A survey and experimental evaluation of image spam filtering techniques

Battista Biggio, Giorgio Fumera, Ignazio Pillai and Fabio Roli

Department of Electrical and Electronic Engineering, University of Cagliari
Piazza d'Armi, Cagliari, 09123, Italy

Abstract

In their arms race against developers of spam filters, spammers have recently introduced the image spam trick to make the analysis of emails' body text ineffective. It consists in embedding the spam message into an attached image, which is often randomly modified to evade signature-based detection, and obfuscated to prevent text recognition by OCR tools. Detecting image spam turns out to be an interesting instance of the problem of content-based filtering of multimedia data in adversarial environments, which is gaining increasing relevance in several applications and media. In this paper we give a comprehensive survey and categorisation of computer vision and pattern recognition techniques proposed so far against image spam, and make an experimental analysis and comparison of some of them on real, publicly available data sets.

Keywords: spam filtering, image spam, document categorisation

1. Introduction

Nowadays, content-based filtering of multimedia documents is a problem of great relevance, given the already huge and yet increasing amount of multimedia contents available on the Internet, and the availability of different kinds of mobile devices capable of producing and accessing such contents. Among others, this raises security issues, like the possibility of perpetrating frauds (i.e., *phishing* emails), and the access to illegal or unsuitable contents by minors. Computer vision and pattern recognition tools are gaining a relevant role in this application field.

A particular instance of this problem, which we focus on in this work, is the *image spam* (short for *image-based spam*) phenomenon. Until a few years ago the content of spam emails was only of textual kind. Accordingly, spam filters analysed only the emails' body text (besides header information) to discriminate between spam and legitimate emails. Text categorisation techniques based on machine learning were also used to this aim (Sahami et al., 1998; Drucker et al., 1999; Graham, 2002a; Robinson, 2003; Meyer and Whateley, 2004). In their arms race against the developers of spam filters, spammers introduced in 2006 the image spam trick, which consists in removing the spam message from the email's body, and embedding it into an image which is sent as attachment. This allows to circumvent the analysis of emails' body text. To detect image spam, computer vision and pattern recognition techniques are also required, and indeed several techniques have been recently proposed. However, the solutions proposed so far exhibit several weaknesses, and their effectiveness has not been thoroughly investigated so far. Moreover, detecting spam emails, and image spam in particular, is complicated by the *adversarial* nature of this task, namely, by the fact that spammers can adaptively manipulate their emails to get them past spam filters, exploiting their knowledge on spam filters themselves.

In this work we make a survey on the computer vision and pattern recognition techniques proposed so far against image spam. We propose a categorisation of these techniques and discuss their potential advantages and disadvantages, including their vulnerability to adversarial data manipulation. We also carry out an experimental evaluation and comparison of some of these techniques, which is lacking in the literature. In particular, our experiments are carried out on large and publicly available data sets of images which were attached to real legitimate and spam emails.

We believe that the scope of our results goes beyond the spam filtering task, and that they could be useful to the more general problem of content-based filtering of multimedia documents. This includes issues like new forms of spam which could soon be spread through new media (for instance, audio-visual spam in systems like Youtube, envisaged by Mehta et al., 2008), as well as issues related to adversarial environments which are common to other content-based filtering tasks, like the detection of adult content in web sites.

The paper is structured as follows. We first give an overview of the spam filtering task and image spam in Sect. 2. Methods for image spam filtering proposed in the literature are reviewed in Sect. 3. The experimental analysis is reported in Sect. 4, and the vulnerabilities of image spam filtering techniques are finally discussed in Sect. 5.

Email address:
{battista.biggio,fumera,pillai,roli}@diee.unica.it (Battista Biggio, Giorgio Fumera, Ignazio Pillai and Fabio Roli)

URL: http://prag.diee.unica.it (Battista Biggio, Giorgio Fumera, Ignazio Pillai and Fabio Roli)

2. Overview of spam filtering and image spam

Spam filters have become the main technological countermeasure against spam. Although they do not affect the root cause of the spam phenomenon, they can at least alleviate some of its consequences, by helping end-users to keep their mailboxes clean (Graham, 2002b). Spam filters are software tools which run on Internet Service Providers (ISPs), corporate email servers, or users' email clients. They are currently made up of several modules which analyse different characteristics of input emails, like sender and recipient addresses, textual content, header format and attachments. The outputs of each module are combined to decide whether labelling an email as spam or legitimate.

During the past fifteen years automatic text categorisation techniques based on machine learning have been widely investigated by several authors, to implement textual content analysis (Sahami et al., 1998; Drucker et al., 1999; Androutsopoulos et al., 2000; Graham, 2002a; Robinson, 2003; Meyer and Whateley, 2004; Koprinska et al., 2007; Lai et al., 2009). These techniques are currently used by most (if not all) spam filters. Examples of popular open-source spam filters which include them are *SpamAssassin*, *SpamBayes*, *Bogofilter*, and *CRM114*. An alternative approach was proposed in Wu and Tsai (2009), based on a set of features which characterise the spammers' behaviour (e.g., forged header identification, suspicious time of delivery), instead of using a text categorisation approach. Similar features are already used by several spam filters, like *SpamAssassin*[1].

The effectiveness of spam filters is undermined by the adversarial environment in which they operate, which gave rise to an arms race between their designers and spammers(Cranor and LaMacchia, 1998; Fawcett, 2003; Weinstein, 2003; Somayaji, 2004). Well known tricks used by spammers to get their emails misclassified as legitimate consist in using fake sender addresses to evade blacklist detection, and misspelling typical "spammy" words to evade keyword detection. Some tricks are specifically targeted against machine learning techniques, like adding to the email's body some text which includes words likely to appear in legitimate emails.[2] The consequence is that spam filters must be constantly updated to keep a fair performance. In particular, this implies a frequent re-training of modules based on machine learning techniques.

Image spam is one of the most recent tricks introduced by spammers. It consists in embedding the spam message into images which are sent as email attachments. Its goal is to circumvent the analysis of the emails' textual content performed by spam filters, including automatic text classifiers. Since attached images are displayed by default by most email clients, the message is directly conveyed to the user as soon as the email is opened. The simplest kind of image spam can be viewed as a screen shot of a plain text written using a standard text editor (see Fig. 1, top). Often spam images are constructed by introducing random changes to a given template image, to make signature-based detection techniques ineffective, and are obfuscated to prevent optical character recognition (OCR) tools from reading the embedded text (see the examples in Fig. 1, middle and bottom). Ironically, some text obfuscation techniques used against OCR tools are very similar to the ones exploited to design CAPTCHAs (Completely Automated Public Turing test to tell Computers and Humans Apart),[3] which have been introduced to protect web-sites from spammers' or hackers' bots (an example is shown in Fig. 1, middle right).

Image spam was largely used in 2006, when it was estimated to be around 30% of the spam volume. Its use has subsequently decreased (its percentage has been estimated to be around 3% in 2008, according to IBM, 2008), mainly in favour of the so-called *URL-based* spam, which consists in a very simple body containing often just a link (URL) to an external web page or image. However the decrease of image spam was not due to a decrease in its effectiveness, since no proper countermeasures had been developed yet, but to efficiency reasons. Indeed, URL-based spam allows spammers to keep the email size lower, and, thus, to send emails at a higher rate. Given the continuous increase in network bandwidth, image spam could soon arise again, as very recent statistics seem to show.[4] Moreover, the problem of detecting image spam has a wider scope than the spam filtering task in itself, since it is an instance of the more general problem of content-based filtering of multimedia documents. Accordingly, we believe that detecting image spam can be still considered a relevant issue to the computer vision and pattern recognition communities.

