Detecting Employee Misconduct and Malfeasance

George Valkanas
Detectica, Inc.
george@detectica.com

Panos Ipeirotis
Detectica, Inc. and New York University

Foster Provost
Detectica, Inc. and New York University

Josh Attenberg
Detectica, Inc.

Jennifer Chin
Detectica, Inc.

Chathra Hendahewa
Detectica, Inc.

Abe Stanway
Detectica, Inc.

Bernando Suryanto
Detectica, Inc.

Bharath Vivekananda Swamy
Detectica, Inc.

ABSTRACT

In the United States financial firms have the regulatory obligation to monitor the communications of their employees (e.g., emails, chats, phone calls) in order to detect misconduct. Some forms of misconduct are illegal activities (e.g., insider trading, bribery) while others are policy violations (e.g., improper security practices, or inappropriate language use). Traditionally, firms have deployed relatively simple rule-based systems for employee surveillance. Such systems generate many false positive alerts and are hard to adapt to the changing environment. Recently, firms have attempted to improve their systems by transitioning from the rule-based techniques to statistical machine learning approaches. However, they still treat the problem of misconduct detection as a single-document classification problem. We present an approach that focuses on actors, connections among actors, and on *cases* of misconduct. Furthermore, we highlight the importance of having a "human-in-the-loop" approach, where humans are both guided by and guide the system at the same time, in order to detect malfeasance faster and to adapt to changing environments. We also discuss how humans can play a key role for detecting shortcomings of existing machine-learning-based malfeasance-detection systems. Our multifaceted approach has been developed and tested in real environments within both massive and smaller financial institutions, and we discuss practical constraints and lessons learned.[1]

ACM Reference format:
George Valkanas, Panos Ipeirotis, Foster Provost, Josh Attenberg, Jennifer Chin, Chathra Hendahewa, Abe Stanway, Bernando Suryanto, and Bharath Vivekananda Swamy. 2018. Detecting Employee Misconduct and Malfeasance. In *Proceedings of WSDM workshop on Misinformation and Misbehavior Mining on the Web, Marina Del Rey, CA, USA, 2018 (MIS2),* 8 pages.
https://doi.org/10.475/123_4

[1]This work was done while all authors were at Detectica, Inc.

MIS2, 2018, Marina Del Rey, CA, USA

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

1 INTRODUCTION

Financial institutions have been the focus of increased regulatory scrutiny over the last decade, with *more than $321 billion dollars*[2] paid in fines, mainly due to problematic and unethical behavior of the firms' employees. One of the key problems that causes these fines is the lack of effective tools to monitor employees' actions, which permits employees to misbehave without being detected. Firms in the financial industry have the obligation to monitor their employees and investigate any potential cases of malfeasance. The types of malfeasance can vary, from bribery and insider trading, to improper promises to clients, violations of information security, to sexual harassment. Unfortunately, the existing systems in place to detect malfeasance still rely on archaic, rule-based technology that generates massive numbers of false alerts and places a heavy burden on human reviewers to reason from alerts on low-level events (e.g., emails) to inferences about the presence of malfeasance.

Recent efforts attempting to replace the rule-based systems with statistical machine learning approaches have had limited success. The key problem is that such systems replicate the approach of the rule-based systems, which are *event focused*—they attempt to classify individual observations of events as malfeasance. Events here could be particular emails, chats, calls, trades, etc. Unfortunately, individual events are rarely sufficient, even for a human, to determine whether malfeasance has occurred.

We take a holistic approach, focusing on compiling and analyzing *cases* of misconduct, including *events*, *actors*, *other entities* (such as companies), and *connections*. Our system ingests digital interactions of employees and systematically analyzes and aggregates them to construct the cases, which are scored and ranked in order of importance. Processing cases, instead of individual events, reduces the effort humans must invest to decide whether there is sufficient evidence to warrant escalation, whether the case requires further evidence gathering, whether it should "brew" for a while longer, or whether it is something innocuous that can be closed without further action.

The usual "label-cases-and-then-learn-models" paradigm tends to produce unsatisfactory results in this domain, due to: the extreme rarity of malfeasance, the lack of a true *instance* on which to base training cases, the non-self-revealing nature of the problem, and the high level of domain knowledge required. Thus, our approach (and philosophy) is that the best training information can be obtained

[2]http://bit.ly/regtech-fines

by observing the investigators work, following a "human-in-the-loop" paradigm, where humans are both guided by and guide the system at the same time. The training information goes beyond the labeling of cases (e.g., "good case", "bad case"), which now can be done by observing the final dispositions of the cases. We also can examine the level of activity on a case, as a proxy for interest or importance. Moreover, humans are better suited to find cases of malfeasance, when provided with good tools [2], and can find important instances that machine learning algorithms miss [1]. Actively engaging them and making them an integral part of the overall process can substantially improve the quality of the resultant system.

Overall, we make the following contributions:

- We present a multifaceted human+machine-learning system, to identify cases of employee misconduct. We discuss knowledge engineering, graph analysis, and actor analysis.
- We present lessons learned from building systems from the ground up for a massive international bank and for smaller financial institutions.
- We provide some preliminary results of our system's performance compared to other alternatives.

2 RELATED WORK

Finding misconduct in digital environments has long been of interest to researchers and practitioners from both academia and industry. Fraud detection was one of the earliest commercial applications of sophisticated machine learning, where systems already were moving from event-based detection to richer case-based detection based on multi-modal sources of evidence [5, 15, 31].

