DICTIONARY LEARNING WITH UNIFORM SPARSE REPRESENTATIONS FOR ANOMALY DETECTION

Paul Irofti, Cristian Rusu, Andrei Pătrașcu

Research Center for Logic, Optimization and Security (LOS), Department of Computer Science, Faculty of Mathematics and Computer Science, University of Bucharest, Romania

ABSTRACT

Many applications like audio and image processing show that sparse representations are a powerful and efficient signal modeling technique. Finding an optimal dictionary that generates at the same time the sparsest representations of data and the smallest approximation error is a hard problem approached by dictionary learning (DL). We study how DL performs in detecting abnormal samples in a dataset of signals. In this paper we use a particular DL formulation that seeks uniform sparse representations model to detect the underlying subspace of the majority of samples in a dataset, using a K-SVD-type algorithm. Numerical simulations show that one can efficiently use this resulted subspace to discriminate the anomalies over the regular data points.

***Index Terms*—** anomaly detection, dictionary learning, sparse representation

1. INTRODUCTION

Dictionary learning (DL) is a decomposition method with many applications to audio and image processing, compression, classification, and computer vision, where it gives better performance than popular transforms. Given the training data, DL builds a dictionary and sparse representations corresponding to data points by minimization of the approximation error, imposing the desired limits on coefficients sparsity.

Intuitively, the generic anomaly detection (AD) problem consists of finding particular points in a given dataset, called anomalies or outliers, that are not conformal to the majority of the rest of the data points (called inliers).

DL algorithms construct representation vectors that have an unstructured support distribution, i.e., the sparsity pattern is unstructured and uses many of the theoretical possible subspaces to represent the data. When done this way, there is no indication of the existence of a common subspace that generates all the training data. But, we now assume that the regular majority of signals is generated by the same atoms.

Prior work. The idea of enforcing uniform support representations is not new, for instance in [1, 2] it is used in feature selection problems in combination with squared Euclidean loss and other various robust losses. The work in [3] introduces the Simultaneous Orthogonal Matching Pursuit (S-OMP) method to solve the sparse approximation step while also balancing the number of elementary signals that are used in the representations. The Joint Sparse Representation (JSR) model analyzed in [4, 5] assumes a multi-class partitioning of the input, where each class spans a low-dimensional subspace. In [6], ℓ_1 penalty-based JSR is used for detecting noisy anomalies with prefixed dictionary. Although their formulations are similar to ours, the authors aim to compute only specific Sparse Representations that highlight diversity in the dataset.

Some references on DL-based AD in images are listed further, however note that these do not tackle the standard DL problem and thus do not generalize well. In [7], local dictionaries with specific structures are enhanced based on information from neighbors for detecting abnormal images. The convolutional sparse coding model is exploited in [8] in order to learn a dictionary of filters used for the same task. Particular DL and SR formulations for AD in network traffic and telemetry are analyzed in [9, 10, 11]. Empirical evidence on detection of anomalous images and electrocardiographic data are given in [12, 13, 14, 10, 15].

Contribution. In this paper, we first reformulate the DL problem and add a row sparsity regularization, by replacing the usual column sparsity penalty. We approach two particular penalties, ℓ_1 norm and ℓ_0-"norm", aiming to enforce an entire row of representations matrix X to be null while at the same time allowing subspace differentiation within the remaining rows. Prior work uses regularization and other techniques to impose row sparsity in X such that the final representations lie on the same subspace; our work also imposes row sparsity but gains a competitive edge by allowing subspace differentiation within the selected rows. By taking ad-

Paul Irofti was supported by a grant of the Romanian Ministry of Education and Research, CNCS - UEFISCDI, project number PN-III-P1-1.1-PD-2019-0825, within PNCDI III. Andrei Pătraşcu was supported by a grant of the Romanian Ministry of Education and Research, CNCS - UEFISCDI, project number PN-III-P1-1.1-PD-2019-1123, within PNCDI III. Paul Irofti and Andrei Pătrașcu were also supported by a grant of the Romanian Ministry of Education and Research, CNCS - UEFISCDI, project number PN-III-P2-2.1-PED-2019-3248, within PNCDI III. Cristian Rusu was supported by the Romanian Ministry of Education and Research, CNCS-UEFISCDI, project number PN-III-P1-1.1-TE-2019-1843, within PNCDI III.

vantage of the sum decomposition of the new proposed regularizer, we devise a K-SVD-type algorithm with similar complexity as the usual K-SVD iteration [16, 17, 18]. We showcase the performance of our approach in Section 4.

