Active Learning for Accurate Analysis of Streaming Partial Discharge Data

Hai-Long Nguyen[1], João Bártolo Gomes[1], Min Wu[1], Hong Cao[2], Jianneng Cao[1], Shonali Krishnaswamy[1]
[1] Institute for Infocomm Research, A*STAR, 1 Fusionopolis Way #21-01 Connexis, Singapore 138632
[2] McLaren Applied Technolgies, APAC, 7 Temasek Boulevard, Suntec Tower One, Singapore 038987
{nguyenhl,bartologjp,wumin,caojn,spkrishna}@i2r.a-star.edu.sg[1], hong.cao@mclaren.com[2]

***Abstract*—Partial discharge (PD) is a phenomenon of electric discharge typically caused by the damaged or aged insulation of high voltage equipment in power grids, such as transformers, switch gears, and cable terminals. In the context of Prognostic and Health Management (PHM), detection and monitoring of PD are important to ensure the reliability of electrical assets and to avoid catastrophic failures. Machine learning techniques have been successfully applied to discover features and patterns that correspond to different types of partial discharges [9], [11]. Recently, PD monitoring systems have being deployed for assessing the health condition of these equipments continuously so that the maintenance would require less human effort and fewer maintenance interruptions to the operation. However, such systems require labeled data to build data models for PD detection and classification. Labeled data is expensive to obtain since it requires domain expert's manual inputs. Minimizing the labeling cost is thus an important issue to solve.To the best of our knowledge, this issue has not been properly addressed in this domain. This paper proposes an active learning (AL) approach for accurate analysis of streaming PD data that aims to train an accurate PD classification model with minimum cost through selecting the most informative instances for the human experts to label. Experimental results show that our method is able to achieve the high classification accuracy of 86.9% with only a small labeling budget of 1%.**

I. Introduction

Partial discharge is a localised electrical discharge, due to the inability of the insulation to withstand the local electrical stress. In the context of Prognostic and Health Management (PHM), PD detection is a key early indicator for electrical failures of electrical assets in power grids. It is important to detect PD in the early stage of insulation damage so that severe power failure or electrical outage can be avoided. Therefore, PD detection has attracted a lot of research attention recently [14], as it can support the reliable performance of electrical assets through condition based maintenance.

Nowadays, advances in information technologies have made possible for the power industry to remotely assess and monitor the condition of the power grid. However, this involves long-term and continuous recording of high-rate data, which easily produces a huge amount of streaming sensory data. Machine learning and data mining techniques have been successfully applied to extract useful features to learn models from such data for PD analysis. However in this process, we need to know the true labels of training PD data for constructing an accurate classifiction PD model. Noticeably, the labeled PD data, which is usually provided by experts in power industry domain, is expensive to acquire due to the human labor cost as well as the required laboratory test.

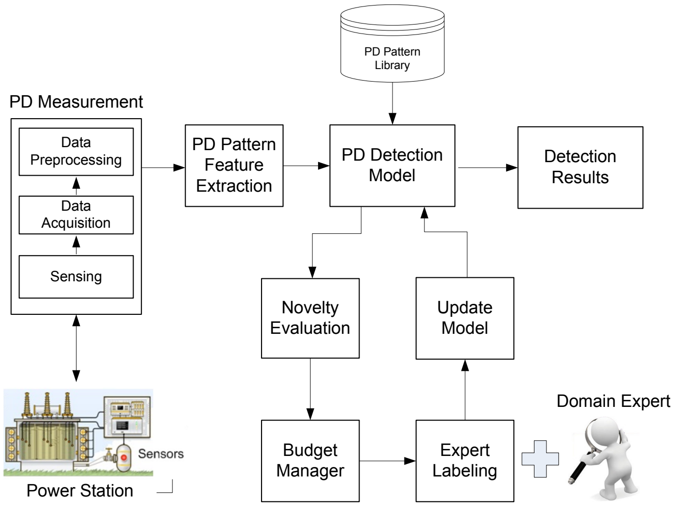

Fig. 1: Active Learning for PD Detection.

To overcome the above-mentioned challenge, we propose to use an active learning approach [18], [19], [4], which is able to train an accurate prediction model with minimum cost of labeling data. The principle is to iteratively query the labels of just the most informative data instance about the decision boundary, and thereby to attain an accurate classifier at significantly lower cost than regular supervised learning. To the best of our knowledge this is the first work proposed to use active learning for PD monitoring.

