. Performance Comparison with the Existing Methods

The scheme proposed in this paper is evaluated using four different steganographic datasets. The experimental results for four steganographic techniques are illustrated in Figure 5.

Figure 5 presents the performance comparison of various methods. The DST-based approach [24] exhibited the lowest accuracy, around 75% on the Geiser dataset and be-

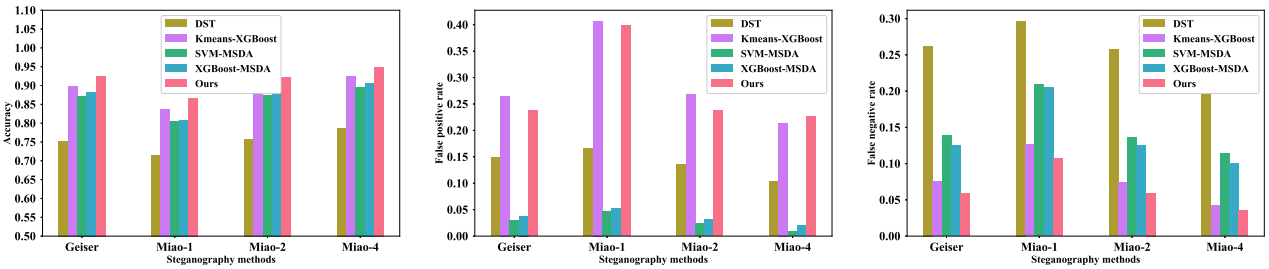
low 80% on others. SVM-MSDA [25] and XGBoost-MSDA [25] demonstrated moderate improvements, with SVM achieving 87.1% on Geiser and XGBoost slightly surpassing it at 88.3%. The Kmeans-XGBoost [25] method outperformed these baselines, reaching nearly 90% on Geiser and maintaining stable performance across datasets. Our proposed method achieved the highest detection accuracy, reaching 92.45% on Geiser and 94.75% on Miao ($\eta = 4$), consistently surpassing all baselines. While Kmeans-XGBoost [25] attained a 5% FNR on Geiser, our approach further reduced it to 3.5% on Miao ($\eta = 4$), demonstrating superior detection capability, especially at low embedding rates. Additionally, our method maintained competitive FPR values, effectively minimizing false positives across all datasets.

DST-based methods [24] encounter significant limitations, particularly at low embedding rates. Their dependence on fixed embedding rate classifiers restricts their ability to accurately identify steganographic samples, often misclassifying low embedding rate instances as cover samples. This results in high FNR values and reduced overall accuracy. While these methods achieve relatively low FPR by correctly classifying most cover samples, their reliance on manual feature extraction and inability to adapt to embedding rate variations constrain their effectiveness.

SVM-MSDA and XGBoost-MSDA [25] rely on handcrafted features and suffer from underfitting due to their training strategies. By merging steganographic samples with different embedding rates into a single training set, they fail to capture the distinct patterns of low embedding rate steganography. This leads to the frequent misclassification of low-rate steganographic samples as cover samples, increasing FNR values. Additionally, their inability to model relationships between different embedding rates further limits detection performance.

The Kmeans-XGBoost detection method [25] achieves competitive performance by utilizing a clustering mechanism. It assigns samples to clusters based on distance measures and applies a classifier within each cluster for detection. Its multi-class training strategy, which treats each embedding rate as a separate class, helps address data imbalance and enhance detection accuracy. However, its reliance on handcrafted features and lack of modeling for embedding rate interdependencies limits its adaptability to highly diverse embedding rates, leading to slightly higher FPR values.

In contrast, our approach addresses these limitations by replacing handcrafted feature reliance with a hierarchical transformer architecture that captures both local and global dependencies across embedding rates. Unlike traditional methods [24,25], which overlook embedding rate correlations, our model employs a multi-task learning framework that integrates classification and regression. This design enables explicit modeling of embedding rate interdependencies, enhancing sensitivity to low embedding rate steganography. As a result, it significantly reduces FNR values while improving overall detection accuracy.



(**a**) The ACC result (**b**) The FPR result (**c**) The FNR result

Figure 5. The accuracy results on different steganography methods.

To provide a more precise evaluation of the model's detection capability under varying embedding rates, we report the recall values separately for each embedding level. Since the primary objective in steganalysis is to correctly detect steganographic samples, recall is an appropriate metric for measuring performance in this context. The results are presented in Figure 6.

