To appear, *International Joint Conference on Artificial Intelligence (IJCAI) 2013.*

One-Class Conditional Random Fields for Sequential Anomaly Detection

Yale Song[1], Zhen Wen[2], Ching-Yung Lin[2], Randall Davis[1]
[1]MIT Computer Science and Artificial Intelligence Laboratory
[2]IBM T.J. Watson Research Center
{yalesong,davis}@csail.mit.edu, {zhenwen,chingyung}@us.ibm.com

Abstract

Sequential anomaly detection is a challenging problem due to the one-class nature of the data (i.e., data is collected from only one class) and the temporal dependence in sequential data. We present One-Class Conditional Random Fields (OCCRF) for sequential anomaly detection that learn from a one-class dataset and capture the temporal dependence structure, in an unsupervised fashion. We propose a hinge loss in a regularized risk minimization framework that maximizes the margin between each sequence being classified as "normal" and "abnormal." This allows our model to accept most (but not all) of the training data as normal, yet keeps the solution space tight. Experimental results on a number of real-world datasets show our model outperforming several baselines. We also report an exploratory study on detecting abnormal organizational behavior in enterprise social networks.

1 Introduction

Consider an espionage case where an insider performs a series of events that seem normal individually, yet appear abnormal only when considered collectively, as for example, logging in to a system late at night, downloading files from a server untouched for a while, and copying large amounts of data to a USB drive. Detecting this type of sequential anomaly is an extremely difficult task because of both the one-class nature of the data [Chandola *et al.*, 2009] and the temporal dependence in observations [Cheng *et al.*, 2009].

One-class learning refers to the process of using training data collected from only one class to predict whether or not a new sample is drawn from the same distribution. Many types of real-world anomalies occur very infrequently, and hence are hard to obtain samples from, e.g., rare diseases and network intrusions. Furthermore, even if there are some examples known as anomalous, they may not represent the underlying distribution of that class accurately, making them unsuitable as training data. For example, malicious hackers may invent new patterns of attack, so a network intrusion detection system trained with only known patterns may not perform reliably on new attacks.

A second source of difficulty in sequential anomaly detection is the temporal dependence in sequential observations. Much research has highlighted the importance of capturing dependence structure in the sequential data [Quattoni *et al.*, 2007]. In many anomaly detection scenarios, in particular, an anomalous pattern is often defined as a series of events that are normal individually but abnormal only collectively. For example, the UNIX command `login` and `passwd` are common and expected input, but many repetitions of them may indicate malicious intent.

Standard machine learning algorithms (e.g., SVM, HMM, and CRF) are not suitable for one-class learning because, to work properly, they require data from all classes. Many efficient algorithms for one-class learning have been proposed. One-class SVM [Schölkopf *et al.*, 2001] uses kernels to capture patterns in a high dimensional feature space, but is not directly applicable to sequential data because it ignores any dependence structure in the data. Numerous attempts have been made to capture dependence structure in the data, e.g., by latent SVM [Yu and Joachims, 2009] and CRF [Lafferty *et al.*, 2001], but their discriminative learning procedure suffers with one-class datasets [He and Garcia, 2009].

This work presents One-Class Conditional Random Fields (OCCRF) as a general-purpose sequential anomaly detection algorithm that deals with the difficulties mentioned above. Our method is inspired by the idea of both one-class SVM [Schölkopf *et al.*, 2001] and CRF [Lafferty *et al.*, 2001]. We follow the learning strategy of Scholkopf *et al.* [2001] and accept most of the training samples as normal, while favoring a simple and tight solution space. Dependence structure in the sequential data is captured using a log-linear model similar to the CRF formulation. The key to our method is a new hinge loss function in a regularized risk minimization framework that maximizes the margin between positive and negative label assignments to the conditional probability distribution.

We report on experiments with four real-world datasets; the results show that our method outperforms several baselines, including Active Outlier [Abe *et al.*, 2006], Local Outlier Factor [Breunig *et al.*, 2000], One-Class SVM [Schölkopf *et al.*, 2001], and HMM [Rabiner, 1989]. We also report on an exploratory study on detecting abnormal organizational behavior in enterprise social networks.

*This work was supported in part by DARPA W911NF-12-C-0028, by ONR #N000140910625, and by NSF IIS-1018055.

2 Previous Work

Anomaly detection has long been an active research topic in data mining and machine learning community; see [Chandola *et al.*, 2009] for a comprehensive survey. A simple approach is to cast the problem as binary classification and to train a standard model using both positive (normal) and negative (abnormal) data. For example, Liao *et al.* [2010] used a CRF to detect anomalies in GPS data. However, these methods require at least one example from each class to work properly. Advanced methods include [Abe *et al.*, 2006] that trains an ensemble of decision trees using one-class data augmented with artificially generated negative samples. However, it is non-trivial to generate sequential data. Our method learns from one-class data and does not require any negative sample during learning process.