Notations. Denote: x_i the i-th row, x^j the j-th column in matrix X.The ordered i-th left and right singular vectors of matrix X are $u^i(X)$ and $v_i(X)$, respectively. The "norm" $\|\cdot\|_0$ counts the number of nonzero elements of a vector. We use $[n] = \{1, \cdots, n\}$ for some $n \geq 1$. The set of column normalized matrices is $\mathcal{N}_{m,n} = \{D \in \mathbf{R}^{m\times n} : \|d^j\| = 1, \forall j \in [n]\}$.

2. PROBLEM FORMULATION

Let Y be the input data, the basic DL problem is:

$$\min_{X, D\in\mathcal{N}_{m,n}} \|DX - Y\|_F^2 + \lambda \sum_{i=1}^{N} \|x_i\|_0$$

where d^j is the j-th atom of the dictionary D and X is the representation matrix. The above ℓ_0 regularization promotes unstructured sparsity in the columns of X. Unfortunately this does not reveal any underlying joint properties of signals Y (Fig.1 left). Existing methods replace this standard regularization in order to promote a similar support among the columns of X (Fig.1 center). Therefore, to benefit from both, our aim is to preserve the sparsity pattern from both coordinates (Fig.1 right). The general model of interest is:

$$\min_{X, D\in\mathcal{N}_{m,n}} \frac{1}{2}\|DX - Y\|_F^2 + \lambda \sum_{i=1}^{n} \phi(\|x_i\|_2). \tag{1}$$

where ϕ is a sparse regularizer. Although a similar intuition is shared by JSR, which aims to obtain sparsity pattern as in the center of Fig. 1, we argue later that our algorithm could preserve the sparsity on both coordinates, illustrated in Fig. 1 (right), by nature of K-SVD iteration. For simplicity we further use notation $F(D, X) := \frac{1}{2}\|DX - Y\|_F^2 + \lambda \sum\limits_{i=1}^{n} \phi(\|x_i\|_2)$. Simultaneous minimization over subsets of $\{x_i\}_{i=0}^N$ that spans multiples rows makes the regularization hard even for convex ϕ. Our further approach involves alternating minimization over one row at each iteration in order to obtain an algorithmic scheme with simple steps.

3. ALGORITHMS AND METHODOLOGY

The K-SVD algorithm introduced in [17] selects at each iteration k an index i_k and, based on the information at previous step $k-1$, minimizes the residual over d^{i_k} and x_{i_k}, while it keeps unchanged $d^j = (d^j)^k$ and $x_j = (x_j)^k$ for $j \neq i_k$. The adaptation of this K-SVD reasoning to our regularized model leads to the following: at iteration k, choose $i \in [N]$

$$(d^{i,k+1}, x_i^{k+1}) = \arg\min_{\|d^i\|=1, x_i} \frac{1}{2}\|d^i x_i - R^k\|^2 + \lambda\phi(\|x_i\|_2) \tag{2}$$

where $\{d^{j,k}, x_j^k\}$ are the $j-$th atom of the dictionary D^k and $j-$th row of X^k, respectively. Also we denote $R^k = Y - \sum_{j\neq i_k} d^{j,k} x_j^k$. If $\lambda = 0$, then $(d^{i,k+1}, x_i^{k+1})$ are the maximal left $u^1(R^k)$ and right $v_1(R^k)$ singular vectors. This iteration guarantees a decrease in the objective function F.

