Transfer Learning on Phasor Measurement Data from a Power System to Detect Events in Another System

Ameen Abdel Hai [1], Taif Mohamed [2], Martin Pavlovski[1], Mladen Kezunovic [2], Zoran Obradovic [1]
[1]Computer and Information Sciences, Temple University, Philadelphia, PA, 19122, USA
{aabdelhai, martin.pavlovski, zoran.obradovic}@temple.edu
[2]Electrical & Computer Engineering, Texas A&M University, College Station, TX, 77843-3128, USA
{taif205, m-kezunovic}@tamu.edu

***Abstract*—The methods for power system event detection using field-recorded data from Phasor Measurement Units (PMUs) often require many labeled events, which can be costly or infeasible to obtain. We show that events in one power system can be accurately detected by reusing a small number of carefully selected labeled PMU data from another without the need for additional labeling. Our transfer learning-based approach outperforms alternative state-of-the-art conventional machine learning (ML) methods on a large PMU historical dataset. We demonstrate this approach with a use case of detecting events from historical PMU data recorded in the Eastern Interconnection in the USA by using similar labeled PMU data from the Western Interconnection. This technique may be propagated to other situations where some of the events' data from one power system may be applied to enhance learning in another.**

***Keywords* – Big data, Power system disturbance, Event detection, Machine learning, Phasor Measurement Unit, Transfer learning.**

I. INTRODUCTION

The reliability of electric power systems may be severely impacted by many events of different types, caused by a variety of factors that occur irregularly over time. Some events such as transmission line faults due to severe weather, vegetation impact, etc. may be local, and some such as fundamental frequency events are system-wide. The volume of data collected by numerous PMUs in a given utility or interconnection system may reach hundreds of terabytes over a single year since data are reported at the rate of 30 to 60 frames per second [1]. The manual means for analysis of such elaborated historical recordings are impractical and efficient automated analysis is needed but finding a solution may be challenging [2].

With the increase in the availability of PMU measurements in electric utilities, there is a potential for the development of new machine learning (ML) applications that could significantly increase the importance of storing and managing the PMU data and provide new predictive decision-making tools [3]. Event detection is one of the most beneficial applications [4] focusing on the identification of instances in PMU measurements that have a significant deviation from the normal operating conditions of the system.

Various event detection approaches based on PMU measurements have been investigated. The event detection method based on the Principal Component Analysis was introduced [5] and was applied for the analysis of cascading events [6]. ML methods such as K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Decision Tree (DT) were also applied [7]. Convolutional Neural Network was used to classify events based on wavelet analysis [8]. Other feature engineering methods include: Detrended Fluctuation Analysis [9] and Signal Energy Transform [10]. Applicability of transfer learning (TL) to transient stability problem was investigated in [11] and applied to the analysis of oscillation events in transmission system [12], and fault detection in distribution systems [13]. Supervised transfer learning was proposed for event type differentiation and was applied on synthetic PMU data [14]. However, accurate event labeling on real-world, field-recorded PMU data remains a challenge.

In our study, we mitigate the essential need for extensive event labeling by utilizing a TL approach that only requires a small number of well-labeled events from one power system to detect events in another without any additional labeling efforts. We utilize a TL method combined with a semi-supervised detector that leverages related labeled instances from a source dataset to the target domain. Selected relates instances aid semi-supervised detectors detect events since it selects instances from the source that are similar to the target domain. Our contribution is in the enhancement of the TL method using a non-redundant approach that does not select duplicates/similar instances in order to improve computation efficiency. We improve the semi-supervised detector by using an alternative similarity measure that is more applicable to the dimensionality of the PMU data. Our approach demonstrates superior performance over various state-of-the-art ML algorithms (unsupervised, semi-supervised, and supervised) when leveraging labeled data from one power system to detect events in another.

II. TRANSFER LEARNING FOR EVENT DETECTION

A. Rationale for Transfer Learning

This material is supported by the Department of Energy under Award DE-OE0000913.