Methods based on kernel density estimation [Parzen, 1962] detect anomalies by fitting a non-parametric model to positive samples and rejecting samples using a statistical hypothesis test, while relative density-based methods work by computing an anomaly score of each sample using its neighbors, with an assumption that normal samples occur in dense neighborhoods. For example, Breunig *et al.* [2000] proposed a method to compute an anomaly score by comparing a local density of each sample to an average local density of its neighbors. These methods have a rigorous theoretical foundation, but are computationally intensive because they require finding k-nearest neighbors for each test sample, which takes $O(N^2)$ with N samples. Our method is learning-based and takes only $O(1)$ at test time.

Scholkopf *et al.* [2001] and Tax and Duin [2004] have independently proposed a support vector algorithm for one-class classification, using kernels to obtain a tight boundary around normal samples in high dimensional feature space. However, these models do not consider dependence structure in the data. Lafferty *et al.* [2001] proposed a discriminatively trained conditional log-linear model for sequence labeling task, while Yu and Joachims [2009] have extended the standard SVM to learn dependence structure in the data. However, these methods are developed for multi-class classification and are not directly applicable to one-class learning. Our method is a combination of both one-class SVM and CRF, and thus can work in sequential anomaly detection setting.

Several efforts have focused on sequential anomaly detection using categorical input attributes. Chandola [2009] evaluated several algorithms to detect anomalies in symbolic sequences (e.g., genetic codes), Sun *et al.* [2006] used probabilistic suffix trees to mine outliers in sequential databases, and Xu [2010] proposed a Markov reward process model for intrusion detection. Unfortunately, these methods do not work on continuous attributes. Our method is based on a log-linear model and can deal with continuous attributes.

Sliding window-based methods have been popular in sequential anomaly detection. Tan *et al.* [2011] used an ensemble of half-space trees with a sliding window to detect anomalies in evolving streaming data. However, since it computes an anomaly score by traversing a tree structure that is bounded by the maximum depth parameter and the size of sliding window, it may not capture long range dependence. Since our method inherits the benefits of CRF models, it can capture long-range dependence in the sequential data.

3 One-Class Conditional Random Fields

We consider unsupervised learning of a classifier $f : \mathcal{X} \to \mathcal{Y}$ from a one-class dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^{d \times t_i}, y_i = +1\}_{i=1}^{N}$, i.e., all of the samples are assumed to have positive labels, but we do not in fact have ground-truth labels. The input domain $\mathcal{X}$ consists of multivariate time-series data, of dimension d and length t_i (the length of each sequence can vary). The output domain $\mathcal{Y} = \{+1, -1\}$ contains, without loss of generality, the *normal* class label +1 (i.e., the class it has learned from) and the *abnormal* class label -1 (i.e., the class it has never seen before).

3.1 The Model

It is trivial to learn a function $f(\mathbf{x})$ that accepts all of the training examples as normal, but it may not generalize well to unseen samples. Similar to [Schölkopf *et al.*, 2001], our learning strategy is to accept *most* of the training examples as normal, while making the solution space as tight as possible.

Let's assume for now that we can compute the conditional probability distribution $p_{\mathbf{w}}(y \mid \mathbf{x})$ with some parameter vector $\mathbf{w}$ (described in Section 3.2). Our learning objective is

$$\min_{\mathbf{w},\xi,\rho} L(\mathbf{w}) = \frac{\lambda}{2} \|\mathbf{w}\|_2^2 + \frac{1}{N} \sum_{i=1}^{N} \xi_i - \rho \tag{1}$$

$$s.t.\ \forall i : \Delta(\mathbf{x}_i; \mathbf{w}) \geq \rho - \xi_i,\ \ \xi_i \geq 0 \tag{2}$$

$$\Delta(\mathbf{x}_i; \mathbf{w}) = p_{\mathbf{w}}(y_i = +1|\mathbf{x}_i) - p_{\mathbf{w}}(y_i = -1|\mathbf{x}_i) \tag{3}$$

where $\|\mathbf{w}\|_2^2 = \sum_l |w_l|^2$ is a squared L_2 norm and ξ_i's are slack variables that are related to the soft margin, i.e., introduced to measure the degree of misclassification.

This objective demands most of the training examples to have a higher probability of being normal ($y_i = +1$), while favoring a tight solution space with L_2 regularization, at the cost of allowing some examples to have a margin $\Delta(\mathbf{x}_i; \mathbf{w})$ smaller than an offset parameter $\rho \in [0, 1)$. It controls the tradeoff between these two goals by specifying the minimum margin between the probability of each sample being classified as normal or abnormal (regardless of its ground truth label). Once we find the solution $(\mathbf{w}, \rho)$, we can set a decision rule for our one-class classifier as

$$f(\mathbf{x}) = sgn\left(\Delta(\mathbf{x}; \mathbf{w}) - \rho\right) \tag{4}$$

As noted above, our model performs unsupervised learning, as we do not have ground-truth labels. Because our learning strategy accepts most (but not necessarily all) of the samples as positive, some samples are allowed to be negative, enabling the learned classifier to be more robust to outliers in training data. In Section 5 we show experimental result that used training data containing true negative labels.

3.2 Computing Conditional Probability

Now we turn to computing the conditional probability distribution $p_{\mathbf{w}}(y \mid \mathbf{x})$ that is required to obtain $\Delta(\mathbf{x}, \mathbf{w})$. We can use either the standard formulation of CRF [Lafferty *et al.*,

2001] or CRF with latent variables [Quattoni *et al.*, 2007]. We briefly introduce formulations of both models, discuss their pros and cons in the context of sequential anomaly detection, and explain why we chose CRF with latent variables.

