

# VBSF: A Visual-Based Spam Filtering Technique for Obfuscated Emails

Ali Hossary<sup>a</sup> and Stefano Tomasin<sup>b</sup>

Dept. of Information Engineering (DEI), University of Padova, Italy  
{ali.hossary, stefano.tomasin}@unipd.it

**Keywords:** Spam Email Detection, Hidden Text Salting, Obfuscated Words, Email Rendering, Machine Learning, Optical Character Recognition (OCR) Convolutional Neural Networks (CNN) and Ensemble Learning.

**Abstract:** Recent spam email techniques exploit visual effects in text messages, such as poisoning text, obfuscating words, and hidden text salting techniques. These effects were able to evade spam detection techniques based on the text. In this paper, we overcome this limitation by introducing a novel visual-based spam detection architecture, denoted as visual based spam filter (VBSF). The multi-step process mimics the human eye's natural way of processing visual information, automatically rendering incoming emails and capturing their content as it appears on a user screen. Then, two different processing pipelines are applied in parallel. The first pipeline pertains to the perceived textual content, as it includes optical character recognition (OCR) to extract rendered textual content, followed by naïve Bayes (NB) and decision tree (DT) content classifiers. The second pipeline focuses on the appearance of the email, as it analyzes and classifies the images of rendered emails through a specific convolutional neural network. Lastly, a meta classifier integrates text- and image-based classifier outputs exploiting the stacking ensemble learning method. The performance of the proposed VBSF is assessed, showing that It achieves an accuracy of more than 98%, which is higher than the compared existing techniques on the designed dataset.

## 1 INTRODUCTION


Emails have witnessed an overwhelming global volume exceeding 200 billion messages daily. However, a staggering 80-90% of this flow comprises spam, which both annoys users and fuels malicious activities like phishing, fraud, and malware dissemination. Conventional anti-spam methods like blacklists and heuristics struggle against this onslaught, prompting the development of scalable and adaptive techniques. Machine learning is the leading approach, with state-of-the-art classifiers achieving over 90% accuracy. The prominence of machine learning in spam defense is driven by factors such as the massive scale of global spam, evolving spam tactics, and ongoing algorithmic advancements tailored for text analysis. Moreover, the exponential growth in computational resources enhances the effectiveness of spam filtering models.


Among advanced spam techniques, visual effects on text messages are particularly challenging for spam detectors, since they defeat text-based detection systems. Such approaches go under the names poisoning text, obfuscated words, and *hidden salt-*

*ing* (Bergholz et al., 2008). Recently, (Sokolov et al., 2020) has shown that spam detection techniques can be evaded by replacing some characters with others that look very similar but come from a different alphabet. For a review of hidden salting tricks see (Jáñez-Martino et al., 2023).

This paper proposes new techniques to detect spam messages, including hidden salting and other visual attack strategies. The introduced solution is a VBSF and it emulates the human visual perception of emails, thus aiming at reading the text as perceived by the human reader. Therefore, visual tricks such as faded colors and small text used to hide part of the text and letting the reader see hidden spam content will also affect our detection technique that will be able to *see* the spam content and predict it exploiting multiple diverse classifiers.

The findings underscore the importance of model composition and the value of incorporating diverse classifiers to achieve superior results. Our enhanced VBSF represents a promising advancement in predictive modeling, offering a pathway for further refinement and optimization of our approach. The resulting decision-making mechanism is robust and provides a final classification accuracy of the meta-classifier sur-

<sup>a</sup>  <https://orcid.org/0009-0002-9227-2662>

<sup>b</sup>  <https://orcid.org/0000-0003-3253-6793>

passing the accuracy of all the base models, exceeding 98%.

The rest of this paper is organized as follows. In Section II we review the existing literature with a focus on works related to our proposed VBSF solution. Section III presents the VBSF technique in detail. We then design its implementation in Section III. The performance results of the proposed solution and existing approaches are presented and discussed in Section V. Lastly, Section VI provides the main conclusions of this work.