Fraud detection work in telecommunications introduced the notion that network analysis can play a substantial role [9, 15]. Network-based techniques, applied on digital interactions, have also been used to identify suspicious individuals or entities in online auctions [29], the online advertising ecosystem [35], the business world [22], and the physical world [24]. Identifying spam sites to improve web search results has been the focus of the TrustRank algorithm [20] and follow-up topical variations [36]. Our system incorporates network analysis; however, our network consists of firm employees, and thus the network incorporates real-world social and professional relationships. Therefore, one of our contributions is to evaluate the usefulness of socio-professional networks to locate suspicious individuals in a corporate setting.

Interest in network-based approaches has been invigorated with the proliferation of social media [19, 37], as the role of users has shifted from passive content consumers to that of active producers. Improved quality can be achieved by knowing about the diffusion process of information posted online, which is the focus of [17].

Social media researchers have also worked on behavioral misconduct between users. In particular, *cyberbullying*, which falls on spectrum of harassment and takes place in the online world, has drawn attention in recent years [7, 33]. The solutions typically involve learning a machine learning model, trained on a large set of online labeled data. Despite the common goal, harassment is only one of the types of misconduct that we are targeting. Furthermore, and though we cannot preclude cyberbullying, the professional environment we operate in drastically reduces the possibility of such phenomena. The fact that in our domain the amount of labelled data that are available is small, compared to what these authors gathered in their research, is partial testament to the rarity of such incidents in the corporate world.

3 A MULTI-FACETED APPROACH

Before diving into the specifics of our design and system, we provide some context. As already mentioned, we target misconduct of employees working in a financial environment. This ranges from (sexual) harassment to leaking sensitive data to attempts at market manipulation or insider trading. In simple terms, we are interested in detecting any inappropriate behavior for which the institution needs to take corrective action.

To that end, our system ingests and systematically analyzes digital communications between employees, who are knowingly under surveillance. Such communications come in several formats, including email, instant messaging and chats through finance-specific terminals. Some of these communications come in near real-time and are full of abbreviations and slang (chat terminals), as they are used among traders and brokers to make and close deals (for example). Others can be more conversational or instructional / informational, with (more) normal language. In addition to the content, each communication comes with associated metadata, such as timestamp, communication channel and thread it belongs to (or chat room, for example). Participating individuals and their roles (sender, direct recipient, bcc recipient) are also specified. The system also has access to information from the Human Resources directory, such as an employee's job title and role, work division, (re)hire date(s), position in the org chart, and so on.

We also have a small amount—relative to total volume—of communications that have been reviewed in the past by analysts after our system or an existing, rule-based system has flagged them as being suspicious. The reviewers may have closed them out as being non-suspicious or passed them along for additional action, after reviewing the communication and other supplementary information to which they have access. We augment this data with additional data and evidence structures that we have designed and which we present next. Later we discuss how we use the evidence.

3.1 Knowledge Engineering

Financial institutions operate within a regulatory framework. To comply with surveillance and supervisory regulations, they rely on the domain knowledge of traders and analysts. Existing systems encode some of this domain knowledge in rules using a syntax akin to regular expressions. In one of the most broadly used existing systems, each incoming message is tested against the entire set of expressions, and as long as it reaches a minimum "score", which almost always means that it matches a minimum number of rules—typically 2 or 3—it is presented to an analyst for further investigation.

Rule-based systems have a long history of development and use [6]. For text documents in particular, building rule-based systems via encoding domain knowledge via dictionaries has been standard practice for years. Much of the appeal of rule-based systems is that they are easy to implement, to extend, and (in principle) to explain to a non-technical audience. However, rules crafted by

humans often have disappointing accuracy due to people's inability to assess large-scale statistical consequences. (How often, when looking for an email that you know is there, has your own email search query returned massive numbers of unexpected false positives?) We return to this below.

Moreover, the rule-based approach becomes increasingly costly over time, as maintaining or modifying existing knowledge bases is an arduous task, e.g., avoiding duplicate entries, dealing with the interactions among rules, updating rules as one gathers statistics on their efficacy, and updating the overall model as the world changes. This effect is magnified in an enterprise setting, where multiple—often disagreeing—stakeholder perspectives need to be taken into account and encoded.

Another well-known problem is that for most rule-based systems a rule needs to match *exactly*. The slightest spelling deviation will cause the match to fail and, in our application, the communication will go through the system without raising any flags. This is a major shortcoming, and the "solution" of including spelling variations in the expressions increases the maintenance costs dramatically. Furthermore, the addition of more words often increases the false positive rate of alerted communications.

Last but not least, all phrases in the rule-base may carry the same weight, regardless of the actual likelihood of malfeasance when they match, or the rule weights do not truly correlate with likelihood of malfeasance. The main reasons that the rules (effectively) carry the same weight are (i) that the expressions are designed by humans without the aid of labeled training data, (ii) the system does not have an effective (probabilistic) score-combining capability, (iii) the systems do not collect statistics on their effectiveness, and (iv) even if they did gather such statistics, the systems do not have a machine-learning component to update the scoring accordingly. Manually assigning weights to thousands of phrases involves a huge human-resource overhead, but the problem is much more fundamental than that. While humans are good at generating ideas of what might indicate malfeasance, they are quite poor at estimating the actual likelihood of malfeasance for a given rule and in particular at anticipating the unintended consequences (i.e., false-positive matches) of a rule.