Proposition 1. *Let $\{D^k, X^k\}_{k\geq 0}$ be the sequence generated by the Algorithm 1. Then the following decrease hold:* $F(D^{k+1}, X^{k+1}) \leq F(D^k, X^k) \qquad \forall k \geq 0.$

Proof. The iteration (2) claims that $(d^{i,k+1}, x_i^{k+1})$ is the minimizer of the right-hand side objective. Therefore, the value of this objective in $(d^{i,k+1}, x_i^{k+1})$ is lower than its evaluation in the previous iterate. Thus, by using this fact we have:

$$\begin{aligned} F(D^{k+1}, X^{k+1}) &= \frac{1}{2}\|d^{i,k+1}x_i^{k+1} - R^k\|^2 \\ &\quad + \lambda\phi(\|x_i^{k+1}\|_2) + \lambda\sum_{j\neq i}\phi(\|x_j^k\|_2) \\ &\leq \frac{1}{2}\|d^{i,k}x_i^k - R^k\|^2 + \lambda\sum_i \phi(\|x_i^k\|_2) = F(D^k, X^k). \end{aligned}$$

□

We further show that the above algorithm allows explicit forms of the solution at each iteration (2) for some important cases when ϕ identifies with the most used sparse penalties.

3.1. Convex $\ell_{2,1}$ regularization

Let $\phi(z) = z$, then in this particular case the regularizer of (1) becomes the widely known sparse penalty $\|X\|_{2,1}$.

Proposition 2. *Let $\phi(x) = x$ and σ_1 be the maximal singular value of R^k, then the closed form solution of K-SVD iteration* (2) *is: if $\sigma_1 \geq \lambda$ then*

$$(d^{i,k+1}, x_i^{k+1}) = \left(u^1(R^k), (\sigma_1 - \lambda)v_1(R^k)\right), \tag{3}$$

otherwise $(d^{i,k+1}, x_i^{k+1}) = (d^{i,k}, 0)$.

Proof. For simplicity, we redenote $d := d^{i,k+1}, x := x_i^{k+1}$ and $t := \|x_i^{k+1}\|$. The singular values of R^k are called σ_i. We represent d and x in the SVD basis of R^k: $d = \sum_{i=1}^m \rho_i u^i(R^k)$ and $x = \sum_{i=1}^N \theta_i v_i(R^k)$ to expand the objective in the new form:

$$F(D^{k+1}, X^{k+1}) = \frac{1}{2}\|R^k\|_F^2 - x^T(R^k)^T d + \frac{1}{2}\|x\|^2 + \lambda\|x\|.$$

Note that we can write $(R^k)^T d = \sum_{i=1}^r \sigma_i v_i(R^k) u^i(R^k)^T d = \sum_{i=1}^r \sigma_i\rho_i v_i(R^k)$. Then the minimization of F becomes:

$$\min_{t\geq 0,\theta,\rho} \lambda t + \frac{1}{2}t^2 - \sum_{i=1}^r \sigma_i\rho_i\theta_i + \|R^k\|_F^2. \tag{4}$$

Further, observe that by the Cauchy-Schwarz inequality: $\left(\sum_{i=1}^r \sigma_i\rho_i\theta_i\right)^2 \leq \left(\sum_{i=1}^r \sigma_i\rho_i\right)^2 \left(\sum_{i=1}^N \theta_i\right)^2 \leq \sigma_1^2 t^2$ yields that $(\theta^*, \rho^*) = (e_1, e_1)$ are optimal for any t in problem (4). Finally, the final form of (4) remains: $\min_{t\geq 0} \ \lambda t + \frac{1}{2}t^2 - \sigma_1(R^k)t + \|R^k\|_F^2$ which has solution $t^* = \max\{0, \sigma_1 - \lambda\}$. Observe that for $\lambda > \sigma_1$, the optimal row x_i^{k+1} is null. □

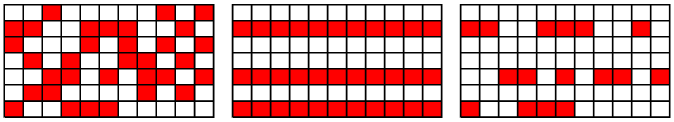

Fig. 1. The non-zeros entries (red squares) in the sparse representation matrix X for: the single measurement vector case (left), the multiple measurement vector – also called the simultaneous – case (center), and the proposed (right).

