Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

# Behavioral Experiments in Email Filter Evasion

**Liyiming Ke** and **Bo Li** and **Yevgeniy Vorobeychik**
Electrical Engineering and Computer Science
Vanderbilt University
Nashville, TN

## Abstract

Despite decades of effort to combat spam, unwanted and even malicious emails, such as phish which aim to deceive recipients into disclosing sensitive information, still routinely find their way into one's mailbox. To be sure, email filters manage to stop a large fraction of spam emails from ever reaching users, but spammers and phishers have mastered the art of *filter evasion*, or manipulating the content of email messages to avoid being filtered. We present a unique behavioral experiment designed to study email filter evasion. Our experiment is framed in somewhat broader terms: given the widespread use of machine learning methods for distinguishing spam and non-spam, we investigate how human subjects manipulate a spam template to evade a classification-based filter. We find that adding a small amount of noise to a filter significantly reduces the ability of subjects to evade it, observing that noise does not merely have a short-term impact, but also degrades evasion performance in the longer term. Moreover, we find that greater coverage of an email template by the classifier (filter) features significantly increases the difficulty of evading it. This observation suggests that aggressive feature reduction—a common practice in applied machine learning—can actually facilitate evasion. In addition to the descriptive analysis of behavior, we develop a synthetic model of human evasion behavior which closely matches observed behavior and effectively replicates experimental findings in simulation.

## Introduction

Email has come to be a mainstay of our daily lives. According to a 2012 McKinsey report people spend on average 28% of workday activities using email. Nie *et al.* have suggested that an average email user loses 10 working days each year dealing with such emails (Nie et al. 2005), and most estimates have placed worldwide cost of spam to over 10 billion dollars (Zeller Jr 2005).

Decades of research have yielded numerous methods for designing spam filters making use of knowledge engineering (Mason 2004) and machine (typically, classification) learning (Sakkis et al. 2003; Carreras and Marquez 2001), with the latter having become the prominent paradigm (Goodman, Cormack, and Heckerman 2007). In simple terms, the machine learning approach works by collecting many labeled instances of emails, where labels correspond to bad (spam) and normal (non-spam, sometimes called ham) emails. Emails themselves are represented quantitatively as feature vectors, with features often corresponding to presence or absence of specific indicator words or phrases, and a classification algorithm, such as Naive Bayes or Support Vector Machine (Bishop 2007) is run on the data to obtain a classifier which, given an arbitrary email instance (coded into features) outputs a decision whether this instance is spam or ham. For a given labeled data set, machine learning algorithms have come to be extremely good at detecting spam. The problem is that spammers have themselves become quite sophisticated in the techniques of *filter evasion*, or manipulating the spam email templates to bypass common filtering techniques. A typical approach in the field is a cat-and-mouse game in which a classifier is re-trained on new data at regular intervals, and spammers routinely change behavior in response (Goodman, Cormack, and Heckerman 2007). Clearly, a more proactive approach is called for, and a literature emerged with a focus on modeling and algorithmic assessment of the *classifier (filter) evasion problem* as well as associated proactive learning algorithm design (Dalvi et al. 2004; Lowd and Meek 2005a; Vorobeychik and Li 2014; Li and Vorobeychik 2014; 2015) In these highly stylized models, a spammer is typically viewed as aiming to minimize the number of edits to an "ideal" spam email template (for example, because modifications to the ideal instance adversely affect the associated response rate), subject to a hard constraint that the email evades the filter. While much progress has been made in considering an explicit model of spammer evasion, the central limitation of all of this literature is that the models are not grounded in actual spammer behavior.

To address this limitation, we present the first human subject study of adversarial evasion of a classification-based spam filter. In our experiments, 265 human subjects, recruited using Amazon Mechanical Turk, each faced a task of editing an initial spam or phishing email template in order to achieve two objectives: first, bypass a filter, and second, remain close to the original. Our treatments were explicitly designed to investigate two hypotheses. The first hypothesis was that adding a small amount of noise to filtering decisions significantly reduces the subjects' ability to evade it. While prior work exists investigating the design of optimal ran-

