Signed Graph Laplacian for Semi-Supervised Anomaly Detection

Jihong Bae
Department of Mathematics
Sungkyunkwan University
Republic of Korea
baeji0904@skku.edu
(0009-0008-9351-6474)

Hyeongcheol Park
MaaS Division
S-Traffic Co., Ltd.
Republic of Korea
hyeongcheol.park@straffic.com
(0000-0003-4507-2991)

Minsoo Chung
MaaS Division
S-Traffic Co., Ltd.
Republic of Korea
minsoochung@straffic.com
(0009-0008-0904-3134)

Muhammad Toaha Raza Khan
Computer Engineering Department
Middle East Technical University
Turkish Republic of North Cyprus
khan@metu.edu.tr
(0000-0003-2180-0923)

Kyungchul Lee
MaaS Division
S-Traffic Co., Ltd.
Republic of Korea
dalma@straffic.com
(0009-0001-7468-9862)

***Abstract*—Anomaly detection is a cutting-edge technology in the fields of healthcare and machine failure detection. It is well known that the performance of anomaly detection can be improved with more labeled data. However, it is common to predict anomalous and normal data in where are large unlabeled data and small labeled data. Generally, using a large amount of labeled data can lead to high accuracy prediction. However, the cost of labeling data is expensive, which can lead to challenges in anomaly detection. To achieve high prediction rate of anomaly detection, it is required to utilize a large amount of unlabeled data. The only way to achieve high rates of anomaly detection using both unlabeled data and labeled data is to use semi-supervised learning. However, if semi-supervised learning is used without data preprocessing, there is a limitation to obtain high detection rates. To perform effectively preprocess, we propose a scheme that leverages graph theory and semi-supervised learning to address the limitation. The proposed scheme uses graph Laplacian to get high accuracy in situations where there is little labeled data and a lot of unlabeled data. We further extend our scheme by considering friendly-antagonistic interactions into graph Laplacian, which is called signed graph Laplacian. We show that using signed graph Laplacian can improve the performance of our anomaly detection scheme. Furthermore, we evaluate our proposed scheme on a variety of validated datasets and show that it outperforms state-of-the-art semi-supervised anomaly detection methods.**

***Index Terms*—anomaly detection, semi-supervised learning, graph Laplacian, signed graph Laplacian, self-training, label propagation, friendly-antagonistic interactions**

I. Introduction

Recent technological advancements have facilitated the diagnosis of brain tumors and other neurological disorders through Magnetic Resonance Imaging (MRI) scans. Additionally, these advancements have enabled remote fault detection in autonomous vehicles and traffic infrastructure. Traditionally, humans have been responsible for identifying health deterioration or mechanical malfunctions. However, human judgment is inherently susceptible to errors, which can lead to incorrect assessments [1]. Erroneous judgements can lead to delayed problem detection, potentially resulting in severe consequences, including loss of life.

To address this issue, anomaly detection has been an active area of research since the 1960s, to empower computers to automatically identify the anomalies [2]–[4]. Researchers have made significant efforts to develop techniques that allow machines to learn and distinguish between normal and abnormal data patterns. These efforts have resulted in the accumulation of big data across various fields. However, these data are typically unlabeled, making it challenging for machines to learn from them as the data does not indicate whether it is normal or abnormal. The process of labelling data is costly, presenting a significant obstacle, namely high cost of performing anomaly detection.

To address these limitations, we devise a semi-supervised learning approach to exploit a large amount of unlabeled data with a small amount of labeled data for anomaly detection. We also performed experimental analysis to demonstrate the cost-efficient solution with high efficiency. The rest of the manuscript is structured as follows. In Section II, we reviewed the literature on anomaly detection, especially the challenges and cost-effectiveness. In section III, we propose our scheme for classifying unlabeled data using the signed graph Laplacian. In section IV, we describe how to perform anomaly detection using machine learning based on the proposed method. In section V, we rigorously evaluate the performance of our proposed scheme. Finally, in section VI, we conclude the paper by summarizing the key findings of our research and discussing the broader implications of our work.