II. Methodology

Our active learning system architecture for on-line PD detection is shown in Figure 1. It consists of two main components, a PD measurement module, where sensors are deployed and PD signals are collected/pre-processed, and a PD detection module, where advanced machine learning techniques are used to learn a model that is able to differentiate PD signals from noise, and even different PD types (e.g., internal, surface, corona PD). The main contribution of the approach is the active learning part with ability of evaluating the novelty of PD signals. Only novel signals are then given to an expert for labeling and are subsequently used to update the model. It is worth noting that not all data is given to the expert.

978-1-4799-1894-215$31.00 ©2015 IEEE

The budget management will set the constraints and only the most informative data instances that is within the budget is finally given to the expert as illustrated in Figure 1.

A. Feature Representation for PD Signals

For the sensory measurements from PD detectors, we generally represent them in two different patterns, namely time-resolved patterns and phase-resolved patterns as shown in Figure 2. Figure 2(A) shows a time-resolved pattern, i.e., a $q-t$ waveform, where q is the amplitude (i.e., the apparent charge or discharge voltage) and t shows the time information. Figure 2(B) demonstrates a phase-resolved pattern (PR pattern), i.e., a tuple (ϕ, q, n) pattern, where ϕ is the phase angle for the PD pulse, q also refers to the apparent charge or discharge voltage and n is the number of pulses.

In Figure 2(A), the waveform shows a single PD pulse, which appears at the time point $t = 2\mu s$, and the whole signal lasts $10\mu s$ with 1000 data points (i.e., the sampling rate is 100 M/s). We can simply digitize this pulse by a 1000 dimension vector consisting of 1000 time-series data points. A longer waveform, which may involve multiple PD pulse, thus requires a higher dimensional vector for its representation. In addition, we can also extract features for pulse shape (e.g., the pulse height, the pulse rise and decay time, etc.) from the time domain [13] and some other properties for pulses from the frequency domain by signal processing methods.

In Figure 2(B), a point (red circle) is a pulse which can be represented as a time signal in Figure 2(A) and thus Figure 2(B) shows a collection of pulses. We can obtain a distribution of maximum amplitude against the phase angle ($q - \phi$ distribution). Statistical moments (e.g., mean, standard deviation, skewness and kurtosis) for $q - \phi$ distribution can be derived from both positive and negative half cycles [5], [3] (positive half cycle means that the voltage is positive in this half cycle). Some additional features (e.g., discharge asymmetry) can be collected to evaluate the differences between the distributions in both positive and negative cycles. In this section, we will introduce the features extracted from both waveforms and phase-resolved patterns.

A common representation for phase-resolved pattern is based on the phase-window method in [7], [10], [6], [12]. The phase-window method divides the power cycle with 360° into several small phase windows and then generates some features for each phase window. For example, we have 360 windows if each phase window has a size of 1°. We can then extract some features in each phase window, e.g., the number of pulses, maximum amplitude and average amplitude. The statistical moments of these features over all the phase windows can be extracted to further represent the given set of pulses [17].

Assume that we have N phase windows and x_i is a specific feature value for the i^{th} phase window. The mean of x_i over these N phase windows would be $\mu = \sum_{i=1}^{N} x_i/N$. If we consider that phase windows may have different importance (let $p(x_i)$ be the importance of the i^{th} phase window), the weighted mean is thus computed in Equation (1). For simplicity, the other statistics, e.g., variance (σ^2), skewness (S_k) and kurtosis (K_u), are all defined without considering the importance for phase windows [5], [3].

$$\text{Weighted Mean } (w\mu): \qquad w\mu = \frac{\sum_{i=1}^{N} x_i p(x_i)}{\sum_{i=1}^{N} p(x_i)} \tag{1}$$

$$\text{Variance } (\sigma^2): \qquad \sigma^2 = \frac{\sum_{i=1}^{N} (x_i - \mu)^2}{N} \tag{2}$$

$$\text{Skewness } (\gamma): \qquad \gamma = \frac{\sum_{i=1}^{N} (x_i - \mu)^3}{\sigma^3 \times N} \tag{3}$$

$$\text{Kurtosis } (\kappa): \qquad \kappa = \frac{\sum_{i=1}^{N} (x_i - \mu)^4}{\sigma^4 \times N} - 3 \tag{4}$$