domization schemes in adversarial settings (Paruchuri et al. 2008; Pita et al. 2009), including work involving human subjects (e.g., (Pita et al. 2010)), ours is the first to investigate how randomization affects decision efficacy, and is also the first to consider randomization in experimental investigation of spam filter evasion. The second hypothesis was that measuring distance to the original in a way that does not penalize the subjects for making word substitutions, such as using synonyms, will significantly improve their performance. We find strong support for the first hypothesis, whereas, somewhat surprisingly, the support for the second hypothesis is mixed. In particular, we observe that randomized filtering significantly reduces the ability of subjects to evade, largely because it significantly increases the fraction of times that participants' submissions were filtered by the system. An additional finding of great practical importance is that increasing the fraction of words in an original email that are included as features in the classifier (filter) also increases the difficulty of the associated task. One of the core guiding principles in applied machine learning is that one should keep the number of features used as small as possible. Our finding, in contrast, suggests that when faced with an adversarial classification problem, such as spam filtering, limiting the number of features can actually make the evasion problem easier.

In addition to our experimental findings, we developed a synthetic model of evasion behavior calibrated using the experimental data. We demonstrate that our model effectively predicts both individual-level behavior, as well as aggregate experimental quantities, allowing us to successfully replicate the experimental findings in simulation. Our synthetic model can thus be utilized in follow-up development of proactive spam filtering methods that explicitly account for human evasion behavior.

## Preliminaries

Before we can properly describe and motivate the experimental setup, we begin by considering in the abstract the problem of *adversarial classifier evasion*, which has been previously studied in several forms from a computational standpoint (Lowd and Meek 2005b; 2005a; Vorobeychik and Li 2014; Li and Vorobeychik 2014). Spam and phishing email filtering is a special case: when machine learning is used to filter spam emails, the spammer changes the email structure (template) in order to evade the filter to ensure successful delivery of the email. The crucial challenge is that evasion alone is not sufficient: one needs to ensure that the original purpose of the email is still fulfilled. Next, we describe how this tradeoff can be quantified.

An instance of interest, such as an email (which may or may not be spam) is represented as a vector of binary features, $x = \{x_1, \ldots, x_n\}$. The corresponding label is encoded as either $+1$, signifying a malicious instance (spam, for example), or $-1$, signifying a normal or benign instance (a regular email). In using machine learning for spam/phishing detection, we start with a training data set of instances $\{(x^i, y^i)\}$ where $x^i$ are the feature vectors and $y^i$ are the corresponding labels, and train a classifier, $g(x)$, which predicts a label for an arbitrary feature vector (email) $x$. Many techniques have been proposed for this problem. For our purposes, we used a Naive Bayes classifier, and used 500 features (words) that had the highest (R)elative (F)requency in spam / non-spam emails respectively. The relative frequency for each word $i$ is defined as $|RF_{-1}(i) - RF_{+1}(i)|$, where $RF_C(i)$ is the relative frequency that word $i$ appears in instances $x$ in class $C$. We used TREC data to train the classifier with the average accuracy of $\sim 91\%$ in five-fold cross-validation.

Given a classifier, $g(x)$, adversarial evasion is commonly modeled as (an adversary) choosing another feature vector $x'$ (for example, a modified spam email template) which is classified as benign and is as close to the original instance $x$ as possible (Lowd and Meek 2005a; Nelson et al. 2012; Li and Vorobeychik 2014). Formally, let $c(x, x')$ be a cost function representing the loss sustained by the adversary from choosing $x'$ instead of $x$. The adversary is solving the following optimization problem:

$$\min_{x' | g(x') = -1} c(x, x').$$

The simplest way to measure this cost is by using a norm, commonly, $l_1$, giving rise to the following cost function:

$$S_1 : \; c(x, x') = \sum_i |x_i - {x_i}'|,$$

where $i$ ranges over the features. In recent work, this cost function has been criticized on the grounds that it penalizes for substitutions among equivalent features (for example, synonyms or 1-letter substitutions in words) (Li and Vorobeychik 2014). This work proposed an alternative cost function defined as follows:

$$S_2 : \; c(x, x') = \sum_i \min_{j \in F_i | {x_j}' \oplus x_j = 1} |x_j - {x_i}'|,$$

where $x_i$ denotes the i*th* feature within the instance $x$; $F_i$ is the equivalence class (i.e., the set of equivalent features) for a feature $x_i$; and $\oplus$ represents the exclusive-or to guarantee that the features are substituted instead of only being deleted. In our experiments we defined the equivalence class of a word to be its synonyms (evaluated using the semantic dictionary *WordNet* (Miller et al. 1990)) and 1- and 2-character substitutions.