3. Image spam detection techniques

In this section we survey and categorise the techniques based on computer vision and pattern recognition, which have been proposed so far against image spam. They can be subdivided in two broad categories: techniques based on OCR tools (Sect. 3.1), and on low-level image features (Sect. 3.2). A previous survey has been given in Hayati and Potdar (2008). We extend it by considering a larger and updated set of works, and by providing an experimental evaluation and comparison of some of the considered techniques.

3.1. OCR-based techniques

OCR-based techniques extract and analyse the text embedded into attached images. While some implementation of these techniques can be found in commercial spam filters (Samosseiko and Thomas, 2006) and in open-source ones like SpamAssassin,[5] they have been investigated only in our previous work (Fumera et al., 2006). Existing techniques are based on the same approaches which are used in spam filters to analyse email's body text: keyword detection and text categorisation.

[1] See `http://spamassassin.apache.org/tests.html`

[2] A comprehensive list of spammers' tricks is reported at `http://www.virusbtn.com/resources/spammerscompendium/index`.

[3] `http://www.captcha.net`

[4] `http://blogs.iss.net/archive/image-spam-rebirth.html`

[5] See `http://wiki.apache.org/spamassassin/CustomPlugins`.

Figure 1: Examples of real spam images taken from the authors' mailboxes, and publicly available: *clean* (top) and *obfuscated* (middle, bottom) images.

Keyword detection. A simple method to find evidence that an email is spam is to check for the occurrence of typical keywords which are most likely to appear in spam emails. This requires a frequent update of the keyword list, and can be easily circumvented by tricks like misspelling common "spammy" words (see the examples in Sect. 2). When applied to text extracted from OCR tools, this technique exhibits the same drawbacks above, and its effectiveness can be undermined also by OCR errors. An implementation of this technique is the *OCR* plug-in of SpamAssassin.[6] Its default keyword list is very short, and can be customised by end users. The *OCR* plug-in has Boolean output, which is set to `True` if at least one of the keywords is detected in the image. The Boolean values are numerically coded as 3 (`True`) and 0 (`False`), to be combined with the outputs of the other SpamAssassin's modules.

To compensate OCR errors, a fuzzy matching between keywords and words extracted by OCR can be considered, instead of the exact word matching. This could also reduce the effect of misspelled "spammy" words. This approach is exploited by another SpamAssassin plug-in, *FuzzyOCR*.[7] Its output is a real number which increases as the number of keywords found in the text extracted by the OCR increases. The underlying rationale is that the more the keywords found, the higher the likelihood that the considered image embeds a spam message.

Text categorisation. In Fumera et al. (2006) we investigated whether the same text categorisation techniques applied to email's body text can be effective also to analyse the text extracted by OCR. We first considered text classifiers trained on text coming from email's body, and tested on text coming from both the email's body and attached images (if any). This method allowed to improve the image spam detection rate, despite OCR errors. We observed that the image spam detection rate further improved, if the text extracted by OCR was processed by a distinct text classifier, trained only on text extracted from images. The reason seems to be that the text used for training and testing was affected by the same kinds of OCR errors, which is in agreement with Ittner et al. (1995). At the time of our work (early 2006) spammers were not adopting any obfuscation technique against OCR, and thus our approach was not investigated against obfuscated images. Our approach was implemented in the *BayesOCR* SpamAssassin plug-in,[8] which feeds the text extracted by OCR to the text classifier included in SpamAssassin (it is trained on email's body text only). The output of this classifier is a real number in $[0, 1]$, to be interpreted as the probability that the considered image embeds a spam message, and is multiplied to a default weight of 4.5 to be combined with the outputs of the other SpamAssassin modules.

3.2. *Techniques based on low-level image features*

Methods based on using low-level image features can be subdivided into two categories: image classification techniques, and near-duplicate detection techniques.

Image classification. Several authors proposed to detect image spam using a discriminative approach: a set of low-level features is extracted from images, and a classifier is trained on a feature vector representation of a collection of images attached to spam and legitimate images. The main distinctive characteristic of the different techniques are the chosen features. They are often defined on the basis of some assumptions about the properties which discriminate spam and legitimate images, derived from the analysis of real spam images. A second characteristic is the classification approach. Most of the works use a two-class classifier, some of them use a one-class classifier (trained on spam images only), while we found just one work in which a multi-class approach is proposed. The effectiveness

[6] `http://wiki.apache.org/spamassassin/OcrPlugin`
[7] `http://fuzzyocr.own-hero.net/`
[8] `http://prag.diee.unica.it/pra/eng/research/doccategorisation/spamfiltering/products/bayesocrplugin`

of the proposed techniques has been almost always evaluated on different data sets (often personal email collections, not publicly available), with different experimental settings and different performance measures, and with no comparison with other works. The reported performances are thus not comparable. The experimental evaluation in Sect. 4 partially addresses this issue. Finally, although the average processing time per image is a critical factor, it was not reported by all authors.

In the following we describe the main works we are aware of. They are summarised in Table 1, where some information on the data sets used and the reported performances is also included, for the sake of completeness.

Some works make the common assumption that the main distinctive characteristics of spam images are the presence of relatively large text areas, and the fact that such images are often computer-generated graphics with specific properties Wu et al. (2005); Aradhye et al. (2005); Liu et al. (2010).

The work by Wu et al. (2005) is the first one in which an image classification technique was proposed. The chosen features were cumulatively computed on all the images attached to an email. Features related to the presence of text are: the number of detected text regions, the fraction of images with detected text regions, and the relative area occupied by text (usually denoted as *text area*). Based on the assumption that many spam images are banners and computer-generated graphics (which are part of advertisements), the ratio of the number of banner and of graphic images to the total number of attached images were also used as features. Banners were detected through their aspect ratio, height, and width. Graphics detection was based on the assumption that computer-generated graphics usually contain homogeneous background and very little texture; it was carried out through texture analysis based on wavelets. The ratio of external images (i.e., images placed on a remote server and linked in the email's body) to the total number of external and attached images was also used as a feature, given that spam emails often contain links to external images. A one-class classifier (a SVM) was used in this work, since a representative set of legitimate emails was deemed difficult to collect.

The features proposed in Aradhye et al. (2005) were separately computed on each individual image. First, the text area, extracted by an ad hoc method, was used. Then, four colour saturation and heterogeneity features were proposed, since it was argued that colour saturation and heterogeneity values in spam images are intermediate between those of legitimate images of natural scenes and legitimate computer-generated graphics. A two-class SVM classifier was used in this work.

Finally, Liu et al. (2010) used features derived from corner and edge detection to characterise text areas, while colour features were used to characterise the graphic properties of spam images. In particular, colour variance, prevalent colour coverage, and the number of colours contained in an image were used as features (with no explicit assumption about their distribution in spam and legitimate images), as well as colour saturation features, with the same rationale as in Aradhye et al. (2005).

In other works only the low-level graphic properties of images were taken into account (Byun et al., 2007; Mehta et al., 2008; Gao et al., 2008; Hsia and Chen, 2009; Zuo et al., 2009b,a). The only exception is Hsia and Chen (2009), where the presence of text regions was used in a pre-filtering step. In particular, the rationale of the techniques in Hsia and Chen (2009); Zuo et al. (2009b,a) is to detect spam images identical or very similar to the ones in the training set, exploiting the fact that they are often sent in batches.

