ONLINE ANOMALY DETECTION WITH EXPERT SYSTEM FEEDBACK IN SOCIAL NETWORKS

Corinne Horn and Rebecca Willett

Department of Electrical and Computer Engineering
Duke University, Durham, NC 27708

ABSTRACT

In this paper, we propose examining the *participants* in various meetings or communications within a social network, and using sequential inference based on these participant lists to quickly and accurately predict anomalies in the *content* of those communications. The proposed approach consists of two main elements: (1) *filtering*, or assigning a belief or likelihood to each successive measurement based upon our ability to predict it from previous noisy observations, and (2) *hedging*, or flagging potential anomalies by comparing the current belief against a time-varying and data-adaptive threshold. The threshold is adjusted based on feedback requested from an expert system. In general, parsing communication data can require nontrivial computational resources, but since parsed data is only used sparingly for feedback, the overall computational complexity of the proposed approach is relatively low. Regret bounds quantify the performance of the proposed approach, and experiments on the Enron email database demonstrate its efficacy.

Index Terms— anomaly detection, exponential families, sequential probability assignment, label-efficient prediction, filtering

1. INTRODUCTION

In a variety of contexts and applications, it is helpful or important to understand behavior patterns and actions within social networks. For instance, one might ask

- whether the focus of political party is shifting with the development of new factions,
- who within a financial firm knows when a product has been designed to favor some customers over others, or
- the impact on the political process of rules governing lobbyist activities.

To address these and other questions, a number of tools are being developed to help analyze enormous volumes of email, voice recordings, text documents, and other evidence. These expert systems, however, typically have a large computational complexity associated with them. In many cases, transcribing voice data, translating foreign languages, or decryption processes introduce significant bottlenecks into the process. Subsequent tools for taking the preprocessed data and automatically generating a "big-picture" summary (i.e. *topic modeling*) can be similarly time consuming or require training time and samples [1].

With these issues in mind, it it clear that somehow *prioritizing* the available data for detailed analysis is an essential step in the timely analysis of communication data. In this paper, we propose examining the *participants* in various forms of communications within a social network, and using sequential inference based on these participant lists to quickly and accurately predict anomalies in the *content* of those communications. Related work [1] has shown that there are meaningful relationships between email distribution lists and content. To ensure that the predicted anomalies are meaningful to an end user, we use an oracle or expert system to provide feedback to the social network analysis. Since the expert system is only used sparingly, the overall computational complexity of the proposed approach is significantly less than analyzing the content of all recorded communications.

There are several salient features of the proposed work that distinguishes it from previous social network analyses and anomaly detection methods:

- Co-occurrences: we develop a novel analysis framework for social network data based on observed communication patterns.
- Online: predictive analysis is performed in real time with each new observation; we require no training or assumption of a stationary underlying process.
- Non-stochastic: unlike particle filtering methods, we do not rely on assumptions of independence or known dynamical models.
- Performance bounds: while most online anomaly detection methods (cf. [2]) have no performance guarantees, we prove bounds on the regret of our approach relative to an oracle.

We know of *no other method* with these features. The main contribution of this paper is a combination of (a) theoretical performance bounds on (b) a novel formulation of social network analysis which (c) has been successfully applied to a real-world data set.

1.1. Problem Formulation

We have a network of n people, and at each time step $t = 1, 2, \ldots$ we observe the participant list of some form of communication (which we generically refer to as a *meeting*) among a subset of these n people. The state of the network at time t is thus described by a binary string $x_t = (x_t(1), \ldots, x_t(n)) \in \{0,1\}^n$, where $x_t(j) = 0$ or 1 according to whether the jth person participated in the meeting at time t. In this paper, we assume that we observe the sequence of x_ts directly (i.e. without noise) for simplicity of presentation[1] Having observed x_t at time t, we would like to infer whether the meeting is somehow *anomalous* or unusual relative to the past observations $x^{t-1} = (x_1, \ldots, x_{t-1})$.

Our inference engine outputs a binary label $\widehat{y}_t \in \{-1, +1\}$ indicating whether or not it has deemed x_t anomalous $(+1)$, or non-anomalous (-1) in addition to a scalar ℓ_t measuring the *degree* of

This work was supported by NSF CAREER Award No. CCF-06-43947 and ARO Grant No. W911NF-09-1-0262 PAR.