II. Related Work

In this section, we present a comprehensive review of the current state-of-the-art in anomaly detection techniques, with a particular focus on the limitations of existing methods. Anomaly detection, also known as outlier detection and novelty detection, has been the subject of numerous studies. However, achieving a high detection rate remains a significant challenge due to factors such as 'Unknownness', 'Heterogeneous anomaly classes', 'Rarity and class impact', and 'Diverse type of anomaly'. To address these challenges, researchers have proposed a novel deep learning-based anomaly detection method, referred to as deep anomaly detection [5].

Despite the advancements, developing effective anomaly detection methods for complex, high-dimensional data is still an open problem. To tackle this issue, the authors of [6] proposed Adversarially Learned Anomaly Detection, a technique that leverages Generative Adversarial Networks (GANs) for anomaly detection and uses adversarially learned features. Additionally, the authors of [7] introduced an enhanced scheme called Cycle-consistent Generative Adversarial Network, which capitalizes on the strengths of GANs in anomaly detection for time series data. These strengths include the ability to learn the distribution of normal data for effective anomaly discrimination and the capability to reverse the generated data back to the original data.

In [8], the authors introduce a technique for anomaly detection in graph data. This method integrates node-based and edge-based approaches, capitalizing on the relational information inherent in graph-based data. Furthermore, [9] presents a strategy to enhance the convergence of the graph Laplacian using a kNN self-tuned kernel. In a separate study, [10] proposes an advanced variant of the graph Laplacian. This variant incorporates both contrast and density affinity, leading to improved performance in semi-supervised learning problems that employ graph spectral analysis. Graph spectral analysis is a robust method for clustering objects based on their similarities, which are represented by a graph connecting these objects [11]–[13]. This enhanced semi-supervised learning (SSL) Laplacian facilitates improved spectral clustering.

In [14], the authors concentrate on fully unsupervised anomaly detection using an unlabeled training dataset, which comprises both normal and abnormal samples. They propose a method to enhance the robustness of one-class classification, which is trained on self-supervised representations (self-training). This method employs a data refinement process to effectively tackle the problem. The fundamental principle of self-training algorithms is based on a semi-supervised learning technique. This technique iteratively learns a classifier by assigning pseudo-labels to a set of unlabeled training samples, provided their margin exceeds a certain threshold [15].

In [16], the authors provide a comprehensive review of self-training schemes for semi-supervised learning, a field that has garnered increasing interest in recent years. They explore various strategies for selecting unlabeled samples for pseudo-labeling and offer an overview of different self-training variants and related schemes found in the literature. The authors also examine recent theoretical advancements in this research area and elucidate the key features of self-training employed in several renowned methods. Recently, self-training has been widely applied in unsupervised domain adaptation, extending beyond the realm of semi-supervised learning [17]–[19].

In [20], the authors introduce a label propagation algorithm based on a bipartite graph to tackle the issue of poor performance of label propagation when certain class regions lack prior labels. The algorithm constructs a bipartite graph using example constraints and devises a two-channel example generation strategy to ensure the representation of all classes in the generated examples. Furthermore, the authors propose a supervised update strategy that utilizes known classes to learn example constraints for supervising missing classes, thereby facilitating the generation of high-quality examples in both channels.

However, these studies face a significant challenge due to the domain shift between the source and target domains. The prediction confidence provided by the trained model may be strongly biased towards the source domain, so it is not reliable for generating pseudo-labels in the target domain. Consequently, the performances of self-training and label propagation can be substantially degraded due to the introduction of substantial errors in the pseudo-labels, stemming from the significant mismatch between the source and target domains. To solve this problem, there is a way to use large amounts of labeled data for anomaly detection. But collecting labeled data is not only expensive but also practically difficult.

We propose two novel deep anomaly detection methods that efficiently classify unlabeled data, unlike existing methods. Our methods apply semi-supervised learning techniques, such as self-training (ST) and label propagation (LP), using graph Laplacian. Our methods achieve superior performance in scenarios with a small amount of labeled data by providing additional class information to unlabeled data, which improves the classification accuracy. Moreover, we propose an improved graph Laplacian that incorporates friendly-antagonistic relationships between data using labeled data. This improvement enhances the performance in situations with a large amount of unlabeled data and a small amount of labeled data. We demonstrate that our improved graph Laplacian can boost the performance of our anomaly detection methods. We evaluate our methods on various validated datasets and show that they outperform state-of-the-art semi-supervised anomaly detection methods.