978-1-6654-6283-9/22/$31.00 ©2022 IEEE
DOI 10.1109/ICMLA55696.2022.00244

Most disturbances detected by the PMUs are typically not labeled. Some labels can be created by utility using SCADA event logs, but such event logs may not be reliable. Event detection tasks are often performed using an unsupervised approach since manually labeling data can be time-consuming and costly. However, unsupervised approaches are not aided by labeled data that allow the possibility of correcting errors. On the other hand, supervised classification approaches require enough accurate labels, which can be infeasible to obtain. Thus, unsupervised as well as supervised ML approaches for event detection in power systems have serious limitations when labels are unavailable or imprecise. We hypothesize that it might be feasible to transfer relevant labeled instances from a source power system to address the problem at the target power system by using minimal labeled data instances.

TL objective is to use similar labeled instances from a related source task to facilitate learning in the target domain based on a minimal amount of labeled data. Because semi-supervised detectors assume the availability of a finite number of labeled data instances, TL is frequently used with semi-supervised learning methods. In such instances, TL utilizes labeled data from a similar dataset to train a model for the unlabeled target dataset [15].

TL is frequently used on datasets that defy conventional ML modeling assumptions. When solving event detection, the following assumptions should be considered: (1) the dimensionality of the feature space of the source and target datasets might differ; (2) covariate shift assumption, i.e., marginal distributions of the source and target datasets might be dissimilar; and (3) concept shift assumption, i.e., conditional distributions might differ since the meaning of an identical behavior might differ in the source and target domains. The second and third assumptions challenge the transfer learning task [16].

B. Related Work

In [17], TL was applied to detect events using PMU measurements by transferring relevant labeled data from a power system collected in one year (2016) to detect events from future instances (2017) in the same power system. In [18] TL technique in conjunction with deep learning model was utilized to enhance the detection of events in one power system using a model pre-trained on another. The use case of using PMU recorded data from the Western and Eastern Interconnections (WI and EI) in the US demonstrates that the use of TL enhances the performance by leveraging labeled data from both WI and EI to enhance the detection on WI. This model transfers parameters of the pre-trained model, trained on EI to be used as the initial parameters of the model trained on the data of the WI. As illustrated in Sec. IV, the quality of data of the EI is poor compared to data from the WI, so detecting events from WI based on EI only might be insufficient. There are some limitations of the study reported in [18]: a) its proposed model does not detect events from one power system based on another without utilizing labeled data from both power systems, b) it utilizes a fully supervised learning estimator and considers only line, generator, oscillation events, and normal/healthy signals.

C. New Event Detection Approach

To address the mentioned gaps, our paper extends and enhances studies reported in [17, 18] by exploring the benefits of knowledge transfer between two independent power systems, such as the WI and EI in the USA using a transfer function combined with a semi-supervised detector to identify events based on minimal labeled data of the source task only, and it downgrades to unsupervised mode if no related labeled data instances were available in the source power system.

To address the mentioned issues, we propose the following two methods based on TL techniques: 1) Spatial transfer, sLocITR (spatial localized instance transfer reduced), which leverages labeled data from one power system to detect events in another system. Our approach does not require target labels, since it relies only on related instances from the source power system to detect events from the target power system, while the study reported in [18] requires target labels since it leverages labeled data from both power systems (source and target) to detect events from the target power system. 2) Spatiotemporal transfer stLocITR, based on leveraging labeled data from one power system integrated with a small number of labeled data from another power system to detect future events. Table I summarizes the major differences between the proposed approach and studies reported in [17, 18].

TABLE I
COMPARING THE PROPOSED METHOD TO TRANSFER LEARNING ALTERNATIVES.

Study	Source	Target	Transfers	Detector	Target Labels
[17]	WI_{past}	WI_{future}	Temporally Related data	Semi-supervised	Not Required
[18]	EI & WI	WI	Parameters	Supervised	Required
sLocITR	WI	EI	Spatially Related data	Semi-supervised	Not Required
stLocITR	WI & EI_{past}	EI_{future}	Spatio-Temporally Related data	Semi-supervised	Not Required

