The ISC Int'l Journal of Information Security
ISeCure

January 2022, Volume 14, Number 1 (pp. 1–9)
http://www.isecure-journal.org

A Binary Relevance Adaptive Model-Selection for Ensemble Steganalysis

Mahdie Abazar [1], Peyman Masjedi [1], and Mohammad Taheri [1,*]
[1] *Computer Science and Engineering and Information Technology Department, Shiraz University, Shiraz, Iran.*

ARTICLE INFO.

Article history:
Received: December 20, 2020
Revised: May 10, 2021
Accepted: May 19, 2021
Published Online: September 6, 2021

Keywords:
Binary Relevance, Ensemble, FLD, Model-Selection, Steganalysis

Type: Research Article

doi: 10.22042/isecure.2021.262990.596

dor: --

ABSTRACT

Steganalysis is an interesting classification problem to discriminate the images, including hidden messages from the clean ones. There are many methods, including deep CNN networks, to extract fine features for this classification task. Also, some researches have been conducted to improve the final classifier. Some state-of-the-art methods use ensemble of networks by a voting strategy to achieve more stable performance. In this paper, a selection phase is proposed to filter improper networks before any voting. This filtering is done by a binary relevance multi-label classification approach. Xu-Net and ResT-Net, the most famous state-of-the-art Steganalysis ensemble models, are considered as the base networks for feature extraction. The Logistic Regression (LR) is chosen here as the last layer of the networks for classification. One large-margin Fisher's linear discriminant (FLD) classifier is trained for each one of the networks to measure its suitability in classifying the query image. The proposed method with different approaches is applied on the BOSSbase dataset and compared to traditional voting and some state-of-the-art related ensemble techniques. The results show significant accuracy improvement of the proposed method in comparison with others.

1 Introduction

Digital steganography is the technique of hiding a secret message into a media file as the cover, such as image, sound, and video. Contrarily, Steganalysis is the process of discovering if a cover has a hidden message. In other words, it reveals the presence of secret messages embedded in the covers. Nowadays, images are used on the worldwide web, storage, and communications more than other media. This is why; image steganalysis and steganography attract much interest in the field of information security.

Steganalysis is usually modeled as a binary classification model to classify an image as a clean image (cover) or an image with a secret message (stego). Feature engineering consisting of generation, selection, and extraction is an important task in Steganalysis[1]. Traditional feature-based methods, Spatial Rich Model (SRM) [2] and its selection-channel awareness version [3], provide various hand-crafted features to be used by a binary classifier. Instead of using a single classifier, Ensemble Classifiers (EC) [4] can look at the problem from different perspectives and achieve better performance.

The huge datasets with the more complicated instances raised more robust methods to analyze them. Deep neural networks (e.g., CNN) can handle huge

* Corresponding author.
Email addresses: mahdie.abazar@gmail.com, peymanmasjedi@gmail.com, motaheri@shirazu.ac.ir
ISSN: 2008-2045

data with complex instances. As mentioned in the following, CNNs were mostly used to analyze covers and detect stego images. These networks automatically extract features that can reduce complexity and dimensionality in comparison to traditional techniques [5].

The most concentrated of recent methods was improving the feature extraction with less focus on improving the classification part. Due to using the merits of features extracted from different viewpoints, some of these methods used more than one network [6–8]. As the final phase, the reasoning is done based on a majority-vote strategy on the classification results of the networks. In most of the recent models, all the networks have a similar contribution to the final prediction. A new method has recently been proposed for weighting these models to improve the classification accuracy on the training set [9]. This approach is called weighted-vote reasoning. Although this weighting could improve the performance, the weights were fixed regardless of the observed cover image. In other words, the weights assign a global degree of trust to each model and do not determine the suitability of the models for each image.

The main supportive idea of this paper is that each network may be converged to extract suitable features for specific regions of data space. Hence, each image is desired to be correctly classified by proper classifiers of the associated region. The proposed method aims to increase the detection accuracy of steganography by selecting proper classifiers for each image. In other words, some fast post-processing technique on the trained classifiers is proposed to choose the best ones for classifying the query image. Then, a simple majority-vote or a weighted-vote may be used as the reasoning method for just the selected classifiers to predict the class label. Two experimental results on two state-of-the-art ensemble Steganalysis (Xu-Net and ResT-Net) showed that the proposed method makes the model more general and improves detection accuracy.

This paper is organized as follows. Related work is presented in Section 2. Section 3 is dedicated to explaining the proposed method. Experimental results are reported in Section 4, and the paper is finally concluded in Section 5.