## 2 RELATED WORK

Several works have addressed the problem of detecting spam when hidden text salting is used. In many cases, machine learning techniques are employed, as they are well known to be effective in several security-related applications (Shaukat et al., 2020).

In (Moens et al., 2010), the rendering process is tapped into. The rendering commands are analyzed to identify sections of the source text (plaintext) that will be invisible to human readers, based on criteria such as text character and background colors, font size, and overlapping characters. Furthermore, the visible text (cover text) is reconstructed from rendering commands, and the character reading order is identified, which may differ from the rendering order. In our study, we also render emails as images to analyze how they are perceived by the human eye. Instead of exploiting rendering commands, we render the whole content and then operate on the rendered image, thus being more flexible and independent of the specific rendering process. Moreover, we exploit powerful tools such as OCR and image analysis by neural networks that have proven to be effective in detection processes.

In (Nam et al., 2022) it has been proposed to use three sub-models to extract three features from images. In particular, two sub-models for text processing extract topic-based features (to identify the main subject of the message) and word-embedding-based features (to capture the meaning and relationships between words in the message) using the text contained in the images extracted by OCR. Then, a convolution-based sub-model extracts convolution-based features from images. Lastly, text and image features extracted from each sub-model are input into the classifier model that decides on the spam nature of the email. Thus, (Nam et al., 2022) proposes a technique for classifying spam images using image and text features extracted from images, which is related to our approach. We use instead three classifiers, each with

its feature extraction method, followed by a stacking meta classifier to consolidate the predictions of sub-models, additionally. Moreover, the focus of (Nam et al., 2022) was on spam images included in emails, while we generated the images from the incoming emails.

In (Biggio et al., 2007) the focus is on detecting spam techniques that hide the real content of the image. The proposed approach aims at identifying a specific characteristic of spam images with embedded text - the presence of content obscuring techniques. The underlying rationale is that images containing embedded text, which are deliberately obscured to render OCR ineffective, are likely to be spam.

In (Naiemi et al., 2019), a method based on the histogram of oriented gradients, HOG, and a support vector machine (SVM) has been used for OCR in images contained in emails. One of the limitations faced by the HOG feature extraction method is its lack of resistance against character variations on scales and translations. The proposed enhanced HOG feature extraction method has been used so that the OCR system of spam has been enhanced by using the HOG feature extraction method in such a way to be both resistant against the character variations on scale and translation and to be computationally cost-effective. Our work focuses on text emails with hidden salting tricks, rather than on emails containing spam images.

Other approaches for spam detection using machine learning approaches include a bio-inspired technique (Gibson et al., 2020), such as particle swarm optimizations and genetic algorithms which are used to optimize the performance of classifiers: it turned out that multinomial NB with the genetic algorithm outperforms the other. Still, no visual tricks were considered in (Gibson et al., 2020).

In (Karim et al., 2021), an unsupervised framework for spam detection is proposed, that resorts to a clustering approach including multiple algorithms. A suitable feature reduction is applied to obtain seven features that represent impactful analytical email characteristics from a multiangular point of view. However, this solution primarily uses the email content (body) and the subject header and does not properly deal with visual tricks.

In our work, we also apply a convolutional neural network (CNN) directly to an image rendering an email. Such an approach has been adopted in other contexts (not related to spam detection). For example, in (Rizky et al., 2023) a CNN is used to recognize text in images. Several modifications of the images have been investigated and the best model turned out to be the VGG-16 architecture along with specific image transformations. The model architecture used in