[1] We note that our framework is also applicable when we observe a *noisy* version of x_t, where each bit $x_t(j)$ may be flipped with probability $p < 1/2$, independently of all other bits. This would occur, for instance, when there is some uncertainty about the identity of some participants. See [3] for details.

anomalousness of each observation. This approach is based on the intuitive idea that a new observation x_t should be declared anomalous if it is very unlikely to occur based on the best probability model that can be assigned to our previously seen observations. The anomaly detection engine may decide to request the correct label whenever it is "unsure" (relative to some measure of confidence) about its decision. An expert topic modeling system then determines the veracity of each anomaly label using the "correct" label $y_t \in \{-1, +1\}$ and gives *feedback* to the anomaly detection algorithm.

This element of receiving and using limited feedback has conceptual connections to "active learning". However, most active learning methods, in contrast to our approach, either (a) are ad-hoc without theoretical performance bounds or (b) make strict assumptions about the underlying stochastic process; in many settings of interest, including ours, such assumptions are not realistic.

1.2. Outline of proposed approach

In this paper, we propose a general methodology for addressing these challenges. We call our proposed framework FHTAGN, or Filtering and Hedging for Time-varying Anomaly recoGNition. More specifically, the two components that make up FHTAGN are:

- *Filtering* — the sequential process of updating *beliefs* on the next state of the system based on the observed past.
- *Hedging* — the sequential process of flagging potential anomalies by comparing the belief against a time-varying threshold.

Having observed x^{t-1} (but not x_t), we can use this observation to assign "beliefs" or "likelihoods" to the next state x_t. Once we have access to x_t, we evaluate the likelihood of x_t as $\widehat{p}_t = p_t(x_t|x^{t-1})$ and declare an anomaly ($\widehat{y}_t = +1$) if $\zeta_t \triangleq \zeta(\widehat{p}_t) > \tau_t$, where τ_t is some threshold and $\zeta : \mathbb{R}^+ \to \mathbb{R}$ is a user-specified monotonically decreasing function (e.g. $\zeta_t = -\log(\widehat{p}_t) = \ell_t$) which should be chosen to sidestep challenging numerical issues when $\widehat{p}_t$ is very small. Declaring observations anomalous if their likelihoods fall below some threshold is a popular and effective strategy for anomaly detection, but setting this threshold is a notoriously difficult problem (cf. [2]). However, if we receive feedback y_t at time t and it differs from our label $\widehat{y}_t$, then we may adjust the threshold *appropriately*.

Rather than explicitly modeling the evolution of the system state and then designing methods for that model, (e.g., using Bayesian updates), we adopt an "individual sequence" perspective in which we develop procedures that perform provably well for any individual sequence in the problem domain. This approach allows us to sidestep challenging statistical issues associated with dependent observations or dynamic and evolving probability distributions. In particular we show that the per-round performance of our filtering technique approaches that of the best *offline* method with access to the entire data sequence, while our thresholding strategy performs better than any static threshold determined in hindsight.

2. FHTAGN METHOD

Filtering. At each time $t = 1, 2, \ldots$, before meeting x_t is observed, we assign a probability density $\widehat{p}_t$ to the possible values of x_t. More specifically, we choose $\widehat{p}_t$ to be a member of an *exponential family*; that is, at time t we select a parameter $\widehat{\theta}_t$ as a function of the past observations x^{t-1} so that $\widehat{p}_t(x) = p_{\widehat{\theta}_t}(x)$ and

$$p_\theta(x) = e^{\langle\theta,\phi(x)\rangle - \Phi(\theta)}, \text{ where } \Phi(\theta) \triangleq \int_{\mathsf{X}} \exp\big(\langle\theta, \phi(x)\rangle\big) d\nu(x)$$

is a normalization constant called the *log partition function* and $\mathsf{X} \triangleq \{0,1\}^n$. Here, $\phi : \mathsf{X} \to \mathbb{R}^n$ maps a meeting pattern x to an n-dimensional vector. For instance, if $\phi(x) = x$, then we implicitly model $\widehat{p}_t$ as a product of Bernoulli marginals, while if $(\phi(x))_{i,j} = x_i x_j$, then we model $\widehat{p}_t$ using an Ising model.