Rules as Background Knowledge for Machine Learning: Rule-based systems certainly have these shortcomings. However, we should be careful not to throw out the baby with the bathwater. The hundreds (or more) of rules in the system exist because of the good-faith efforts of domain experts (in our case, analysts and traders) to tailor the systems to detect malfeasance. Thus, the expressions encode valuable domain knowledge developed over years. In a domain such as this with few positive instances, it would be a tall order for a machine learning system on its own to rediscover all the useful knowledge encoded in the rule base. Even with the aid of knowledge engineers, such knowledge would be difficult for a data science team to obtain from scratch without substantial effort.

Our approach is to use the rule base as prior knowledge on which the machine learning can build. The notion of extracting the knowledge of a rule-based system for use in a machine learning system is not new [12], but we approach the problem differently than the prior work. Our approach is to code each rule as a feature generator. This will allow us both to create a high-fidelity replica of the existing system, in cases where user acceptance requires continuity with prior operations [12], but also to integrate the background knowledge from the rule base with arbitrary additional features, in arbitrary machine learning models.

Based on this approach, our first improvement is to *learn weights* for each rule from our ground truth dataset. In addition, we have introduced a tool that acts as a comprehensive rule-evaluation dashboard. The tool provides information for both old and new rules alike, offering a testbed for trying out new phrases prior to actually adding them to a rule-based system (either traditional or as features for machine learning).

The main focus is:

- The *weight* of the rule, learned from ground truth data. This can be, for example, the logistic regression [10] coefficient.
- The rule's *success rate statistics* [32], also collected from the ground truth information.
- An estimate of the number of messages that the rule will match, collected via statistics on the messages that we have.

The first two values provide quantitative evidence on the effectiveness of the rule. The third acts as a proxy for the amount of work that will be added (reduced) from the inclusion (exclusion) of the rule. (NB: This ignores rule overlap.) Furthermore, given the extremely low base rate of positive instances in the domain, this also acts as a reasonable estimate of the rule's false positive rate.

Sub-Rule Suggestions: Finally, one of our major advances is a novel technique to aid users in updating existing rules. The current approach to rule updating is based primarily on intuition, as with creating new rules. Once written, rules are vetted by expert examination, deployed and evaluated over a (very) long period of time. There are two main ways rules fail: *i*) they are too specific, matching too few (often *zero*) communications, *ii*) they are too general, matching *too many* communications. In addition to generating rule-matching statistics, we introduce a technique to *recommend* contextually similar phrases that can substitute for parts of the original rule. To provide this functionality, after parsing and tokenizing the rules, we utilize distributed neural network embeddings (cf., [26]) to suggest contextually similar phrases that could augment the original version and present these suggestions to the user. We learn the embeddings from a large sample of (many months of) communications data. Our sample contains communications from all channels, i.e., chats, emails, etc.

By looping humans in this process, the rule is screened before moving forward: the embeddings propose contextual alternatives, but not necessarily direct substitutes or synonyms. Therefore, the user can tweak a suggestion prior to accepting it, while filtering out meaningless ones. Importantly, this technique is both *data-driven*, relying on massive data on real communications, and expert-filtered. As language use evolves over time, new embeddings are learned from incoming data, and our suggestions are able to follow the linguistic patterns and trends of the continuously changing environment.

3.2 Actor-Centric Evidence

As discussed at the outset, the business goal is not simply to classify events (such as individual electronic communications), but to monitor for and investigate malfeasance. Treating the problem as email classification leads to poor performance and user satisfaction,

because it is not well aligned with the actual business goal. Our system instead focuses on the problem of building and investigating cases of potential malfeasance, that can incorporate a wide variety of evidence.

One important source of evidence is *information on the individuals involved*. The system can include individuals in a case in different ways, for example, because they were participants in a flagged communication or trade (or are connected to such people, as we will discuss later). We can think of the system as including a dossier on every individual,[3] which contains a variety of information that might be brought to bear to judge the suspiciousness of a case. This information can include details on an individual's role and position in the organization, whether she is working on a private deal (and involving what), her prior history of alerts and escalations, aspects of her personality and psycho-emotional state, etc. Let's spend some time digging into this last category.

According to empirical research on white-collar crimes, some dimensions of personality are more strongly associated with workplace misconduct. In particular, for 3 of the 5 traits of the OCEAN model [14, 18], *conscientiousness, agreeableness* and *emotional stability*, how strongly one exhibits the trait (inversely) correlates with likelihood of workplace misconduct [21]. The OCEAN model is widely accepted as an accurate taxonomy of an individual's personality, and has been validated several times across domains [4]. Similar research in behavioral psychology has shown that employees' reactions are driven by a combination of their personality and external circumstances. For instance, employees who are under stress or feel mistreated appear to have a higher propensity to engage in misconduct [13, 16]. However, the breaking point is different for each individual and depends on personal characteristics.

Our system estimates a user's psycho-emotional state by tracking a set of psychological and emotional categories $\mathcal{E}$, extracted from the individual's communications over time.[4] Examples of these categories are *stress*, *sadness*, *anger* and *joy* to name a few. It is worth noting that researchers in the past have correlated such categories with the way employees collect information within financial firms [34]. Also, these psycho-emotional categories can be manifestations of personality traits; for example, a particular emotional response to a situation can be a facet of *neuroticism*. In our context, communications that exhibit an abnormal degree of emotional charge, conditioned on normal behavior of the user, may be of interest to an analyst.