3.2. Nonconvex regularizers

There is wide evidence that nonconvex regularizers guarantees in some cases better performance, than convex ones, on unstructured sparse optimization problems [4]. We elaborate the explicit form of iteration (2) when ϕ is the ℓ_0 regularization, i.e. $\phi(x) = \|x\|_0$. At iteration k, the index $i \in [n]$ is chosen and the following subproblem is solved:

$$(d^{i,k+1}, x_i^{k+1}) = \arg \min_{\|d^i\|=1, x_i} \frac{1}{2}\|d^i x_i - R^k\|^2 + \lambda \| \|x_i\|_2 \|_0$$

Proposition 3. *Let $\phi(x) = \|x\|_0$, then the closed form solution of K-SVD iteration* (2) *is:*

$$(d^{i,k+1}, x_i^{k+1}) = (u^1(R^k), v_1(R^k)), \tag{5}$$

assuming $\frac{1}{2}\|u^1(R^k)v_1(R^k) - R^k\|^2 \leq \frac{1}{2}\|R^k\|^2 - \lambda$. *Otherwise* $(d^{i,k+1}, x_i^{k+1}) = (d^{i,k}, 0)$.

Proof. Obviously, when $\phi(x) = \|x\|_0$ there are only two possible cases: (i) the solution x_i^{k+1} is nonzero and, thus, is the same with the usual K-SVD update $\left(u^1(R^k), v_1(R^k)\right)$, see [16]; (ii) x_i^{k+1} is null when this value guarantees a larger descent on the local objective function, i.e. $\frac{1}{2}\|R^k\|^2 \leq \frac{1}{2}\|u^i(R^k)v_i(R^k) - R^k\|^2 + \lambda$. □

The penalty parameter λ represents the only degree of freedom that influences the number of 0-rows in the optimal X^*. Equivalently, large values of λ yields an increasing number of ignored atoms in the final sparse representations of Y. Since the large energy of row x_i might reflect a large importance of atom d^i, its elimination is undesirable. Thus, we might consider a truncated ℓ_2 norm, that promotes sparsity on the low-norm rows in X. In this case, the particular penalty $\phi(z) = \ell_\epsilon(z) := \min\{|z|, \epsilon\}$ would penalize only the components that are below threshold ϵ. Although iteration (2), with this form of ϕ, keeps a simple and explicit form, our experiments did not showed any improvement over ℓ_1 and ℓ_0 models.

3.3. The proposed algorithm

The proposed procedure is given in Algorithm 1 with separate: (*Training Procedure*) and (*Anomaly Detection*) sections.

In the training phase, we assume that an initial dictionary with n atoms is available (possibly, a random dictionary) and that the dataset Y was already split into the training and test sets of dimension N and $\hat{N}$, respectively. The training set does not contain any anomalies. The first step of the algorithm is to construct the sparse representations via the greedy Orthogonal Matching Pursuit (OMP) [19] algorithm that iteratively seeks a separate s-sparse representation for each signal in Y. Based on the sparsity pattern computed by OMP, each one of the following iterations will update all the atoms of the dictionary by the standard K-SVD approach (an SVD step on the residual matrix R^k but only on the columns that use the current atom) modified to take into account one of the regularizers we described in Section 3. The effect of the regularizer is to apply a joint sparsity constraint on the rows of X^k while preserving the sparsity pattern of the remaining rows (see Figure 1) originally decided by the OMP algorithm. We highlight that OMP runs a single time, at the start of Algorithm 1 and not with every iteration. The support set $\mathcal{I}$ contains the indices of the rows from X which are non-zero.