In the above definitions, skewness and kurtosis are calculated with respect to a reference normal distribution. Skewness is a measure of asymmetry or degree of tilt of the data with respect to normal distribution, which has a skewness of zero. Negative values for the skewness indicate data that are skewed left (the left tail is longer than the right tail) and positive values indicate data that are skewed right. Kurtosis is an indicator of sharpness of distribution. If a distribution has the same sharpness as normal distribution, the kurtosis is zero. Negative values for kurtosis indicate a distribution flatter than normal distribution, while positive values indicate a sharper distribution.

PD pulses generally occur in both positive and negative halves of voltage cycle. Some features, including discharge asymmetry ($Asym$) in Equation (5) and cross-correlation factor (F) in Equation (6), can be extracted to tell the differences between two halves. In Equation (5), N^+ is the number of phase windows in positive voltage cycle. $Q^+ = \sum_{i=1}^{N^+} x_i^+$ and x_i^+ refers to a specific feature (e.g., average pulse charge, maximum pulse charge or the number of pulses) in i^{th} positive phase window. N^- and Q^- are similarly defined. In Equation (6), both positive and negative halves of voltage cycle have N phase windows. x_i^+ and x_i^- are the same specific feature, but refer to its values in positive and negative power cycles, respectively.

$$Asym = \frac{Q^-/N^-}{Q^+/N^+} \tag{5}$$

$$F = \frac{\sum_{i=1}^{N} x_i^+ x_i^- - \frac{1}{N} \sum_{i=1}^{N} x_i^+ \sum_{i=1}^{N} x_i^-}{\sqrt{\left[\sum_{i=1}^{N} (x_i^+)^2 - \frac{1}{N}(\sum_{i=1}^{N} x_i^+)^2\right]\left[\sum_{i=1}^{N} (x_i^-)^2 - \frac{1}{N}(\sum_{i=1}^{N} x_i^-)^2\right]}} \tag{6}$$

Now, we consider 3 features from the phase resolved pattern as 3 distributions, e.g., the number of pulses, maximum amplitude and average amplitude. Moreover, we have 4 types of statistics, i.e., mean, variance, skewness and kurtosis, which are calculated separately for these 3 distributions. Note that all these features are calculated for positive and negative voltage cycles. Hence, we have 3 × 4 × 2 = 24 features in total.

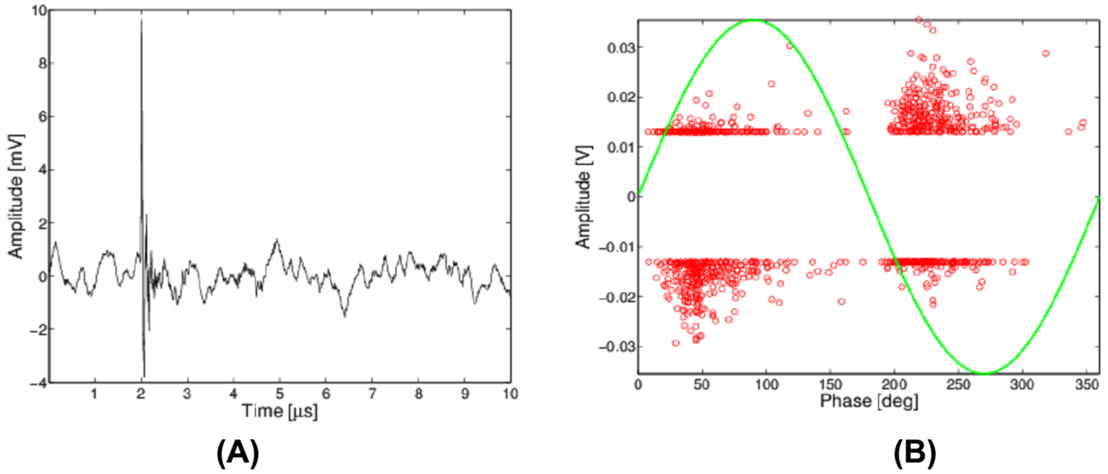

Fig. 2: (A) Time-resolved pattern (waveform) and (B) Phase-resolved pattern.