To estimate a user's psycho-emotional state, we collect the messages $\mathcal{M}_u = \{m_1^u, m_2^u, \cdots\}$ sent out by user (employee) u. We run each message m_j^u through a custom psychometrics extractor, extending the approach in [30]. This yields an associative map per message, linking a psychometric category e to the degree that it (the category) is present in the message m_i^u – for example, the fraction of tokens in the message that belong to category e. Formally

$$PsyEval(m_i^u) = \{(e, s_e) | e \in \mathcal{E}, s_e \in [0, 1]\}$$

We then approximate the user's psychoemotional state by *aggregating* (averaging, taking the max, etc.), for each category separately, the score of all their messages. To reduce computational costs and to better track a user's state, we discretize time into non-overlapping intervals of size Δ. A message m, sent at time $m(t)$, falls in the time slot for which $t \leq m(t) < t + \Delta$. The aggregation step focuses on the messages of the most recent interval, thereby providing the most recent approximation of the user's psychoemotional state. The overall psychoemotional state of a user is given by the respective u_e scores, one per tracked category e, computed as

$$u_e = \frac{\sum_{m_i \in \mathcal{M}_u} PsyEval(m_i).s_e}{|\mathcal{M}_u|}$$

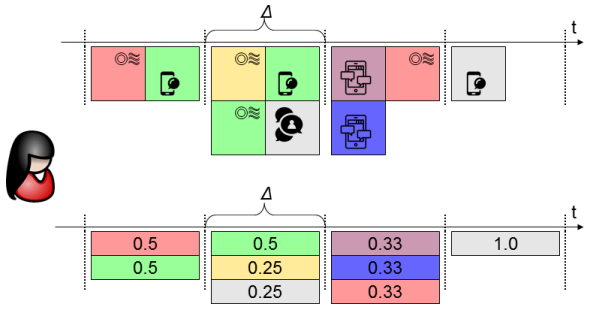


Figure 1: Graphic representation of how psychometrics of an individual are aggregated over time

Figure 1 shows the process for a single user in a simplified setting. The upper part contains the messages (rectangles) sent by a user via several communications channels. Dotted lines illustrate time slots and each message falls into exactly one, yet each slot may contain an arbitrary number of messages. For simplicity in the example, each message is associated entirely with a single category e, i.e., $s_e(m_j^i) = 1$. We use color coding to distinguish between different categories. The resulting psychoemotional state of the user, for each time slot, is shown in the lower part of the figure.

It is important to stress two things. First, a user's psychoemotional state can be useful as evidence in a machine learning framework, but the actual scores should not be interpreted as definitive personality assessments. Analysis of the language used in electronic communications creates *proxies* for personality assessments (as opposed to, say, using standardized personality questionnaires); therefore, the generated scores come with a degree of uncertainty. Second, we advise against making decisions based solely on psychoemotional evidence. The value of such information is to flesh out a rich case, or in extreme cases to initiate an investigation.

Additional Actor-Centric Evidence Psychoemotional analysis can be a useful tool for analysts to review evidence or initiate an investigation. However, as discussed above, this is not the only actor-based information of relevance. To help both the system and the analysts make informed decisions regarding an employee's behavior, we collect as much data as possible at the actor-level, and create other telling evidence structures—such as "prior escalation" and "prior false positive" scores.

[3] As well as other entities, like companies, but we will focus on employees here.
[4] Note that these indicators aggregate across many "events" (communications), similarly to how in fraud-detection systems account-level information can give a more holistic picture than looking only at individual transactions [15].

3.3 Network Approach

Interaction networks have been used in the past to identify individuals associated with suspicious activities in various domains [9, 15, 19, 22, 24, 29, 35–37]. Their common denominator is the reliance on the structural properties of the network, rather than (or in addition to) the properties (e.g., content) of the nodes. The empirical effectiveness of these techniques for other malfeasance detection tasks makes network analysis an important potential evidence source to consider for this application.

Ideally we would know the strength of the socio-professional relationships among all the employees, and then we could derive information such as: how closely related are you to known bad actors? Such guilt by association is the basis of the prior work referenced above on using network analysis for malfeasance. In addition, prior work has shown specifically that bad actors in financial institutions are closely connected in the social network [28]. As a different use of the network analysis, we could augment the socio-professional network among employees with other sorts of entities, such as departments, companies, and even pseudo-entities such as "private information". After such augmentation, we could derive information such as "the individual who executed this suspicious trade is how close to the company whose security was traded?" The result could be a score, which then could be used as a feature in the suspicion model. Alternatively, the result could be a (set of) path(s) showing the strongest connections. For example, "Joe is strongly connected to Jane who is strongly connected to Harry who has been emailing the company in question."

Unfortunately, the actual strength of the socio-professional connection between employees is not available.[5] The communication network among employees could be used to proxy for the socio-professional network. In this case, part of the analytical design would entail deciding on how to estimate the strength of the relationship between two individuals from their communication patterns. Obvious choices would be the number of communications, or the total length of the communications. However, strength of relationship does not necessarily correlate well with raw communication counts—especially communications via work communications systems. The closest friends may not communicate much or at all via work systems.