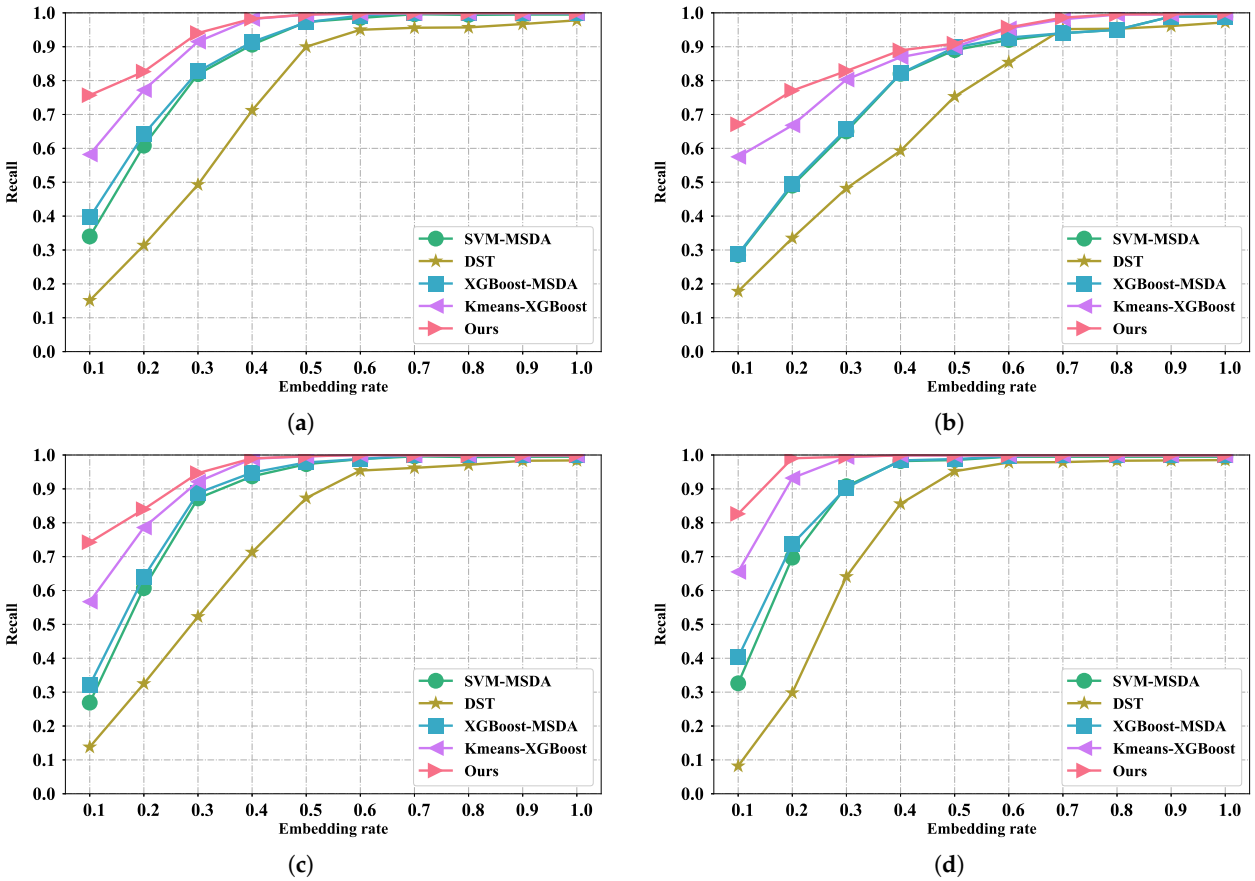


Figure 6. The recall result with each embedding rate on multiple steganography methods. (**a**) The recall result with each embedding rate on Geiser's steganography method; (**b**) the recall result with each embedding rate on Miao's (η = 1) steganography method; (**c**) the recall result with each embedding rate on Miao's (η = 2) steganography method; (**d**) the recall result with each embedding rate on Miao's (η = 4) steganography method.

As illustrated in Figure 6, two key observations can be drawn: (i) When the embedding rate is below 60%, our method consistently achieves the highest recall among all compared approaches, demonstrating superior detection capability in low embedding rate conditions. Specifically, for the Geiser steganography method at an embedding rate of 10%, the DST-based method yields a recall below 15%, while SVM-MSDA and XGBoost-MSDA achieve recall values below 40%. In contrast, our method attains a recall exceeding 70%. (ii) A positive correlation is observed between the embedding rate and recall, and the performance gap between the proposed method and the DST-based method gradually decreases as the embedding rate increases.