In Byun et al. (2007) four properties were argued to be specific of image spam: colour characteristics (such as discontinuous distributions, high intensity, and dominant peaks), characterised with colour moments computed in the HSV space; colour heterogeneity (which is deemed to be more uniform in spam images), characterised by RMS differences between the original and the quantised images; "conspicuousness" (intended as the presence of highly contrasted colours aimed at making the spam message easily noticeable), characterised by colour saturation features; and self-similarity (based on the observation that different regions in the same spam image often exhibit similar characteristics, contrary to legitimate images), measured through a log-Gabor filter bank on predefined image blocks. A multi-class approach was proposed in this work, based on the rationale that several sub-classes of both spam and legitimate images exist, exhibiting large intra-class variations. Five sub-classes of spam images were considered: synthetic images containing text with simple or complex background, synthetic images without text, and non-synthetic images with sexual or non-sexual content. In the case of legitimate images, three sub-classes were considered: photos, maps and directions, and cartoons. A maximum figure of merit classification algorithm was used, and a comparison with the techniques in Wu et al. (2005) and Aradhye et al. (2005) was also reported.

Gao et al. (2008) argued that spam images are mainly artificially generated, and can be discriminated from legitimate images (supposed to be typically photographs of natural scenes) through their texture statistics. To this aim, the global colour and gradient orientation histograms were proposed as features. A two-class probabilistic boosting tree was used as classifier.

A similar assumption was made by Mehta et al. (2008), who argued that spam images are often artificial, and contain clearer and sharper objects than legitimate images; thus, their colour distribution should be less smooth. In particular, the authors suggested that the highest discriminant capability is attained by low-level features which capture how an image is perceived by the end user, and that such features also make it more difficult for a spammer to manipulate his images to get them misclassified as legitimate. The proposed features are related to colour, shape and texture. A two-class SVM classifier with the RBF (non-linear) kernel was used.

The method proposed in Hsia and Chen (2009) pre-filters images on the basis of the extent of detected text regions. Images are fed to a two-class classifier (a SVM), only if their relative text area exceeds a predefined threshold, otherwise they are labelled as legitimate. The approach used to build the classifier is based on characterising spam images through visual bag-of-words. This approach is analogous to the bag-of-words representation of text, and is used for object recognition as well as in image and video retrieval tasks. Also in this case the rationale is to detect near-duplicate spam images.

The works in Zuo et al. (2009b,a) were based on the same rationale above. In particular, the presence of random variations introduced by spammers to circumvent signature-based filters was considered, including translation, rotation, scaling, local changes and addition of random noise. The features proposed in Zuo et al. (2009b) are based on the Fourier-Mellin invariant descriptor, which is a translation, scaling and rotation invariant function, widely used in watermark detection and fingerprint verification systems. The local invariant features MSER and SURF were instead used in Zuo et al. (2009a). In both works a one-class SVM classifier was used.

Dredze et al. (2007) and Uemura and Tabata (2008) proposed a rather different approach, based on the use of image metadata.

The main goal in Dredze et al. (2007) was to build a fast classifier, avoiding computationally costly image processing operations. The corresponding choice of features was based on an analogy with the bag-of-words representation of text, where a large number of individual words are used as features: although the discriminant capability of every single word is very low, the one of a large set of words may be very high. The proposed approach consists in exploiting a large set of so-called "incidental features" which are computationally cheap to compute: image metadata, and information like image height, width, aspect ratio, format extension (e.g., gif, jpg), and file size. An additional motivation for using metadata is that they can reveal the use of ad hoc software tools by spammers to forge their images (so-called "ratware", Stern, 2008). Some visual features were also considered: average red, green and blue values, features based on edge detection, and the prevalent colour coverage, with no specific assumption on their discriminant capability, as well as the colour saturation features of Aradhye et al. (2005). Maximum entropy and Naïve Bayes classifiers were used, as well as decision trees, which allow to compute only a subset of features per testing image. The processing time at classification phase was further improved through an ad hoc feature selection step. The reported image processing time at classification phase varies from 2.5 to 4.4 milliseconds, depending on the classifier.

Uemura and Tabata (2008) used only few image metadata, making assumptions on their discriminant capability: the file name (which is likely to be the same for identical or similar images), the file size (which is deemed to be lower in spam images than in legitimate ones), the image area in pixels, and the image compressibility (which is supposed to be higher for spam images, since their content is typically simpler than that of legitimate images). The proposed classification approach was different than in other works: such features were fused together with the text features and fed to the same "Bayesian filter", implemented according to Robinson (2003).

Finally, in our previous works (Biggio et al., 2007, 2008) we focused on devising features capable of detecting spam images with obfuscated text (see Fig. 1, bottom). The underlying assumption is that the text in legitimate images (if any) is unlikely to be obfuscated with an "adversarial" kind of noise. In Biggio et al. (2007) we developed three features aimed at detecting text areas which contain characters exhibiting "atypical" shapes, resulting from the use of different obfuscation techniques observed in real spam images, like lines of the same colour as the background running over the text, and non-uniform backgrounds. Such shapes are originated from broken characters, or groups of connected characters, and are emphasised by image binarization. We characterised them through a modification of the *perimetric complexity* measure (Baird and Chew, 2003; Pelli et al., 2006). Although these features turned out to be not capable of detecting spam images with obfuscated text directly, we found that they allowed to discriminate between low-level characteristics of text embedded into *generic* (even non-obfuscated) spam images and into legitimate ones. These features provided a good discriminant capability between spam and legitimate images, when used in a two-class SVM classifier together with some features proposed in other works: the image aspect ratio, the relative area of text regions, the number of different colours present in the image, and the prevalent colour coverage (Biggio et al., 2008). Extracting our features took about 0.03 seconds per image. Text localisation was much slower instead (about 1.2 seconds per image), although our code was not optimised: for instance, in Hsia and Chen (2009) a processing time of 0.160 seconds was reported for text localisation. We also developed a plug-in for SpamAssassin based on these features, called *Image Cerberus*.[9]

Near-duplicate detection. These techniques are analogous to content-based image retrieval (CBIR) techniques, which aim at finding images that "look alike" a query image. They are based on the same rationale of some image classification techniques mentioned above: spam images are often generated from a common template, are randomised to evade signature-based detection, and are sent in batches to many users. Thus, images derived from the same template are visually similar ("near-duplicate"), and can be recognised by a comparison with a data base of known spam images. In particular, in Mehta et al. (2008) it is argued that the huge volume of spam makes it inevitable for spammers to send many similar images generated from a common template; hence, near-duplicate detection techniques are believed to remain effective over time.

This approach is implemented in Wang et al. (2007) by exploiting one or several feature sets, possibly extracted by existing image spam filters (in this case they are used only as feature extractors, not as classifiers). The distance between a query image and the templates in the data base is computed separately in each feature space, and compared to a threshold (a different one for each feature space) to decide whether it is spam or legitimate. Labels obtained from different feature spaces are then combined using logical operators (OR, or AND), or by voting. Three kinds of features were experimentally evaluated, derived from colour histograms, Haar wavelet transform, and edge orientation histograms. The reported processing time per image is of the order of milliseconds.