III. METHODOLOGY

A. Compression and Unification of Data Dimension

To transfer labeled instances from one power system to another, we project time windows (*TWs)* from the source and target datasets of the two power systems with different numbers of PMUs to latent spaces of unified dimensions while preserving the properties of the original data. This is achieved by an Autoencoder, i.e., an unsupervised Neural Network (NN) for dimensionality reduction. Autoencoders utilize multiple neural computing layers to learn non-linear transformations of data to a latent space [19]. Other dimensionality reduction techniques such as Principal Component Analysis (PCA) were also considered but failed to learn a representation that preserves the properties of the original data since such techniques are limited to linear transformations only [5]. The feature vectors (*TWs*) from both datasets were extended to 200 dimensions by padding with zeros, thus standardizing the number of dimensions in the two

datasets. In the use case with the data from WI and EI, two fully connected layers with batch normalization were used to learn how to unify the 35-dimensional feature vectors from both WI and EI datasets. To enhance the performance of the ML models, the unified data were scaled to a standard range using Standard Scaler [17], defined as $z = \frac{x-\mu}{\sigma}$.

B. Comparative Analysis

A comparative study was conducted to evaluate the applicability and effectiveness of the TL method when leveraging labeled data from one power system to detect events from another. A multitude of ML algorithms of various types (unsupervised, supervised, and semi-supervised learning) were used as baselines.

Unsupervised learning-based baseline methods find hidden patterns in data without using any labeled data instances, which is the only option when labels are not available. On the contrary, learning from normal data alone is based on the premise that the occurrence of events is uncommon and infrequent, which field-recorded PMU data contradicts. The performance of the algorithms is degraded by the scarcity of labeled data instances [20]. The isolation nearest neighbor ensembles (iNNE) method [21], and the k-nearest neighbor outlier (kNNO) detection method [4] are used in our study for comparison.

We also considered *supervised learning* baselines that learn from labeled cases. Moreover, supervised learning methods assume that the marginal distributions of the WI source training data and the EI target test data are similar (no covariate shift assumption) which PMU data of both power systems contradict [17]. Hence, these approaches might be infeasible to train on one power system and detect events from another, due to the scarcity of labels and distributional difference. We utilized some of the most common and state-of-the-art traditional supervised learning methods and compared them with alternative learning types, including Logistic Regression (LR), Support Vector Machine (SVM), K-Nearest-Neighbor (KNN), and Multilayer Perceptron (MLP), available through the Scikit-Learn library [22].

Semi-supervised learning baseline methods are often used when larger amounts of labeled data are difficult or impractical to obtain, which is applicable for PMU data. The aim is to learn a classifier from integrated labeled and unlabeled instances to outperform unsupervised and supervised learning algorithms when training with an insufficient amount of labeled data [23]. For comparison, two semi-supervised learning detectors were employed to classify *TWs* as events or normal operations. One detector is the semi-supervised k-nearest neighbor anomaly (SSKNNO) detection algorithm [20]. This method downgrades to unsupervised mode if no related labeled data instances are found in the labeled source data. This is determined by a specified threshold. Since the algorithm utilizes a distance metric, the number of labeled *TWs* does not influence the detection tasks when well-labeled *TWs* from the same signal pattern are available in the source labeled data. This algorithm uses the Euclidean distance to compute the similarity between *TWs*. This similarity measure is not applicable when the sizes (dimensions) of *TWs* are large, hence, we modified and enhanced the algorithm by replacing the Euclidean distance with the Manhattan distance, since Manhattan distance achieved better performance. Another semi-supervised baseline method is the semi-supervised detection of outliers (SSDO), which is derived from k-means clustering [24].

The proposed TL approach consists of two steps. 1) A relevant subset of labeled *TWs* is selected from WI (source dataset D_s) to transfer to the target dataset D_t which contains unlabeled *TWs* from EI. The algorithm transfers a certain *TW* from the WI to EI if its local data distribution is similar in EI and there is a lack of labeled cases in that region of EI feature space. The transfer function employs unsupervised techniques since labels for *TWs* in D_t are unavailable and the labeled *TWs* in D_s do not affect the transfer decision. 2) A semi-supervised learning algorithm computes an anomaly score based on a nearest-neighbor technique that considers the related *TWs* that were transferred from WI and the unlabeled target *TWs* from EI. This algorithm takes as an input a partly labeled dataset containing the transferred labeled *TWs* merged with the unlabeled target *TWs* [17].