To further evaluate the effectiveness of our approach and compare it with existing steganalysis methods, we present receiver operating characteristic (ROC) curves for detecting four steganographic techniques, as shown in Figure 7. The results clearly show that our method is effective and consistently outperforms the baseline approaches.

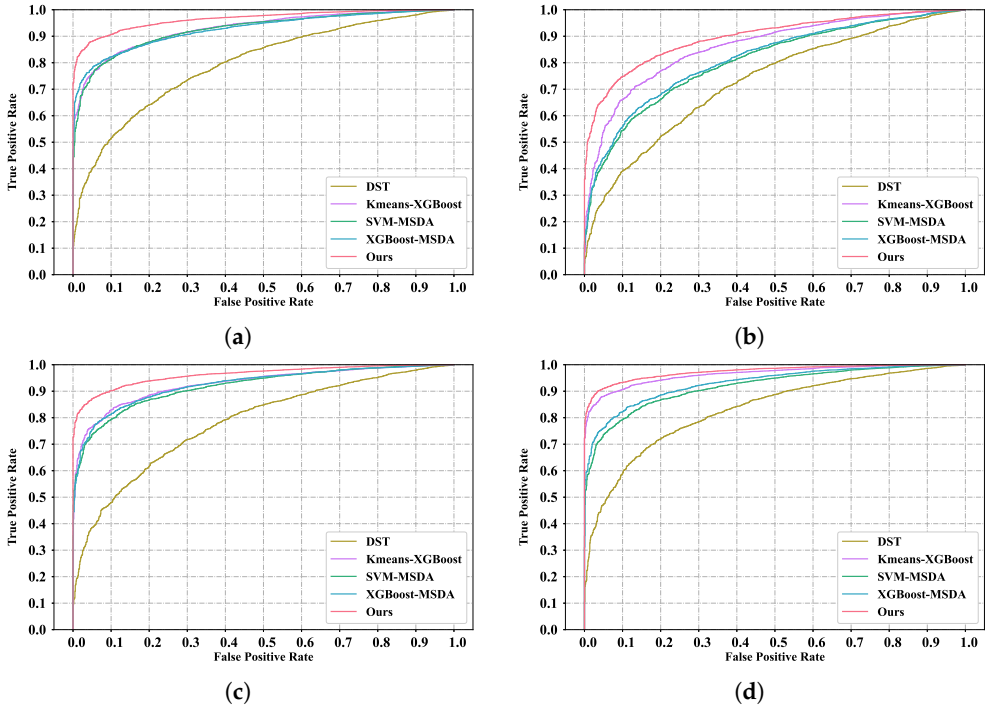


Figure 7. The ROC results of multiple steganography methods. (**a**) The ROC result on Geiser's steganography method; (**b**) the ROC result on Miao's ($\eta = 1$) steganography method; (**c**) the ROC result on Miao's ($\eta = 2$) steganography method; (**d**) the ROC result on Miao's ($\eta = 4$) steganography method.

5.7. Ablation Study

This ablation study evaluated the impact of each component within the proposed multi-scale transformer framework on detection performance. By selectively removing key elements—including the subframe-level transformer, pairwise subframe-level transformer, frame-level transformer, and regression task—the study quantifies their individual contributions to classification accuracy. This analysis underscores the significance of the hierarchical Transformer structure and multi-task learning in enhancing steganographic detection. The results are presented in Table 3.

The results highlight the contribution of each component to detection performance. Removing the subframe-level transformer reduces accuracy from 92.45% to 92.01%, indicating its role in capturing local features. Excluding the pairwise subframe-level transformer lowers accuracy to 91.86%, demonstrating its importance in modeling inter-subframe dependencies. Eliminating the frame-level transformer results in a more significant drop to 91.72%, emphasizing its role in capturing the global context. Removing the regression task decreases accuracy to 91.90%, confirming the benefit of incorporating continuous embedding rate information. The full model achieves the highest accuracy of 92.45%, demonstrating the collective effectiveness of all components.

Table 3. Detection accuracy with different model variants.