III. Signed graph Laplacian for classification of unlabeled data

In this section, we propose graph Laplacian and signed graph Laplacian methods for classifying unlabeled data. Graph Laplacian is a method for classifying graph data using the connection information between nodes in a graph. Signed graph Laplacian is a method for classifying graph data that contains signed relations (friendly-antagonistic interactions)

between nodes [21]–[23]. Based on the Laplacian graph, these relations are created by providing additional class information to unlabeled nodes. In this paper, we use the term *signed relations* to refer to the friendly-antagonistic interactions in a graph.

A. Classification using the Graph Laplacian (CGL)

In this section, we use graph Laplacian to preprocess the data. To construct a graph Laplacian, we use k-Nearest Neighborhood (kNN) to connect each node. Graph Laplacian is a matrix that represents the structural properties of a graph. Based on the connection information between nodes, we identify the structural characteristics of the graph using spectral coordinates. These coordinates can be used to classify graph data. The algorithm using spectral coordinates of graph Laplacian is as follows [24]:

Algorithm 1: Spectral Coordinates of Graph Laplacian

Input: Set of points $X = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^m$ to be clustered into k-subsets.
Output: Clustering assignment for each point in X.
Step 1: Construct the adjacency matrix $A \in \mathbb{R}^{n \times n}$ with $A_{ij} = e^{-\frac{\|x_i - x_j\|^2}{\sigma}}$ for $i \neq j$, and $A_{ii} = 0$.
Step 2: Form the diagonal matrix D with D_{ii} as the sum of the i-th row (or column) of A.
Step 3: Construct the Laplacian matrix $L = D - A$.
Step 4: Find $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_k$, the k largest eigenvectors of L (ensuring orthogonality for repeated eigenvectors). Form the matrix $E = [\mathbf{e}_1\ \mathbf{e}_2\ \ldots\ \mathbf{e}_k]$.
Step 5: Treat each row of E as a point in $\mathbb{R}^k$ and cluster them into k clusters using the kNN method or another algorithm.
Step 6: Assign the original point x_i to cluster j if and only if row i of matrix E was assigned to cluster j.

However, graph Laplacian has a drawback that it cannot accurately classify the data when the distributions of data from different classes overlap. It also does not capture the signed relations between nodes, which can indicate friendly or antagonistic interactions. To address these limitations, we use a scheme called *signed graph Laplacian*, which incorporates signed relations by providing additional class information to unlabeled nodes.

B. Classification using the Signed Graph Laplacian (CSGL)

Signed graph Laplacian is a scheme that improves the limitations of graph Laplacian in Section III-A. Signed graph Laplacian is a matrix that represents the structural properties of a graph by considering both the connection information between nodes and the signed relations between nodes. It classifies graph data using spectral coordinates similarly to graph Laplacian. However, unlike graph Laplacian, it has the advantage of providing additional class information to unlabeled nodes, making it easier to infer the relationship between labeled points. It also has the advantage of performing more accurate clustering by reflecting the signed relations between nodes. The algorithm using spectral coordinates of signed graph Laplacian is as follows:

Algorithm 2: Spectral Coordinates of Signed Graph Laplacian

To enhance Algorithm 1, the signed graph Laplacian algorithm incorporates the signed relations of the labeled points.
Input: Set of points $X = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^m$ to be clustered into k-subsets.
Output: Clustering assignment for each point in X.
Step 1: Form the adjacency matrix $A \in \mathbb{R}^{n \times n}$ defined by $A_{ij} = e^{-\frac{\|x_i - x_j\|^2}{\sigma}}$ if $i \neq j$, and $A_{ii} = 0$.
Step 2: If x_i and x_j are labeled points for some i, j, set $A_{ij} = +1$ when in the same class and $A_{ij} = -1$ when in different classes.
Step 3: Define the diagonal matrix D as D_{ii} being the sum of the i-th row (or column) of A.
Step 4: Based on Steps 1 and 2, construct the signed Laplacian matrix $S = D - A$.
Step 5: Find $\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_k$, the k largest eigenvectors of S (ensuring orthogonality for repeated eigenvectors). Form the matrix $F = [\mathbf{f}_1\ \mathbf{f}_2\ \ldots\ \mathbf{f}_k]$.
Step 6: Treat each row of F as a point in $\mathbb{R}^k$ and cluster them into k clusters using the kNN method or another algorithm.
Step 7: Assign the original point x_i to cluster j if and only if row i of matrix F was assigned to cluster j.