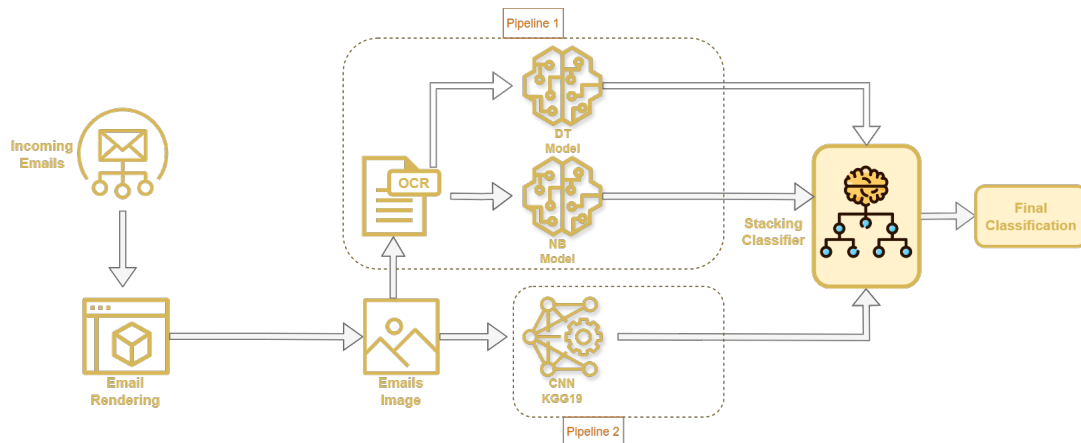


Figure 1: The proposed VBSF solution.

this study could be a valuable resource for developing future text detection systems.

### 3 VISUAL-BASED FILTERING TECHNIQUE

Spammers often use hyper text markup language (HTML) techniques such as hidden or invisible text, HTML comments, or misleading formatting to trick traditional text-based classifiers. At the same time, to convey the spam message, such tricks should provide a final image of the email that is clearly readable to the human reader. The basic idea of VBSF is to first render the email as an image, then perform spam detection on the image: this enables the spam detector to operate on the same input as provided to the human reader. This approach enhances the classifier's ability to detect spam accurately, as it considers the visual presentation of the email, uncovering potential malicious elements that might be hidden in the HTML code.

In detail, first, the email is rendered as an image: this includes interpreting HTML commands (on fonts, colors, and page layout), adding attached images, etc. Then, two spam detection techniques are applied in parallel on the obtained image, each denoted as a *pipeline*. The first pipeline is based on the *perceived content*, obtained through an OCR: the extracted text content is then fed to a content-based classification system based on NB and DT classifiers. The latter pipeline is based on the *visual appearance* and classifies images of the email content by using aCNN.

Lastly, a meta classifier combines the outputs of both text- and image-based classifiers by a stacking ensemble learning method. Through experiments, we observed a remarkable increase in testing accuracy.

Integrating the DT classifier proved to be particularly impactful, contributing to a significant enhancement in predictive performance.

Fig. 1 shows the workflow of the proposed VBSF, which is composed of the following elements:

- Rendering of the email as an image
- First pipeline: text extraction using OCR, followed by content-based filters
- Second pipeline: image-based classifier (by a CNN model), applied to the email image
- Meta classifier, utilizing stacking ensemble method, as an ensemble learning technique

Each element of the VBSF solution is described in detail below.

#### 3.1 Email Rendering

The email rendering step generates the image of the email, as it would be shown to the end human reader. This includes the formatting of text and page according to the HTML format, the inclusion of images, etc. Hidden salting tricks are also exploiting such formatting parts so that once the email is shown to the reader, it shows content (such as sentences or images) that are hard to identify in the original HTML document (Bergholz et al., 2008). The rendering can be easily obtained with one of the several tools that allow the conversion of an HTML file into an image (as it was rendered in a browser or email reader). The obtained image is then processed in the forthcoming steps.

### 3.2 First Pipeline: OCR Engine and Textual-Based Classification

The first pipeline aims at detecting spam by first converting the image of the email with OCR and then applying spam detectors on the obtained text. Noting that this step is not the inverse of email rendering, as the text captured by the OCR is very close to what the human eyes perceive, and can be very different from the textual content of the HTML files, due to the hidden salting tricks.

#### 3.2.1 NB and DT Classifiers for Spam Filtering

Once the image has been converted into the *perceived text*, we apply text-based spam classifiers to detect the presence of spam. In particular, we consider the NB classifier as one of the most renowned and effective content-based spam filters. We also use a DT classifier which operates in a different manner and is also very effective.