Index	Network Description	Accuracy
#1	without the subframe-level transformer	0.9201
#2	without the pairwise subframe-level transformer	0.9186
#3	without the frame-level transformer	0.9172
#4	without the regression task	0.9190
#5	the whole proposed method	0.9245

6. Conclusions

To address the challenge of detecting steganographic content in AMR-coded speech streams with unknown embedding rates, this paper has proposed a multi-scale transformer-based framework with a multi-task learning mechanism that combines classification and regression tasks. The model automatically learns discriminative representations from raw speech data, capturing subtle embedding patterns without relying on handcrafted features. Its hierarchical architecture enables effective multi-scale feature extraction across local, pairwise, and global contexts. Experimental results confirmed the robustness and generalization ability of the proposed approach across a wide range of embedding rates, thus successfully achieving the primary research objective. However, the current framework was developed under the assumption that the steganographic embedding method is known during training, which limits its applicability when encountering unknown or unseen hiding techniques. In future work, we aim to enhance the model's generalization ability by incorporating self-supervised or contrastive learning strategies to discover embedding-invariant features, thereby improving its effectiveness against unknown steganographic methods in real-world scenarios.

Author Contributions: All authors contributed to the preparation of this paper. C.S. proposed the method, conducted the literature review, designed and performed the experiments, analyzed the results, and wrote the manuscript. A.A. and N.S. supervised the research, offering guidance and suggestions for improvement. N.A.R. assisted with analyzing the experimental results and provided recommendations for revisions. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are publicly available but are available upon request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Xiang, X.; Tan, Y.; Qin, J.; Tan, Y. Advancements and challenges in coverless image steganography: A survey. *Signal Process.* **2024**, *228* , 109761.
2. Setiadi, D.R.I.M.; Ghosal, S.K.; Sahu, A.K. AI-Powered Steganography: Advances in Image, Linguistic, and 3D Mesh Data Hiding—A Survey. *J. Future Artif. Intell. Technol.* **2025**, *2*, 1–23.
3. Biswal, M.; Shao, T.; Rose, K.; Yin, P.; Mccarthy, S. StegaNeRV: Video Steganography using Implicit Neural Representation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 888–898.
4. Li, N.; Qin, J.; Xiang, X.; Tan, Y. Robust coverless video steganography based on pose estimation and object tracking. *J. Inf. Secur. Appl.* **2024**, *87*, 103912.
5. Wu, J.; Wu, Z.; Xue, Y.; Wen, J.; Peng, W. Generative text steganography with large language model. In Proceedings of the 32nd ACM International Conference on Multimedia, Melbourne, Australia, 28 October–1 November 2024; pp. 10345–10353.
6. Li, F.; Wei, P.; Fu, T.; Lin, Y.; Zhou, W. Imperceptible Text Steganography based on Group Chat. In Proceedings of the 2024 IEEE International Conference on Multimedia and Expo (ICME), Niagra Falls, ON, Canada, 15–19 July 2024; pp. 1–6.
7. Su, W.; Ni, J.; Hu, X.; Li, B. Efficient Audio Steganography Using Generalized Audio Intrinsic Energy With Micro-Amplitude Modification Suppression. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 6559–6572.
8. Zhuo, P.; Yan, D.; Ying, K.; Wang, R.; Dong, L. Audio steganography cover enhancement via reinforcement learning. *Signal Image Video Process.* **2024**, *18*, 1007–1013.
9. Kheddar, H.; Hemis, M.; Himeur, Y.; Megías, D.; Amira, A. Deep learning for steganalysis of diverse data types: A review of methods, taxonomy, challenges and future directions. *Neurocomputing* **2024**, *581*, 127528.
10. Guo, F.; Sun, S.; Weng, S.; Yu, L.; He, J. A two-stream-network based steganalysis network: TSNet. *Expert Syst. Appl.* **2024**, *255*, 124796.