The features proposed in (Mehta et al., 2008) were based on the assumption that spam images are often artificially generated, and are related to spatial information (pixel coordinates), colour and texture. Their distribution was approximated with Gaussian mixture models. The distance between images was

[9] `http://imagecerberus.sourceforge.net`

measured using the Jensen-Shannon divergence, given that it is symmetric, unlike commonly used measures of distance between probability distributions like the Kullback-Leiber divergence.

In (He et al., 2009) images are processed in two steps. First, a very fast comparison is made between file properties of the input and template images: file size, image width and height, bit depth, and aspect ratio. If they are deemed similar, another comparison is carried out on gray-level or colour image histogram, using measures like histogram intersection and Euclidean distance. This technique is claimed to be fast, although the processing time is not reported.

We finally mention the technique proposed in (Qu and Zhang, 2009), in which the image similarity is computed on the basis of colour moments, texture, and shape (edge) features. The label produced by a two-class SVM classifier trained on spam and legitimate images is considered as an additional feature.

3.3. Discussion

All the techniques described above exhibit pros and cons, which however can be discussed mainly on a theoretical basis, given the absence of a common and complete empirical evaluation.

OCR-based techniques allow to analyse the high-level content of many spam images, namely, the conveyed textual message (if any). A potential advantage is that they are unlikely to produce false positives (legitimate emails misclassified as spam), given that it is improbable that images attached to legitimate emails embed "spammy" keywords. However, their effectiveness is potentially affected by OCR errors, which depend on many factors like font face and size, image background and resolution, and the characteristics of the OCR itself. Nevertheless, some robustness to OCR errors was observed in Fumera et al. (2006), when the extracted text was processed using the same text classifiers used for the emails' body text. The more serious drawback is perhaps the high computational cost, which may lead to an unacceptable processing time, especially on email servers. Finally, OCR tools can be made ineffective by obfuscation techniques like the ones already used by spammers, which do not affect image processing techniques based on low-level features, instead. Therefore, although OCR-based techniques can be viewed as complementary to low-level image processing techniques, their practical use in spam filters can be at most envisaged as the last phase of a multi-stage processing approach, in which computationally cheaper techniques are used at earlier stages, and the classification of an image is postponed to a later stage, only in case of uncertainty.

Besides not being affected by text obfuscation, in principle the main advantage of image classification techniques is their generalisation capability, due to the use of machine learning algorithms. This can allow them to infer hidden patterns of spam and legitimate images, and to recognise even new kinds of image spam (i.e., images seemingly very different from training ones). However, this capability strongly depends on the choice of proper features. As pointed out before, in many works the choice of features is based on assumptions about the properties which best discriminate spam and legitimate images. Although such properties are derived from the observation of real spam images, it is not unlikely that they can change over time. In particular, this may happen due to the adversarial nature of the spam filtering problem (see Sect. 5).

Near-duplicate detection techniques may exhibit a low false positive rate, given that they label as spam only images similar to known spam images. However, for the same reason they are not intrinsically able to detect new kinds of image spam; thus, they do not exhibit (at least in principle) the generalisation capability of classification techniques based on learning algorithms. The choice of a proper feature set is a critical issue, analogously to image classification techniques.

The processing time at operation phase of techniques based on low-level features mainly depends on the computation of feature values. According to processing times reported by some authors, the classification algorithm has a much smaller impact. In some works both features cheap to compute and fast classification algorithms like decision trees or the Naïve Bayes are explicitly considered (Dredze et al., 2007; Wang et al., 2007; Uemura and Tabata, 2008), while in some cases the processing time is not discussed. A potential drawback of near-duplicate detection techniques is that, although computing a distance measure between two images can be faster than running a classification algorithm (like a SVM with a non-linear kernel), the distance has to be computed for each template image. If the template set is not small enough, this may significantly affect the processing time.

Finally, we point out that all techniques proposed so far lack of any analysis on their vulnerability to adversarial data manipulation. This issue is discussed in Sect. 5.

4. Experimental evaluation

In this section we report an experimental evaluation and comparison among the two kinds of techniques which are potentially capable of detecting also spam images not seen before, namely, the ones based on OCR and on image classifiers. For both kinds of techniques, we considered a representative one for each different approach or type of feature, among the ones identified in Sect. 3. Our choice was also limited to techniques for which enough implementation details were available. Among OCR-based techniques, we chose the *FuzzyOCR* and *BayesOCR* plug-ins of SpamAssassin, which are based on keyword detection and text classification, respectively. Note that analysing the behaviour of these plug-ins is interesting also because SpamAssassin is the most widely used filter, according to the 2009 Spam Survey by the European Network and Information Security Agency (ENISA).[10] Among image classification techniques, we chose the ones in Aradhye et al. (2005) (whose features are related to text areas and to graphics), Dredze et al. (2007) (mainly metadata) and Biggio et al. (2008) (character shape and "generic" visual features). They will be denoted respectively as 'Aradhye', 'Dredze' and 'Image Cerberus'. The

[10] `http://www.enisa.europa.eu/act/res/other-areas/anti-spam-measures/studies/spam-survey`

OCR + Text categorisation

Ref.	*Classifier*	*Data sets and size*	*Spam*	*Ham*	*Results*	*Processing time (ms)*
Fumera et al. (2006)	SVM	*b*,P	445 5608	4852 9526	TP=0.77–0.81, FP=0.01	

Image classification

Ref.	*Kind of feature* TA	LL	IS	RS	M	O	*Classifier*	*Data sets and size*	*Spam*	*Ham*	*Results*	*Processing time (ms)*
Wu et al. (2005)	+	+					One-class SVM	*b*,P	10^4	1428	TP=0.81–0.95, FP=0.01–0.06	500
Aradhye et al. (2005)	+	+					SVM	P	1245	1486	TP=0.81–0.87, FP=0.12–0.27	
Liu et al. (2010)	+	+					SVM	*a*,*b*,P	3112 8719	4041	TP=0.97–0.98, FP=0.01–0.02	
Byun et al. (2007)		+		+			Multi-class max. figure of merit	*b*,P	669 1249	1625 288	TP=0.81–0.86, FP=0.06–0.19	
Gao et al. (2008)		+	+				Prob. boosting tree	P	928	830	TP=0.89, FP=0.01	400
Hsia and Chen (2009)		+	+				SVM	*a*,*b*,P	13000	12500	Acc.=0.90–0.98	160
Zuo et al. (2009b)		+	+				One-class SVM	*b*,*d*,P	1712 11256	2570 15304	Prec.=0.99, Rec.=0.79–0.84	190
Zuo et al. (2009a)		+	+				One-class SVM	*a*,*b*,P	3885	2839	Acc.=0.98	
Mehta et al. (2008)		+					SVM	*a*,*b*,*c*,P	3239 1071 10623	5373	Acc.=0.95–0.98	
Dredze et al. (2007)		+			+		Max. Entropy Naïve Bayes Decision Tree	*a*,*b*,P	3239 9503 12742	2550	Acc.=0.93–0.96	2.5–4.4
Uemura and Tabata (2008)		+			+		Bayesian filter	P	800	600	TP=0.90–0.99, FP=0.01–0.02	270
Biggio et al. (2008)	+	+				+	SVM	*a*,P	3239 8549	2550 2006	TP=0.94–0.98, FP=0.02–0.05	1200

Near-duplicate detection

Ref.	*Kind of feature* TA	LL	IS	RS	M	O	*Distance measure*	*Data sets and size*	*Spam*	*Ham*	*Results*	*Processing time (ms)*
Wang et al. (2007)		+	+				Manhattan	P	1071	10^7	TP=0.63–0.96, FP=0–0.173	50
He et al. (2009)		+	+				Hist., Euclidean	*b*,P	1977	8000	Acc.= 0.81–0.98	
Qu and Zhang (2009)		+	+				Euclidean (weight.)	*b*,P	9235	2021	TP=0.93, FP=0.05	
Mehta et al. (2008)		+	+				Jensen-Shannon	*c*,P	1004		TP=0.76–0.84	

Kinds of features

TA	Text area
LL	Low-level image properties (colour, texture, etc.)
IS	Image similarity
RS	Image regions similarity
M	Image metadata
O	Text obfuscation

Data sets

P	Personal collection(s)
a	Dredze et al. (2007)
b	SpamArchive
c	Princeton spam corpus
d	Caltech-256

Results

Acc.	Classification accuracy
TP	True positive rate
FP	False positive rate
Prec.	Precision
Rec.	Recall

Table 1: A summary of the main characteristics of the image spam detection techniques described in the text.