Given a constant $H > 0$, we restrict the parameter θ to lie in Λ, a closed, convex subset of $\Theta \triangleq \Big\{\theta \in \Theta : \nabla^2\Phi(\theta) \succeq 2H I_{d\times d}\Big\}$, where $I_{d\times d}$ denotes the $d \times d$ identity matrix, and the matrix inequality $A \succeq B$ denotes the fact that $A - B$ is positive semidefinite. We can measure the inaccuracy of our prediction $\widehat{p}_t$ using a negative log loss function:

$$\ell_t(\theta) \triangleq -\langle\theta, \phi(x_t)\rangle + \Phi(\theta) \quad \text{and} \quad L_{\theta^T}(x^T) \triangleq \sum_{t=1}^{T} \ell_t(\theta_t).$$

Our assumption on Θ guarantees that the log-partition function Φ and the loss functions $\ell_t(\theta)$ are *strongly convex* over $\Lambda \subseteq \Theta$. At time t, we use the gradient of this loss function to update our prediction parameter $\widehat{\theta}_t$ according to

$$\widehat{\theta}_t = \arg\min_{\theta\in\Lambda} \Big[\eta_t \Big\langle \theta, \nabla\Phi(\widehat{\theta}_{t-1}) - \phi(x_{t-1})\Big\rangle + D(\widehat{\theta}_{t-1}\|\theta)\Big], \quad (1)$$

where η_t is a positive step size and $D(\cdot\|\cdot)$ is the Kullback-Leibler divergence. The optimization can be reduced to simple update steps as seen in Algorithm 1, which require only the current observation x_t and the probability density $\widehat{p}_t$ estimated at the previous time; it is not necessary to keep all observations in memory to ensure strong performance.

Algorithm 1 Label-efficient anomaly detection

Parameters: real numbers $\eta > 0$, $\tau_{\max} > \tau_{\min}$
Initialize: $\tau_1 \in [\tau_{\min}, \tau_{\max}]$, $\widehat{\theta}_1 \in \Lambda$
for $t = 1, 2, \ldots$ **do**
  Acquire new observation x_t
  Incur the cost $\ell_t(\widehat{\theta}_t) = -\log\widehat{p}_t = -\langle\widehat{\theta}_t, \phi(x_t)\rangle + \Phi(\widehat{\theta}_t)$
  Set $\widehat{\mu}_t = \nabla\Phi(\widehat{\theta}_t)$ and $\widehat{\mu}'_{t+1} = \widehat{\mu}_t - \eta_t\nabla\ell_t(\widehat{\theta}_t)$
  Set $\widetilde{\theta}_{t+1} = \nabla\Phi^*(\widehat{\mu}'_{t+1})$ and $\widehat{\theta}_{t+1} = \arg\min_{\theta\in\Lambda} D(\widetilde{\theta}_{t+1}\|\theta)$
  Set $\zeta_t = \zeta(\widehat{p}_t)$.
  if $\zeta_t > \tau_t$ **then** let $\widehat{y}_t = 1$ **else** let $\widehat{y}_t = -1$
  Draw a Bernoulli random variable U_t such that
  $\Pr[U_t = 1|U^{t-1}] = 1/(1 + |\zeta_t - \tau_t|)$
  if $U_t = 1$ **then** Request feedback y_t and let
  $\tau_{t+1} = \arg\min_{\tau\in[\tau_{\min},\tau_{\max}]}(\tau - \tau_t - \eta y_t 1_{\{\widehat{y}_t \neq y_t\}})^2$
  else let $\tau_{t+1} = \tau_t$
end for

Azoury and Warmuth [4] proposed and analyzed an algorithm similar to eqn. (1) in the setting of online density estimation over an exponential family; however, they did not consider noisy observations and only proved regret bounds for a couple of specific exponential families.

Hedging. The second ingredient of FHTAGN is *hedging*, i.e., sequential adjustment of the critical threshold τ_t, such that whenever $\zeta_t > \tau$ we declare an anomaly. In order to choose an appropriate level τ_t, we rely on feedback from an expert system y_t which indicates whether our anomaly flag is accurate. In addition, we limit the amount of feedback we receive from the expert system by randomly requesting a label with probability U_t that depends on ζ_t and τ_t. When we receive feedback, we update our value of τ_t according to the correctness of $\widehat{y}_t$, as detailed in Algorithm 1.