11. Samukic, A. UMTS/IMT-2000 Standardisation: 3 GPP Third Generation Partnership Project: Development of Standards for the New Millennium. In *Wireless Multimedia Network Technologies*; Springer: Berlin/Heidelberg, Germany, 2000; pp. 75–93.
12. Huang, Y.; Liu, C.; Tang, S.; Bai, S. Steganography integration into a low-bit rate speech codec. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 1865–1875.
13. Yan, S.; Tang, G.; Sun, Y.; Gao, Z.; Shen, L. A triple-layer steganography scheme for low bit-rate speech streams. *Multimed. Tools Appl.* **2015**, *74*, 11763–11782.
14. Liu, P.; Li, S.; Wang, H. Steganography in vector quantization process of linear predictive coding for low-bit-rate speech codec. *Multimed. Syst.* **2017**, *23*, 485–497.
15. Liu, P.; Li, S.; Wang, H. Steganography integrated into linear predictive coding for low bit-rate speech codec. *Multimed. Tools Appl.* **2017**, *76*, 2837–2859.
16. Huang, Y.; Tao, H.; Xiao, B.; Chang, C. Steganography in low bit-rate speech streams based on quantization index modulation controlled by keys. *Sci. China Technol. Sci.* **2017**, *60*, 1585–1596.
17. Geiser, B.; Vary, P. High rate data hiding in ACELP speech codecs. In Proceedings of the 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, NV, USA, 31 March–4 April 2008; pp. 4005–4008.
18. Miao, H.; Huang, L.; Chen, Z.; Yang, W.; Al-Hawbani, A. A new scheme for covert communication via 3G encoded speech. *Comput. Electr. Eng.* **2012**, *38*, 1490–1501.
19. Miao, H.; Huang, L.; Shen, Y.; Lu, X.; Chen, Z. Steganalysis of compressed speech based on Markov and entropy. In Proceedings of the Digital-Forensics and Watermarking: 12th International Workshop, IWDW 2013, Auckland, New Zealand, 1–4 October 2013; Revised Selected Papers 12; Springer: Berlin/Heidelberg, Germany, 2014; pp. 63–76.
20. Ren, Y.; Cai, T.; Tang, M.; Wang, L. AMR steganalysis based on the probability of same pulse position. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 1801–1811.
21. Tian, H.; Wu, Y.; Chang, C.C.; Huang, Y.; Chen, Y.; Wang, T.; Cai, Y.; Liu, J. Steganalysis of adaptive multi-rate speech using statistical characteristics of pulse pairs. *Signal Process.* **2017**, *134*, 9–22.
22. Tian, H.; Liu, J.; Chang, C.C.; Chen, C.C.; Huang, Y. Steganalysis of AMR speech based on multiple classifiers combination. *IEEE Access* **2019**, *7*, 140957–140968.
23. Sun, C.; Tian, H.; Chang, C.C.; Chen, Y.; Cai, Y.; Du, Y.; Chen, Y.H.; Chen, C.C. Steganalysis of adaptive multi-rate speech based on extreme gradient boosting. *Electronics* **2020**, *9*, 522.
24. Tian, H.; Sun, J.; Huang, Y.; Wang, T.; Chen, Y.; Cai, Y. Detecting steganography of adaptive multirate speech with unknown embedding rate. *Mob. Inf. Syst.* **2017**, *2017*, 5418978.
25. Sun, C.; Tian, H.; Mazurczyk, W.; Chang, C.C.; Quan, H.; Chen, Y. Steganalysis of adaptive multi-rate speech with unknown embedding rates using clustering and ensemble learning. *Comput. Electr. Eng.* **2023**, *111*, 108909.
26. Zhang, Y.; Yang, Q. An overview of multi-task learning. *Natl. Sci. Rev.* **2018**, *5*, 30–43.
27. Zhang, Y.; Yang, Q. A survey on multi-task learning. *IEEE Trans. Knowl. Data Eng.* **2021**, *34*, 5586–5609.
28. Wang, H.; Nie, F.; Huang, H.; Risacher, S.; Ding, C.; Saykin, A.J.; Shen, L. Sparse multi-task regression and feature selection to identify brain imaging predictors for memory performance. In Proceedings of the 2011 IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 557–562.
29. Han, K.; Xiao, A.; Wu, E.; Guo, J.; Xu, C.; Wang, Y. Transformer in transformer. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 15908–15919.
30. Liu, Z.; Ning, J.; Cao, Y.; Wei, Y.; Zhang, Z.; Lin, S.; Hu, H. Video swin transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 3202–3211.
31. Xiao, B.; Huang, Y.; Tang, S. An approach to information hiding in low bit-rate speech stream. In Proceedings of the IEEE GLOBECOM 2008—2008 IEEE Global Telecommunications Conference, New Orleans, LA, USA, 30 November–4 December 2008; pp. 1–5.
32. Lai, S.; Liu, K.; He, S.; Zhao, J. How to generate a good word embedding. *IEEE Intell. Syst.* **2016**, *31*, 5–14.
33. Yin, Z.; Shen, Y. On the dimensionality of word embedding. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 895–906.
34. Lin, Z.; Huang, Y.; Wang, J. RNN-SM: Fast steganalysis of VoIP streams using recurrent neural network. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 1854–1868.