2 Related Work

Recently, several steganalysis methods has been proposed based on Deep Learning methods. These methods use convolutional neural networks (CNN) in particular. Compared to traditional methods that manually extract features, CNN-based Steganalysis methods automatically extract features. Tan and Li [10] introduced a stacked convolutional auto-encoder to detect any secret messages in stego. Qian *et al.* [11] proposed a CNN-based steganalysis called Qian-Net. They designed five convolutional groups; each contains a convolutional layer, Gaussian activation function and average pooling. Also, a fixed 5×5 high pass filter, called KV filter, was used as the preprocessing layer. Further, Qian *et al.* [12] proved transfer learning is useful for CNN-based Steganalysis to detect stego at low payloads.

Xu *et al.* [6] proposed an effective and powerful CNN with five convolutional layers and average pooling groups called Xu-Net. Xu-Net is a well-known CNN that is considered as the base model for recent CNN-based steganalyzers [7–9, 13]. More details of Xu-Net are explained in Section 2.1.

Xu *et al.* [7] also introduced an ensemble model based on Xu-Net to achieve better accuracy. Wu *et al.* then proposed a CNN model based on residual learning and reduced detection error rates when cover images and stego images are paired [14, 15]. Ye *et al.* [16] used Truncation Linear Unit (TLU) as an activation function in the design of CNN-based models. Ye-Net surpassed the classic SRM on resampled and cropped images with selection channel awareness and data augmentation.

Yedroudj *et al.* [13] introduced a combined network based on Xu-Net and Ye-Net, with 30 high pass filters based on SRM filters as a preprocessing layer, and ReLU activation function in the layers. They also incorporate TLU in their CNN (like Ye-Net) in the first layers and used five convolutional layers, batch normalization, and ABS layer (like Xu-Net). Li *et al.* [8] introduced CNN-based Steganalysis based on Xu-Net, which is called ResT-Net. They proposed three subnets with different preprocessing steps. The first subnet applied 6×6 Gabor filters, the second one applied Linear-SRM filters, and the third one applied non-linear SRM filters. Then, Rectified Linear Unit (ReLU), Sigmoid and Hyperbolic Tangent (TanH) activation functions were used together in some of the convolutional layers of Xu-Net to build its subnets. More details of ResT-Net are explained in Section 2.1.

Boroumand *et al.* [17] proposed a deep residual architecture to minimize heuristics and externally enforced elements usage for both spatial-domain and JPEG steganography. The key part of that work is to prevent suppression of the stego signal by expanding the front part of the detector that "computes noise residuals" in which pooling has been disabled. Although an ensemble model based on Xu-Net and Rest-Net is proposed in this paper, any other steganalysis model can also be used as the base learner. Besides, the proposed model is highly related to Binary Relevance (BR) as a specific technique of multi-label classification. This section is followed by two subsections including more related works: Base Learners (Xu-Net and ResT-Net) and multi-label classification.

2.1 Base Learners (Xu-Net and ResT-Net)

Both Xu-Net and ResT-Net proposed an ensemble steganalysis, which uses 5 and 3 subnets, respectively. Xu-Net is a typical framework of feature extraction by a deep CNN. It applies a fixed high pass filter (HPF) layer for the preprocessing step to transform the image into residual noise. Xu-Net consists of five serial convolutional groups. Each group starts with a convolutional layer to generate a feature map for each group. TanH is used in the first and second groups as a non-linear activation function, but a ReLU is used in the other groups. Batch Normalization (BN) is used to prevent falling into a poor local minimum and learn optimal scales and biases for feature maps. Also, average pooling is applied as the last layer of each group, and then the output is fed to a fully connected layer. At last, a soft-max layer computes the probability of each class as the classifier.

Xu-Net uses 10000 images with a size of 512×512 from the BOSSbase v1.01 dataset [18]. These images are divided into 5000 tests and 5000 training instances. Each training or test image is originally clean. S-UNIWARD [19] and HILL [20], which are spatial domain content-adaptive algorithms, have been applied to create stego images. Therefore, each part has 5000 pairs of clean-stego images (totally 10000 training images). With the same manner, 5000 pairs of clean-stego test images are generated. In the training phase, five Xu-Net were learned; each one got 4000 pairs of images for training, and the rest of them (1000 pairs) were used for validation. In the test (prediction) phase, the outputs of all networks are averaged to classify each test image. In this paper, the same as Xu-Net, five networks have been trained as the base learners.