Instead of using the communications network as a proxy for the relationship network, we use the communications network as a base data source from which to estimate the relationship network. Specifically, we build a computational, generative model of communications given (latent) relationship strength, and then fit the model to an organization's communications data. As we have found in practice, the learned latent relationship strengths can even reveal close friends who never communicate, because of how they are embedded in the social network. (For example, it gives strong relationship strength to traders who sit next to each other, yet never communicate electronically.) In the empirical results, below, we show that guilt-by-association is substantially stronger on the learned latent relationship network than it is on the observed communications network.

[5] One might think about mining proxies for this strength from online social network systems; we take a different approach in this work.

3.4 Explaining the Reasons for "Alerts"

When systems need to be deployed in production, for a variety of reasons the decisions made by the system need to be explained to stakeholders [25]. For example, managers need to sign off on the deployment of the system, analysts can be more efficient and effective if they understand the reasons why an alert is generated, and data scientists can debug the models/knowledge more effectively if they understand why it made the decision(s) that it did.

When a case has enough evidence to present to an analyst, the decision can be explained following the "evidence counterfactual" framework [8, 25]. Specifically, the system can examine internally the evidence present in the case and ask the question: what is a minimal set of evidence, such that if it were not present, then the case would not have been presented? The assumption is that the different pieces of evidence can be interpreted (more or less) by examination: words or phrases in communications, a high stress score, proximity in the network to bad actors, proximity to someone with private information, and so on. Many of these pieces of evidence considered in isolation would not generate sufficient suspicion to create an alert presented to an analyst. Further, given an alert, some evidence might be superfluous. The evidence counterfactual explanations show minimum sets of evidence that were sufficient to increase the case's "suspicion score" above the alerting threshold.

This method of explaining decisions was applied successfully previously for malfeasance detection [22], as well as for finding inappropriate content [25] and other applications [8, 27]. Note that the problem of explaining the reasons for why a *decision* was made is different from the problem of explaining a decision-making model, as has been discussed in detail elsewhere [25].

3.5 Rare Positives and Unknown Unknowns

Our setting exhibits three related problem characteristics that tend to vex AI systems, and especially machine learning systems.
(1) Positive instance are *extremely* rare—far rarer than most machine learning research considers even when addressing "class imbalance." A large financial institution might have a quarter of a million employees. Hopefully only a tiny fraction of them are bad actors! But the "instances" considered by an AI system would be aggregations of employee behavior. If we consider employee-days, then we might have a hundred million or more instances a year, but only a tiny fraction of them would exhibit malfeasance (we hope!). And if we were to consider individual evidence scoring problems, such as individual communications, then the base rate is infinitesimal.
(2) The problem is not "self-revealing," meaning that unlike many machine learning problems, we cannot simply wait and get training labels on historical cases (as would be the case, for example, with credit-card fraud). A common solution is then to have humans label data, but because of the prior challenge (extremely low base rate), human labeling fails for the reasons that have been elaborated previously in similar domains [2]. A machine learning researcher might then turn to active learning, but alas active learning also fails, as again has been discussed previously [3].
(3) The world changes, and the world even changes in direct response to the models that are put in place [15]. Such non-stationarity or concept drift is a challenge generally, but it is a particular challenge in non-self-revealing domains, because one cannot easily

monitor for it. If the rate of catching bad guys goes down, is that because they have changed their behavior? Or is that because you're doing a really good job, and are deterring bad behavior?

These three challenges conspire not only to make the knowledge engineering and machine learning difficult, they bring to the fore a key problem for machine learning systems that has received scant attention: the problem of Unknown Unknowns [1, 23]. Specifically, it is very difficult to know what such a system is missing, especially when it is confident that it is correct.[6]

Unfortunately, in this domain the Unknown Unknowns are absolutely key. When regulators hand out billion-dollar fines, it's not because analysts didn't get around to closing out the last false positives; it is because malfeasance was discovered that the regulators deem the firm was not even looking for.

Our human-in-the-loop approach was designed with these challenges in mind. We provide a rich search interface so that investigators can find cases of malfeasance efficiently and effectively on their own, in essence providing "guided learning" [2] training data to the system in the normal course of their activities. Furthermore, it has been shown that when humans are challenged to "beat the machine", i.e., to find cases of malfeasance that the machine misses but is confident that it is correct, they reveal very different sorts of cases than are otherwise found [1]. Beat the machine also is naturally implemented once one integrates the human investigators with the AI system.

4 EXPERIMENTAL RESULTS

Due to confidentiality and legal requirements, we cannot show results on actual data from financial institutions, nor can we report actual numbers of cases or base rates. However, the results we show here, generated on data drawn from similar distributions, qualitatively match results from real data, and the plots are similar. The conclusions we draw regarding the effectiveness of the techniques hold on the real data as well as the simulated data.

Domain Knowledge and Machine Learning. Let's first return to knowledge engineering, and in particular getting leverage from the existing rule-based systems. Given the time and monetary investments that have been put into these systems, it is interesting to see whether they can be improved through our proposed framework. We note that in such strictly regulated environments, like finance and compliance, continuity with prior operations is fundamental. Implicitly, this means that any new system should be as good as what is currently in place. Of course, this requirement is not necessarily quantified through a single value.

Nevertheless, keeping the number of false positive alerts at the same, if not lower, levels as the current system is a solid starting point. As alerted communications must be reviewed by humans, more alerts mean more reviewers, thereby raising the actual business cost of operating the system. Therefore, there is a clear, though indirect, relation between the alerts raised by the system and the actual cost of reviewing alerts. [7]

[6] Active learning can possibly address the known unknowns, if the low base rate can be dealt with, but such methods actually actively steer attention *away* from the Unknown Unknowns.