### 3.3 Second Pipeline: Image-Based Classification with CNN

The second pipeline aims to detect spam directly from the *appearance* of the email as rendered in the image, including colors and other visual objects. Here we use a specific computer vision CNN model as a very effective tool for image classification in similar contexts. The CNN, serves as a visual perception model, learns patterns and features crucial for distinguishing between the images of spam and ham emails.

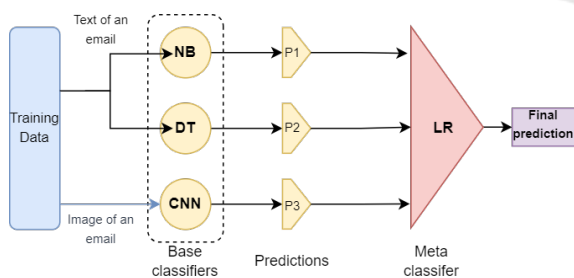


Figure 2: Stacking classifier architecture.

### 3.4 Final Classification

Since we have two pipelines aiming at providing a classification of an email, and to consolidate the classification predictions from both pipelines, which are predictions of baseline models, we incorporated a meta-classifier (Wolpert, 1992) utilizing the stacking ensemble method, as depicted in Fig. 2. The stacking classifier architecture elements are as follows:

- Baseline classification models: the NB, DT and CNN models.
- Predictions of the base models: the training predictions of the models in binary form, also called stacking features, which are fed as input training data for the meta classifier
- Meta classifier: a logistic regression (LR) classifier is chosen, trained on the training predictions of the base classifiers, in its output the final binary prediction of the whole architecture.

The LR classifier is fed with the predictions generated by the CNN classifier, DTclassifier, and the NB classifier, the use of a diverse set of base classifiers, leveraged their diverse nature to improve final predictive accuracy.

The utilization of Logistic Regression classifier as the stacking classifier further refines the integration process, providing a well-balanced synthesis of the predictions from baseline models, ensuring a more comprehensive and nuanced analysis of the input data, leading to a more reliable and informed final classification outcome. Interestingly, we experimented with various models as potential stacking meta-classifiers, such as SVM, DT, and random forests. However, after rigorous evaluation, LR emerged as the best fit.

## 4 VBSF ENVIRONMENT SETUP

In this section, we describe our dataset and the Environment of the VBSF technique including fine tuning the Neural Network model process and its setup used for performance evaluation in the next section.

### 4.1 Dataset Collection and Preparation

A mix of publicly available datasets has been used together with a dataset of hand-crafted emails for specific testing purposes. In particular, we considered the Enron 1 and Enron 4 and pre-processed Spam Assassin email corpus. The combination of parts of the three datasets was the best fit for our proposed model and led to better generalization: indeed, Enron 1 and 4 have enough textual features, while they lack colors and visual features, while Spam Assassin is rich in colors and visual features but alone was not big enough.

The combined dataset had imbalanced class distributions (40% spam and 60% ham emails), so we increased the number of spam emails to balance the dataset and prevent overfitting toward the majority class. As a result, we have 4009 benign emails (ham)

Table 1: Accuracy of VBSF-Pipeline 1 (after applying OCR) vs existing normal Text-Based Detection (without OCR).

ML Model	Accuracy	
	Text-based spam detector	VBSF - Pipeline 1
<b>NB</b>	94 %	96 %
<b>DT</b>	95 %	97 %
<b>LR</b>	96 %	97 %
<b>SVM</b>	80 %	96 %
<b>AdaBoost</b>	96 %	96 %
<b>KNN</b>	89 %	91 %

Table 2: VBSF Accuracy With Different Meta Classifiers.

Meta Classifier Model used for VBSF	VBSF final test accuracy	False Positive Rate	False Negative Rate
LR	98.3%	1.2%	0.5%
Random Forest	97.3%	1.7%	1.0%
DT	96.6%	2.3%	1.1%

and 3800 spam emails. Additionally, a few samples have been minimally modified by applying some spam tricks, trying to emulate an adversary behavior, such as spam word spacing using HTML comments, ham and spam word injection, and modification of the size and bold effects. Most of these samples succeeded in misleading existing classifiers.