As mentioned before, each subnet of ResT-Net is based on Xu-Net and contains five groups of layers. Its main contribution is using three activation functions. In the second and fourth layers of this work, ReLU, Sigmoid, and TanH activation functions are used parallelly and then concatenated results are fed to the next layer. Each subnet produces 256 features and classifies the image by using a fully connected and a soft-max layer. The training of ResT-Net contains two parts. In the first part, all subnets are trained with the same data for 1000 epochs. After training all subnets, the classification part of each subnet is ignored. In the second part, concatenated results of the three subnets are fed to another fully connected layer with 768 input neurons for classification as an ensemble steganalysis. In the last part, all subnets' parameters are fixed, and just classification layer parameters are trained for 50 epochs. In this paper, only the three subnets are used and trained as the proposed method's base learner.

2.2 Multi-Label Classification (MLC)

For a set of labels L, Single-Label classification is the task of assigning one and only one label to each instance x. If $|L| = 2$, the learning problem is known as a binary classification. Totally, in supervised learning, a model is developed to learn the task of classification from a set of training labeled instances. Many machine learning algorithms focus on solving binary classification problems. Multi-Label Classification (MLC) is the task of assigning $L(x) \subseteq L$ to each instance x. Recently, MLC has got much attention due to increasing number of interesting applications.

Wu *et al.* [21] proposed an MLC method called Binary Relevance (BR) for training a separate binary classifier for each label. In other words, the problem was transformed into $|L|$ separate binary classifiers, each one of which predicts if the instance belongs to the associated label.

As mentioned, the proposed method selects a subset of networks proper to classify each image. In other words, a subset of networks (labels) should be assigned to each image. This view is the most important contribution of this paper. Generally, model selection is considered as a multi-label classification in this paper.

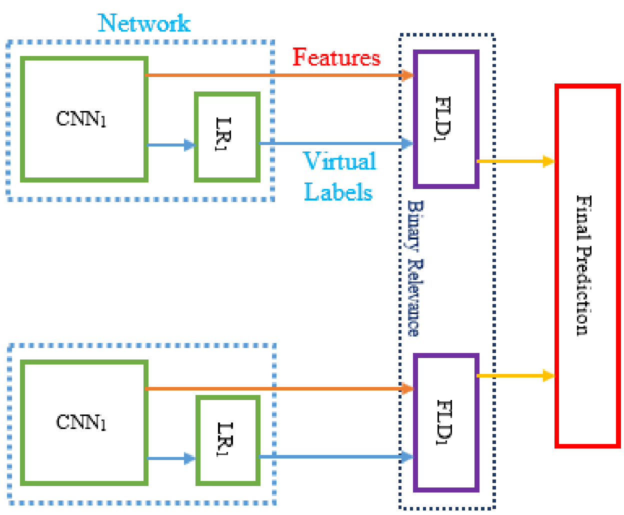

Figure 1. The proposed steganalysis scheme

3 Proposed Method

In this work, a new ensemble method is proposed to improve detection accuracy. The important part of this work is using a selection phase to filter unsuit-

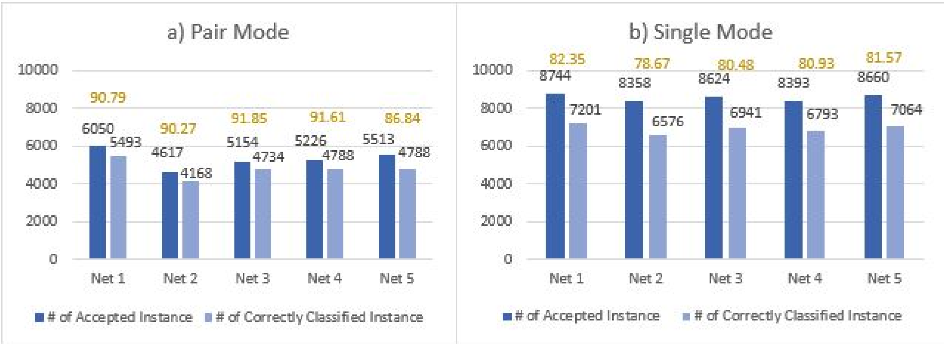

Figure 2. Number of accepted, correctly predicted instances and accuracy for each base learner (Xu-Net) in two labeling modes: a) pair mode, b) single mode