[7] The cost of reviewing and investigating a single serious alert is estimated at a few hundred dollars.

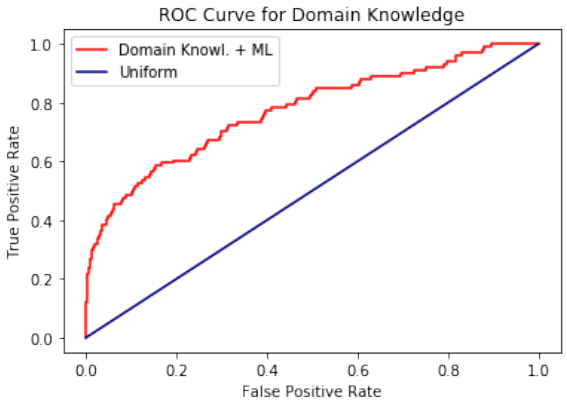


Figure 2: ROC curve when combining domain knowledge with Machine Learning

Figure 2 shows the results of combining the domain knowledge from the prior rule based system with machine learning trained on case outcomes, computed with 10-fold cross validation, incorporating the improvements that we discussed in Section 3.1 The ROC curve shown is generated a little differently from what we're used to, so let us explain. For these results we consider only the features created from the prior rule base, so the machine-learned model will only give a non-zero score to cases where the rule-based model gave a non-zero score. Therefore, all the false negatives and true negatives of the original system will also be false negatives and true negatives (respectively) of this component of the new system. The ROC curve shown is built on the population of cases for which the existing system created an alert. The positives are the alerts from the prior system that were deemed by experts to be interesting enough to pursue further; the negatives were the alerts that were deemed to be false alarms. So the ROC curve shows how well the new system can score (rank) the prior binary alerts.

As we can see from the ROC curve, the machine learning method does indeed learn to rank the truly interesting "cases" above the non-interesting ones, with a substantial concentration at the top of the ranked list. The current system "scores" each communication only by possibly issuing a binary alert, and does not perform numeric scoring or ranking by likelihood of malfeasance; thus for reference we plot its *expected behavior* as the diagonal (blue) line, indicating that one could achieve any (FP,TP) performance along the line via uniform random sampling of alerts. The red line shows the performance of our approach for combining domain knowledge and machine learning.

Network Analysis. We now turn our attention to the network analysis. Our interest here is to validate, whether the socio-professional network can provide additional evidence that might increase or decrease the suspicion of a case. Although network structures have been used successfully to identify malfeasance in other domains, as discussed above, it is unclear whether that will be the case for our domain as well. Our preliminary results, shown in Figure 3, answer this question in the affirmative.

As already explained in Section 3.3, we do not have direct knowledge of the strength of the socio-professional relationships between

employees. We estimate relationship strength in two ways. First we use the amount of communication between employees as a proxy for the strength of their relationship. Let's call this *local proximity*, as it is based on the pairwise communications. Shortcomings of local proximity include: it only considers direct, observed communications, and the amount of direct communication does not necessarily represent relationship strength.

The second approach, which we described in detail above, is to fit a model of latent relationship strength to the observed communication network, inferring the latent relationship strengths between each pair of employees. Let's call the resulting network the *global proximity* network. In both networks each pair of nodes (employees) is associated with a score, which is the estimated strength of their socio-professional relationship. This score can be used in various ways, such as to rank the neighbors of a focal node n, showing the other employees with which n has the strongest relationships.

We can now ask: do bad or suspicious actors cluster on these networks, similar to what has been observed in other malfeasance detection applications? More specifically, do bad actors tend to be more closely related to other bad actors? If so, does it matter which sort of relationship network one uses?

To evaluate these questions we take historical "escalation" as our approximate indication of bad behavior.[8] We mark employees (nodes) in the network as "positive" or "suspicious" if they have had a case escalated in the past. Now we can generate an ROC curve *for each node in the graph*–showing how the bad vs. good neighbors are ranked by their proximity (relationship strength). We can then average the ROC curves across multiple employees, which we have done for our evaluation. Figure 3 shows the results for the two different methods of estimating socio-professional relationship strength. Let's examine each in turn.

For the local proximity method, Figure 3a shows two lines: the red line is the average of the ROC curves for which the node under consideration, called an *ego node*, has been flagged as suspicious. To generate that ROC curve we iterate over the nodes who have been escalated and compute their individual ROC curves as described above. We then collect all of them and average them, which returns the red line. We do the same thing for "normal" employees, i.e., those who have never been escalated, to generate the green curve.

From the two ROC curves shown, we observe that local proximity indeed tends to rank suspicious neighbors more highly for suspicious egos, but this is due to the fact that suspicious neighbors are further away from normal egos; for the suspicious egos, suspicious neighbors are more-or-less randomly ranked by local proximity (on average). This is a fairly negative result for using the *communications* graph for this sort of network inference, as it seems to show that there is little signal in being close to a prior bad actor. However, the problem is with the choice of the communications graph, not with using network inference.

If we instead use the latent socio-professional relationship graph described above (global proximity), we see a very different result. Figure 3b shows the average ROC curves for normal and suspicious nodes using the *global proximity* network. We see two things: (*i*) the difference between the two lines is more pronounced, and (*ii*) the ROC curve for suspicious nodes is higher than the uniform (random) line—clearly in the positive portion of ROC space. In other words, using global proximity, suspicious egos are substantially closer to suspicious neighbors, while suspicious neighbors are randomly scattered among the neighbors of normal egos. This provides evidence to affirm that our generative network model may offer additional information in finding cases of malfeasance.