LocIT transfers a *TW* from D_s to D_t if the distributions of two subsets are sufficiently similar. This is measured by comparing the first and second order statistics using a subset of the nearest neighbor in D_s and D_t and is computed as

$$d_1(N1, N2) = \left\| \frac{1}{k} \left(\sum_{x_i \in N1} x_i - \sum_{x_j \in N2} x_j \right) \right\|_2 . \quad (1)$$

Equation 1 defines the location distance which is the l2-norm of the difference of the centroids (i.e., arithmetic mean) between two neighborhoods N_1 and N_2. High values of d_1 imply less similarity and reduce the chance of a transfer. The distance between the covariance of two neighborhoods (correlation distance) is computed as:

$$d_2(N1, N2) = \frac{\|C_{N1} - C_{N2}\|_F}{\|C_{N1}\|_F} \quad (2)$$

where $\|\cdot\|_F$ is the Frobenius norm (the Euclidean norm of a matrix), and C is the covariance matrix. High values of d_2 imply that the localized distributions of the D_s and D_t differ. The TL algorithm transfers a *TW* from D_s if the distance to its nearest neighbor in D_s and D_t is similar to the average distance between any two neighboring instances in D_t.

Transfer Function. SVM classifier is trained by combining the values of d_1 and d_2. The classifier predicts if a source *TW* fits in the target by considering at the correlation and location distance between the neighborhood sets of the *TW*. The size of the neighborhood is specified to control the strictness of the instance transfer. A positive training example is created for each *TW* in D_t by identifying its nearest neighbor in D_t and computing d_1 and d_2 on D_t. The negative training examples are created by computing for each *TW* in D_t a feature vector consisting of the distances between the neighborhood sets of target *TW* and its farthest neighbor. Finally, each *TW* from D_s is predicted using a trained SVM classifier on D_t using the negative and positive training examples [17].

The utilized transfer function selects and transfers all related data instances, including duplicates and instances that are very similar in distance. Due to the distance-based nature of the semi-supervised detector, redundant instances do not contribute to the classification task, but increase the complexity of the algorithm and reduce efficiency. Thus, we modified and enhanced the TL method to exclude similar instances/duplicates. The Euclidean distance was utilized to compute the mean μ of the transferred *TWs,* and iteratively, compute the Euclidean distance of two independent *TWs*. Then, we excluded all *TWs* for which their $distance < \mu$, indicating a certain instance does not contribute to the classification task of the TL method, since another *TW* of the same quality (pattern) was found.

IV. Data Processing

PMU Data. We utilize historical field measurements collected over two years, 2016-2017 from 38 PMUs placed in the WI, and from 178 PMUs placed in the EI in the U.S. electric power system. The measurements from EI are collected at 30 frames per second (fps), while measurements from WI are collected at 30 fps or 60 fps. Locations of PMUs and the system topology are not provided to us. Some outliers, data duplicates and missing data are observed in both datasets but do not affect our method significantly [25]. Non-uniform number of PMUs and data quality issues make this event detection task complex. WI dataset contained higher quality measurements than the EI dataset, since EI contains missing data ranging from ~1% to ~70%, whereas missing data of WI ranges from ~1% to ~30%. Thus, we utilize labeled data from WI to detect events from EI, without using any labeled data from EI.

Event Log. Both WI and EI datasets contain phasor measurements associated with line outages, transformer outages, and fundamental frequency deviations that are labeled in the event log. Visual inspection of these events revealed that some events evolve from one type to another, hence, they were considered "complex" events. Complex events include events labeled generator, capacitor, bus, and oscillation. The provided event log most likely was obtained from the SCADA data, and therefore it contains temporally imprecise event labels (start time with a precision of 1-minute). In addition, due to the sparsity of PMU locations in the network, log events did not necessarily occur in the vicinity of the PMUs used in this study. To improve the temporal precision of the log events, visual inspection was performed by the domain expert on our team. Then, we used a more precise start time of the events confirmed through visual inspection. The study reported in [17] experimented various dimensions of *TWs;* 2-second *TWs* resulted in performant classification results; hence, the dimension of 2-seconds was used. Table II presents the number of labels used for each proposed method.