## Experiment Design

Since we intend to study adversarial evasion of spam filters which use classification learning, our ideal source of subjects is spammers or phishers.[1] It is clearly infeasible to obtain enough subjects from this population for an experiment. As a proxy, we use human subjects recruited using Amazon Mechanical Turk, a popular crowd-sourcing platform that is commonly utilized by behavioral science researchers

[1] We are assuming that spam filter evasion, and, indeed, evasion of machine learning methods in general, is largely a manual process. To our knowledge, no general-purpose tools exist to automate such a process, although *theoretical models* of spammer evasion have received some attention, as described above.

to recruit and pay human subjects (Von Ahn and Dabbish 2008). While not ideal, there is now substantial evidence that results from the experiments using Amazon Mechanical Turk are often indistinguishable from those found in physical laboratories (Horton, Rand, and Zeckhauser 2011; Suri and Watts 2011). To collect data, we built a Rails application and ran it on the Amazon Web Services EC2, while storing data on Amazon Web Services RDS. In all, we recruited 265 participants for the study who have jointly completed 482 tasks (described below), with each subject performing at most two tasks, both necessarily distinct. No task could be repeated by the same subject.

After signing up for the experiment, each participant received a simple and brief English language test (see the Supplement for details).[2] Passing this test qualified them for participation in the experiment. At this point, subjects were invited to read the tutorial describing the experimental setup (see the Supplement for details). A participant was randomly assigned two tasks (corresponding to experimental treatments). Each task entailed a sequence of 20 submissions of manipulated instances of an "ideal" email by the subjects.[3] For each submission the subjects saw an interface similar to the one shown in Figure 1.

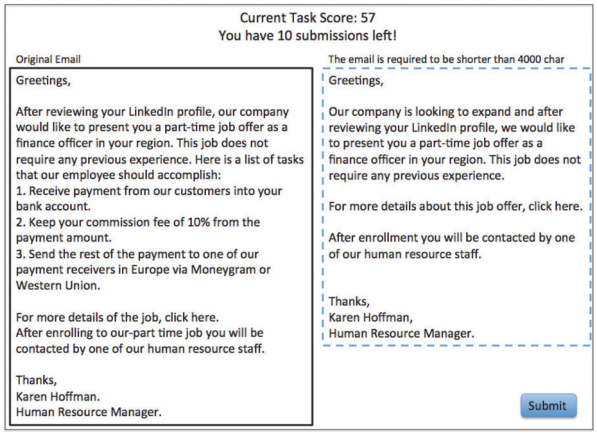


Figure 1: Example interface. The left window is the original or "ideal" email. The right window is a free text form for the submission. Once the participant has written the new email (which may involve copying and pasting portions of the original on the left), they click "Submit" to submit it to the system for scoring. In this case, the prior submission bypassed the filter, receiving a score of 57. The participant can make at most 20 submissions. In this example, the participant has made 10 submissions thus far, with 10 more remaining.

Each task/treatment included three randomly generated pieces:

1. An "ideal" email instance, visible to the subjects,
2. a classifier filtering submissions, not visible to the subjects, and
3. a scoring function, not visible to the subjects.

An "ideal" email instance corresponds in the evasion model above to the ideal feature vector $x$; in order to maximize their score, the subjects had to craft an email which was close to this ideal. Ideal instances for the experiments were chosen from 10 pre-selected spam and phishing emails (see the Supplement for details). The actual classifier used in each task was fixed to the model described above. What we varied was whether or not noise was added to a classification output $g(x)$ for a given instance (submission) $x$. Specifically, in the "noise" treatment the output of the classifier was flipped with 10% probability. The baseline (no noise) treatment, on the other hand, used the classifier as is for all submissions. Finally, the scoring function was chosen uniformly at random between $S_1$ and $S_2$ described above. The subjects were not initially told the specifics of the classifier (nor whether or not it included noise) or of the scoring function (they could request this information after the experiment).

After each submission, a participant received immediate feedback about whether or not the submission was filtered (filtered submissions received zero score), and, if it was not filtered, the score of their last submission, as well as their current best score. For a given submission $e$, if the cost function (randomly chosen for that task) evaluated to $c$, the score was calculated as $r(e) = 100 \times e^{-1.5c}$. The best score obtained over all 20 submissions, $r_{max}$, was used to determine their payment. Specifically, payment for a task was in the range $\$1 - \$5$, with the actual payment computed as $\$1 + 0.04 \times r_{max}$.