Once the training is complete and we have the dictionary D and the set $\mathcal{I}$, in the AD step we use the OMP algorithm to compute the sparse representations X_{test} on the test dataset $\hat{Y}$ and the we classify individually the data points as anomalies when in their sparse support there is a single atom from D which is not in the set $\mathcal{I}$. Note that standard DL algorithms, such as K-SVD, produce in general a uniformly distributed support across the representations X which make them unfeasible for AD: if $\mathcal{I} = [n]$ then $\mathcal{J}$ would always be included in $\mathcal{I}$ at step 16 and thus no anomalies would be detected.

Algorithm 1: Uniform DL Representation for AD

Data: train set $Y \in \mathbf{R}^{m \times N}$, test set $\tilde{Y} \in \mathbf{R}^{m \times \tilde{N}}$, $D^1 \in \mathbf{R}^{m \times n}$, sparsity s, iterations K,
Result: anomalies $\mathcal{A}$

1 *Training Procedure*
2 Representation: $X^1 = \text{OMP}(Y, D^1, s)$
3 **for** $k \in \{1, \ldots, K\}$ **do**
4   Error: $E^k = Y - D^k X^k$
5   **for** $i \in \{1, \ldots, n\}$ **do**
6   Atom error: $R^k = E^k + d^{i,k} x_i^k$
7   SVD rank-1 approximation: $R^k \approx u^1 \sigma_1 v_1$
8   K-SVD update: $(d_{\text{SVD}}^{i,k+1}, x_{i,\text{SVD}}^{k+1}) = (u^1, \sigma_1 v_1)$
9   Regularization: apply (3) or (5)
10   New error: $E^k = R^k - d^{i,k+1} x_i^{k+1}$
11   Uniform Support: $\mathcal{I} = \{i \mid \|x_i\|_0 \neq 0\}$

12 *Anomaly Detection*
13 Representation: $X_{\text{test}} = \text{OMP}(\tilde{Y}, D^{K+1}, s)$
14 **for** $i \in [\tilde{N}]$ **do**
15     $\mathcal{J} = \{j \mid x_{\text{test},j}^i \neq 0\}$
16   **if** $\mathcal{J} \nsubseteq \mathcal{I}$ **then** $\mathcal{A} = \mathcal{A} \cup \{i\}$

Table 1. Maximum AD accuracy, standard deviation in parenthesis and running times for real datasets.

Dataset	$(m, N, \text{outliers})$	DL-$\ell_0(\sigma)$		DL-$\ell_1(\sigma)$		OC-SVM		LOF		IForest	
satellite	(36, 6435, 2036)	**0.8059** (0.059)	0.57s	0.8020 (0.059)	0.77s	0.6391	0.36s	0.5677	0.2s	0.7062	0.14s
shuttle	(9, 49097, 3511)	0.8155 (0.109)	1.16s	0.9262 (0.107)	1.18s	0.6322	2.40s	0.5269	0.1s	**0.9771**	0.21s
pendigits	(16, 6870, 156)	0.7679 (0.110)	0.19s	**0.8822** (0.108)	0.29s	0.7748	0.03s	0.5895	0.01s	0.8612	0.11s
speech	(400, 3686, 61)	0.5510 (0.008)	2.96s	0.5485 (0.022)	6.89s	**0.5917**	0.03s	0.5	0.02s	0.5289	0.2s
mnist	(100, 7603, 700)	0.5882 (0.015)	7.31s	**0.5917** (0.013)	27.0s	0.5576	0.9s	0.5736	0.05s	0.5255	0.2s