Feature Extraction. For each *TW* and the selected *PMU* device, we calculated the *Rectangle Area (RA)* features using the frequency and positive-sequence voltage magnitude as:

$$RA_{PMU,TW} = (f_{max} - f_{min}) * (V_{max} - V_{min}) \quad (3)$$

where f_{max} and f_{min} are the maximum and minimum frequency values, and V_{max} and V_{min} are the maximum and minimum positive sequence voltage magnitudes, respectively [17]. Datasets from WI comprise of feature vectors of 38 RA values, where each RA value corresponds to a certain PMU in a certain *TW*. Similarly, feature vectors from EI comprise the feature vectors of 178 RA values.

TABLE II
NUMBER OF LABELS PER CATEGORY FROM BOTH WI AND EI DATASETS.

Method	# Event Labels from WI	# Normal Labels from WI	# Event Labels from EI	# Normal Labels from EI
sLocITR	1038	1846	0	0
stLocITR	1038	1846	849	762

V. Experimental Setup

We propose two TL methods based on different splits of the source and target datasets. 1) *Spatial transfer*, sLocITR, where labeled *TWs* were selected from D_s which consisted of *TWs* from WI and were used to detect events in D_t, which consisted of unlabeled *TWs* from EI. In this experiment, D_s contained the entire data of the WI, while D_t contained the entire data of the EI. 2) *Spatiotemporal,* stLocITR, where $D_s = WI \cup EI_{2016}$; $D_t = EI_{2017}$; where *WI* denotes the entire *TWs* of the WI, EI_{2016} denotes the *TWs* of the EI collected from 2016, used to detect events in EI_{2017} which denotes the *TWs* of the EI from 2017.

We answer the following empirical questions: 1) How does the proposed TL method perform compared to alternative baselines? 2) How does the number of labeled source data selected from D_s affect the classification accuracy for events in the target domain D_t? The results validate our hypothesis and illustrate the benefits of employing TL techniques in conjunction with a semi-supervised detector to leverage knowledge and detect events based on minimal labeled data. To address question 2, we selected the top p related instances excluding redundant/similar instances to experiment how the proportion of labeled data affects the performance; where $p \in \{20, 51, 103, 259, 415, 570, 726\}$ corresponding to 1% to 25% of labeled source data instances.

The performance of the TL algorithm was evaluated by comparing it to common conventional ML algorithms of varying learning types described in Sec. III (i.e., unsupervised, supervised, and semi-supervised). The following metrics were used to evaluate the algorithms: The area under the receiver operating characteristic (AUROC), Precision, Recall, and F-1 score [26].

VI. Results and Discussion

A. WI and EI Distribution Comparison

To validate the applicability of the TL on PMU data, we utilized Kolmogorov-Smirnov (KS) metric to test if the cumulative distribution functions of the source WI and target EI datasets are similar. KS metric was applied to compare two independent samples on the source and target system, where the source is represented as a 1-dimensional array that contains features from the WI and the target is a 1-dimensional array that contains features from the EI. We obtained p-values by iteratively computing similarities between two independent samples. The maximum p-value was $2.7e^{-13}$, thus, since the obtained p-value is very small, we can safely reject the null hypothesis, implying distributions of WI and EI are different.

TABLE III
COMPARATIVE ANALYSIS OF THE UTILIZED TRANSFER LEARNING METHODS TO VARIOUS BASELINES USING THE SELECTED LABELED *TWS* FROM D_S