The standard CRF formulates the conditional probability distribution $p_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x})$ as

$$p_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x}) = \frac{\exp\left\{\mathbf{w}^\top \cdot \Phi(\mathbf{y}, \mathbf{x})\right\}}{\sum_{\mathbf{y}' \in \mathcal{Y}} \exp\left\{\mathbf{w}^\top \cdot \Phi(\mathbf{y}', \mathbf{x})\right\}} \tag{5}$$

where $\mathbf{y}$ is a vector of length t, and the feature function $\Phi(\mathbf{y}, \mathbf{x})$ is defined as

$$\Phi(\mathbf{y}, \mathbf{x}) = \sum_j \phi(y_j, \mathbf{x}) + \sum_{j,k} \phi(y_j, y_k) \tag{6}$$

with singleton features $\phi(y_j, \mathbf{x})$ and pairwise features $\phi(y_j, y_k)$; specific definitions of the features depend on applications, we define these in Section 4.2.

Using the CRF, we can compute $\Delta(\mathbf{x}, \mathbf{w})$ by obtaining $p_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x})$ via Equation 5, taking an average of the marginals $p_{\mathbf{w}}(y \mid \mathbf{x}) = \frac{1}{t}\sum_{j \in t} p_{\mathbf{w}}(y_j \mid \mathbf{x})$, and finally computing the difference $p_{\mathbf{w}}(y_i = +1|\mathbf{x}_i) - p_{\mathbf{w}}(y_i = -1|\mathbf{x}_i)$.

This standard CRF formulation has the advantage that we can use many existing convex optimization algorithms with theoretically sound convergence bounds. However, previous research suggests that many real-world problems may not be easily formulated as a simple convex optimization problem without forcing a reduction in the expressiveness of the models [Do and Artieres, 2012]. Specifically, evidence has shown that, despite making the problem no longer convex, incorporating latent variables to the model to capture hidden dependence structure in the data often leads to better performance [Quattoni *et al.*, 2007; Yu and Joachims, 2009].

A CRF with a set of latent variables $\mathbf{h} \in \mathcal{H}$ is formulated in [Quattoni *et al.*, 2007] as

$$p_{\mathbf{w}}(y \mid \mathbf{x}) = \frac{\sum_{\mathbf{h}} \exp\left\{\mathbf{w}^\top \cdot \Phi(y, \mathbf{h}, \mathbf{x})\right\}}{\sum_{y' \in \mathcal{Y}, \mathbf{h}} \exp\left\{\mathbf{w}^\top \cdot \Phi(y', \mathbf{h}, \mathbf{x})\right\}} \tag{7}$$

where the feature function $\Phi(y, \mathbf{h}, \mathbf{x})$ is defined as

$$\Phi(y, \mathbf{h}, \mathbf{x}) = \sum_j \phi(y, h_j, \mathbf{x}) + \sum_{j,k} \phi(y, h_j, h_k) \tag{8}$$

with singleton features $\phi(y, h_j, \mathbf{x})$ and pairwise features $\phi(y, h_j, h_k)$. Using the additional set of latent variables, we can expect that our model is more expressive (and as a result, the computed $p_{\mathbf{w}}(y \mid \mathbf{x})$ is more accurate) because, unlike CRFs, each observation within a sequence is allowed to have a different label. This is especially crucial in sequential anomaly detection, where there may exist several possible descriptions of "normal" sequential patterns.

For these reasons, we use a CRF with latent variables in our experiments, though one can use the standard CRF to compute the conditional probability distribution $p_{\mathbf{w}}(y \mid \mathbf{x})$.

3.3 Solving Regularized Risk Minimization

We can cast our objective in Equation 1 as a regularized risk minimization problem,

$$\min_{\mathbf{w}} L(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \frac{1}{N}\sum_{i=1}^{N} l(\mathbf{x}_i, y_i; \mathbf{w}) \tag{9}$$

$$l(\mathbf{x}_i, y_i; \mathbf{w}) = \max\left[0, \log\left(\frac{1+\rho}{1-\rho}\right) - \Gamma(\mathbf{x}_i; \mathbf{w})\right] \tag{10}$$

$$\Gamma(\mathbf{x}_i; \mathbf{w}) = \log\left(\frac{p_{\mathbf{w}}(y_i = +1 \mid \mathbf{x}_i)}{p_{\mathbf{w}}(y_i = -1 \mid \mathbf{x}_i)}\right) \tag{11}$$

where $l(\mathbf{x}_i, y_i; \mathbf{w}) \geq 0$ is a hinge loss that penalizes the cases when the constraints in Equation 2 is violated. It is derived from $l(\mathbf{x}_i, y_i; \mathbf{w}) = \max[0, \rho - \Delta(\mathbf{x}_i; \mathbf{w})]$; we converted this to a log scale for numerical stability. Note that our loss function is undefined when $p_{\mathbf{w}}(y_i \mid \mathbf{x}_i)$ is either 0 or 1; we require it to be in the range of an open bounded interval (0,1).