Consequently, the signed graph Laplacian overcomes the limitations of the graph Laplacian and performs superiorly in classifying data, even when the distributions of data from different classes overlap. In Section IV, we introduce a scheme for performing anomaly detection by applying the techniques.

IV. Semi-supervised learning with CGL and CSGL for anomaly detection

In this section, we present our method for anomaly detection that leverages unlabeled data classified by the schemes proposed in the previous section.

A. Semi-supervised learning algorithm

We use two semi-supervised learning algorithms, namely label propagation and self-training, that are compatible with CGL and CSGL. Label propagation is an algorithm that classifies data using graph theory and gradually assigns labels to unlabeled data that are similar to labeled data [2], [25]. Self-training is an algorithm that exploits the structural information of unlabeled data [26]. Since the data generated by CGL and CSGL capture the structural information of the original data, self-training can effectively utilize this information for classification. The learning process of anomaly detection consists of common operations and algorithm-specific operations.

B. Common Operations

To perform anomaly detection, we first generate data for semi-supervised learning based on the graph Laplacian and signed graph Laplacian proposed in Section III. We then apply label propagation and self-training to learn from this data. The common operations are as follows:

1) **Input Data:** We input data that contains normal and anomaly information. This data consists of a mixture of labeled and unlabeled data, assuming a scenario with abundant unlabeled data and a limited amount of labeled data—a common situation in the real world.
2) **Adjust Dimensions:** We adjust the dimensions of the data to reduce the dimensionality. This can lower the time complexity and prevent eigenvalues from diverging.
3) **Calculate Consistency:** We calculate the consistency between labeled and unlabeled data using weighted kNN [27]. This method can estimate the similarity between data points based on their distances. Range search and kNN are other possible methods for calculating consistency [27], [28], but they may have drawbacks such as producing isolated nodes or dissimilar connections.

C. Operations for Label Propagation

The operations performed by label propagation are as follows:

1) **Kernel Selection:** We use kNN as the kernel of label propagation, which can calculate the similarity between nodes based on their distance. Radial basis function (RBF) kernel is another option, but it may cause excessive or insufficient connections depending on the range parameter [27], [28].
2) **Label Propagation Formula:** We apply the label propagation algorithm based on the graph Laplacian matrix L. The formula for propagating the labels from the labeled data X to the unlabeled data is given by:

$$F = (I - \alpha L)^{-1} Y,$$

where Y is the initial label matrix, F is the final label matrix, and α is a damping factor. In component form, the label propagation formula can be written as:

$$y_i^{(t+1)} = \sum_{j \in N(i)} \alpha_{ij} y_j^{(t)},$$

where $y_i^{(t+1)}$ is the label of node i at iteration $t+1$, $N(i)$ is the set of neighbors of node i, α_{ij} is the weight of the edge between nodes i and j, and $y_j^{(t)}$ is the label of node j at iteration t.

D. Self-Training Operations

The following steps describe the self-training operations:

1) **Perform Self-Training:** Perform self-training using the k-best criterion [16], [26]. This criterion selects the k most confident predictions from the unlabeled data and adds them to the labeled data. It is more robust and flexible than the threshold criterion, which requires a well-calibrated classifier.
2) **Train Support Vector Machine (SVM) Model:** Train a support vector machine (SVM) model on the labeled data using the following objective function:

$$\theta = \arg\min_{\theta} \left[\frac{1}{2} \|\theta\|^2 + C \sum_{i=1}^{m} \max(0, 1 - y_i(\theta \cdot \mathbf{x}_i + b) \right],$$

where θ is the model parameter, $\mathbf{x}_i$ is a data point, y_i is the label of $\mathbf{x}_i$, and C is the regularization parameter.