## 4.2 VBSF Setup

For the OCR of the obtained image, we resort to Google Tesseract (Smith, 2007) (PyP, ).

For the second pipeline that classifies the image into the two spam and ham classes, we resort to the CNN VGG-19 neural network that utilizes small  $3 \times 3$  filters across all convolutional layers, resulting in optimal performance reflected in its low error rate (Zheng et al., 2018). We fine-tuned the VGG-19 model through various settings to attain optimal predictive performance. Several key hyperparameters were precisely adjusted, including the learning rate, the number of training and fine-tuning epochs. Additionally, we incorporated data augmentation techniques to further enhance the model's ability to generalize patterns from our dataset. Choosing an appropriate learning rate was a critical step in achieving the best results. Fig. 3 shows the classification accuracy heatmap for the VGG19 model. We note that through a systematic exploration of various learning rates and epoch numbers, we identified the spot that led to a remarkable accuracy and low validation loss, without overfitting or underfitting the data.

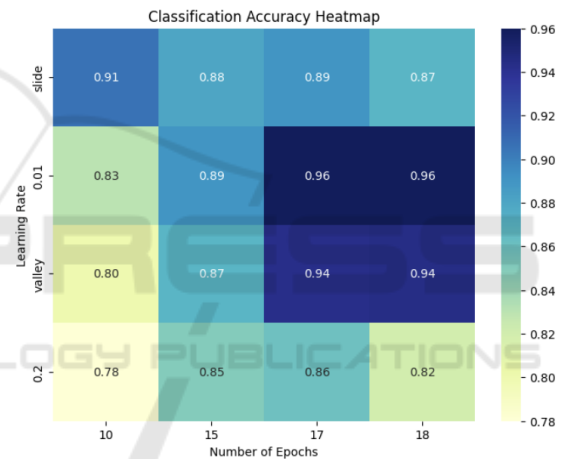


Figure 3: Classification accuracy heatmap for the VGG19 model.

## 5 NUMERICAL RESULTS

We first assess the accuracy of the first pipeline *without the DT branch* and with different classifiers on the OCR output. In particular, we consider the NB, DT, LR, SVM, AdaBoost (Freund and Schapire, 1997), and k-nearest neighbor (KNN) classifiers. For comparison purposes, we also apply the same classifiers directly to the original HTML text file, thus without passing through the visual representation and OCR.

Table 1 shows the accuracy of both the VBSF and text-based detectors for the various classifiers. Interestingly, when applied to the raw emails (source etext) in our new dataset, the performance of conventional NB and DT classifiers did not match those on well-known email datasets. However, when we

used the NB and DT classifiers on emails that went through the OCR after rendering and capturing, both classifiers demonstrated a remarkable performance improvement. The adaptation of OCR technology appeared to enhance the classifier's ability to discern spam characteristics within the text, showcasing the versatility of the NB and DT classifiers in the context of our VBSF's first pipeline. This nuanced observation underscores the importance of tailoring spam filters to the unique characteristics of the dataset at hand, optimizing their performance for diverse sources and formats of email content.

Now, we assess the performance of the VBSF solution. Table 2 shows the test accuracy of the meta classifier after augmenting the first pipeline of the VBSF. Several meta-classifiers underwent testing, again, among which LR produced superior performance compared to others reaching more than 98% accuracy, hence it was selected as the preferred choice.

Through experimentation and evaluation, we observed a remarkable increase in testing accuracy. The integration of the DT classifier proved to be particularly impactful, contributing to a significant enhancement in predictive performance. These findings underscore the importance of model composition and the value of incorporating diverse classifiers to achieve superior results. Our enhanced variant of VBSF represents a promising advancement in predictive modeling, offering a pathway for further refinement and optimization of our approach.

