Image spam filtering by detection of adversarial obfuscated text

F.Roli, B.Biggio, G.Fumera, I.Pillai, R.Satta

Department of Electrical and Electronic Engineering – University of Cagliari
Piazza d'Armi – I-09123 Cagliari (Italy)
Phone +39 070 6755779 Fax +39 070 6755782
{roli, battista.biggio, fumera, pillai}@diee.unica.it, riccardo.satta@tiscali.it

In recent years anti-spam filters have become necessary tools for Internet service providers to face up to the continuously growing spam phenomenon. Spam filtering is clearly a pattern recognition task in adversarial environment, as spammers are adversaries who continuously change the tricks used for evading classifiers (i.e., anti-spam filters) [1]. Many spammers' tricks are based on techniques that we can call of "content obfuscation". For example, in order to evade filters which analyse email content, spammers use content obscuring techniques, by misspelling words and inserting HTML tags inside words, so avoiding automatic detection of typical spam keywords. One of the last tricks used by spammers, named image-based spam (shortly image spam), consists in embedding the spam message into attached images to circumvent content-based filtering modules. This is a successful trick, as it makes all current techniques based on the analysis of digital text in the subject and body fields of emails ineffective. In [2], the authors proposed an approach based on OCR tools and text categorization techniques which showed to be effective on the first generation of image spam, which exhibited a low degree of obfuscation. Other researchers proposed alternative approaches to defeat image spam [3], and the popular Spam Assassin filter was recently equipped with some plugins that use OCR algorithms. However, spammers started to obfuscate the text embedded into images to make it unreadable by OCR algorithms. The authors have recently carried out a systematic evaluation of OCR performance on spammer-obscured text images showing that spammers can evade OCR tools quite easily using obscured text images. In fact, spammers can obscure text embedded into images in so many different ways that approaches aimed to extract and analyse it have poor chances of success. For this kind of image spam, we advocated the use of approaches which take into account explicitly the adversarial environment, namely, approaches which recognise spam images by detecting the presence of obscured text (i.e., by detecting the presence of adversarial actions) [5].

In this paper we give two contributions to image spam filtering as a pattern recognition task in adversarial environment. First, we show by experiments that filtering of adversarial obscured images can be an extremely difficult task, if spammers' actions for evading classifiers are not taken into account explicitly. In particular, we report results of a systematic evaluation of OCR performance on spammer-obscured text images. We tested performance of *gocr* and *Tesseract* which are two OCR tools freely available, used in some plugins of Spam Assassin. To assess OCR performance on a large and "controlled" database of obscured images, we implemented a software module that allows us generating automatically obscured text images by controlling the type and the level of obscuration. Controlling the obscuration level allowed us evaluating how and if OCR performance changes as the obscuration level increases. Performance of *gocr* and *Tesseract* were assessed in terms of the word error rate (WER) as a function of the obscuration level. Our results show that spammers can evade OCR tools quite easily using obscured text images without compromising human readability. Secondly, we propose a new approach to filter obscured spam images based on the detection of obfuscated text, namely, an approach which takes into account explicitly the adversarial environment, and it is based on authors' previous works [2,4,5]. We devised three features aimed at detecting some kinds of text image defects (e.g., character fragmentation or fusion, presence of background noise around text) that are caused by obfuscation techniques used by spammers and compromise OCR effectiveness. Two of our features are based on the *perimetric complexity* measure, which is able to discriminate images of clean characters from fragmented or fused characters and background noise components in the binarized image. Figure 1 outlines the extraction of one of our features, while Figure 2 shows an example of the three feature values on a spam image and on a legitimate image. To assess the performance of our features we report experimental results on a data set of real spam and legitimate images.

References

[1]J. Graham-Cumming, The spammer compendium, available at http://www.jgc.org/tsc.html.

[2]G. Fumera, I. Pillai, F. Roli, Spam filtering based on the analysis of text information embedded into images, Journal of Machine Learning Research, Vol. 7, pp. 2699-2720, 2006 (paper available at http://jmlr.csail.mit.edu/).

[3]M. Dredze, R.Gevaryahu, A. Elias-Bachrach, Learning fast classifiers for image spam, Fourth conference on email and anti-spam, CEAS 2007, Mountain View, California, August 2-3, 2007 (paper available at http://www.ceas.cc/).

[4]G.Fumera, I.Pillai, F.Roli, B.Biggio, Image spam filtering using textual and visual information, MIT Spam Conference 2007, Cambridge, USA, March 2007 (paper available at http://www.spamconference.org/).

[5]B.Biggio, G.Fumera, I. Pillai, F. Roli, Image spam filtering by content obscuring detection, Fourth conference on email and anti-spam, CEAS 2007, Mountain View, California, August 2-3, 2007 (paper available at http://www.ceas.cc/).