After applying these steps, we obtain an anomaly detection model to classify the data into normal and abnormal classes.

V. Experiment

We aim to apply our proposed scheme to the health diagnosis and equipment anomaly detection industry. In this section, we conduct anomaly detection experiments using data from these fields to evaluate the performance of semi-supervised learning with CGL and CSGL.

A. Dataset

We used two datasets from Kaggle for our experiments: 'Machine Predictive Maintenance Classification' [29] and 'Heart Failure Prediction Dataset' [30]. We refer to them as 'maintenance' and 'heart failure' respectively. These datasets are suitable for anomaly detection tasks in the domains of health diagnosis and equipment maintenance.

B. Design of experiment

We split each dataset into 80% for training and 20% for validation. We performed semi-supervised learning with only 0.5% of the training data labeled and the rest unlabeled. We applied two semi-supervised learning algorithms: label propagation and self-training. We also generated different types of data for semi-supervised learning based on the graph Laplacian and signed graph Laplacian methods proposed in Section III. We denote the data types as follows:

1) Original dataset (Org): The original data without any transformation.
2) Laplacian dataset (L): The data transformed by the graph Laplacian method.
3) Signed Laplacian dataset (sL): The data transformed by the signed graph Laplacian method.

We combined each data type with each semi-supervised learning algorithm and obtained six variants: 'LP+Org', 'LP+L', 'LP+sL', 'ST+Org', 'ST+L' and 'ST+sL'. We used kNN as the kernel for both label propagation and self-training. We varied the value of k from 5 to 25 with a step size of 5 and repeated each experiment 100 times for each k. The source code, the datasets, and the intermediate data are available at [31].

C. Computing environment

As the size of data used in semi-supervised learning increases, more memory is required. For fast data preprocessing and semi-supervised learning, high-speed processor is required. The computer used in this paper is a general PC, but it is important to determine an appropriate computing resource considering the size of the data generated in an environment using this technology and the size of the data that needs to be processed [32]. The specifications of the computer used in the experiment are shown in Table I.

TABLE I
COMPUTER SPECIFICATION

Component	Specification
CPU	AMD Ryzen 9 5900X
RAM	DDR4-3200 96 GB
GPU	NVIDIA RTX 2060 12GB
OS	Ubuntu 22.04 LTS
Used Libraries	Python 3.11, scikit-learn 1.3.2, SciPy 1.11.3, pandas 2.1.1, NumPy 1.26.1

D. Experimental results

As depicted in Figure 1 and Figure 2, the prediction accuracy was higher when the data for semi-supervised learning was generated using the signed graph Laplacian. This suggests that preprocessing the data to account for the distribution of normal and abnormal data can enhance the results of semi-supervised learning.

The figures also demonstrate that adjusting the k-value appropriately can increase prediction accuracy by ensuring that nearby nodes in the same class are well-connected. Conversely, an inappropriate k-value can lead to disconnected nodes or connections between nodes of different classes, thereby reducing accuracy.

In these experiments, the 'ST+sL' method achieved an accuracy rate of 100%. The high detection rate of 'ST+sL' is likely due to its ability to approximate nonlinear data to linear data using the signed graph Laplacian, which is compatible with self-training using SVM as a base classifier.