Because individual tasks elicited highly divergent performance by the subjects, our reported results use *normalized* scores, $S(t,i) = 90\frac{r(t)-low(i)}{high(i)-low(i)} + 10$, where $high(i)$ and $low(i)$ represents the highest and lowest score among all submissions for task $i$, respectively.

## Results

### Randomization makes evasion more difficult

Since we randomly assigned subjects to randomized vs. non-randomized classifier treatments, we are able to definitively establish the impact that adding a small amount of noise to the filter has on the ability of subjects to evade it. Table 1 supports the hypothesis that adding noise to a filter reduces the effectiveness of human subjects to evade it.

| Category | $S_1$ | $S_2$ | Overall Average |
|---|---|---|---|
| Noise-Free Filter | 24.02 | 30.06 | 26.87 |
| Noisy Filter | 19.98 | 20.58 | 20.29 |

Table 1: Average subject scores for the two scoring functions in the noise-free and noisy filter treatments.

In particular, the overall average for the noise-free filter is over 30% higher than when noise is added. This result is

[2]Online supplement can be found at http://vorobeychik.com/2016/evasionSupplement.pdf.

[3]The first five submissions were "trials" and were not used towards calculating the final score.

statistically significant ($p < 0.01$). Moreover, the result remains significant for the two scoring functions individually. In addition, we find that the scoring function $S_2$ which does not penalize for substitutions tends to yield higher scores for the subjects. This effect is substantial (25% improvement in average normalized score) and significant ($p < 0.01$) in the noise-free filter treatments. Surprisingly, the effect is quite small (3%) and not statistically significant when noise is present. Next we delve deeper in the details of participant behavior to try to shed some light on these results.

The most natural hypothesis into why randomization has a significant impact on the ability to evade the filter is that it makes the task of learning how the filter operates significantly more challenging. The subject behavior bears this out: 77% of submissions were filtered in the noise-free environment, compared to 82% when noise was present (the comparison is significant with $p < 0.01$). Expanding this result by submission (Figure 2), we can observe that not only does the noise-free environment appear to promote much faster learning of how to evade a classifier by the subjects, but the relative difference appears to increase with experience (even as there is clear indication of learning to evade in both cases). Interestingly, we did not observe a similar

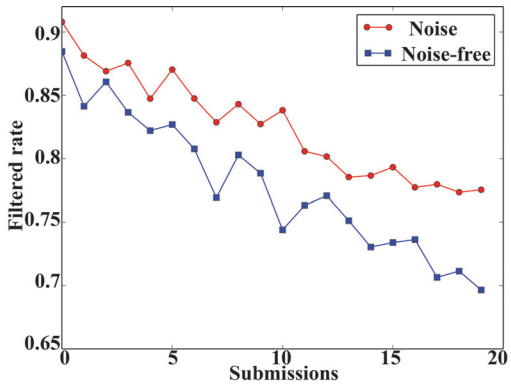


Figure 2: Fraction of submissions that were filtered in the noise-free setting (the blue lower line) and when noise was randomly added to filter output (the red higher line).

pattern when it came to scores received on non-filtered submissions (filtered submissions, of course, received a score of zero) or in time taken to make a decisions: in both cases, differences were not significant between the noise-free and noisy settings, and there was no clear long-term trend. Thus, for example, there was little evidence that scores earned on non-filtered submissions improved with experience (see the Supplement for details). Similarly, while the first few submissions took longer than others on average, thereafter the overall trend with experience was minimal (see the Supplement for details). To sum up, it appears that the reduced ability to successfully evade the classifier was the primary effect of randomization on evasion performance.

To further investigate the source of the difficulty that participants faced in the evasion task under randomization, we consider a simple logistic regression model in which the output is the probability that a submission is filtered in the noisy setting. In this model we consider four variables: a binary indicator whether or not a previous submission was flipped (due to noise), the number of prior submission that were flipped, the submission number (higher number indicates a later submission and thereby indicates greater experience), and feature density (fraction of the words in the "ideal" email that are features in the classification-based email filter). The results are shown in Table 2. Here $x_1$ is a binary

| | Coefficient | Std Error | $z$ | p-value |
|---|---|---|---|---|
| $x_1$ | 3.11 | 0.15 | 21 | $< 0.01$ |
| $x_2$ | 0.38 | 0.025 | 15 | $< 0.01$ |
| $x_3$ | -0.39 | 0.021 | -18 | $< 0.01$ |
| $x_4$ | 7.85 | 1.11 | 7.1 | $< 0.01$ |
| constant | -1.53 | 0.33 | -4.7 | $< 0.01$ |