Method	Learning Type	Model	AUC	Precision	Recall	F1
Spatio-temporal	Transfer Learning	stLocITR	**0.90**	0.90	0.90	0.90
	Semi-supervised	SSKNNO	0.85	0.86	0.86	0.86
		SSDO	0.84	0.86	0.85	0.85
	Supervised	RF	0.79	0.79	0.79	0.79
		KNN	0.79	0.80	0.78	0.79
		MLP	0.74	0.82	0.73	0.77
		SVM	0.72	0.81	0.70	0.75
	Unsupervised	kNNO	0.79	0.80	0.79	0.79
		iNNE	0.74	0.75	0.73	0.74
Spatial	Transfer Learning	sLocITR	**0.87**	0.87	0.87	0.87
	Semi-supervised	SSKNNO	0.84	0.84	0.84	0.84
		SSDO	0.83	0.85	0.84	0.84
	Supervised	RF	0.75	0.77	0.74	0.75
		KNN	0.72	0.75	0.71	0.73
		MLP	0.68	0.77	0.66	0.71
		SVM	0.65	0.77	0.63	0.69
	Unsupervised	kNNO	0.77	0.79	0.76	0.77
		iNNE	0.74	0.76	0.73	0.74

B. *Transfer Learning versus Baseline Event Detectors*

Table III presents and compares the performance of the proposed TL methods stLocITR and sLocITR to alternative baselines of various learning types. Consistent results demonstrate the effectiveness of the proposed methods and show that both methods outperformed fully supervised, semi-supervised, and unsupervised algorithms. The sLocITR method selected and transferred 570 (out of 2,884) related data instances (543 abnormal events + 27 normal) excluding redundant instances from WI to detect events from EI. stLocITR transferred additional 362 (out of 1,611) temporally disjoint related cases from EI, resulting in increased AUROC by 11% when compared to the best performing supervised and unsupervised learning algorithms, and a 5% improvement when compared to the best performing semi-supervised learning algorithms. sLocITR increases the AUROC by 12% when compared with the best supervised learning algorithm, 10% improvement when compared with the best unsupervised learning algorithm, and 3% improvement when compared with the best semi-supervised learning algorithm. Unsupervised learning algorithm, kNNO outperformed supervised variant using the Spatial split, indicating that the source and target label sets differ significantly. In other words, there are many input-output relationships in the target domain that do not have similar counterparts in the source. However, the underlying anomaly patterns remain similar. Unsupervised learning is based on detecting anomaly patterns only from the input signals, whereas supervised algorithms attempt to learn the relationship between the input signals and the output labels, which might be misinforming for some cases due to the distributional difference (label sets) of both interconnections.

Experiments provide evidence that TL-based methods are more accurate than unsupervised, supervised, and semi-supervised alternatives for detecting events from one power system based on labeled data of another.

C. *The Effect of Using Various Quantities of Labeled Data*

Often, obtaining event logs or labeled data for event detection tasks is non-trivial or costly. Thus, we studied the effect of using various amounts of labeled source data to assess what number of labeled data is adequate to detect events from the EI of the U.S.A. based on minimal labeled data from the WI of the U.S.A. (Spatial Split). We selected from D_s 20, 51, 103, 259, 415, 570, and 726 events to detect events from the target data D_t. We repeated the experiments 10 times and reported AUROCs, and their corresponding two-sided confidence intervals calculated at 95% confidence level, presented in the shaded area of Fig. 1. We selected the best methods from various learning types (i.e., fully supervised, semi-supervised, and unsupervised) and compared them with the proposed TL method stLocITR.

Fig. 1 shows that the TL method outperforms supervised learning on a large benchmark since there is a distributional difference between the D_s and D_t. Results show that the TL method outperforms baselines with varying quantities of labeled data incorporated. The straight line of the unsupervised learning algorithm kNNO with no labels incorporated is included for comparison. When sufficient labeled data are incorporated, semi-supervised SSKNNO outperforms unsupervised learning. The increase in labeled source data is not found to increase the performance of the supervised algorithm, since the source and target label sets differ greatly. This study demonstrates that transferring 570 labeled data instances from the WI are sufficient to detect events from the 3,085 instances of the EI PMU data. We randomly select a proportion of labeled data from D_s to train supervised and semi-supervised learning algorithms, whereas the TL algorithm uses the most relevant instances from D_s. When comparing sLocITR with a supervised learning algorithm, Fig. 1 shows that selecting the top relevant instances results in not only better performance, but a more stable model since sLocITR has a significantly lower two-sided confidence interval than RF. Table IV illustrates event types when transferring the top selected 100, 300, and 500 instances.