Data set	Spam images	Legitimate images
Dredze	3,114	1,742
Dredze + Personal	6,636	1,742
TREC 2007	6,854	732

Table 2: Number of images in the three data sets used in our experiments.

Classifier	Parameters
SVM	RBF kernel with $\gamma \in \{0.01, 0.1, 0.5, 1.0, 10\}$ $C \in \{0.1, 1.0, 10.0\}$
C4.5	confidence factor ranging from 0.05 to 0.65 (step 0.10)

Table 3: Values considered for classifier parameter estimation.

experiments were carried out on the same, publicly available data sets, using the same experimental settings and performance measures.

4.1. Experimental setup

Our experiments were carried out on different data sets of images extracted from *real* spam and legitimate emails, which are publicly available at `http://prag.diee.unica.it/pra/eng/spamRepository`. One data set is made up of personal emails collected by Dredze et al. (2007). We will denote it as 'Dredze'. Another data set was made up by the legitimate images of the 'Dredze' data set and a corpus of personal spam images collected by the authors between 2004 and 2007, and used in Biggio et al. (2008). It will be denoted as 'Dredze + Personal'. A third data set consists of images extracted from the publicly available TREC 2007 spam corpus (Cormack, 2007), made up of messages collected in 2007 from different accounts, including honeypots. For a fair comparison, we removed from all data sets the images which could not be processed by the original code for feature extraction provided in Dredze et al. (2007), due to limitations of the Java library used in that code. The size of the resulting data sets is summarised in Table 2.

These data sets exhibit different characteristics. Many legitimate images of TREC 2007 are identical or very similar, since they were collected from online services and mailing lists, and no more than about 10% of them are natural images. Legitimate images of the Dredze data set are more variable, instead, and include photos, scanned images, graphics, icons and banners. About 30% of them are natural images, and about 50% contain text. Spam images exhibit a rather larger variability in all data sets, and especially in our corpus, which was collected over a longer time period. Among spam images with embedded text, in the Dredze data set about 25% are obfuscated, while this percentage drops to about 15% in our data set as well as in TREC 2007.

The default configuration settings for the SpamAssassin plug-ins were used, including the keyword lists. Among the two open-source OCR tools available, *gocr* (v. 0.44) and *tesseract* (v. 2.01), we considered the former, since it exhibited the best performance. The text classifier used by *BayesOCR* was trained on the body text of 5,168 personal spam emails collected by the authors between January and July 2006, and the first (in chronological order) 3,201 legitimate emails of the TREC 2006 corpus (Cormack, 2006).

The image classification techniques were implemented according to the corresponding works. A SVM with radial basis function (RBF) kernel was used for the techniques in Aradhye et al. (2005) and Biggio et al. (2008). It was implemented with the LibSVM software (Chang and Lin, 2001). A decision tree was used for the technique in Dredze et al. (2007), implemented with the C4.5 software (Quinlan, 1993).

The classification performance was measured using the Receiver Operating Characteristic (ROC) curve, using a 5-fold cross validation. Classifier parameters were set by an inner 5-fold cross validation. The parameters which minimised the sum of the false positive (FP) error rates corresponding to true positive (TP) classification rates ranging from 0.70 to 0.95 (with steps of 0.025) were chosen, through a grid search among the values listed in Table 3. The rationale was to prefer classifiers exhibiting low FP rates for a range of reasonably high values of the TP rate, since in spam filtering FP errors are much more harmful than false negative ones.

4.2. Experimental results

In Fig. 2 we report the average ROC curves attained by the considered techniques on the three data sets. Only the most interesting portion is shown, corresponding to low FP rates. The standard deviation turned out to be negligible; thus, it is not reported.

It can be seen that Image Cerberus almost always outperformed the other techniques, except for high FP rates in the Dredze and Dredze+Personal data sets, where the classifier of Dredze et al. (2007) performed slightly better. The relative performance of the 'Dredze' and 'Aradhye' techniques depend on the data set, and in one case (on the 'Dredze' data set) also on the operational point. In some cases they are outperformed by the OCR-based techniques. Among OCR-based techniques, *FuzzyOCR* almost always outperformed *BayesOCR*. Note that these techniques were not able to attain TP rates as high as the other ones. The reason is that they gave the lowest score (corresponding to the lowest "spamminess" degree) to all images in which no text was detected, which included both spam and legitimate images. The only way to attain a higher TP rate than the one shown in Fig. 2 was to trivially label all images as spam, which also increased the FP rate to 1.0.

Let us analyse in more details these results, focusing on the assumptions underlying the considered techniques.

Consider first that in several cases the OCR-based techniques attained lower TP rates than image classification ones, being equal the FP rate. The reason is that the OCR-based techniques misclassified as legitimate several spam images without text (e.g., icons, logos and photos), which were correctly labelled by image classification techniques. This is reasonable for FuzzyOCR, since no spam keywords can be found in those images. For BayesOCR, this depends on the fact that most of the legitimate images in the training set do not contain text.

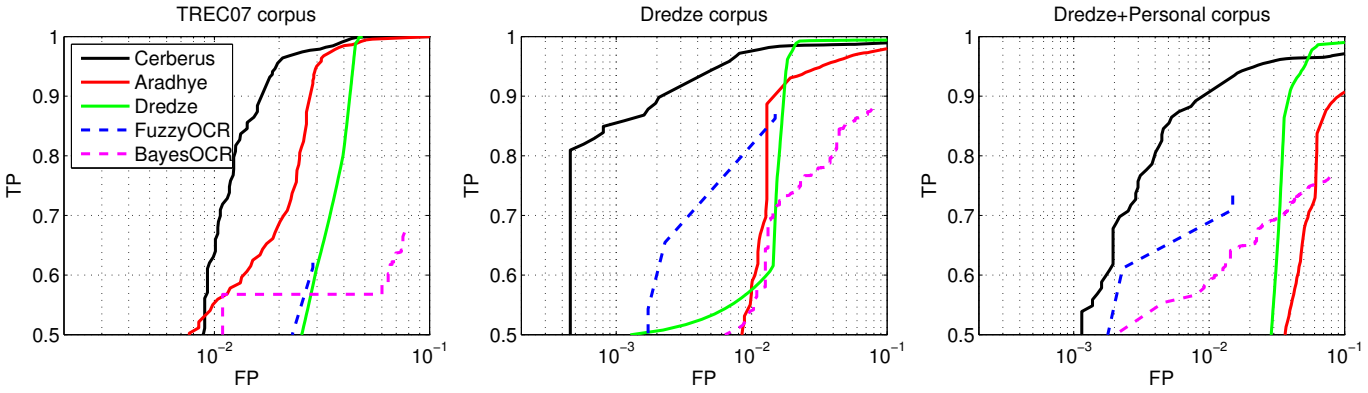

Figure 2: Average ROC curves attained on the three data sets by the five considered techniques for image spam detection.