3. PERFORMANCE BOUNDS

In the present individual-sequence setting, the performance of a given prediction strategy is compared to the best performance achievable on $\boldsymbol{x}$ by any strategy lying in the above exponential family. Suppose that the horizon T is fixed in advance. Given a prediction strategy $\widehat{p}^T = \{\widehat{p}_t\}_{t=1}^T$, we can define the *filtering regret* w.r.t. $p^T = \{p_t\}_{t=1}^T$ in the above exponential family as

$$R_T^{(F)}(\widehat{p}^T; x^T, p^T) = L_{\widehat{\theta}^T}(x^T) - L_{\theta^T}(x^T). \tag{2}$$

We can establish a sub-linear $O(\sqrt{T})$ regret bound against *time varying* strategies $\theta^T \in \Lambda^T$, provided the variation is sufficiently slow. The precise result can be stated as follows:

Theorem 1 (Filtering regret against time-varying strategies) *Let $\widehat{\theta}^T$ be the sequence of parameters in Λ computed from the sequence x^T using the procedure shown in Algorithm 1 with step sizes $\eta_t = 1/\sqrt{t}$. Then, for any sequence $\theta^T \in \Lambda$, we have*

$$R_T^{(F)}(\widehat{p}^T; x^T, p^T) \preceq \sqrt{T} V_T(\theta^T)$$

where $V_T(\theta^T) \triangleq \sum_{t=1}^T \|\theta_t - \theta_{t+1}\|$ is the variation *of θ^T.*

(The notation $a_T \preceq b_T$ means that there exists some $C > 0$ such that $a_n \leq C b_n$ for n sufficiently large.) The proof relies on the strong convexity of Φ, as well as the more basic machinery of [5], [6]. See [3] for details.

For the hedging step, we are interested in the number of mistakes made by the forecaster over T time steps, $\sum_{t=1}^T 1_{\{\widehat{y}_t \neq y_t\}}$. We would like to obtain regret bounds relative to any fixed threshold $\tau \in [\tau_{\min}, \tau_{\max}]$ that could be chosen in hindsight after having observed the entire sequence of transformed probability assignments $\{\zeta_t\}_{t=1}^T$ and sporadic feedback $\{y_t\}_{t=1}^T$. (note that some y_t's may be "empty," reflecting the lack of availability of feedback at the corresponding times). Ideally, for any static τ, we would like to bound

$$\sum_{t=1}^T 1_{\{\mathrm{sgn}(\tau_t - \zeta_t) \neq y_t\}} - \sum_{t=1}^T 1_{\{\mathrm{sgn}(\tau - \zeta_t) \neq y_t\}}. \tag{3}$$

However, analyzing this expression is difficult owing to the fact that the function $\tau \mapsto 1_{\{\mathrm{sgn}(\tau - \zeta_t) \neq y\}}$ is not convex in τ. To deal with this difficulty, we use the standard technique of replacing the comparator loss with the convex *hinge loss* $(1 - (\tau - \zeta)y)_+$, where $(\alpha)_+ \triangleq \max\{0, \alpha\}$. Thus, instead of (3), we bound the *hedging regret*

$$R_T^{(H)}(\widehat{\tau}^T; \zeta^T, y^T, \tau) \triangleq \sum_{t=1}^T 1_{\{\mathrm{sgn}(\tau_t - \zeta_t) \neq y_t\}} - \sum_{t=1}^T (1 - (\tau - \zeta_t) y_t)_+. \tag{4}$$

Theorem 2 (Hedging regret) *Fix a time horizon T and consider the sequence $\widehat{\tau}^T$ computed via Algorithm 1 with parameter $\eta = 1/\sqrt{T}$. Then*

$$\mathbb{E}\left[R_T^{(H)}(\widehat{\tau}^T; \zeta^T, y^T, \tau)\right] \leq \sqrt{T}.$$

where the expectation is taken with respect to $\{U_t\}_t$.

The proof of this theorem is a modification of the proof of Theorem 12.5 in [7]. See [3] for details.