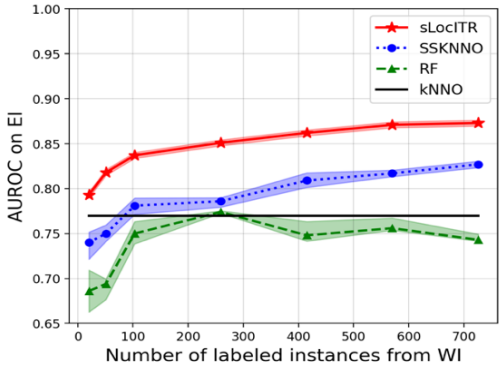

Fig 1. Comparing the performance of the proposed method sLocITR to baselines based on varying number of labeled source data evaluated using AUROC and their corresponding two-sided confidence interval calculated at 95% confidence level.

TABLE IV.
EVENTS TRANSFERRED PER CATEGORY AMONG TOP 100, 300, AND 500.

# Labeled Events	Line	Frequency	Transformer	Complex	Normal
100	68	15	3	6	8
300	165	71	3	45	16
500	269	103	3	103	22

This experiment shows that supervised learning algorithms are infeasible when leveraging knowledge from one interconnection to another due to covariate and concept shift assumptions and when labels are scarce and difficult to obtain.

D. *Misclassified Events*

To further comprehend the errors made by the TL method, a domain expert visually inspected the misclassified *TWs.* The most common occurrence of these *TWs* is the presence of low-frequency *oscillations* that the algorithm was unreliable in detecting as only 0.3% of all events in WI were labeled as oscillations even though these events are more common. Low-frequency oscillation events are difficult to capture because their impact is most obvious after performing modal analysis.

VII. CONCLUSION

This study shows that the TL method yields a substantial increase in AUROC compared with other state-of-the-art ML algorithms (fully supervised, semi-supervised, and unsupervised). Experiments show that this method is more feasible than alternative baselines when conventional ML modeling assumptions are violated and outperforms the baselines when reusing labeled data instances from one power system to detect events from another. Furthermore, this method can detect events based on a small amount of transferred relevant labeled data from another power system.

DISCLAIMER

REFERENCES

[1] A. G. Phadke and T. Bi, "Phasor measurement units, WAMS, and their applications in protection and control of power systems," J. Modern Power Systems and Clean Energy, vol. 6, no. 4, pp. 619–629, Jul. 2018.

[2] NASPI Data Network Management Task Team, "NASPI 2020 Survey of Industry Best Practices for Archiving Synchronized Measurements," NASPI Technical Report, Nov. 2020, NASPI-2020-TR-024.

[3] North American SynchroPhasor Initiative, "Data Mining Techniques and Tools for Synchrophasor Data," NASPI-2018-TT-007, January 2019.

[4] S. Ramaswamy, R. Rastogi, K. Shim, "Efficient Algorithms for Mining Outliers from Large Data Sets," ACM SIGMOD, 2000.

[5] L. Xie, Y. Chen, P. R. Kumar, "Dimensionality Reduction of Synchrophasor Data for Early Event Detection: Linearized Analysis," IEEE Trans. Power Systems, vol. 29, no. 6, Nov. 2014, pp. 2784-2794.

[6] M. Rafferty, X. Liu, D. M. Laverty, S. McLoone, "Real-Time Multiple Event Detection and Classification Using Moving Window PCA," IEEE Trans. Smart Grid, vol. 7, no. 5, pp. 2537-2548, Sep 2016.

[7] S. Brahma, R. Kavasseri, H. Cao, N. R. Chaudhuri, T. Alexopoulos, Y. Cui, "Real-Time Identification of Dynamic Events in Power Systems Using PMU Data, and Potential Applications-Models, Promises, and Challenges," IEEE Trans. Power Delivery, vol.32, no. 1, Feb. 2017.

[8] S. Wang, P. Dehghanian, L. Li, "Power Grid Online Surveillance through PMU-Embedded Convolutional Neural Networks," IEEE Trans. Industrial Applications, vol. 56, no. 2, Mar.-Apr. 2020, pp. 1146-1155.