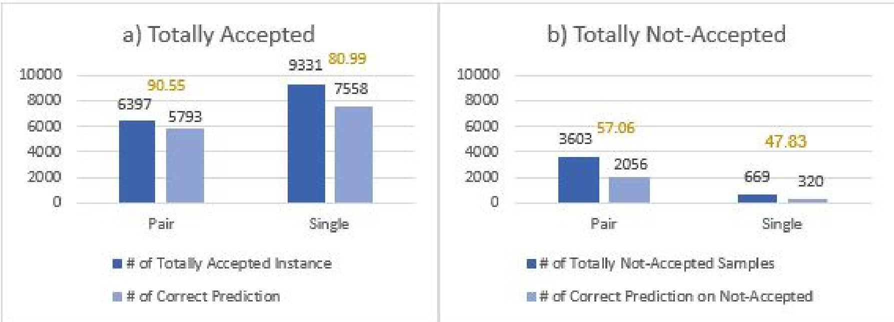

Figure 3. a) Number of totally accepted, and b) Number of not-accepted instances, and accuracy based on the labeling mode

able networks before voting; so a binary relevance multi-label classification approach is used for filtering. Xu-Net and subnets of ResT-Net are used as the base learners as two different models in the proposed method. At first, all the base learners are trained separately. The first model used five separate Xu-Nets with different initial weights, and for the second model, three subnets of ResT-Net with diverse pre-process steps are used. Then, for each subnet, a Logistic Regression (LR) is tuned as the last layer of the network to classify the query image based on the extracted features. After training all the networks with their specific conditions, a binary relevance model should be trained. The classification result of each subnet on a specific training pair determines the suitability of associated subnet on that instance. Then, a binary classifier is used to learn the suitability of each subnet on the training pairs. Fisher Linear Discriminant (FLD) classifier is used, in this paper, as the binary classifier. In the testing phase, networks are evaluated on each test image by associated FLDs. One or more proper networks are picked by the binary relevance model to vote on the class label of that image. Based on this evaluation, each network is suitable or not for each training image. In a multi-label approach, L=(N_1,N_2,...,N_5) where, N_i represents i^{th} network. For each image x, its associated labels $L(x) \subseteq$ L, determine which networks are successful to correctly classify the image. In other words, assume $y\left(x\right)$ *and* $N_i\left(x\right) \in \{clean, stego\}$ are the correct class and the predicted class of N_i for the image x, respectively. Then, the labeling is done as shown in Equation 1.

$$N_i \in L\left(x\right) \Leftrightarrow N_i\left(x\right) = y(x) \tag{1}$$

In order to make difference, $y(x)$ is called from now on steganalysis class label and $L(x)$ is called the *virtual labels*. The labeling strategy presented in Equation 1 is called Single-Mode where, each image is labeled separately. Assume, x_c and x_s are paired clean and stego versions of an image (i.e. $y\left(x_c\right) = clean$ and $y\left(x_s\right) = stego$). In another labeling strategy called Pair-Mode, a network is proper if it can correctly classify both clean and stego images of a pair as shown in Equation 2.

$$\begin{aligned} L\left(x_c\right) = L\left(x_s\right)\ \&\ N_i \in L\left(x_c\right) \Leftrightarrow \\ N_i\left(x_c\right) = clean\ \ \&\ N_i\left(x_s\right) = stego \end{aligned} \tag{2}$$

Finally, this multi-label classification is solved by the explained BR model. In other words, after labeling the training images based on Single-Mode or Pair-Mode, an FLD classifier is trained for each virtual label (network) on all the training pairs. This FLD evaluates whether this network is suitable for the given image or not. The proposed architecture is shown in Figure 1.

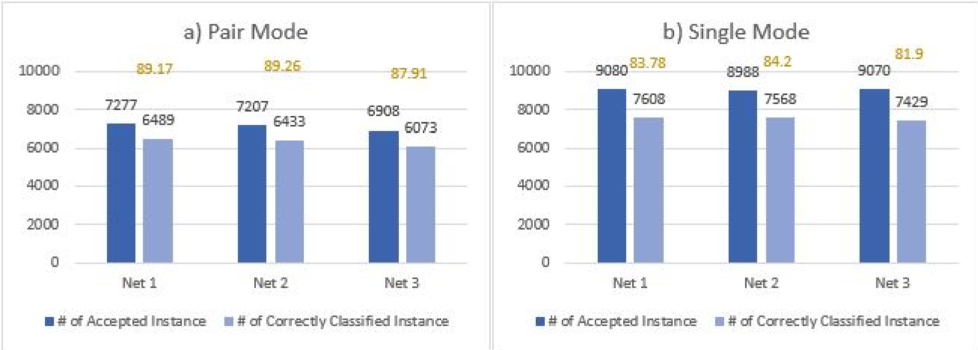

Figure 4. Number of accepted, correctly predicted instances and accuracy for each base learner (ResT-Net) in two labeling mode: a) pair mode, b) single mode