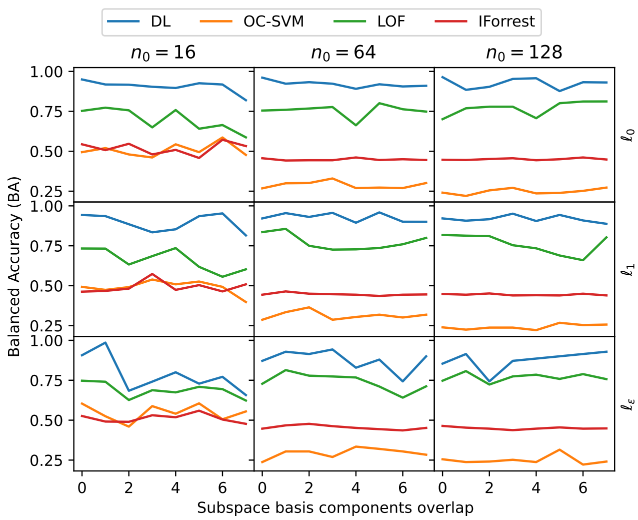

Fig. 2. Average accuracy of AD for synthetic data.

4. NUMERICAL EXPERIMENTS[1]

In this section, we provide synthetic and real-world numerical experiments to evaluate the performance of the proposed algorithm. We also compare against some of the state-of-the-art methods from the AD literature, but not with standard DL algorithms as they are not fit for AD. Throughout our experiments we use $K = 20$ and $s = 0.2\sqrt{m}$ and use 90% of the available inliers for training (the outliers are not included). We run on an AMD Ryzen Threadripper PRO 3955WX with 512GB of memory using Python 3.9.7 and Scikit-learn 1.0.

Our first experiment is based on synthetic data. We generate two dictionaries of sizes n_n for inliers and n_0 for outliers from which we generate $N + \tilde{N}$ signals with $m = 64$. Each signal is produced by randomly choosing s atoms from one of the dictionaries that produce a linear combination together with their associated coefficients drawn from the normal distribution. To harden the problem we also create an overlap between the atoms of the two dictionaries. We split the resulting dataset into the training set Y, and the testing data set $\tilde{Y}$ built from the outliers together with the remaining inliers. In the paper the outliers represent 10% of the total amount of testing signals. We have tested with similar results anomaly planting from 1 to 20 percent. DL starts with an $n = 128$ normalized randomly generated dictionary ($n \gg \max\{n_n, n_0\}$).

Figure 2 presents 9 rounds of experiments where we vary from $n_0 = 16$ to $n_0 = 128$ on the columns. The rows represent experiments with the regularizations from Section 3. Each plot presents results with different degrees of overlap between the original generating dictionaries and their effect on the balanced accuracy (BA). We choose BA because it averages the sensitivity and specificity of our models thus giving the reader a sense of both false positives and false negatives (the undetected anomalies).

For our second experiment, we have chosen 5 datasets that belong to the publicly available Outlier Detection DataSets (ODDS)[2].We chose these datasets to span a wide range of available features and number of outliers. In all cases, in the testing phase we use 10% of the inlier data and all the outlier data. We present the results in Table 1, including the standard deviation σ for our methods shown in parenthesis. The proposed method, with both regularizers ℓ_0 and ℓ_1, performs best in 3 out of 5 cases and stays competitive in the other two but has the largest running time among the methods we consider. The parameter λ of the proposed method is optimized by using a grid search. The competing methods, One Class - Support Vector Machine (OC-SVM) [20], Local Outlier Factor (LOF) [21], and Isolation Forest [22], were optimized through an extensive grid-search across multiple kernels, metrics and hyper-parameters (OC-SVM did not always convergence on Shuttle and MNIST).

5. CONCLUSIONS

In this paper we propose a new dictionary learning based anomaly detection scheme with uniform sparse representations. Our algorithm starts with an initial sparse support and proceeds only with regularized rank-1 update iterations. Avoiding sparse representation on each dictionary learning iteration allows us to guarantee a descent on the objective function regularized by $\ell_{2,0}$, $\ell_{2,1}$, and $\ell_{2,\epsilon}$ penalties. We also provide numerical experiments that confirm our method.

In the future we plan on providing an in-depth analysis of the parameters effect on the anomaly detection task in order to establish a more rigorous detection scheme.