## 6 CONCLUSIONS

We have proposed a new approach to detect emails that use visual tricks (or hidden salting tricks) and HTML-related tricks, to convey spam messages to end users. By employing a multi-step process imitating the natural processing of visual information by the human eye, alongside text extraction of email snapshots using OCR followed by textual content classification using an NB classifier, augmented by a DT classifier, our system efficiently cleans and analyzes email text content. Moreover, integrating a CNN as a visual perception classification model enhances the system's ability to discern between spam and legitimate emails based on visual features and cues.

A remarkable strength of our proposed solution lies in its adaptability to the dynamic nature of spamming techniques, especially the visual ones. The proposed model includes parsing all HTML tags and formatting the content according to their specifications. Whether it's normal content, known spam content

hiding tricks, or crafty spam tactics, all elements are visually visible and ready for further investigation. By integrating text-based and image-based classifiers in a meta-classifier using stacking ensemble learning, our system achieves a very good final classification accuracy exceeding 98%. This holistic approach enhances both the accuracy and the resilience against evolving spam tactics.

## REFERENCES

- Bergholz, A., Paass, G., Reichartz, F., Strobel, S., Moens, M.-F., and Witten, B. (2008). Detecting known and new salting tricks in unwanted emails. In *Proc. Fifth Conference on Email and Anti-Spam (CEAS)*, volume 9.
- Biggio, B., Fumera, G., Pillai, I., and Roli, F. (2007). Image spam filtering using visual information. In *Proc. Int. Conf. on Image Analysis and Processing (ICIAP 2007)*, pages 105–110.
- Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139.
- Gibson, S., Issac, B., Zhang, L., and Jacob, S. M. (2020). Detecting spam email with machine learning optimized with bio-inspired metaheuristic algorithms. *IEEE Access*, 8:187914–187932.
- Jáñez-Martino, F., Alaiz-Rodríguez, R., González-Castro, V., Fidalgo, E., and Alegre, E. (2023). A review of spam email detection: analysis of spammer strategies and the dataset shift problem. *Artificial Intelligence Review*, 56(2):1145–1173.
- Karim, A., Azam, S., Shanmugam, B., and Kannoorpatti, K. (2021). An unsupervised approach for content-based clustering of emails into spam and ham through multi-angular feature formulation. *IEEE Access*, 9:135186–135209.
- Moens, M.-F., De Beer, J., Boiy, E., and Gomez, J. C. (2010). Identifying and resolving hidden text salting. *IEEE Trans. on Info. Forensics and Security*, 5(4):837–847.
- Naiemi, F., Ghods, V., and Khalesi, H. (2019). An efficient character recognition method using enhanced HOG for spam image detection. *Soft Computing*, 23(22):11759–11774.
- Nam, S.-G., Jang, Y., Lee, D.-G., and Seo, Y.-S. (2022). Hybrid features by combining visual and text information to improve spam filtering performance. *Electronics*, 11(13).
- Rizky, A. F., Yudistira, N., and Santoso, E. (2023). Text recognition on images using pre-trained CNN. *arXiv*.
- Shaukat, K., Luo, S., Varadharajan, V., Hameed, I. A., and Xu, M. (2020). A survey on machine learning techniques for cyber security in the last decade. *IEEE Access*, 8:222310–222354.

- Smith, R. (2007). An overview of the Tesseract OCR engine. In *Proc. Int. Conf. on Document Analysis and Recognition (ICDAR 2007)*, volume 2, pages 629–633.
- Sokolov, M., Olufowobi, K., and Herndon, N. (2020). Visual spoofing in content-based spam detection. In *Proc. Int. Conf. on Security of Information and Networks*, page 3–3.
- Wolpert, D. H. (1992). Stacked generalization. *Neural networks*, 5(2):241–259.
- Zheng, Y., Yang, C., and Merkulov, A. (2018). Breast cancer screening using convolutional neural network and follow-up digital mammography. In *Proc. Computational Imaging III, SPIE Commercial and Scientific Sensing and Imaging*.