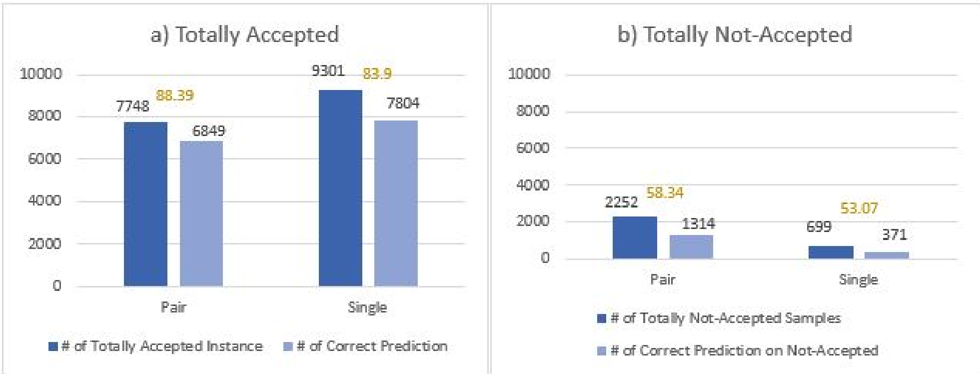

Figure 5. a) Number of totally accepted, and b) Number of not-accepted instances, and accuracy based on the labeling mode

As mentioned, each CNN-based Steganalysis consists of two essential parts: feature extraction and classification. Basically, the output of the feature extraction part of i^{th} steganalyzer (which are Xu-Net or ResT-Net subnets in this paper) is given to two classifiers: LR_i, which is the classification part of i^{th} steganalyzer, and FLD_i, which is the corresponding base classifier of BR model. The former aims to predict the steganalysis label of the input image. The latter predicts whether LR_i can correctly classify the image. Suppose the second classifier can perfectly detect the error of the first one. In this case, the correct steganalysis label can always be predicted by toggling the first output in the case of having an error.

The reader may think that, both of these classifiers take similar features. Why should FLD_i generate additional information rather than LR_i? The difference is laid on that FLD_i uses two outputs, ground truth and the output of LR_i to generate the virtual label whereas LD_i just uses the ground truth as the output. The virtual label can be considered as the accuracy (complement of error) of the first classifier. The following scenario may be helpful.

Assume, it is desired to estimate a random variable $z \sim \mathcal{N}\left(z|\mu, \delta^2\right)$. Also assume that the estimator is a simple model that always generates a constant output $\tilde{z}$. In order to minimize the squared error, the best output of the estimator is the expected value of z (i.e. $\tilde{z} = \mu$). Then, the squared error of the estimation is itself a random variable $e = (z-\mu)^2$. A second estimator tries to estimate e by a constant value $\tilde{e}$. With the same reasoning, $\tilde{e}$ should be the expected value of e equal to the variance of z as shown in Equation 3.

$$\tilde{e} = E_e\left(e\right) = E_z\left((z-\mu)^2\right) = E_z\ z^2 - \left(E_z z\right)^2 = \delta^2 \quad (3)$$

In other words, the second estimator approximates the uncertainty of the first one. It is trivial that this estimated values μ and δ^2 have independent information. For distributions with large variance, the second estimator determines a large uncertainty on the estimation of the first one, such that the user may prefer to ignore it. With the same manner, the FLD classifier can be considered as the error estimator of the LR in the proposed method.

In the prediction phase, each test image is given to each network. Result features of the network are independently given to both LR and FLD classifiers. The FLD classifiers determine the proper networks. These networks are called Accepted networks for that image. Also, that image is denoted by the Accepted instance for those networks. Most of the images have at least one Accepted network. Such images are called Totally Accepted images and others are denoted by Totally Not-accepted ones.

Based on the results, there are two strategies for reasoning: Only-Accepted and All-Nets. In the Only-Accepted strategy, just the Accepted networks of input image x are used for classification. The output of the LR classifiers associated with these networks votes to determine the final steganalysis label. This voting may be Majority voting or Weighted voting. In the former reasoning, the major class label is selected, and all Accepted networks have the same weights in the prediction. In Weighted voting, the absolute value of the output of LR classifiers is chosen as the weight of that network. Then , the label with maximum accumulated weight is assigned to the image.