A notable exception to this behaviour can be observed in the Dredze+Personal data set, in which for very low FP rates the OCR-based techniques largely outperform the 'Aradhye' and 'Dredze' image classification techniques. The reason is that some spam images, including banners, icons and scanned images, were close to some legitimate images in the feature spaces used by the 'Aradhye' and 'Drezde' techniques, and were misclassified as legitimate. Moreover, many spam images with obfuscated text were misclassified by OCR-based techniques, but not by image classification ones. Notably, the latter are not influenced by text obfuscation techniques, and do not rely only on the presence of text. Nevertheless, we observed that some obfuscated images were correctly detected by OCR-based techniques, too. To better understand whether the obfuscation techniques observed in real spam emails are actually able to evade OCR-based techniques, besides other factors which may affect OCR performance, like font face, font size and image resolution, we carried out further experiments on artificially generated spam images. In particular, we mimicked three different kinds of obfuscation techniques observed in real spam images, like the ones shown in Fig. 1 (bottom). We found evidence that the considered OCR-based plug-ins can be evaded by such obfuscation techniques, even when they are able to correctly extract the text from the corresponding clean images, characterised by the same resolution, font face and font size. Finally, the reason why *FuzzyOCR* often outperformed *BayesOCR* is that its fuzzy word matching better compensates OCR errors. Moreover, the text classifier used by *BayesOCR* is trained on emails' body text, which is not affected by OCR errors.

Consider now the image classification techniques. The assumption in Aradhye et al. (2005) about the prevalence of text regions in spam images than in legitimate ones turned out to be true on the three data sets considered. However, the assumption that in spam images colour saturation and heterogeneity values are intermediate between the ones of legitimate natural images and computer-generated graphics turned out not to hold. Nevertheless, we observed that the clusters of spam and legitimate images were rather well separated in the considered feature space. The features used in Dredze et al. (2007) are mainly derived from image metadata. We found evidence that these features actually improved the discriminant capability of the corresponding classifier, with respect to the low-level image features alone used in Dredze et al. (2007). For instance, on the largest Dredze+Personal data set they allowed to increase the TP rate of about 0.3 for FP rates below 0.05. The effect is that the techniques in Dredze et al. (2007) and Aradhye et al. (2005) (which share some low-level image features) performed rather similarly on our data sets. Finally, it turned out that the main reason why Image Cerberus almost always outperformed the two techniques above, is that it allowed to better discriminate among spam and legitimate images containing text. In particular, the other image classification techniques misclassified as spam several legitimate images containing text with complex shapes or backgrounds, which was part of photographs (e.g., a road sign), postcards, or playbills (see Figure 3). These images were correctly labelled by Image cerberus, instead. This fact agrees with the observations we made in Biggio et al. (2008), as mentioned in Sect. 3.2.

Consider finally the image processing time at classification phase. The one of OCR-based techniques was rather high as expected, about 1.5-2 seconds per image, on average. In the case of image classification techniques, the classification was always very fast (it took few milliseconds), while the feature extraction step was computationally cheap only for the features by Dredze et al. (2007) (few milliseconds as well, see also table 1). In particular, using a 2.26 GHz Intel CPU, the localisation of text areas required both by the 'Aradhye' and 'Image Cerberus' techniques took 1.17 seconds per image. Computing feature values took 2.47 seconds for 'Aradhye', including 2.20 seconds for a pre-processing step required by the heterogeneity features, and 0.03 seconds for 'Image Cerberus'. Accordingly, only the processing time of the 'Dredze' technique is compatible with a real word setting, although the code for the other two techniques was not fully optimised in our experiments. In particular, we believe that text localisation in 'Aradhye' and 'Image Cerberus' could be significantly speeded up.

We conclude by reminding the reader an observation made in Sect. 3.3: the two kinds of techniques considered in our experiments are based on different and somewhat complementary information sources (the text extracted from images, and low-level or metadata image features), and thus their combination may further improve their image spam detection capability.

Figure 3: Examples of legitimate images which were correctly classified only by Image Cerberus.

This is further suggested by the above results, since none of the considered techniques consistently outperformed all the other ones, except for 'Image Cerberus'. Accordingly, in the next section we give a preliminary experimental investigation of this issue.

4.3. Combination of OCR-based and image classification techniques

All the techniques considered in the above experiments provide a real-valued score, besides a class label. Both score-level and decision-level combining rules can thus be used in principle, among the ones proposed in the MCS field Kittler et al. (1998). Although we argued in Sect. 3.3 that a serial combination is more suitable, due to the high computational cost of OCR tools, in this preliminary investigation we consider a simpler parallel combination, only to assess whether combining may be beneficial in terms of classification performance. We also consider the average score fusion rule, which is one of the most simple and widely used rules for combining classifiers. Since the scores produced by the considered techniques have different ranges, we first normalised them in $[0, 1]$. In Fig. 4 we report the ROC curves obtained by combining all the possible pairs of an OCR-based plug-in and an image classification technique, among the ones considered in the above experiments, with the same experimental setting.

It can be seen that the ROC curves of the combined systems are often better or at least close to the ones of the individual ones. This happens especially when *FuzzyOCR* is used, which is in agreement with the fact that it outperformed *BayesOCR*. The main exceptions can be observed on the TREC 2007 data set, where the performances of the 'Aradhye' and 'Image Cerberus' techniques were worsened by the combination with both OCR-based plug-ins.

Although these experiments are very limited, they nevertheless provide some evidence that the combination of the two kinds of techniques can provide a better discriminant capability between spam and legitimate images. This result has also implications on the robustness to adversarial data manipulations: this point will be discussed in the next section.

5. Vulnerabilities of image spam detection techniques

When a system is being designed to operate in an adversarial environment, it is necessary to analyse its vulnerabilities to adversarial actions which may undermine it, and their potential impact to to its performance.

There are at least two kind of adversarial actions which can be targeted against two potential vulnerabilities of OCR-based techniques. One is obfuscating text to prevent OCR tools from recognising it. Another one is using the same tricks which are currently used in the emails' body text against keyword detectors and text classifiers, namely, misspelling "spammy" words and adding legitimate-looking text. While to our knowledge the latter kind of trick has not been observed yet in image spam, the former vulnerability has already been exploited by spammers. As mentioned in Sect. 4.2, we found that obfuscation techniques used by spammers are actually able to evade current OCR-based techniques.

To our knowledge, the only trick used so far by spammers against techniques based on low-level image properties is the randomisation of a template image, to evade signature-based detectors. Near-duplicate detection techniques have been proposed just to counteract such trick. No evidence of tricks against near-duplicate detection and image classification techniques have been observed so far. Thus, we can only discuss their potential vulnerabilities.