Table 2: Results of the logistic regression model for the probability of a filtered submission.

indicator whether or not a previous submission was flipped (due to noise). $x_2$ is the number of prior submission that were flipped. $x_3$ is the submission number. $x_4$ is the feature density (fraction of the words in the "ideal" email that are features in the classification-based email filter). While all coefficients are highly significant, of particular interest are the first two: whether the prior submission was flipped, and the number of previously flipped submissions. The former increases log-odds by 3.11, or increasing the odds ratio (probability of failed evasion divided by probability of success) by more than 20. Thus, it is quite clear that adding a noise to a particular submission results in a substantial short-term impact on the ability to evade a classifier. In addition, there is a significant longer-term impact: each flip increases the odds ratio of failure to success for any subsequent submission by $\sim$1.5.

Next, we turn to the issue of the impact that a substitution-aware scoring function ($S_2$) has on performance by considering two micro-level manipulations that participants performed to the "ideal" email as well as subsequent submissions: word additions and deletions. We first look at how these evolved over time.

Figure 3 shows that the first several submissions exhibit considerable manipulation, but after the third submission, the numbers of both additions and deletions remains relatively stable, with slightly fewer words added than deleted in each submission except the first few.

Overall, we found the number of additions to be higher than deletions ($p < 0.01$). We can also observe that additions comprise a significant fraction of the previous submission, in many cases over 40% of the content. In contrast, the fraction of words deleted was typically around 20%. (The rest of the content is, of course, unchanged from the previous submission). Taking the deletions and additions together, we next compare the total number of *edits* for the noise-free and noisy filter settings. When no noise is added to the filter, the subjects made, on average, nearly 48 edits per submission. In contrast, when noise was present, this number dropped to 45.86 (the difference is significant with $p < 0.01$): somewhat surprisingly, this indicates that the

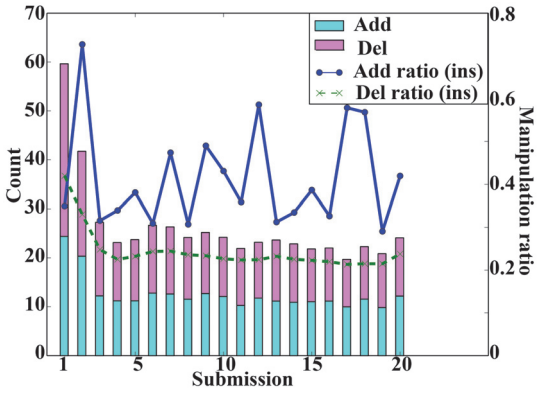


Figure 3: Additions and deletions of words to prior submission over time (i.e., over a sequence of submissions). The number of additions is the lower light blue bar, and the heavy blue line represents additions as a fraction of total number of words in the prior submission. The number of deletions is the purple bar at the top, and the dashed green line corresponds to the deletions as a fraction of words in the prior submission.

participants were engaged is slightly less exploration in the noisy-filter treatments, accounting to some extent for the fact that they had significantly more trouble evading the filter in this setting. Moreover, we found that the total fraction of edits which involved substitutions was significantly higher in the noise-free treatment (0.52 vs. 0.47; $p < 0.01$). Again, this points to significantly increased activity and engagement by the subjects in the noise-free treatment (and, perhaps, less confusion about how to effectively manipulate the template). In addition, this helps explain the surprising result that the difference between $S_2$ and $S_1$ treatments was much smaller when noise was present: the substitution-aware scoring function $S_2$ will reward subjects especially for using substitutions in their manipulations, and these were far more prominent under the noise-free treatment.

While substitutions clearly played an important role in subject behavior under all treatments, the composition was somewhat surprising: synonyms accounted for less than 3% of all edits, as compared to character substitutions, which amounted to over 45% of edits. Although some of the character substitutions may have been incidental, rather than deliberate (many words are only a few characters apart), many were clearly deliberate; for example, over 22% of 2-character substitutions cannot be found in the dictionary, suggesting that subjects deliberately misspelled words to evade the classifier.