[8]In fact, a particular escalation may turn out to be benign. Therefore, we might more accurately talk about whether suspicious behavior clusters on the network rather than whether bad behavior clusters on the network. That said, the escalations very often result in at least a reprimand or a talking-to, so as long as we are not associating bad behavior exclusively with criminal behavior, but also with perhaps unintentional rule-breaking, then using the term "bad actor" may be reasonable.

5 DISCUSSION

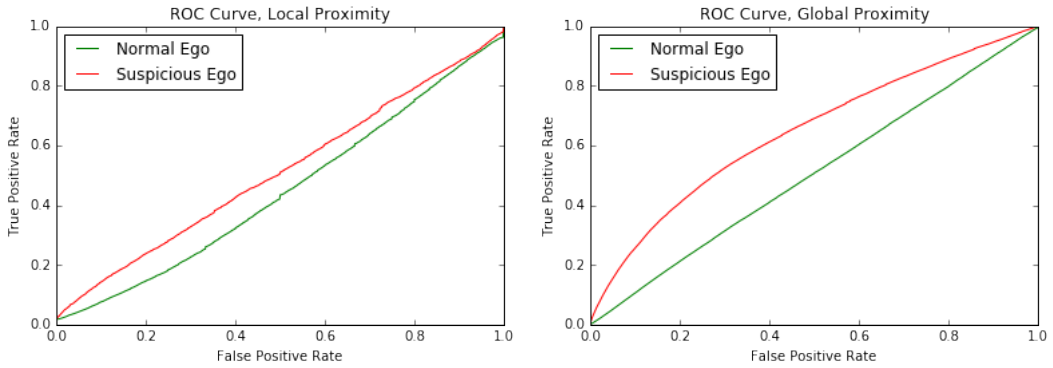
We presented details of a multi-faceted system built for detecting and investigating employee malfeasance. Such a system is crucial for regulatory compliance for financial firms. It is likely that as bad behavior by employees increasingly reflects on the firms that employ them, industries beyond finance will turn to technological solutions to help detect and hopefully deter bad behavior.

Our approach is a holistic, evidence-based approach, identifying *cases* of malfeasance, as opposed to the traditional approach of treating the problem as single-document classification. We systematically analyze broad data to add alternative perspectives, including *actor*-level information, *domain knowledge*, and *socio-professional network* analysis. We also present preliminary results of our improvements on integrating domain knowledge and machine learning, and we demonstrate the effectiveness of our generative model for inferring (latent) socio-professional relationship strength.

Thinking about lessons learned, the difficulty of acquiring and processing the required data is important to keep in mind as it impacts not only time and cost management, but also the system's effectiveness. In data science we must not be misled by the relative ease of gathering data that we see in ecommerce and adtech, where the entire process takes place within modern, interconnected systems. Enterprise applications can involve disconnected processes that are much more difficult to instrument for data gathering. Stakeholders need to be willing to make the investments in data assets needed to build more effective detection and investigation systems, including evaluation platforms that allow experimentation with different alternatives on the way to building an effective system.

Data acquisition is particularly vexing in "non-self-revealing" problems such as this, where one normally does not know the ground truth of cases that are not investigated. A key to dealing with this lack of data is a divide-and-conquer approach: (i) look to do well for the cases for which you have/can acquire ground truth data (or appropriate proxies), and (ii) separately begin to put in place methods for addressing the "unknowns" (both known unknowns and unknown unknowns). Also, key to acceptance, use, deployment, and debugging of systems for detecting malfeasance is the ability to explain the decisions made by the system.

Finally, as such systems become used increasingly by firms to monitor employee behavior, we (as a community) should add them to the discussion of the ethical implications of the application of AI and machine learning systems. The employees of financial firms are well aware that their communications are being monitored, and that the firms have a legal requirement to monitor employee behavior for various sorts of malfeasance. As such systems proliferate into other



(a) Local Proximity **(b) Global Proximity**

Figure 3: Average ROC curve, among all individuals, for the two types of network proximity

sorts of firms, the firms should inform the employees clearly both of the policies and laws, and also of the surveillance and monitoring that is being undertaken. However, the ethical implications go beyond issues of privacy and confidentiality. Systems that learn from historical data can unknowingly build in various biases, that can lead to discriminatory and other unwanted actions [11].

Acknowledgements The authors thank David Boyhan for many fruitful discussions about malfeasance detection and associated systems. The authors thank George A. Kellner and Andre Meyer for faculty fellowships. This paper's descriptions of models and methods are intended for a research audience and do not necessarily describe any models/methods used by any company.