In principle, near-duplicate detection techniques may be evaded by avoiding using the sample template for different images. This is however unrealistic given the high volume of spam, as observed in Mehta et al. (2008). Another possibility is to introduce larger changes to the image template. To this aim, some knowledge on the feature space and the distance measure used by near-duplicate detection techniques may also be exploited. Such knowledge is in principle not difficult to obtain, although it may be rather difficult to exploit.

As explained in Sect. 3.2, image classification techniques are usually based on some assumption on the properties which discriminate between spam and legitimate images. However, spam images, and even more legitimate ones, are too broad and not well-defined categories, whose properties are likely to change

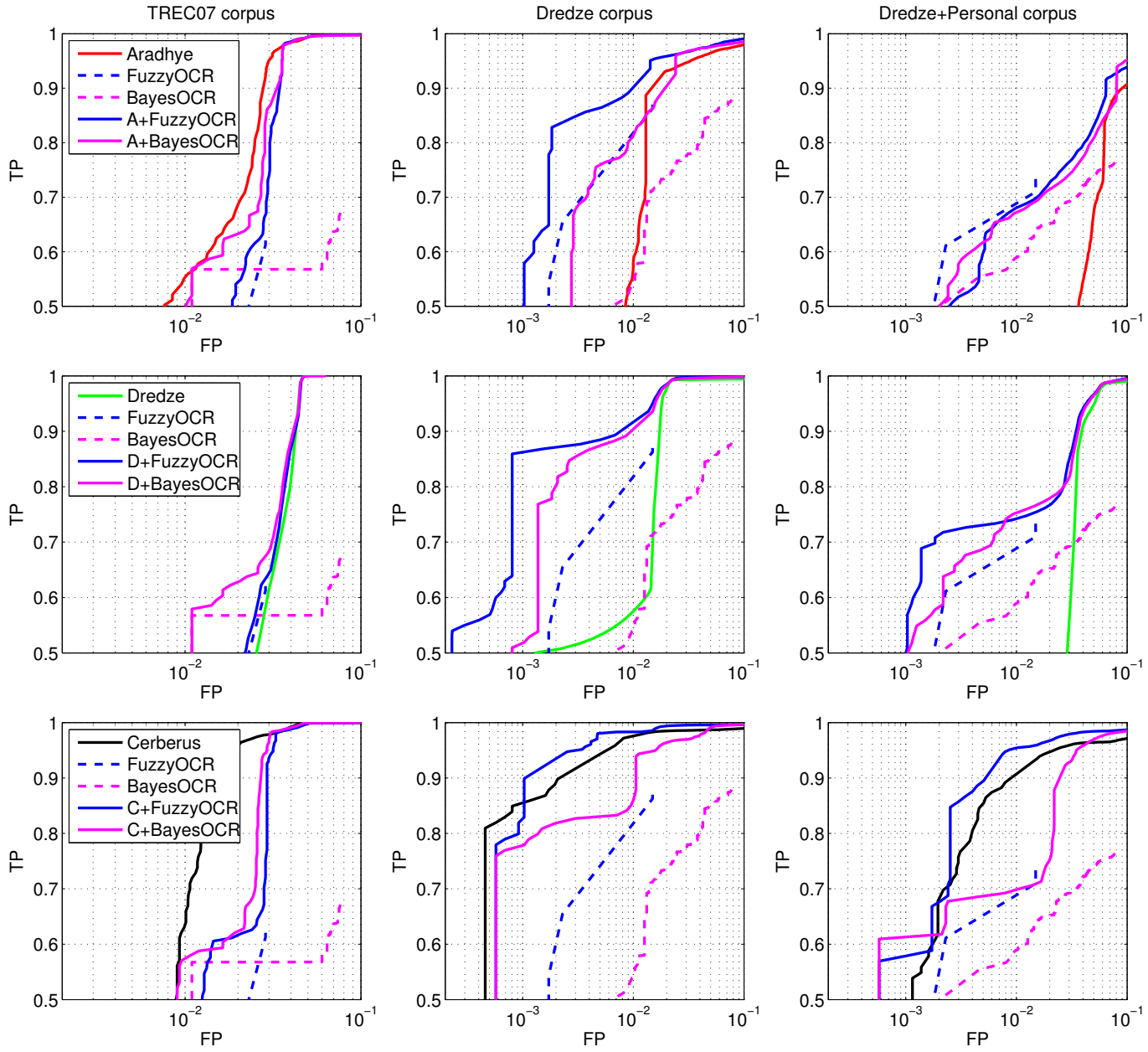

Figure 4: Average ROC curves obtained on the three data sets by combining the three image classification techniques and the two OCR-based plug-ins. The ROC curves of the individual techniques are also reported, for an easier comparison.

over time. In particular, spam images may be constructed with the aim of having them appear similar to legitimate ones in the chosen feature space, in an analogous way in which images can be obfuscated to make OCR tools ineffective. Manipulating their features to this aim is not difficult in principle, especially for features like the colour distribution proposed by several authors. For instance, the spam message can be placed over a photography. Manipulating the metadata features proposed in Dredze et al. (2007); Uemura and Tabata (2008) may be even simpler than low-level image features. Obviously some knowledge on the features is required, as for near-duplicate detection techniques. However, in this case spammers must also estimate which values such features exhibit on legitimate images.

Finally, note that the widely used approach of combining different techniques to improve the detection capability of spam filters may also be exploited to improve their robustness to adversarial data manipulation, as proposed by some authors. In the specific case of image spam, we provided in Sect. 4.3 some evidence that the combination of OCR-based and image classification techniques can improve the detection capability, by exploiting the high-level analysis made possible by text recognition, for images clean enough, and the low-level image features, for obfuscated images.

6. Conclusions

Image spam detection is an interesting instance of the problem of content-based filtering of multimedia data, which is becoming increasingly relevant, and is of interest also for the computer vision and pattern recognition research communities. This motivated the survey and the experimental analysis given in this work.

We reviewed the techniques for image spam filtering based on computer vision and pattern recognition methods proposed so far, pointing out the main approaches and assumptions on which they are based, as well as the kind of information they exploit. We also discussed their advantages and drawbacks, including their potential and already exploited vulnerabilities to adversarial data manipulation, given the adversarial nature of the spam filtering problem. Finally, we provided an experimental analysis and comparison of some representatives of two kinds of image spam filtering techniques, on three publicly available data sets of real spam and legitimate images. Such a comparison was still lacking in the literature, and allowed also to evaluate the underlying assumptions of the considered techniques on real data. We also gave a preliminary evaluation of the potential effectiveness of combining techniques based on complementary information, namely OCR-based and image classification techniques, which may provide also a way to increase their robustness in an adversarial setting.

We believe that our analysis and experimental results of image spam filtering techniques may provide useful hints to the development of pattern recognition techniques for content filtering in multimedia data, in particular for adversarial environments.

Acknowledgements

We would like to thank H. Stern for useful discussions, and M. Dredze for making his data set publicly available and sending us his code for feature extraction. This work was partly supported by a grant from Regione Autonoma della Sardegna awarded to B. Biggio and I. Pillai, PO Sardegna FSE 2007-2013, L.R.7/2007 "Promotion of the scientific research and technological innovation in Sardinia".

References

Androutsopoulos, A., Koutsias, J., Cbandrinos, K. V., Spyropoulos, C. D., 2000. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages. In: Proc. ACM Int. Conf. on Res. and Dev. in Information Retrieval, pp. 160–167.