## Positive feedback improves engagement in the task

Next we investigate the extent to which receiving a positive feedback on prior submission impacts human subject performance. We quantify feedback as the difference between the score for prior submission $r(e_{i+1})$ and the score received for the immediately preceding submission $r(e_i)$. We say that feedback is *positive* when this difference is positive ($r(e_{i+1}) - r(e_i) > 0$), and it is *negative* when this difference is negative ($r(e_{i+1}) - r(e_i) < 0$). When these scores are both zero, we say that feedback is *null*, whereas when they are both equal and positive ($r(e_{i+1}) = r(e_i) > 0$), we call it *equal*. We find that receiving positive feedback on prior submission leads subjects to spend *more time* on the following submission ($p < 0.01$; see Figure 4), and obtain a higher score. In contrast, receiving null or equal feedback leads to less time spent on the following submission. This observation is quite surprising: one would think that particularly null feedback (corresponding to several filtered submissions in a row) would cause the subjects to spend more time contemplating a better evasion strategy, but we see the opposite. Interestingly, our findings are consistent with Mason and Watts (2012). Briefly, Mason and Watts consider a problem of exploring a complex landscape in human subject experiments on a network, where participants could observe what their network neighbors have found. One of their key findings is that when a subject's neighbors find good solutions, the subject engages in significantly *more* exploration of the landscape. Taken together, their findings and ours suggest that positive feedback serves as an important psychological motivator of engagement in a task, which in turn improves performance.

## Feature reduction makes adversarial evasion easier

There was significant variability among the 10 tasks (original "ideal" emails) in terms of apparent difficulty of evasion. To understand the source of this variability, we consider the relationship between *feature density*, or the fraction of words in the ideal instance which are used as features in the classifier (filter), and average as well as maximum score for the task.

Figure 4 shows that higher feature density leads to a lower score, appearing to make evasion more difficult for the subjects. Table 2, which features a logistic regression model of

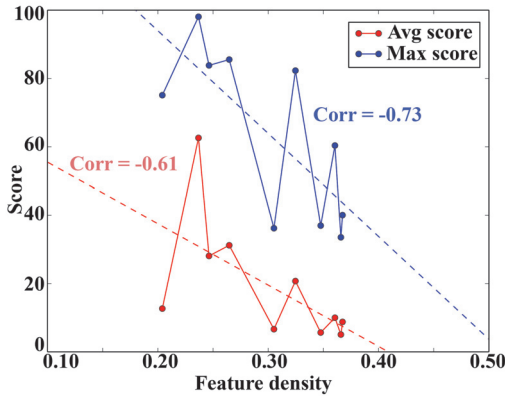


Figure 4: The relationship between maximum and average score for each task and the feature density of the corresponding "ideal" email. Feature density is the fraction of words in the ideal email that correspond to features in the classifier (filter).

the probability that a submission is filtered, offers stronger

evidence: the coefficient corresponding to feature density is again high and significant (and with submission-level granularity we now have much more data to justify this conclusion): it seems clear that increasing feature density makes adversarial evasion far more challenging.

The observation that increased feature density increases difficulty of evasion has important ramifications for the use of machine learning tools in spam/phish filtering tasks specifically, and intrusion detection more generally. One of the most important "best practice" principles in applied machine learning is feature reduction, or limiting the total number of features used, for example, through the use of regularization or other means of feature selection (Bishop 2007). Clearly, when we reduce the size of the feature set used for filtering, we also reduce feature coverage of spam instances, that is, we reduce feature density. Our observation here suggests, therefore, that feature reduction can make adversarial evasion easier.

### Synthetic model of human evasion behavior

One of our motivations for engaging in human subject evasion experiments was to develop and calibrate a synthetic model of evasion dynamics. We propose the following composite model of behavior:

1. For each feature predict (independently of other features) whether its value is changed, and
2. For each feature word which is predicted to be deleted, predict whether it is substituted for (we assume that substitutions are chosen outside of the feature vector).