B. Budget Manager

Since our goal is to maximize PD classification accuracy while keeping the labeling costs fixed within an allocated budget. Here, the budget B is defined as the percentage of instances that can be labeled for a given time period, which represents the time commitment of the expert in relation to the overall set of instances. Once all budget is spent, we can not ask the expert for labels anymore.

C. Novelty Evaluation

Uncertainty sampling is a commonly used active learning strategy [18], [19]. The idea is to select instances, which the current classifier is the least confident about their labels. In the PD monitoring system, this means that instances with certainty below a predefined threshold θ should be selected. A common way to measure uncertainty is to use the posterior probability estimation of the current PD model for the unlabeled instance and thus select the ones that satisfy the following condition ($P_L(y|x_t) < \theta$), where L is the trained classifier, x_t is the unlabeled instance at instant t, and θ is a pre-defined threshold.

In our algorithm, the threshold θ is adaptive; it adjusts itself depending on the incoming data to align with the budget. The threshold θ increases to be able to capture the most uncertain instances when a classifier becomes more certain (stable situations). Moreover, it decreases to query the most uncertain instances first, when a change happens and suddenly a lot of labeling requests appear. Algorithm 1 shows details of our algorithm. Given a small adjusting step s, the threshold θ decreases by a portion of $(1-s)$ when uncertainty of classification appears (lines 4-9). This aims to request more labels of incoming instances for updating the trained classifier L. On the other hand, threshold θ decreases by a portion of $(1-s)$ when the trained model is sure about its classification (lines 11-12).

Algorithm 1 Active Learning(x_t, B, L, s) [19]

Input: x_t - incoming instance, B - labeling budget, L - trained classifier, s - adjusting step
Output: $label \in \{true, false\}$ indicates whether to request the true label y_t

```
1:  Initialize total labeling cost $u$ = 0, labeling threshold $\theta$ = 1.
2:  if $(u/t < B)$ then
3:      $\hat{y_t} = argmax_y P_L(y|x_t)$, where $y \in \{1, \ldots, c\}$ is one of the class labels.
4:      if $P_L(\hat{y_t}|x_t) < \theta$ then
5:          $u = u + 1$ labeling cost increase,
6:          $\theta = \theta(1 - s)$ the threshold decreases,
7:          Request for label of $x_t$
8:          Update classifier $L$ with $x_t$
9:          return true
10:     else
11:         $\theta = \theta(1 + s)$ make the uncertainty region wider.
12:         return false
13:     end if
14: else
15:     return false
16: end if
```

III. Experimental Results

A. Data Collection

The PD data was collected from utility industry. The data are captured into pulses, each of which has the largest reading (trigger) as shown in Figure 2(A). Then, we aggregate 300 data pulses into a data acquisition or an instance alternatively. In power industry domain, experts usually analyze and label a PD instance based on its phase-resolved representation, where the maximum value and phase angle of each PD data pulse are extracted for visualization. Then, we generate features for each data acquisition according to the feature generation

Algorithm 2 Random(x_t, B, L)

```
Input: $x_t$ - incoming instance, $B$ - labeling budget, $L$ - trained classifier
Output: $label \in \{true, false\}$ indicates whether to request the true label $y_t$
generate a uniform random variable $\xi_t \in [0,1]$
if $\xi_t < B$ then
    Request for label of $x_t$
    Update classifier $L$ with $x_t$
    return true
else
    return false
end if
```

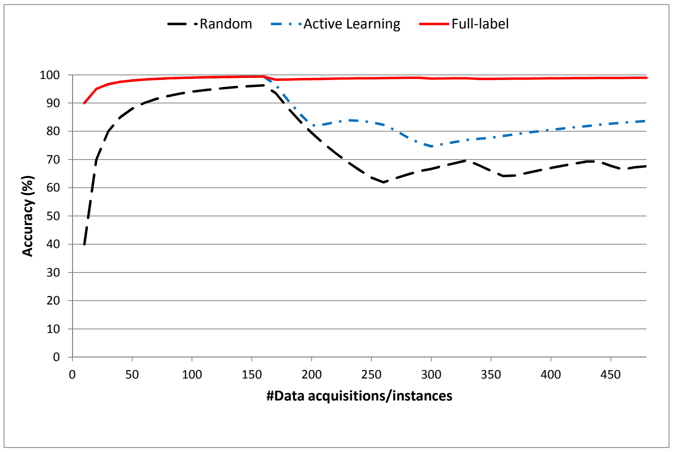

Fig. 3: Accuracy comparison among different methods.