[9] M. Khan, P. M. Ashton, M. Li, G. A. Taylor, I. Pisica, J. Liu, "Parallel Detrended Fluctuation Analysis for Fast Event Detection on Massive PMU Data," IEEE Trans. Smart Grid, vol. 6, no. 1, Jan 2015.

[10] R. Yadav, A. K. Pradhan, I. Kamwa, "Real-Time Multiple Event Detection and Classification in Power System using Signal Energy Transformations," IEEE Trans. Industrial Informatics, vol. 15, no. 3, Mar. 2019, pp. 1521-1531.

[11] S. Jafarzadeh; N. Moarref; Y. Yaslan; V. M. Istemihan Genc, "A CNN-Based Post-Contingency Transient Stability Prediction Using Transfer Learning," 11th Int'l Conf. Electrical and Electronics Engineering (ELECO), Bursa, Turkey, Nov. 2019.

[12] Zakaria El Mrabet; Daisy Flora Selvaraj; Prakash Ranganathan, "Adaptive Hoeffding Tree with Transfer Learning for Streaming Synchrophasor Data Sets," 2019 IEEE Int'l Conf. Big Data, Los Angeles, CA, USA, Dec. 2019.

[13] Y. Zhang, X. Wang, Y. Luo, Y. Xu, J. He, G. Wu, "A CNN Based Transfer Learning Method for High Impedance Fault Detection," 2020 IEEE, PESGM, Montreal, QC, Canada, Aug. 2020.

[14] H. Li, Z. Ma, Y. Weng and E. Farantatos, "Transfer Learning for Event-Type Differentiation on Power Systems," IEEE Int'l Conf. SGSMA, Split, Croatia, May 2022.

[15] J. Van Haaren, A. Kolobov, J. Davis, "TODTLER: Two-order-deep transfer learning," 29th AAAI Conf. Artificial Intelligence, 2015.

[16] W.M. Kouw, M. Loog, "An Introduction to Domain Adaptation and Transfer Learning," arXiv:1812.11806, 2018.

[17] A.A. Hai, T. Dokic, M. Pavlovski, T. Mohamed, D. Saranovic, M. Alqudah, M. Kezunovic, Z. Obradovic, "Transfer Learning for Event Detection from PMU Measurements with Scarce Labels," in IEEE Access, Vol. 9, Sept. 10, 2021, pp. 127420-127432.

[18] J. Shi, K. Yamashita, N. Yu, "Power System Event Identification with Transfer Learning Using Large-scale Real-world Synchrophasor Data in the United States," IEEE ISGT, 2022.

[19] W. Wang, et. al, "Generalized Autoencoder: A Neural Network Framework for Dimensionality Reduction," IEEE Conf. Computer Vision and Pattern Recognition, Workshops, 2014, pp. 490-497.

[20] V. Vercruyssen, W. Meert, J. Davis, "Transfer Learning for Anomaly Detection through Localized and Unsupervised Instance Selection," AAAI Conf. Artificial Intelligence, 2020.

[21] TR. Bandaragoda, KM. Ting, D. Albrecht, FT. Liu, Y. Zhu, JR. Wells, "Isolation-based Anomaly Detection using Nearest-Neighbor Ensembles," Computational Intelligence. 2018; 34: 968– 998.

[22] Scikit-learn – Machine Learning in Python, [Online] Available: https://scikit-learn.org/stable/.

[23] X. Zhu, A. Goldberg, "Synthesis Lectures on Artificial Intelligence and Machine Learning," 2009, vol. 3, no. 1, pp. 1-130.

[24] V. Vercruyssen, W. Meert, G. Verbruggen, K. Maes, R. B.umer, J. Davis, "Semi-Supervised Anomaly Detection with an Application to Water Analytics," IEEE Int'l Conf. Data Mining, Singapore, 2018.

[25] Cheng, Z., Hu, Y., Obradovic, Z., Kezunovic, M. "Using Synchrophasor Status Word as Data Quality Indicator: What to Expect in the Field?," IEEE Int'l Conf. SGSMA, Split, Croatia, May 2022.

[26] D.M.W. Powers, "Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation," J. Machine Learning Technologies, vol. 2, 2008.