	FHTAGN	Best static threshold
number of errors	73	143
number of false alarms	35	96
number of misses	38	47

Table 1. Performance comparison for FHTAGN and the best static threshold. Feedback was requested for 91 of the 902 days considered, and only 523 of the 902 days had their text parsed.

4. EXPERIMENTAL RESULTS

In order to validate our algorithm, we have conducted experiments using the Enron e-mail database [1], which is publicly available at `http://www.cs.cmu.edu/~enron`. The Enron corpus consists of approximately $500,000$ e-mails involving 151 known employees and more than $75,000$ distinct addresses, between the years 1998 and 2002. We use email timestamps in order to record users that were active in each day, either sending or receiving emails. This was done for $1,177$ days, starting from Jan 1, 1999. We removed days during which no email correspondence occurred, and we consolidated each weekend's emails into the preceding Friday's observation vector, resulting in a total of 902 days in our dataset. In this setting, we let $\phi(x) = x$ leading to a dimensionality of $d = n = 75,511$, and $\zeta_t = -(8\text{e-}3)\log \widehat{p}_t$. τ was initialized at $\zeta(\widehat{p}_1)$. We set $\eta = 11.2$. Feedback was received when requested according to Algorithm 1.

We generate oracle or expert feedback based on the email text as follows. If feedback is requested at time t, we generate word count vectors h_t using the 12,000 most frequently appearing words (to avoid memory issues and misspelled words) for days $t-10, ...t$. Then we average the difference in word counts between day t and each of of previous 10 days as follows: $e_t = (1/10)\sum_{i=t-10}^{t-1} \|h_t - h_i\|_1$, where $\|\cdot\|_1$ represents the ℓ_1 norm. When the prediction error e_t is sufficiently high, we consider day t to be anomalous according to our expert. Note that this expert only needs to process a limited amount of data to deliver feedback, thus reducing the total amount of computational resources needed to open, decrypt, transcribe, or translate documents.

The prediction error e_t and the threshold determining y_t is plotted in Figure 1, right bottom, but note that only a fraction of these values need to be computed to run FHTAGN. A variety of other expert systems for determining anomalous emails or documents based on their contents could be used instead of our keyword predictor; a useful survey of the many such anomaly detection methods available is presented in [2]. Exploring the role and performance of different expert systems in the context of FHTAGN is beyond the scope of this paper, but is an important avenue for future research.

Our results are summarized in Table 1 and Figure 1. As predicted by our theoretical results, FHTAGN performs very well relative to a comparator online anomaly detection method which consists of comparing ζ_t to the best static threshold which could be chosen in hindsight at time T with full knowledge of all filtering outputs and feedback provided to FHTAGN.

As shown in the left plot in Figure 1, the threshold τ_t adapts to feedback regarding false alarms and missed anomalies from the oracle. Moreover, some of the true anomalies in this example are contextual in that they do not always correspond to large values of ζ_t, but rather to large values relative to neighboring observations. Letting the threshold τ_t change over time allows FHTAGN to adapt to an expert's evolving notion of what is anomalous.