method mentioned Section II-A. We divide the power cycle with 360^o into 360 small phase windows with a size of 1^o. For each phase window, we calculate three features: the number of pulses, maximum amplitude, and average amplitude. Then, we extract statistical moments for each of these features over all the phase windows, including mean, variance, skewness, and kurtosis. Moreover, PD pulses generally occur in both positive and negative halves of voltage cycle. That means we have $3 \times 4 \times 2 = 24$ features for each data instance. In total, we have 476 data instances, including 256 noise instances and 220 PD instances.

B. Performance Comparison

We set our algorithm's parameters as follows: the base leaner is the nearest neighbor classifier (1-NN) [1], adjustment step s is 0.01 and the budget is set to a low $B = 1\%$. To validate our model, we use a prequential evaluation method, where we test an arriving instance first, and if we decide to pay the cost for its label then we use it to update the current model.

We compare our algorithm with a random query selection method and a full-label method. Details of the random method is given in Algorithm 2. It generates a random variable ξ and compares to the budget parameter B. If the random variable ξ is less than B, it asks for label of the upcoming instance. The full-label method asks for label of every upcoming instance. It plays a role as an upper bound of the two algorithms. Figure 3 shows accuracy comparison of these three methods. In average, the random method is the worst one with accuracy of 75.47%. It suffers low accuracy at the beginning and gets betters later when it gets sufficient informative instances for training. However, its accuracy dramatically drops when new PD type appears at data acquisition 160. Since the budget we set is very low 1%, the random algorithm does not have enough good samples to update its model. The full-label method achieves the best accuracy of 98.44% since it asks for label of each sample. However, this approach is not applicable in real-life application since it labels each sample. Our active learning method is superior to the random method and is close to the full-label method. According to Figure 3, the accuracy curve of the active learning method is consistently above the accuracy curve of the random method. On average, the active learning method can attain high accuracy of 86.87%, which is better than the random method by 11.4%. Moreover, it is worthy noting that the active method quickly recovers from changes of PD distributions when a new PD type appears at the instance 160.

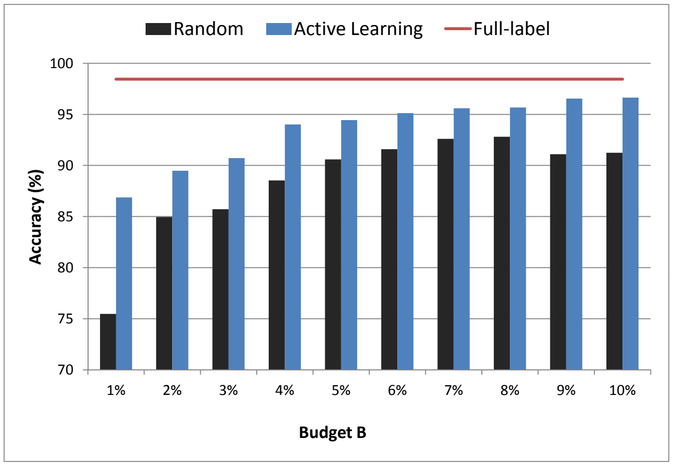

Fig. 4: Sensitivity of the budget.

C. Sensitivity Analysis

In this section, we examine the sensitivity of the two algorithms regarding to the budget parameter B. We keep other parameter unchanged and vary the budget parameter B from 1% to 10% with a step of 1%.

Figure 4 shows the sensitivity of the two algorithms regarding to the budget parameter. We also plot accuracy of the full-label method on top for ease of comparison. We can easily observe that accuracy of the two algorithms increase when the budget increases. However, accuracy of the random method slightly drops at a threshold B of 9% comparing to 8%. A possible explanation is that although the random algorithm requests more labeled instances, some informative instances are not requested due to this random process.