To solve Equation 9 we use the bundle method [Teo *et al.*, 2010; Do and Artieres, 2012], which converges to a solution with an accuracy ε at the rate $O(1/\varepsilon)$ for general non-differentiable convex problems (note that our hinge loss is non-differentiable at the hinge point). The method aims at iteratively building an increasingly accurate piecewise quadratic lower bound of $L(\mathbf{w})$ based on its subgradient $\partial_{\mathbf{w}} L(\mathbf{w})$. The subgradient of $l(\mathbf{x}_i, y_i; \mathbf{w})$ is obtained as

$$\partial_{\mathbf{w}} l(\mathbf{x}_i, y_i; \mathbf{w}) = -\partial_{\mathbf{w}} \Gamma(\mathbf{x}_i; \mathbf{w}) \tag{12}$$

The specific form of $\partial_{\mathbf{w}} \Gamma(\mathbf{x}_i; \mathbf{w})$ depends on whether we use Equation 5 or Equation 7 to compute $p_{\mathbf{w}}(y|\mathbf{x})$. For the former case,

$$\partial_{\mathbf{w}} \Gamma(\mathbf{x}_i; \mathbf{w}) = \Phi(+\mathbf{1}, \mathbf{x}) - \Phi(-\mathbf{1}, \mathbf{x}) \tag{13}$$

and for the latter case,

$$\partial_{\mathbf{w}} \Gamma(\mathbf{x}_i; \mathbf{w}) = \alpha(+1) - \alpha(-1) \tag{14}$$

$$\alpha(y') = \sum_h \mathbb{E}_{h \backsim p_{\mathbf{w}}(h|y', \mathbf{x})}[\Phi(y', h, \mathbf{x})]$$

Since we are interested in time-series data as input, we can restrict the underlying graph structure as a linear chain and use an efficient exact inference algorithms, such as belief propagation [Pearl, 1982], to obtain the marginal probabilities $p_{\mathbf{w}}(y|\mathbf{x})$ and $p_{\mathbf{w}}(h|y, \mathbf{x})$.

4 Experiments

Table 1 shows the descriptive statistics of the datasets we used in our experiments. Following the typical experimental setting in one-class learning literature, for multi-class datasets we set one of the classes as normal and all other classes as abnormal, and we repeated this for all classes. We compared our approach to four well-established baseline methods that are publicly available. Below we detail the datasets, describe the baselines and the parameter values we validated, explain the experimental methodology, and discuss results.

4.1 Datasets

CUAVE [Patterson *et al.*, 2002][1]: This dataset contains audio-visual data of ten spoken digits (zero through nine). We used the clean version (i.e., no noise added) of the isolated digits collection from individual speaker data. Audio features contain 14 MFCCs and their first and second derivatives, resulting in a 42D feature vector. Visual features are computed from a 16x16 gray-scale mouth subregion and contain 35 DCT coefficients and their first derivatives, resulting

[1] We based our data on the version of [Saenko and Livescu, 2006]

Data	$\mathcal{Y}$	d	N	avg(t_i)	avg(N_+)	avg(N_-)
C	10	112	1,790	45.86	179	1,611
A	6	20	724	25.14	121	603
N	24	20	9,600	49.51	400	9,200
S	2	29	17,087	29.32	16,921	166

Table 1: Descriptive statistics of the datasets we used (C: CUAVE, A: ArmGesture, N: NATOPS, S: SOIT). $\mathcal{Y}$ is the number of classes, d the number of features, N the number of sequence samples, avg(t_i) is the average length of sequence, avg(N_+) and avg(N_-) are the average number of normal and abnormal samples when one class is selected as normal and all others as abnormal.

in a 70D feature vector. PCA was used to reduce the dimension to 20, which accounted for 95% of the variation.

ArmGesture [Quattoni *et al.*, 2007]: This dataset contains six classes of upper body gestures. Observation features are 2D joint angles and 3D euclidean coordinates for left/right shoulders and elbows, resulting in a 20D feature vector. We normalized each dimension to have a mean zero and standard deviation one.

NATOPS [Song *et al.*, 2011]: This dataset contains twenty-four classes of body-and-hand gestures used by the US Navy in aircraft handling aboard aircraft carriers. Body features include 3D velocities of four body joints (left/right elbows and wrists), resulting in a 12D feature vector. Hand features include probability estimates of four predefined hand shapes (opened/closed palm, and thumb up/down), resulting in an 8D feature vector. We normalized each dimension to have a mean zero and standard deviation one.

SOIT: The Synthetic Organizational Insider Threat (SOIT) dataset contains data created by a research consortium focusing on insider threat detection. To generate a benchmark dataset, the consortium appointed a group of domain experts (in computer security, counter intelligence, data simulation, etc) to create synthetic data simulating the digital footprints of employee's organizational behavior. In our experiments, we used a subset of the data on 1,000 employee's activities over 500 days, which contained records of approximately 2.6M emails, 0.5M file access, 28.4M web access, 0.9M log on/off, etc. The subset contained 70 realistic insider threat scenarios (e.g., corporate espionage, information leakage) for which exact periods were available. The 29D feature vector included various features related to email, web access, file access, session logs, etc. We divided each employee's sequence of records by finding sequences containing at most 30 days with the same label. This resulted in 16,921 normal instances and 166 anomalies. We normalized each dimension to have a mean zero and standard deviation one.