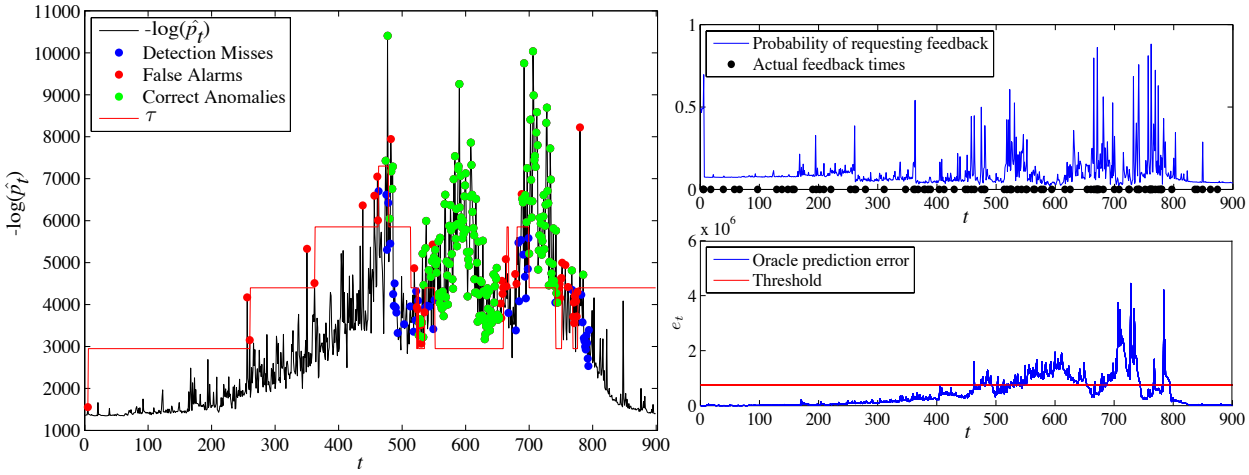

Fig. 1. Online anomaly detection results on Enron corpus. Left plot displays filtering output, locations of correctly identified anomalies (according to our oracle), erroneously identified anomalies, and the dynamic threshold. Top right plot displays the dynamic probability of requesting feedback and the times at which feedback was provided. Bottom right plot displays the oracle's prediction error, used to determine "ground truth" anomalies.

The right upper plot of Figure 1 shows the probability of requesting feedback over time and the days on which feedback is requested; as expected, feedback is less likely when ζ_t is very far from the current threshold choice τ_t. There are a total of 91 feedback requests over 902 days, and because of the sliding window used by our oracle to determine the true labels y_t, a total over 523 of the 902 days required text parsing (and, generally speaking, any overhead associated with processing the documents).

Some of the most anomalous events detected by our proposed approach correspond to historical events. For instance, consider the following events.

- **Dec. 1, 2000:**
  Days before "California faces unprecedented energy alert" (Dec. 7) and energy commodity trading deregulated in Congress. (Dec. 15); see http://www.pbs.org/wgbh/pages/frontline/shows/blackout/california/timeline.html.
- **May 9, 2001:** "California Utility Says Prices of Gas Were Inflated" by Enron collaborator El Paso,blackouts affect upwards of 167,000 Enron customers; see http://archives.cnn.com/2001/us/05/08/calif.power.crisis.02/.
- **Oct. 18, 2001:** Enron reports $618M third quarter loss, followed by later major correction; see http://www.justice.gov/enron/exhibit/04-27/BBC-0001/Images/24379.001.PDF.

These examples indicate that the anomalies in social network communications detected by FHTAGN are indicative of anomalous events of interest to the social network members.

5. CONCLUSIONS

This work highlights the importance of social network structure as a predictor of paradigm shifts in the topics of interest to that network. This revelation is particularly significant when limited resources are available for transcribing or translating recordings of a meeting; by studying changes in *who* is participating in meetings, we can accurately infer when the *content* of the meetings is similarly shifting, and use relatively little feedback to fine-tune our precision. Feedback can be generated using an expert system for topic modeling. We show that by using expert systems to provide limited feedback on an as-needed basis, we can harness the benefit of their analysis using a relatively small amount of computation.

6. ACKNOWLEDGEMENTS

The authors would like to thank Maxim Raginsky and Jorge Silva for their insight during several valuable discussions.

7. REFERENCES

[1] A. McCallum, X. Wang, and A. Corrada-Emmanuel, "Topic and role discovery in social networks with experiments on Enron and academic email," *Journal of Artificial Intelligence Research*, vol. 30, pp. 249–272, 2007.

[2] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection - a survey," *ACM Computing Surveys*, vol. 41, no. 3, 2009.

[3] M. Raginsky, R. Willett, C. Horn, J. Silva, and R. Marcia, "Sequential anomaly detection in the presence of noise and limited feedback," Submitted., 2010.

[4] K. S. Azoury and M. K. Warmuth, "Relative loss bounds for on-line density estimation with the exponential family of distributions," *Machine Learning*, vol. 43, pp. 211–246, 2001.

[5] P. Bartlett, E. Hazan, and A. Rakhlin, "Adaptive online gradient descent," in *Advances in Neural Information Processing Systems (NIPS)*, Cambridge, MA, 2008, vol. 20, pp. 65–72, MIT Press.

[6] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient descent," in *Proc. Int. Conf. on Machine Learning (ICML)*, 2003, pp. 928–936.

[7] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning and Games*, Cambridge Univ. Press, 2006.