For task (1) we develop $n$ independent dynamical models, one corresponding to each binary feature of the classifier, to predict the evolution of the feature vector with the sequence of individual submissions using a Support Vector Machine (SVM) (Bishop 2007). Features used for this prediction task were taken from a combination of features for the previous two submissions, demographics, as well as feedback, including previous score and whether or not the prior submissions were filtered. For task (2) we developed an independent SVM model for each deleted word predicting occurrence of a substitution. Our models were able to recover underlying behavior with high accuracy in cross-validation (average accuracy was over $97\%$). However, this in itself is an insufficient criterion for our purposes: a useful dynamic model of behavior should also successfully replicate the experimental findings described earlier, both qualitatively and quantitatively. To exhibit that the model successfully does so, Table 3

| Category | $S_1$ | $S_2$ | Overall Average |
|---|---|---|---|
| Noise-Free Filter | 28.63 | 31.61 | 29.94 |
| Noisy Filter | 20.51 | 21.08 | 20.75 |

Table 3: Predicted average subject scores for the two scoring functions in the noise-free and noisy filter treatments.

shows the predicted results of our 2x2 treatment (noise-free vs. noisy filter, $S_1$ vs. $S_2$ scoring). To generate these results, the synthetic model was run starting from the ideal email instance, where features were deterministically chosen to be flipped in each iteration based on the model prediction. This process results in a sequence of feature vectors generated by the model, which were then scored as described above to obtain the values in the table. Comparison to Table 1 suggests that the results of the synthetic model closely mirror actual observations of subject behavior in the experiment. In addition, we compare in Figure 5 predicted and actual average normalized scores by submission for randomized (noise-free) and non-randomized filters. The predicted and

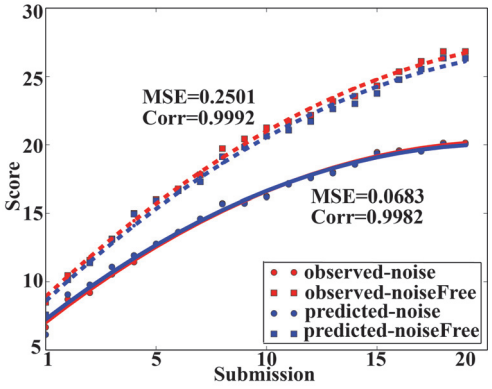


Figure 5: Comparison between experimentally observed scores and those based on the synthetic model of behavior as a function of the submission sequence for noisy and noise-free settings.

observed average scores match rather closely (nearly perfect correlation and small mean-squared error), and the expected gap between randomized and non-randomized is clearly visible in simulated behavior just as it is in real.

## Discussion

Machine learning algorithms have come to be used widely in settings which involve inherently adversarial interactions. One of the most important such settings is spam and phishing email filtering, where evasion has been an important and well documented consideration (Goodman, Cormack, and Heckerman 2007; Colbaugh and Glass 2012). Our results offer both qualitative practical guidance for the design of robust email filtering systems based on machine learning methods, and explore human behavior in the context of adversarial interaction with a machine learning system. In particular, we can offer several practical pieces of advice: first, filtering systems should embed a small amount of noise in order to increase the level of difficulty for spammers and phishers in designing templates to evade these systems; and second, special care must be taken in training classifiers for such systems to not be overly aggressive in removing "unnecessary" features. Of course, in both cases one should not over-compensate: introducing noise into classification decisions necessarily decreases accuracy, and the system designer must balance this degradation with increased robustness to adversarial evasion. Similarly, while our experiments demonstrate the perils of reducing the feature space too aggressively, one clearly must also consider the traditional is-

sues of overfitting in designing robust and effective learning systems in adversarial environments.

From the perspective of individual behavior, our findings reinforce a surprising finding due to Mason and Watts (2012) that observation of good outcomes (we call this positive feedback) increases individual engagement in the task. Indeed, Mason and Watts contextualized this finding entirely in a social network setting, whereas our results suggest that this phenomenon is more fundamental.

Finally, our results have significant bearing on the substantial literature preoccupied with designing high-quality randomization schemes in security (Paruchuri et al. 2008; Pita et al. 2009; 2010; Jajodia et al. 2012) and machine learning (Vorobeychik and Li 2014; Li and Vorobeychik 2015). In much of this work, randomization schemes are developed under the assumption that the adversary is fully rational. Our experiments demonstrate that randomization takes on additional importance with a human adversary. In particular, the introduction of noise appears to significantly hamper the ability of the subjects to learn how to evade the classification-based filter. Moreover, the effect of noise is not merely short-term (immediately following the noisy feedback) but has a significant lasting impact on performance well after the perturbation had been introduced.

## Acknowledgments

This was was partially supported by the NSF (CNS-1238959, IIS-1526860), ONR (N00014-15-1-2621), AFRL (FA8750-14-2-0180), Sandia National Laboratories, and Symantec Labs Graduate Research Fellowship.