4.2 Models

Active Outlier (AO) [Abe *et al.*, 2006][2]: This method, based on an ensemble-based minimum margin active learning, augments the given one-class dataset with synthetic abnormal samples generated by rejection sampling. We use the decision tree as base learner, and assume the sampling distribution as uniform. We varied the number of ensemble learners `nbLearners`=[4 8 12] and the outlier `threshold` from 0.1 to 0.9, increasing by 0.1.

Local Outlier Factor (LOF) [Breunig *et al.*, 2000][2]: This density-based method measures the degree of a sample being an outlier. It is defined as the average ratio of the local density of a sample $\mathbf{x}$ and those of $\mathbf{x}$'s `MinPts`-nearest neighbors. We compute the LOF of each test sample according to the training dataset. After the LOFs of all test samples are computed, we normalize them to have the value between [0, 1] and classify each sample as abnormal if the normalized LOF is higher than a `threshold`. Following the guideline described in [Breunig *et al.*, 2000], we set the lower bound `MinPtsLB`=10 and the upper bound `MinPtsUB`=30, with a step size of 5. The outlier `threshold` was varied from 0.1 to 0.9, increasing by 0.1.

Hidden Markov Model (HMM) [Rabiner, 1989][3]: We trained HMMs using normal training sequences, and computed $p(y|\mathbf{x})$ of each test sequence as the normalized negative log-likelihood. Sequences were classified as abnormal if $p(y = -1 \mid \mathbf{x}) > 0.5$. We varied the number of hidden states $|\mathcal{H}|$=[4 8 12] and the number of Gaussian mixtures per state `nbMixtures`=[1 2 3].

One-Class SVM (OCSVM) [Schölkopf *et al.*, 2001][4]: This support vector method estimates a subset of input space such that a test sample lies outside the subset equals a pre-specified parameter $\rho \in$ [0,1). As in [Schölkopf *et al.*, 2001], we used a Gaussian kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2)$, which makes the data always separable from the origin in feature space. We validated the kernel width γ=[.01 .1 .5 1 2] and the offset parameter ρ from 0.1 to 0.9, increasing by 0.1.

One-Class CRF (OCCRF): For our OCCRF, we used three types of feature functions defined in [Quattoni *et al.*, 2007]. Let $\mathbb{1}[\cdot]$ be an indicator function, and $y' \in \mathcal{Y}$ and $h', h'' \in \mathcal{H}$. The *observation* function $\phi(h_t, \mathbf{x}_t) = \mathbb{1}[h_t = h']\mathbf{x}_t$ captures the compatibility between a latent variable h_t and an observation $\mathbf{x}_t$; the *label* feature function $\phi(y, h_t) = \mathbb{1}[y = y' \wedge h_t = h']$ captures the compatibility between a label y and a latent variable h_t; and the *transition* feature function $\phi(y, h_s, h_t) = \mathbb{1}[y = y' \wedge h_s = h' \wedge h_t = h'']$ captures the compatibility among a label y and two latent variables h_s, h_t. We varied the number of latent states $|\mathcal{H}|$=[4 8 12], the regularization factor λ=[.01 .1 .5 1 2], and the margin offset ρ from 0.1 to 0.9, increasing by 0.1.

4.3 Methodology

Similar to [Ghasemi *et al.*, 2012], we measure the performance of models using the F1 score in the top k returned results (F1@k), setting the k to the number of abnormal samples in the test set. We also show both the precision and recall in the top k (Prec@k and Rec@k) for completeness.

We performed 10-fold cross validation, where one-tenth of the normal samples are used for training split, and the rest is split evenly between validation and test splits, repeating 10 times. Note that the training split contained only normal

[2]We used an implementation of [Erdogan, 2011].

[3]We used an implementation of [Murphy, 1998].

[4]We used an implementation of [Chang and Lin, 2011].