Aradhye, H., Myers, G., Herson, J. A., 2005. Image analysis for efficient categorization of image-based spam e-mail. In: Proc. Int. Conf. on Document Analysis and Recognition, pp. 914–918.

Baird, H. S., Chew, M., 2003. Baffletext: a human interactive proof. In: Proc. IS&T/SPIE Document Recognition & Retrieval Conf.

Biggio, B., Fumera, G., Pillai, I., Roli, F., 2007. Image spam filtering using visual information. In: 14th Int. Conf. on Image Analysis and Processing. IEEE Computer Society, pp. 105–110.

Biggio, B., Fumera, G., Pillai, I., Roli, F., 2008. Improving image spam filtering using image text features. In: Proc. 5th Conf. on Email and Anti-Spam (CEAS).

Byun, B., Lee, C.-H., Webb, S., Pu, C., 2007. A discriminative classifier learning approach to image modeling and spam image identification. In: Proc. 4th Conf. on Email and Anti-Spam (CEAS).

Chang, C.-C., Lin, C.-J., 2001. LibSVM: a library for support vector machines, http://www.csie.ntu.edu.tw/ cjlin/libsvm/.

Cormack, G. V., 2006. Trec 2006 spam track overview. In: Voorhees, E. M., Buckland, L. P. (Eds.), TREC. Vol. Special Publication 500-272. National Institute of Standards and Technology (NIST).

Cormack, G. V., 2007. Trec 2007 spam track overview. In: Voorhees, E. M., Buckland, L. P. (Eds.), TREC. Vol. Special Publication 500-274. National Institute of Standards and Technology (NIST).

Cranor, L. F., LaMacchia, B. A., 1998. Spam! Commun. of ACM 41(8), 74–83.

Dredze, M., Gevaryahu, R., Elias-Bachrach, A., 2007. Learning fast classifiers for image spam. In: Proc. 4th Conf. on Email and Anti-Spam (CEAS).

Drucker, H., Wu, D., Vapnik, V. N., 1999. Support vector machines for spam categorization. IEEE Trans. on Neural Networks 10(5), 1048–1054.

Fawcett, T., 2003. "In vivo" spam filtering: a challenge problem for KDD. SIGKDD Explor. Newsletter 5(2), 140–148.

Fumera, G., Pillai, I., Roli, F., 2006. Spam filtering based on the analysis of text information embedded into images. Journal of Machine Learning Research (special issue on Machine Learning in Computer Security) 7, 2699–2720.

Gao, Y., Yang, M., Zhao, X., Pardo, B., Wu, Y., Pappas, T. N., Choudhary, A. N., 2008. Image spam hunter. In: Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP). IEEE, pp. 1765–1768.

Graham, P., 2002a. A plan for spam, http://paulgraham.com/spam.html.

Graham, P., 2002b. Will filters kill spam?, http://www.paulgraham.com/wfks.html.

Hayati, P., Potdar, V., 2008. Evaluation of spam detection and prevention frameworks for email and image spam: a state of art. In: Proc. 10th Int. Conf. on Information Integration and Web-based Applications & Services. ACM, pp. 520–527.

He, P., Wen, X., Zheng, W., 2009. A simple method for filtering image spam. In: 2009 th IEEE/ACIS Int. Conf. on Computer and Information Science. IEEE, pp. 910–913.

Hsia, J., Chen, M., 2009. Language-model-based detection cascade for efficient classification of image-based spam e-mail. In: IEEE Int. Conf. on Multimedia and Expo, pp. 1182–1185.

IBM, 2008. The 2008 x-force threat and risk report, http://www-935.ibm.com/services/us/iss/xforce/trendreports/.

Ittner, D. J., Lewis, D. D., Ahn, D. D., 1995. Text categorization of low quality images. In: 4th Annual Symp. on Document Analysis and Information Retrieval, pp. 301–315.

Kittler, J., Hatef, M., Duin, R. P., Matas, J., 1998. On combining classifiers. IEEE Trans. on Pattern Analysis and Machine Intelligence 20(3), 226–239.

Koprinska, I., Poon, J., Clark, J., Chan, J., 2007. Learning to classify e-mail. Information Sciences 177(10), 2167–2187.

Lai, C.-C., Wu, C.-H., Tsai, M.-C., 2009. Feature selection using particle swarm optimization with application in spam filtering. Int. Journal of Innovative Computing 5(2), 423–432.

Liu, Q., Qin, Z., Cheng, H., Wan, M., 2010. Efficient modeling of spam images. In: Int. Symp. on Intelligent Information Technology and Security Informatics. IEEE Computer Society, pp. 663–666.

Mehta, B., Nangia, S., Gupta, M., Nejdl, W., 2008. Detecting image spam using visual features and near duplicate detection. In: Proc. 17th Int. Conf. on World Wide Web. ACM, pp. 497–506.

Meyer, T. A., Whateley, B., 2004. Spambayes: Effective open-source, bayesian based, email classification system. In: First Conf. on Email and Anti-Spam (CEAS).

Pelli, D. G., Burns, C. W., Farell, B., Moore-Page, D. C., 2006. Feature detection and letter identification. Vision Research 46, 4646–4674.

Qu, Z., Zhang, Y., 2009. Filtering image spam using image semantics and near-duplicate detection. In: Proc. 2nd Int. Conf. on Intelligent Computation Technology and Automation - Vol. 1. IEEE Computer Society, pp. 600–603.

Quinlan, J. R., 1993. C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Robinson, G., 2003. A statistical approach to the spam problem. Linux J. 2003(107), p. 3.

Sahami, M., Dumais, S., Heckerman, D., Horvitz, E., 1998. A bayesian approach to filtering junk e-mail. AAAI Tech. Rep. WS-98-05, Madison, Wisconsin.

Samosseiko, D., Thomas, R., 2006. The game goes on: An analysis of modern

spam techniques. In: Virus Bulletin Conference.
Somayaji, A., 2004. How to win an evolutionary arms race. IEEE Security and Privacy 2(6), 70–72.
Stern, H., 2008. A survey of modern spam tools. In: Proc. 5th Conf. on Email and Anti-Spam (CEAS).
Uemura, M., Tabata, T., 2008. Design and evaluation of a bayesian-filter-based image spam filtering method. In: Int. Conf. on Information Security and Assurance, pp. 46–51.
Wang, Z., Josephson, W., Lv, Q., Charikar, M., Li, K., 2007. Filtering image spam with near-duplicate detection. In: Proc. 4th Conf. on Email and Anti-Spam (CEAS).
Weinstein, L., 2003. Inside risks: Spam wars. Comm. of ACM 46(8), p. 158.
Wu, C., Tsai, C., 2009. Robust classification for spam filtering by back-propagation neural networks using behavior-based features. Applied Intelligence 31(2), 107–121.
Wu, C.-T., Cheng, K.-T., Zhu, Q., Wu, Y.-L., 2005. Using visual features for anti-spam filtering. In: Proc. IEEE Int. Conf. on Image Processing, Vol. III. pp. 501–504.
Zuo, H., Hu, W., Wu, O., Chen, Y., Luo, G., 2009a. Detecting image spam using local invariant features and pyramid match kernel. In: Proc. 18th Int. Conf. on World Wide Web. ACM, pp. 1187–1188.
Zuo, H., Li, X., Wu, O., Hu, W., Luo, G., 2009b. Image spam filtering using fourier-mellin invariant features. In: Proc. Int. Conf. on Acoustics, Speech, and Signal Processing. IEEE, pp. 849–852.