REFERENCES

[1] J. Attenberg, P. Ipeirotis, and Foster Provost. 2015. Beat the Machine: Challenging Humans to Find a Predictive Model's Unknown Unknowns. *J. Data and Information Quality* 6, 1 (mar 2015).
[2] J. Attenberg and F. Provost. 2010. Why label when you can search?: alternatives to active learning for applying human resources to build classification models under extreme class imbalance. In *SIGKDD '10*.
[3] J. Attenberg and F. Provost. 2011. Inactive learning?: difficulties employing active learning in practice. *ACM SIGKDD Explorations* 12, 2 (2011).
[4] M. R. Barrick and M. K. Mount. 1991. The Big Five Personality Dimensions and Job Performance: A Meta-Analysis. *Personnel Psychology* 44, 1 (1991).
[5] R. J Bolton and D. J Hand. 2002. Statistical Fraud detection: A Review. *Statistical science* (2002).
[6] B. G. Buchanan and E. Shortliffe. 1984. *Rule-based expert systems*. Vol. 3.
[7] D. Chatzakou et al. 2017. Mean Birds: Detecting Aggression and Bullying on Twitter. In *ACM WebSci '17*.
[8] D. Chen et al. 2017. Enhancing transparency and control when drawing data-driven inferences about individuals. *Big Data* 5, 3 (2017).
[9] C. Cortes, D. Pregibon, and C. Volinsky. 2001. Communities of interest. *Advances in Intelligent Data Analysis 2001* (2001).
[10] D. R. Cox. 1958. The regression analysis of binary sequences. *Journal of the Royal Statistical Society. Series B (Methodological)* (1958).
[11] B. d'Alessandro, C. O'Neil, and T. LaGatta. 2017. Conscientious Classification: A Data Scientist's Guide to Discrimination-Aware Classification. *Big data* 5, 2 (2017).
[12] A. Danyluk, F. Provost, and B. Carr. 2002. Handbook of Data Mining and Knowledge Discovery. (2002).
[13] S. W. DeMore, J. D. Fisher, and R. M. Baron. 1988. The Equity-Control Model as a Predictor of Vandalism Among College Students1. *Journal of Applied Social Psychology* 18, 1 (1988).
[14] J. M Digman. 1990. Personality structure: Emergence of the five-factor model. *Annual review of psychology* 41, 1 (1990).
[15] T. Fawcett and F. Provost. 1997. Adaptive Fraud Detection. *Data Mining and Knowledge Discovery* 1, 3 (1997).
[16] R. Folger and R. Cropanzano. 2001. Fairness theory: Justice as accountability. *Advances in organizational justice* 1 (2001).
[17] S. Goel, A. Anderson, J. Hofman, and D. J Watts. 2015. The structural virality of online diffusion. *Management Science* 62, 1 (2015).
[18] L. R. Goldberg. 1990. An alternative "description of personality": the big-five factor structure. *Journal of personality and social psychology* 59, 6 (1990).
[19] Z. Gyongyi, P. Berkhin, H. Garcia-Molina, and J. Pedersen. 2006. Link spam detection based on mass estimation. In *VLDB '06*.
[20] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. 2004. Combating web spam with trustrank. In *VLDB '04*.
[21] LM Hough, JD Kamp, and BN Barge. 1988. Utility of temperament, biodata, and interest assessment for predicting job performance: a review and integration of the literature. *ARI Research Note, ADA* 178944 (1988).
[22] E. Junqué de Fortuny, M. Stankova, J. Moeyersoms, B. Minnaert, F. Provost, and D. Martens. 2014. Corporate residence fraud detection. In *SIGKDD '14*.
[23] H. Lakkaraju, E. Kamar, R. Caruana, and E. Horvitz. 2017. Identifying Unknown Unknowns in the Open World: Representations and Policies for Guided Exploration.. In *AAAI '17*.
[24] S. A Macskassy and F. Provost. 2005. Suspicion scoring of networked entities based on guilt-by-association, collective inference, and focused data access. In *Int. Conf. on Intelligence Analysis*.
[25] D. Martens and F. Provost. 2014. Explaining Data-driven Document Classifications. *MIS Quarterly* 38, 1 (2014).
[26] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS '13*.
[27] J. Moeyersoms, B. d'Alessandro, F. Provost, and D. Martens. 2016. Explaining Classification Models Built on High-Dimensional Sparse Data. *ICML 2016 Workshop on Human Interpretability in Machine Learning* (2016). arXiv:1607.06280
[28] J. Neville et al. 2005. Using Relational Knowledge Discovery to Prevent Securities Fraud. In *SIGKDD '05*.
[29] S. Pandit, D. H. Chau, S. Wang, and C. Faloutsos. 2007. Netprobe: A Fast and Scalable System for Fraud Detection in Online Auction Networks. In *WWW '07*.
[30] J. W. Pennebaker, R. L. Boyd, K. Jordan, and K. Blackburn. 2015. *The development and psychometric properties of LIWC2015*. Technical Report.
[31] F. Provost. 2002. [Statistical Fraud Detection: A Review]: Comment. *Statistical science* (2002).
[32] F. Provost and T. Fawcett. 2013. *Data Science for Business: What you need to know about data mining and data-analytic thinking*.
[33] E. Raisi and B. Huang. 2016. Cyberbullying identification using participant-vocabulary consistency. (2016). arXiv:1606.08084
[34] D. M. Romero, B. Uzzi, and J. Kleinberg. 2016. Social Networks Under Stress. In *WWW '16*.
[35] O. Stitelman, C. Perlich, B. Dalessandro, R. Hook, T. Raeder, and F. Provost. 2013. Using co-visitation networks for detecting large scale online display advertising exchange fraud. In *SIGKDD '13*.
[36] B. Wu, V. Goel, and B. D Davison. 2006. Topical trustrank: Using topicality to combat web spam. In *WWW '06*.
[37] C. Yang, R. Harkreader, J. Zhang, S. Shin, and G. Gu. 2012. Analyzing spammers' social networks for fun and profit: a case study of cyber criminal ecosystem on twitter. In *WWW '12*.