Datasets	Models	Validation Split			Test Split		
		Prec@k	Rec@k	F1@k	Prec@k	Rec@k	F1@k
CUAVE	AO	.9527 ± .02	.5426 ± .09	.6865 ± .07	.9592 ± .02	.5359 ± .09	.6827 ± .08
	LOF	.9127 ± .02	.4597 ± .15	.5948 ± .13	.9340 ± .02	.4801 ± .15	.6173 ± .13
	HMM	.9598 ± .04	.1552 ± .04	.2593 ± .05	.9890 ± .01	.1547 ± .04	.2642 ± .06
	OCSVM	.9025 ± .01	.7372 ± .10	.8048 ± .06	.8896 ± .01	.7433 ± .10	.8034 ± .06
	OCCRF	**.9037 ± .01**	**.8763 ± .06**	**.8875 ± .03**	**.9008 ± .01**	**.8836 ± .06**	**.8900 ± .03**
ArmGesture	AO	.9954 ± .01	.7272 ± .16	.8257 ± .12	.9932 ± .01	.7406 ± .16	.8367 ± .11
	LOF†	.9944 ± .01	.7512 ± .09	.8398 ± .07	.9939 ± .01	.7662 ± .09	.8487 ± .07
	HMM	.9976 ± .01	.1712 ± .11	.2680 ± .14	.9606 ± .05	.1672 ± .12	.2568 ± .15
	OCSVM	.5917 ± .09	.3869 ± .12	.4550 ± .11	.5701 ± .10	.3696 ± .12	.4333 ± .12
	OCCRF	**.8397 ± .00**	**.9999 ± .00**	**.9114 ± .00**	**.8454 ± .00**	**1.000 ± .00**	**.9153 ± .00**
NATOPS	AO	.9042 ± .20	.6755 ± .41	.6957 ± .40	.8958 ± .23	.6756 ± .41	.6957 ± .40
	LOF	.9893 ± .00	.5799 ± .07	.7099 ± .05	.9916 ± .00	.5802 ± .07	.7097 ± .06
	HMM	.9908 ± .00	.2972 ± .05	.4427 ± .06	.9957 ± .00	.2877 ± .05	.4294 ± .06
	OCSVM†	.9582 ± .00	.9218 ± .01	.9387 ± .01	.9577 ± .00	.9223 ± .01	.9387 ± .00
	OCCRF	**.9609 ± .00**	**.9343 ± .03**	**.9464 ± .02**	**.9605 ± .00**	**.9329 ± .03**	**.9454 ± .02**
SOIT	AO	.0000 ± .00	.0000 ± .00	.0000 ± .00	.0000 ± .00	.0000 ± .00	.0000 ± .00
	LOF	.0000 ± .00	.0000 ± .00	.0000 ± .00	.0000 ± .00	.0000 ± .00	.0000 ± .00
	HMM	.1809 ± .12	.7000 ± .26	.2604 ± .10	.2718 ± .32	.3667 ± .20	.2518 ± .21
	OCSVM	.0132 ± .01	.7250 ± .45	.0247 ± .02	.2012 ± .28	.5121 ± .28	.1748 ± .08
	OCCRF	**.6725 ± .05**	**.7745 ± .08**	**.7154 ± .01**	**.6607 ± .02**	**.7554 ± .09**	**.7024 ± .03**

Table 2: A summary of experimental results on both validation and test splits, showing our OCCRF consistently outperforming four other baseline models in terms of the F1@k score. The differences were statistically significant, except for the LOF on the ArmGesture dataset and the OCSVM on the NATOPS dataset (marked with †).

samples; only the validation and test splits contained abnormal samples. The optimal hyper-parameter setting of each fold was selected based on the highest F1@k on the validation split. For temporal models, i.e., HMM and OCCRF, we treated each sequence as an individual sample (as was given). For non-temporal models, i.e., LOF, AO, OCSVM, we treated each frame of a sequence as an individual sample; when computing an anomaly score of a test sequence, we computed the score of each frame and took an average of them.

4.4 Results and Discussion

Table 2 shows experimental results on both validation and test splits. It shows that our OCCRF outperformed all four baseline models across all datasets in terms of F1@k scores. A two-sample t-test was used to determine if the differences between our model and the baselines are statistically significant in terms of the F1@k scores; most of the comparisons showed statistical significance except for the LOF on the ArmGesture dataset and the OCSVM on the NATOPS dataset (marked with † in Table 2).

On the three multi-class datasets (i.e., CUAVE, ArmGesture, NATOPS), the HMM showed high precision rates but low recall rates, resulting in low F1@k scores. This indicates that training a standard multi-class classification model using one-class data does not perform well. The three other baselines (i.e., AO, LOF, OCSVM) performed similarly depending on datasets, but no model performed consistently well across datasets, indicating high sensitivity of the models to the data type. The performance of our OCCRF was consistent across datasets and hence indicates low sensitivity compared to the baselines. These findings were consistent between the validation and test splits, showing that there was no sampling bias between the two splits.

The class distribution of the SOIT dataset was quite different from the rest. The mean percentage of abnormal samples in the test split was only 1.08% for the SOIT dataset (83 anomalies out of 7,698 samples), while it was 90.09% for the CUAVE dataset, 83.00% for the ArmGesture dataset, and 95.82% for the NATOPS dataset. The SOIT dataset, therefore, better reflected a realistic scenario of sequential anomaly detection, where only a fraction of observations are true anomalies. Our results show that both the AO and the LOF failed to detect *any* of the 83 anomalies during testing (both on validation and test splits); the HMM and the OCSVM also showed quite unsatisfactory performance compared to our OCCRF model.

5 A Qualitative Exploratory Study

In this section we report on an exploratory study on detecting employees' abnormal organizational behavior in enterprise social networks [Lin *et al.*, 2012]. Specifically, we aim to identify two types of anomalies: *positive* anomalies, employees who can potentially contribute to the success of the organization; and *negative* anomalies, employees who are in need of attention and help, or who can become a possible insider threat. To give a loose definition of the two types of anomalies, we assume that positive anomalies result in promotion and/or change in job role/location, and that negative anomalies are indicated by resignation or new employees (joined the company within a year) changing their managers. We assume that such anomalies can be observed from organizational behavior, and that email conversation is a good proxy for it since employees spend a lot of their work time on email.