In the All-Nets strategy, the Not-accepted networks also contribute to the voting but by toggling their predictions. This strategy is based on assuming that a non-selected network misclassifies the image. This assumption is incorrect if the training has been done in Pair-Mode. For example, assume a network N_i classifies both x_c and x_s as *clean*. Then, N_i is labeled as improper for x_c where it is really proper. This is why; All-Net strategy is just used after Single-Mode training.

In the case of the Only-Accepted strategy, there may be no network selected for an image (Totally Not-accepted instances). In this situation, two reasoning methods are proposed: Direct and Toggling. In the former, all networks contribute to voting. In the latter approach, all networks are also used but with toggled predictions (i.e., the same as All-Net strategy for these instances). With the same reasoning, the Toggling method is not used in Pair-Mode.

4 Experiments

The BOSSbase v1.01 dataset consisting of 10,000 gray-scale images of size 512×512 is used in this paper for evaluation. For the first experiment, five Xu-Net models are trained and used as the base learners of the proposed method. Then, three subnets of Rest-Net are trained for the second experiment. The proposed method is evaluated on clean images and associated stego versions generated by S-UNIWARD with 0.4 bits per pixel (bpp) payload. Half of the data, 5000 pairs of stego and clean images, are randomly chosen and are used for training, and the rest is used for evaluation.

For training Xu-Nets, training data is divided into 5 folds. Each fold is once left out as the validation, and the remained 4000 pairs are used to train one of the CNNs. Hyperparameters are set the same as Xu-Net. The networks are trained for 960 epochs with batch size 64, and the learning rate is set to 0.001 and is scheduled to decrease by 0.9 after every 5000 steps. For training ResT-Net subnets, 4000 pairs of images were selected as the training data and 1000 pairs used as the validation data. Subnets are trained with the same hyperparameters as the ResT-Net. All subnets are trained for 1000 epochs with batch size 40. The learning rate is set to 0.001, and the learning decay is set to 90% every 5000 training steps. All models are implemented in PyTorch. All the experiments are done using the Tesla K80 graphic card on Google Colab.

Training the proposed BR for network selection is done in both Single-Mode and Pair-Mode. Five FLDs are trained for five Xu-Nets and three other FLDs are trained for corresponding subnets of ResT-Net. All the test images are given to the networks, LRs, and corresponding FLDs classifiers with different proposed evaluation strategies. Only-Accepted and All-Nets strategy are used for reasoning. For the instances with no selected network in Only-Accepted strategy, Direct or Toggled reasonings may be used. After determining the networks and associated predictions in the voting, two voting methods, weighted-vote and majority-vote are applied.

At first, the proposed selection networks method is checked out. Figure 2 shows the details of five Xu-Nets results. It represents the number of accepted instances, the number of correctly classified instances, and base learner's accuracy separately. This report is provided for both labeling modes. In Pair-Mode, the FLD learns to accept a smaller set of instances, just the pairs that are correctly classified by the LR. As expected in the testing phase, the number of accepted instances is also less in Pair-Mode in comparison with the Single-Mode. However, the accuracy of Pair-Mode on the accepted instances is higher than Single-Mode (i.e., more Precision and less Recall). In both labeling modes, the accuracy of networks is improved compared to the base models.

Figure 3 represents more details of Xu-Net results with a general outlook. These charts present the number of Totally accepted and Totally not-accepted instances and the number of correctly predicted instances to analyze the results in two labeling modes (Pair-Mode and Single-Mode). As expected, the number of totally accepted and correctly predicted instances in Pair-Mode are less than ones in Single-Mode. Although, the accuracy of Pair-Mode, on both Totally accepted instances and Totally not-accepted ones, is higher than Single-Mode, the overall accuracy of Single-Mode is higher. In other words, Pair-Mode has a better prediction (but) in a smaller set of instances. As shown, in Pair-Mode, more than 1/3 of in-

Table 1. Result of different proposed strategy with two voting method in case of using Xu-Net as base learner

Model		Accuracy	
Labeling-Mode	Reasoning-Method	Majority-Vote	Weighted-Vote
Pair	Only-Accepted Direct	78.01	78.49
Single	Only-Accepted Direct	78.33	78.78
Single	All Nets	78.88	79.25
Single	Only-Accepted Toggling	78.56	79.07

Table 2. Result of different proposed strategy with two voting method in case of using subnets of ResT-Net as base learner