## References

Bishop, C. 2007. *Pattern Recognition and Machine Learning*. Springer.

Carreras, X., and Marquez, L. 2001. Boosting trees for anti-spam email filtering. *arXiv preprint cs/0109015*.

Colbaugh, R., and Glass, K. 2012. Predictive defense against evolving adversaries. In *IEEE International Conference on Intelligence and Security Informatics*, 18–23.

Dalvi, N.; Domingos, P.; Sanghai, S.; Verma, D.; et al. 2004. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 99–108. ACM.

Goodman, J.; Cormack, G. V.; and Heckerman, D. 2007. Spam and the ongoing battle for the inbox. *Communications of the ACM* 50(2):25–33.

Horton, J. J.; Rand, D. G.; and Zeckhauser, R. J. 2011. The online laboratory: Conducting experiments in a real labor market. *Experimental Economics* 14(3):399–425.

Jajodia, S.; Ghosh, A. K.; Subrahmanian, V.; Swarup, V.; Wang, C.; and Wang, X. S., eds. 2012. *Moving Target Defense II: Application of Game Theory and Adversarial Modeling*. Springer.

Li, B., and Vorobeychik, Y. 2014. Feature cross-substitution in adversarial classification. In *Advances in Neural Information Processing Systems*, 2087–2095.

Li, B., and Vorobeychik, Y. 2015. Scalable optimization of randomized operational decisions in adversarial classification settings. In *International Conference on Artificial Intelligence and Statistics*. to appear.

Lowd, D., and Meek, C. 2005a. Adversarial learning. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, 641–647. ACM.

Lowd, D., and Meek, C. 2005b. Good word attacks on statistical spam filters. In *CEAS*.

Mason, W., and Watts, D. J. 2012. Collaborative learning in networks. *Proceedings of the National Academy of Sciences* 109(3):764–769.

Mason, J. 2004. The spamassassin homepage. *http. V/Spamassassin. org/index. html*.

Miller, G. A.; Beckwith, R.; Fellbaum, C.; Gross, D.; and Miller, K. J. 1990. Introduction to wordnet: An on-line lexical database*. *International journal of lexicography* 3(4):235–244.

Nelson, B.; Rubinstein, B.; Huang, L.; Joseph, A.; Lee, S.; Rao, S.; and Tygar, J. D. 2012. Query strategies for evading convex-inducing classifiers. *Journal of Machine Learning Research* 13:1293–1332.

Nie, N. H.; Simpser, A.; Stepanikova, I.; and Zheng, L. 2005. Ten years after the birth of the internet, how do americans use the internet in their daily lives. *Stanford Institute for the Quantitative Study of Society*.

Paruchuri, P.; Pearce, J. P.; Marecki, J.; Tambe, M.; Ordóñez, F.; and Kraus, S. 2008. Playing games with security: An efficient exact algorithm for Bayesian Stackelberg games. In *Proceedings of the Seventh International Conference on Autonomous Agents and Multiagent Systems*, 895–902.

Pita, J.; Jain, M.; Ordóñez, F.; Portway, C.; Tambe, M.; Western, C.; Paruchuri, P.; and Kraus, S. 2009. Using game theory for los angeles airport security. *AI Magazine* 30(1):43–57.

Pita, J.; Jain, M.; Ordonez, F.; Tambe, M.; and Kraus, S. 2010. Robust solutions to stackelberg games: Addressing bounded rationality and limited observations in human cognition. *Artificial Intelligence Journal* 174(15):1142–1171.

Sakkis, G.; Androutsopoulos, I.; Paliouras, G.; Karkaletsis, V.; Spyropoulos, C. D.; and Stamatopoulos, P. 2003. A memory-based approach to anti-spam filtering for mailing lists. *Information Retrieval* 6(1):49–73.

Suri, S., and Watts, D. J. 2011. Cooperation and contagion in web-based, networked public goods experiments. *PLoS One* 6(3):e16836.

Von Ahn, L., and Dabbish, L. 2008. Designing games with a purpose. *Communications of the ACM* 51(8):58–67.

Vorobeychik, Y., and Li, B. 2014. Optimal randomized classification in adversarial settings. In *International Joint Conference on Autonomous Agents and Multiagent Systems*, 485–492.

Zeller Jr, T. 2005. Law barring junk e-mail allows a flood instead. *The New York Times* 1.