Because it is desirable to detect anomalies well *before* an actual action is executed, we set our goal as an early detection of the four above anomalous action categories, i.e., promotion, change in job role/location, resignation, new employee change manager. To this end, we compute an anomaly score $p(y = -1 \mid \mathbf{x})$ of each employee during a specific period, single out employees with high anomaly score, and look up the corporation's database to see if the employees match one of the four anomalous action categories *after* that period.

To evaluate our model's ability to detect sequentially anomalous patterns, we compare the list of anomalous employees obtained from our OCCRF to that of OCSVM. Specifically, we check to see if there are sequentially anomalous traits of emails from each employee that are detected by OCCRF but not by OCSVM.

5.1 Dataset and Methodology

We collected two year's worth of email communication of 8,952 volunteer employees within a global technology company that spans over 400,000 employees and 70 countries. After eliminating spam emails and mass announcements, the dataset contained approximately 20M email samples. Each sample contained the sender, receiver(s), time stamps, and the body of email. We augmented each sample with sender's personal attributes, including job role, departmental affiliation, and report-to relation with managers. Email addresses were hashed to ensure privacy. Because of the exploratory nature of this dataset, the ground-truth labels were not available; however, the majority of employees (more than 60%) remained in stable positions during this period.

We used the data from the last quarter of the first year, which contained about 1M email records from 5,047 employees. This period was particularly interesting because it was about the peak of financial crisis, where there could be more anomalies (especially resignations). Since our goal was to identify anomalous employees based on a series of emails, we defined sequence as emails from each employee sorted in chronological order. The content of each email was processed using LDA [Blei *et al.*, 2003] with 200 latent topic classes to extract a 200D feature vector, with each dimension indicating the weight to each topic class.[5] The median length of sequence was 110. PCA was used to reduce the dimension of the LDA feature vector to 100, which accounted for 90% of the variation in the dataset.

Experiments were performed following the 10-fold hold-out approach, where one-tenth of the dataset was held out for testing and the rest was used for training, repeating 10 times. We trained both the OCSVM and the OCCRF assuming most of the data in the training split is positive. For OCSVM, we set the offset parameter ρ=0.9 and the Gaussian kernel width γ=0.1. For OCCRF, we set the number of latent states $|\mathcal{H}|$=12, the regularization factor λ=0.1, and the margin offset parameter ρ=0.9. There was no cross-validation of the parameters because the ground-truth label was not available. After evaluating each test split, we picked those employees with an anomaly score $p(y = -1 \mid \mathbf{x})$ higher than 0.9.

[5]We used PLDA+ [Liu *et al.*, 2011] with a default configuration.

5.2 Result and Discussion

Our OCCRF identified 151 employees as anomalous, of which 45 employees were positive anomalies (26 promotions, 19 changes in job role/location) and 67 employees were negative anomalies (53 resignations, 14 new employees changed manager); the remaining 39 employees were normal, i.e., falsely identified as anomalies. This resulted in a precision@k rate of 0.7417, where k is set to the number of employees whose anomaly score was higher than 0.9 (151). On the other hand, the OCSVM identified 21 employees as anomalous, of which 3 employees were positive anomalies (2 promotions, 1 change in job role/location), 9 employees were negative anomalies (7 resignations, 2 new employees changed manager), and 9 falsely identified anomalies, resulting in a precision@k rate of 0.5714. This comparison shows that our OCCRF identified more anomalies with higher precision.

Since the majority type of anomalies detected using OCCRF was in the category 'resignation' (53 out of 112), we further analyzed the input sequence (LDA features) of the 53 resigned employees. To understand the characteristic of each sequence, we examined the maximum weighted LDA topic in each frame. Interestingly, 9 out of 53 sequences contained a topic class *interview* (topic words: candidate, position, recruit) and later followed by a topic class *decision* (topic words: position, potential, proposal, recommend), indicating that those employees have sent emails related to job interview, and later sent other emails related to the decision of interviews. Although this is a simple and illustrative example, the OCSVM did not show such cases, which demonstrates the ability of our model to capture anomalous temporal patterns.[6]

6 Conclusion

We presented One-Class Conditional Random Fields for sequential anomaly detection. It follows the learning strategy of Scholkopf *et al.* [2001] and accepts most of the training examples as normal, while making the solution space as tight as possible. Our main contribution is an extension of this learning strategy to the temporal sequence domain using CRFs. We developed a hinge loss in a regularized risk minimization framework that maximizes the margin between each sequence being classified as "normal" and "abnormal," which allows our model to deal with one-class data and capture the temporal dependence structure. Experimental results on various real-world datasets show our model outperforming several state-of-the-art baseline methods.

One advantage of our model is the ability to make use of various advances in CRF models, as for example the use of kernels [Lafferty *et al.*, 2004] or neural networks [Peng *et al.*, 2009] to capture non-linear relationship in complex real-world data. We look forward to experimenting with these extensions on various real-world anomaly detection scenarios.

[6]Subsequent investigation using the corporation's database revealed that resigned employees identified by OCCRF have sent many emails to their weak-ties within and outside of the company, with whom they had little interactions before. To determine the weak-ties we used the tie strength measure in [Lin *et al.*, 2012].