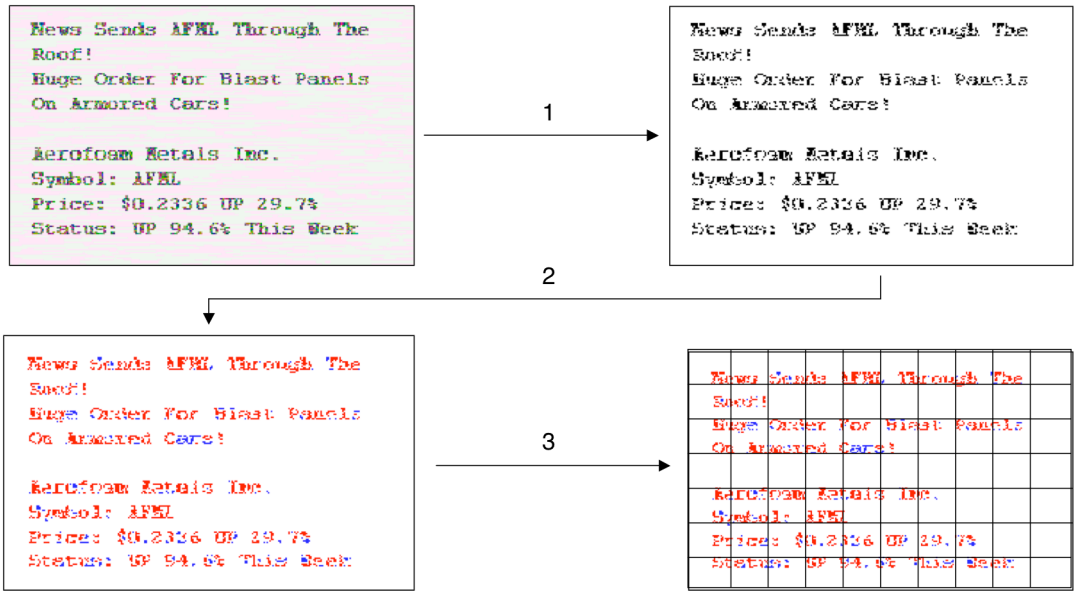

Figure 1. Computation of the feature aimed at detecting fragmented characters and small background components around text. Step 1: binarization of input spam image. Step 2: perimetric complexity is computed for each component in the binarized image, so allowing to identify character-like (red) and noisy patterns or fragmented characters due to text obfuscation (blue). Step 3: the feature value is defined as the average fraction of noisy ("spammer-obscured") patterns across predefined image cells.

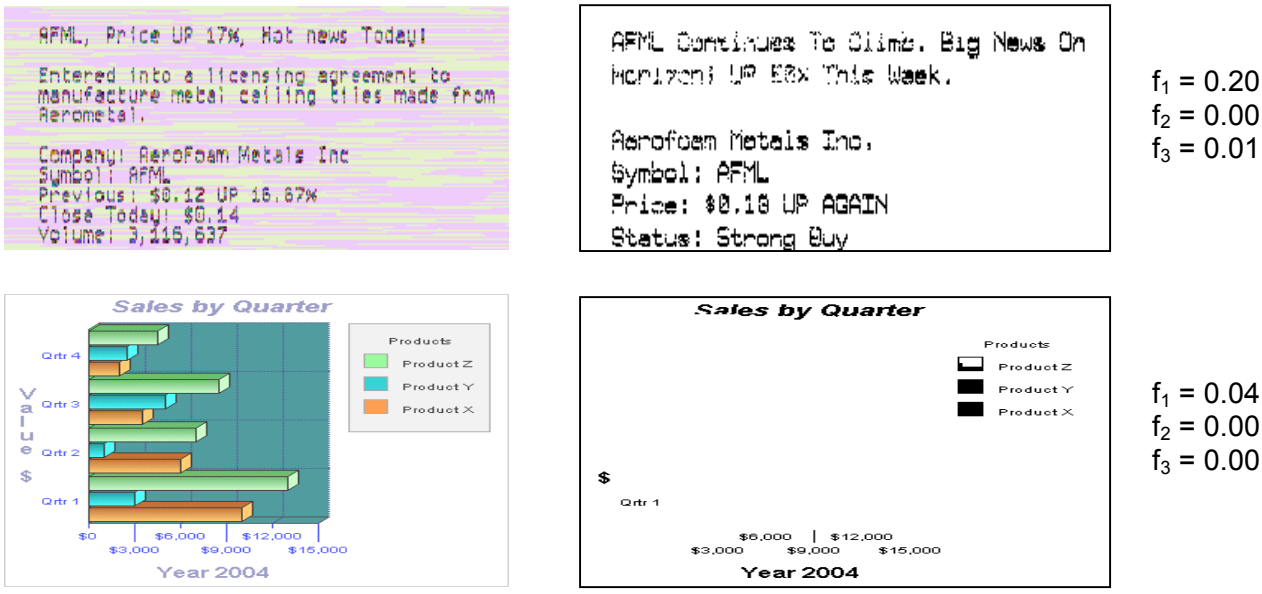

Figure 2. Examples of spam (top) and legitimate images (bottom), with the corresponding binarized images (computed only on text areas) on which our features are computed, and the values of our features. All the three features range in [0,1], the value of 0 meaning the absence of the considered kind of image defect due to text obfuscation. The feature f_1 measures the amount of character fragmentation and of background noise around text (see Figure 1); f_2 measures the amount of character merging (possibly through large background noise components); f_3 measures the amount of text hidden by large background components (a typical consequence of non-uniform image background). The spam image is affected by the kind of image defect measured by the f_1 measure: note the corresponding value of f_1, which is higher than in the legitimate image in which the text is clean.