Model		Accuracy	
Labeling-Mode	Reasoning-Method	Majority-Vote	Weighted-Vote
Pair	Only-Accepted Direct	81.67	81.63
Single	Only-Accepted Direct	81.74	81.75
Single	All-Nets	81.13	81.13
Single	Only-Accepted Toggling	81.31	81.32

Table 3. Comparison of proposed method with Xu-Net [6], Abazar method [9] and ResT-Net [8]

Model	Accuracy
(Single, Only-Accepted, All-Nets) (Weighted Vote) Xu-Net	79.25
Xu-Net [6]	78.16
Weighted Ensemble(Weighted-Vote) [9]	78.04
Weighted Ensemble(Single-Winner) [9]	78.2
(Single,Only-Accepted,All-Direct)(Weighted-Vote)ResT-Net	81.75
ResT-Net[8]	81.46

stances cannot use the FLD information such that it cannot outperform Xu-Net without weighted-voting.

Details of each base learner are investigated in Figure 4, the same as Figure 2, but in this case, base learners are ResT-Net. Subnets of Rest-Net are stronger than Xu-Net, according to their accuracies individually. Also, the subnets of ResT-Net are wider and have more parameters than Xu-Net. Each subnet of ResT-Net accepts more instances than Xu-Net, and also predicts more instances truly. Figure 5, like Figure 3, shows the number of totally accepted and Totally not-accepted instances in Pair-Mode and Single-Mode for subnets of ResT-Net in the proposed method. In this figure, the number of totally accepted instances that are accepted by at least one network is more than Xu-Net. In short, according to Figures 2, 3, 4, 5, Rest-Net base learners are stronger than Xu-Net base learners and have higher accuracies.

The final experimental results of the proposed method are shown in Table 1 and Table 2. In these results, Xu-Net and Rest-Net are used as the base learners of the ensemble method. In addition, the proposed model selection methods are applied to achieve the final prediction. Table 3 compares the results of the proposed method with the base methods, as the most famous state-of-the-art Steganalysis and recently weighted voting methods presented in [9]. Xu-Net uses a simple average function for the ensemble of five networks. ResT-Net trains a fully connected layer to ensemble three subnets. In [9] method, the networks have an input-independent weight in reasoning and the result is achieved by both weighted vote and Single-Winner methods. Based on Table 1, toggling the prediction of Not-accepted networks significantly affects the classification accuracy. For the Only-Accepted strategy, toggling is used only for Totally Not-accepted instances in the 4th row. However, in the All-Nets strategy, classification of all the images is done by toggling for the unselected networks. As can be seen, these two methods have maximum accuracy. Unlike Table 1, Table 2 shows that toggling the prediction of Not-Accepted networks does not improve the classification accuracy and Only-Accepted Direct strategy in both Pair and Single-mode achieves better results. The base learners of this experiment were strong individually. It may be the main reason that, the toggling does not correct the prediction and degrades the results. As a rule, weak learners have a better improvement for ensemble than strong learners. Hence, building the ensemble method using Xu-Net as the base learner has better growth than Rest-Net respect to associated base models. Table 3 compares the results of the proposed method with other methods. The best results of Table 1 and Table 2 are collected and shown in Table 3, although some other results of these Tables were also better than the base models. As a final result, the proposed ensemble method based on the model selection, could improve both base learners Xu-Net and Rest-Net. Totally however, Rest-Net achieves a better result due to its structure and strong base learners.

5 Conclusion and Future Work

In this paper, inspired by multi-label classification, a new ensemble steganalysis has been proposed using an FLD classifier as the final layer to select the proper base learners per image. Two training modes, Pair and Single have been introduced with some reasoning strategies. Each of these approaches may use major-voting or weighted-voting. The results show that, in the case of using Xu-Net as the base learners, the ensemble model usually has better performance than the base models. In using ResT-Net subnets, most of the strategies achieve better results than the base model. In the future, these techniques can be applied to other state-of-the-art base learners. Weighted-voting with adaptive weights can be extended to reasoning strategies. Other binary classifiers or even other multi-label classification systems can also be used.

References

[1] Jian Ye, Jiangqun Ni, and Yang Yi. Deep learning hierarchical representations for image ste-

ganalysis. *IEEE Transactions on Information Forensics and Security*, 12(11):25452557, 2017.

[2] Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868882, 2012.

[3] Tomas Denemark, Vahid Sedighi, Vojtech Holub, Remi Cogranne, and Jessica Fridrich. Selection-channel-aware rich model for steganalysis of digital images. *2014 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2014.