References

[Abe *et al.*, 2006] Naoki Abe, Bianca Zadrozny, and John Langford. Outlier detection by active learning. In *KDD*, 2006. 1, 2, 4

[Blei *et al.*, 2003] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3, 2003. 6

[Breunig *et al.*, 2000] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof: Identifying density-based local outliers. In *SIGMOD*, 2000. 1, 2, 4

[Chandola *et al.*, 2009] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3), 2009. 1, 2

[Chandola, 2009] Varun Chandola. *Anomaly detection for symbolic sequences and time series data*. PhD thesis, University of Minnesota, 2009. 2

[Chang and Lin, 2011] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM TIST*, 2(3):27, 2011. 4

[Cheng *et al.*, 2009] Haibin Cheng, Pang-Ning Tan, Christopher Potter, and Steven A. Klooster. Detection and characterization of anomalies in multivariate time series. In *SDM*, 2009. 1

[Do and Artieres, 2012] Trinh-Minh-Tri Do and Thierry Artieres. Regularized bundle methods for convex and non-convex risks. *Journal of Machine Learning Research*, 13, 2012. 3

[Erdogan, 2011] Goker Erdogan. Outlier Detection Toolbox in Matlab, 2011. 4

[Ghasemi *et al.*, 2012] Alireza Ghasemi, Hamid R. Rabiee, Mohammad Taghi Manzuri, and Mohammad Hossein Rohban. A bayesian approach to the data description problem. In *AAAI*, 2012. 5

[He and Garcia, 2009] Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.*, 21(9), 2009. 1

[Lafferty *et al.*, 2001] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001. 1, 2, 3

[Lafferty *et al.*, 2004] John D. Lafferty, Xiaojin Zhu, and Yan Liu. Kernel conditional random fields: representation and clique selection. In *ICML*, 2004. 7

[Liao *et al.*, 2010] Zicheng Liao, Yizhou Yu, and Baoquan Chen. Anomaly detection in gps data based on visual analytics. In *IEEE VAST*, 2010. 2

[Lin *et al.*, 2012] Ching-Yung Lin, Lynn Wu, Zhen Wen, Hanghang Tong, Vicky Griffiths-Fisher, Lei Shi, and David Lubensky. Social network analysis in enterprise. *Proceedings of the IEEE*, 100(9), 2012. 5, 6

[Liu *et al.*, 2011] Zhiyuan Liu, Yuzhou Zhang, Edward Y. Chang, and Maosong Sun. PLDA+: Parallel Latent Dirichlet Allocation with Data Placement and Pipeline Processing. *ACM TIST*, 2011. Software available at http://code.google.com/p/plda. 6

[Murphy, 1998] Kevin Murphy. Hidden Markov Model (HMM) Toolbox for Matlab, 1998. 4

[Parzen, 1962] Emanuel Parzen. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962. 2

[Patterson *et al.*, 2002] Eric K. Patterson, Sabri Gurbuz, Zekeriya Tufekci, and John N. Gowdy. Cuave: A new audio-visual database for multimodal human-computer interface research. In *ICASSP*, 2002. 4

[Pearl, 1982] Judea Pearl. Reverend bayes on inference engines: A distributed hierarchical approach. In *AAAI*, 1982. 3

[Peng *et al.*, 2009] Jian Peng, Liefeng Bo, and Jinbo Xu. Conditional neural fields. In *NIPS*, 2009. 7

[Quattoni *et al.*, 2007] Ariadna Quattoni, Sy Bor Wang, Louis-Philippe Morency, Michael Collins, and Trevor Darrell. Hidden conditional random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(10), 2007. 1, 3, 4

[Rabiner, 1989] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, 1989. 1, 4

[Saenko and Livescu, 2006] Kate Saenko and Karen Livescu. An asynchronous dbn for audio-visual speech recognition. In *SLT*, 2006. 4

[Schölkopf *et al.*, 2001] Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7), 2001. 1, 2, 4, 6

[Song *et al.*, 2011] Yale Song, David Demirdjian, and Randall Davis. Tracking body and hands for gesture recognition: Natops aircraft handling signals database. In *FG*, 2011. 4

[Sun *et al.*, 2006] Pei Sun, Sanjay Chawla, and Bavani Arunasalam. Mining for outliers in sequential databases. In *SDM*, 2006. 2

[Tan *et al.*, 2011] Swee Chuan Tan, Kai Ming Ting, and Fei Tony Liu. Fast anomaly detection for streaming data. In *IJCAI*, 2011. 2

[Tax and Duin, 2004] David M. J. Tax and Robert P. W. Duin. Support vector data description. *Machine Learning*, 54(1), 2004. 2

[Teo *et al.*, 2010] Choon Hui Teo, S. V. N. Vishwanathan, Alex J. Smola, and Quoc V. Le. Bundle methods for regularized risk minimization. *Journal of Machine Learning Research*, 11, 2010. 3

[Xu, 2010] Xin Xu. Sequential anomaly detection based on temporal-difference learning: Principles, models and case studies. *Appl. Soft Comput.*, 10(3), 2010. 2

[Yu and Joachims, 2009] Chun-Nam John Yu and Thorsten Joachims. Learning structural svms with latent variables. In *ICML*, page 147, 2009. 1, 2, 3