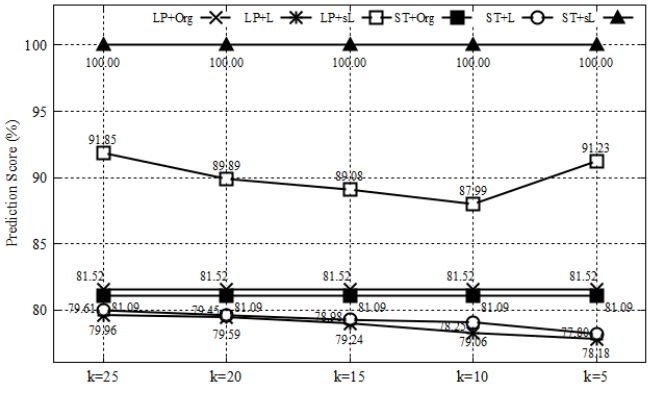

Fig. 1. Experiment on datasets of heart failure

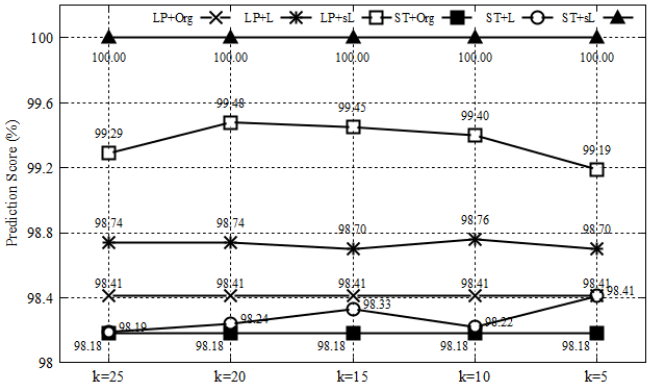

Fig. 2. Experiment on datasets of maintenance

VI. CONCLUSION

The proposed scheme is well-suited for environments where unlabeled data is abundant. It can achieve high accuracy at a low cost by performing semi-supervised learning with a large amount of unlabeled data and a small amount of labeled data. This is due to the scheme's use of the signed graph Laplacian's characteristics to classify unlabeled data into normal and abnormal categories.

However, the proposed scheme has several limitations. One such limitation is the high computational complexity involved in calculating the eigenvalues, which can lead to significant time and memory consumption when classifying unlabeled data from millions of inputs. Another limitation is that the scheme does not directly convert unlabeled data to labeled data. To address these limitations, we plan to conduct research on techniques for reducing complexity and auto-labeling.

In conclusion, the proposed scheme presents a promising approach to anomaly detection in environments where unlabeled data is accumulated. Our future research will aim to address the identified limitations and enhance the scheme's practicality for anomaly detection.

ACKNOWLEDGMENT

This work was supported by S-Traffic Co., Ltd.

REFERENCES

[1] Helen Higham, Charles Vincent, "Human Error and Patient Safety", Textbook of Patient Safety and Clinical Risk Management, Springer Nature, December 2020.
[2] Xiaojin Zhu, Zoubin Ghahramani, "Learning from labeled and unlabeled data with label propagation. Technical Report CMU-CALD-02-107," Carnegie Mellon University, June 2002.
[3] Frank E. Grubbs, "Procedures for Detecting Outlying Observations in Samples," Technometrics, vol. 11, no. 1, pp. 1-21, 1969.
[4] Raghavendra Chalapathy, Sanjay Chawla, "Deep Learning for Anomaly Detection: A Survey," arXiv:1901.03407, January 2019.
[5] Guansong Pang, Chunhua Shen, Longbing Cao, Anton Van Den Hengel, "Deep Learning for Anomaly Detection: A Review," ACM Computing Surveys, no. 38, Issue 2, vol. 54, pp. 1-38, March 2021.
[6] Houssam Zenati, Manon Romain, Chuan-Sheng Foo, Bruno Lecouat, Vijay Chandrasekhar, "Adversarially Learned Anomaly Detection," 2018 IEEE International conference on data mining. pp. 727-736, November 2018.

[7] Alexander Geiger, Alfredo Cuesta-Infante, Kalyan Veeramachaneni, "Adversarially learned anomaly detection for time series data," The International Conference on Learning Representations 2020, September 2019.

[8] Noble, Caleb C., Diane J. Cook, "Graph-based anomaly detection," KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 631-636, August 2003.

[9] Xiuyuan Cheng, Hau-Tieng Wu, "Convergence of Graph Laplacian with kNN Self-tuned Kernels," Information and Inference: A Journal of the IMA, vol. 11, no. 3, pp. 889-957, September 2022.

[10] Or Streicher, Guy Gilboa, "Graph Laplacian for Semi-Supervised Learning," International Conference on Scale Space and Variational Methods in Computer Vision. Cham: Springer International Publishing, pp. 250-262, January 2023.