[1] Python code at https://github.com/pirofti/AD-USR-DL

[2] http://odds.cs.stonybrook.edu/

6. REFERENCES

[1] F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint l-2, 1-norms minimization," *Advances in neural information processing systems*, vol. 23, 2010.

[2] G. Lan, C. Hou, F. Nie, T. Luo, and D. Yi, "Robust feature selection via simultaneous sapped norm and sparse regularizer minimization," *Neurocomputing*, vol. 283, pp. 228–240, 2018.

[3] J.A. Tropp, A.C. Gilbert, and M.J. Strauss, "Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit," *Signal Processing*, vol. 86, no. 3, pp. 572–588, 2006.

[4] H. Zhang, J. Li, Y. Huang, and L. Zhang, "A nonlocal weighted joint sparse representation classification method for hyperspectral imagery," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 6, pp. 2056–2065, 2013.

[5] J. Li, H. Zhang, L. Zhang, and L. Ma, "Hyperspectral anomaly detection by the use of background joint sparse representation," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 6, pp. 2523–2533, 2015.

[6] A. Adler, M. Elad, Y. Hel-Or, and E. Rivlin, "Sparse coding with anomaly detection," *Journal of Signal Processing Systems*, vol. 79, no. 2, pp. 179–188, 2015.

[7] S. Biswas and R.V. Babu, "Sparse representation based anomaly detection with enhanced local dictionaries," in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 5532–5536.

[8] D. Carrera, G. Boracchi, A. Foi, and B. Wohlberg, "Detecting anomalous structures by convolutional sparse models," in *2015 International Joint Conference on Neural Networks (IJCNN)*, 2015, pp. 1–8.

[9] T. Kierul, M. Kierul, T. Andrysiak, and L. Saganowski, "Sparse representation and dictionary learning for network traffic anomaly detection," in *Theory and Applications of Dependable Computer Systems*, W. Zamojski, J. Mazurkiewicz, J. Sugier, T. Walkowiak, and J. Kacprzyk, Eds., Cham, 2020, pp. 344–354, Springer International Publishing.

[10] J. Xing and C. Wu, "Detecting anomalies in encrypted traffic via deep dictionary learning," *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 734–739, 2020.

[11] B. Pilastre, L. Boussouf, S. Escrivan, and J.Y. Tourneret, "Anomaly detection in mixed telemetry data using a sparse representation and dictionary learning," *Signal Processing*, vol. 168, pp. 1–10, Mar. 2020.

[12] T. Andrysiak, "Sparse representation and overcomplete dictionary learning for anomaly detection in electrocardiograms," *Neural Computing and Applications*, vol. 32, pp. 1269–1285, 2018.

[13] G. Boracchi, D. Carrera, and B. Wohlberg, "Novelty detection in images by sparse representations," in *2014 IEEE Symposium on Intelligent Embedded Systems (IES)*, 2014, pp. 47–54.

[14] Y. Yuan, D. Ma, and Q. Wang, "Hyperspectral anomaly detection via sparse dictionary learning method of capped norm," *IEEE Access*, vol. 7, pp. 16132–16144, 2019.

[15] W. Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked rnn framework," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 341–349.

[16] B. Dumitrescu and P. Irofti, "Regularized K-SVD," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 309–313, 2017.

[17] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[18] P. Irofti and A. Băltoiu, "Unsupervised dictionary learning for anomaly detection," *arXiv preprint arXiv:2003.00293*, 2020.

[19] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *27th Asilomar Conf. Signals Systems Computers*, Nov. 1993, vol. 1, pp. 40–44.

[20] B. Schölkopf, R.C. Williamson, A.J. Smola, J. Shawe-Taylor, J.C. Platt, et al., "Support vector method for novelty detection.," in *NIPS*. Citeseer, 1999, vol. 12, pp. 582–588.

[21] M.M. Breunig, H.P. Kriegel, R.T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 93–104.

[22] T.L. Fei, M.T. Kai, K. M. Ting, and Z. Zhi-Hua, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*, 2008, pp. 413–422.