[4] Jan Kodovsky, Jessica Fridrich, and Vojtch Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 7(2):432444, 2012.

[5] Tabares-Soto Reinel, Ramos-Pollan Raul, and Isaza Gustavo. Deep learning applied to steganalysis of digital images: A systematic review. *IEEE Access*, 7:6897068990, 2019.

[6] Guanshuo Xu, Han-Zhou Wu, and Yun-Qing Shi. Structural design of convolutional neural networks for steganalysis. *IEEE Signal Processing Letters*, 23(5):708712, 2016.

[7] Guanshuo Xu, Han-Zhou Wu, and Yun Q.. Shi. Ensemble of cnns for steganalysis. *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security*, 2016.

[8] Bin Li, Weihang Wei, Anselmo Ferreira, and Shunquan Tan. Rest-net: Diverse activation modules and parallel subnets-based cnn for spatial image steganalysis. *IEEE Signal Processing Letters*, 25(5):650654, 2018.

[9] Tayebe Abazar, Peyman Masjedi, and Mohammad Taheri. An efficient ensemble of convolutional deep steganalysis based on clustering. *2020 6th International Conference on Web Research (ICWR)*, 2020.

[10] Shunquan Tan and Bin Li. Stacked convolutional auto-encoders for steganalysis of digital images. *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, 2014.

[11] Yinlong Qian, Jing Dong, Wei Wang, and Tieniu Tan. Deep learning for steganalysis via convolutional neural networks. *Media Watermarking, Security, and Forensics 2015*, 2015.

[12] Yinlong Qian, Jing Dong, Wei Wang, and Tieniu Tan. Learning and transferring representations for image steganalysis using convolutional neural network. *2016 IEEE International Conference on Image Processing (ICIP)*, 2016.

[13] Mehdi Yedroudj, Frederic Comby, and Marc Chaumont. Yedroudj-net: An efficient cnn for spatial steganalysis. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[14] Songtao Wu, Shenghua Zhong, and Yan Liu. Deep residual learning for image steganalysis. *Multimedia Tools and Applications*, 77(9):1043710453, 2017.

[15] Songtao Wu, Sheng-Hua Zhong, and Yan Liu. Residual convolution network based steganalysis with adaptive content suppression. *2017 IEEE International Conference on Multimedia and Expo (ICME)*, 2017.

[16] Jian Ye, Jiangqun Ni, and Yang Yi. Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security*, 12(11):25452557, 2017.

[17] Mehdi Boroumand, Mo Chen, and Jessica Fridrich. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5):11811193, 2019.

[18] Patrick Bas, Tom Filler, and Tom Pevn. break our steganographic system: The ins and outs of organizing boss. *Information Hiding Lecture Notes in Computer Science*, page 5970, 2011.

[19] Vojtch Holub, Jessica Fridrich, and Tom Denemark. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014(1), 2014.

[20] Bin Li, Ming Wang, Jiwu Huang, and Xiaolong Li. A new cost function for spatial image steganography. *2014 IEEE International Conference on Image Processing (ICIP)*, 2014.

[21] Jian Wu, Shiqian Zhao, Victor S. Sheng, Jing Zhang, Chen Ye, Pengpeng Zhao, and Zhiming Cui. Weak-labeled active learning with conditional label dependence for multilabel image classification. *IEEE Transactions on Multimedia*, 19(6):11561169, 2017.

Mahdie Abazar received the B.S degree in Computer Engineering and Information Technology from the Persian Gulf University, Bushehr, Iran, in 2016 and the M.S degree in Computer Engineering,Secure Computing from Shiraz University, Shiraz, Iran, in 2020. Her research interests include Steganalysis, Steganography, Information Security, Machine Learning, Image Processing, and Deep Learning.

Peyman Masjedi received His B.S. and M.S degrees in Computer Engineering and Information Technology, Secure Computing from Shiraz University, Shiraz, Iran, in 2016 and 2020, respectively. He is currently pursuing the Ph.D degree in Artificial Intelligence at Shiraz University, Shiraz, Iran. His research interests include Steganography, Information Security, Machine Learning and Optimization.

Mohammad Taheri was born in 1983. He achieved BS., MS. and Ph.D. degrees as an outstanding student in Computer Science Dept. of Shiraz University (Iran) since 2001 until 2013. He started his job, as the faculty member (2014), with researches in machine learning, fuzzy systems, large margin classifiers, optimization, modeling and information hiding.