[11] Jingzhi Tu, Gang Mei, Francesco Piccialli, "An improved Nyström spectral graph clustering using k-core decomposition as a sampling strategy for large networks," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 6, pp. 3673-3684, June 2022.

[12] Harry Sevi, Matthieu Jonckheere, Argyris Kalogeratos, "Generalized Spectral Clustering for Directed and Undirected Graphs," arXiv preprint arXiv:2203.03221, March 2022.

[13] Slawomir Wierzchoń , Mieczyslaw Kłopotek, "Modern Algorithms of Cluster Analysis," Studies in Big Data, vol. 34, January 2018.

[14] Jinsung Yoon, Kihyuk Sohn, Chun-Liang Li, Sercan Ö. Arik, Chen-Yu Lee, Tomas Pfister, "Self-supervise, Refine, Repeat: Improving Unsupervised Anomaly Detection," arXiv e-prints, arXiv: 2106.06115, June 2021.

[15] David Yarowksy, "Unsupervised word sense disambiguation rivaling supervised methods," ACL '95: Proceedings of the 33rd annual meeting on Association for Computational Linguistics, pp. 189-196, June 1995.

[16] Massih-Reza Amini, Vasilii Feofanov, Loic Pauletto, Emilie Devijver, Yury Maximov, "Self-Training: A Survey," arXiv preprint arXiv:2202.12040, September 2023.

[17] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, "Asymmetric tri-training for unsupervised domain adaptation," Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 2988–2997, August 2017.

[18] Yang Zou, Zhiding Yu, B.V.K. Vijaya Kumar, Jinsong Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," Proceedings of the European conference on computer vision, pp. 289-305, September 2018.

[19] Yang Zou, Zhiding Yu, Xiaofeng Liu, B.V.K. Vijaya Kumar, Jinsong Wang, "Confidence regularized self-training," Proceedings of the IEEE/CVF international conference on computer vision, pp. 5982-5991, October 2019.

[20] Yaoxing Li, Liang Bai, "Label Propagation Based on Bipartite Graph," Neural Processing Letters, vol. 55, no. 6, pp. 7743–7760, December 2023.

[21] Ulrike von Luxburg, "A Tutorial on Spectral Clustering," Statistics and computing, vol. 17, pp. 395-416, August 2007.

[22] Claudio Altafini, "Consensus Problems on Networks with Antagonistic Interactions," IEEE transactions on automatic control, vol. 58, no. 4, pp. 935-946, October 2012.

[23] Jerome Kunegis, Stephan Schmidt, Andreas Lommatzsch, Jurgen Lerner, Ernesto W. De Luca, Sahin Albayrak, "Spectral Analysis of Signed Graphs for Clustering, Prediction and Visualization", Proceedings of the 2010 SIAM international conference on data mining, Society for Industrial and Applied Mathematics, pp. 559-570, April 2010.

[24] Andrew Ng, Michael Jordan, Yair Weiss, "On Spectral Clustering: Analysis and an algorithm," Advances in neural information processing systems, vol. 14, December 2001.

[25] "1.14.2. Label Propagation," scikit-learn 1.3.2 user guide.

[26] "1.14.1. Self Training," scikit-learn 1.3.2 user guide.

[27] Mohammad Reza Abbasifard, Bijan Ghahremani, Hassan Nader, "A Survey on Nearest Neighbor Search Methods," International Journal of Computer Applications, vol. 95, no. 25, June 2014.

[28] Martin Dietrich Buhmann, "Radial basis functions," Acta numerica, pp. 1-38, January 2000

[29] Shivam Bansal, "Machine Predictive Maintenance Classification," Kaggle, November 2021.

[30] Fedesoriano, "Heart Failure Prediction Dataset," Kaggle, September 2021.

[31] "signed-graph-laplacian," hcpark1993, GitHub, December 2023.

[32] Tahseen Khan, Wenhong Tian, Guangyao Zhou, Shashikant Ilager, Mingming Gong, Rajkumar Buyya, "Machine learning (ML)-centric resource management in cloud computing: A review and future directions," Journal of Network and Computer Applications, vol. 204, August 2022.