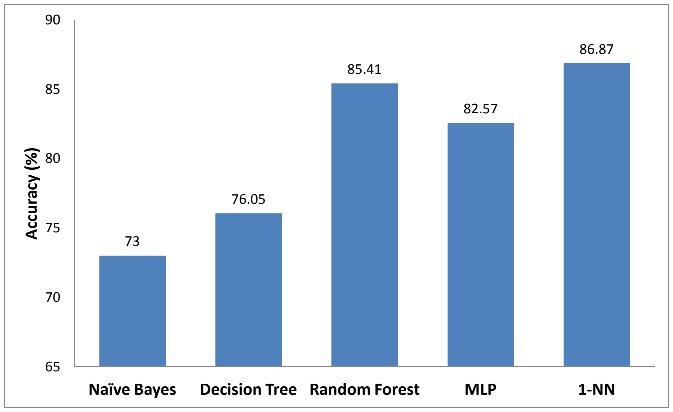

Fig. 5: Accuracy comparison among different classifiers.

The active learning algorithm is consistently better than the random algorithm. The more budget is given, the closer performance of the active learning method is to the full-label's performance. With a budget of 10%, the active learning algorithm can get accuracy of 96.64%, which is only 1.8% lower than the performance of the full-label method.

D. *Comparison among Different Classifiers*

In this section, we compare accuracy among different classifiers. We select five classifiers in our experiment, including Naïve Bayes (NB) [8], decision tree [15], random forest [2], multi-layer perceptron (MLP) [16], and k-NN [1] classifiers.

We observe from Figure 5 that the Naïve Bayes classifier is the worst one since it assumes independent relationship among attributes of PD data. Decision tree classifier is better than the Naïve Bayes classifier; however, its accuracy is still low at 76.05%. Moreover, MLP shows significant improvement comparing to the decision tree classifier by obtaining accuracy of 82.57%. The random forest classifier is the second best classifier with accuracy of 85.41%. The k-NN classifier achieves the best accuracy of 86.87%.

IV. Conclusion

In this paper, we addressed a practical issue of minimizing labeling cost for current PD detection systems. To overcome this challenge, we have proposed an active learning method that selects most informative instances for labeling. Experimental results show that our algorithm is able to achieve good accuracy. With a small budget constraint of 1%, the active learning method helps to achieve accuracy of 86.87%, which is 11.4% higher to the random method's accuracy. Moreover, we also examine our method with different base classifiers, such as Naïve Bayes, decision tree, random forest, MLP, and k-NN. Among these classifiers, k-NN shows its best performance. In future work, we intend to deploy our algorithm in large-scale power grids with real-life applications.

Acknowledgment

This work was supported by Energy Market Authority (EMA) grant: NRF2012EWT-EIRP002-044.

References

[1] D. Aha and D. Kibler. Instance-based learning algorithms. *Machine Learning*, 6:37–66, 1991.

[2] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[3] R. Candela, G. Mirelli, and R. Schifani. Pd recognition by means of statistical and fractal parameters and a neural network. *Dielectrics and Electrical Insulation, IEEE Transactions on*, 7(1):87–94, 2000.

[4] H. Cao, C. Bao, X.-L. Li, and Y.-K. Woon. Class augmented active learning. In *Proceedings of the 2014 SIAM International Conference on Data Mining (SDM)*, pages 1–9. SIAM, 2014.

[5] E. Gulski and A. Krivda. Neural networks as a tool for recognition of partial discharges. *Electrical Insulation, IEEE Transactions on*, 28(6):984–1001, 1993.

[6] L. Hao and P. Lewin. Partial discharge source discrimination using a support vector machine. *Dielectrics and Electrical Insulation, IEEE Transactions on*, 17(1):189–197, 2010.

[7] L. Hao, P. Lewin, and S. Dodd. Comparison of support vector machine based partial discharge identification parameters. In *Electrical Insulation, 2006. Conference Record of the 2006 IEEE International Symposium on*, pages 110–113. IEEE, 2006.

[8] G. H. John and P. Langley. Estimating continuous distributions in bayesian classifiers. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Mateo, 1995. Morgan Kaufmann.

[9] K. Lai, B. Phung, and T. Blackburn. Application of data mining on partial discharge part i: predictive modelling classification. *Dielectrics and Electrical Insulation, IEEE Transactions on*, 17(3):846–854, 2010.

[10] K. Lai, B. Phung, and T. Blackburn. Application of data mining on partial discharge part i: predictive modelling classification. *Dielectrics and Electrical Insulation, IEEE Transactions on*, 17(3):846–854, 2010.

[11] H. Ma, J. C. Chan, T. K. Saha, and C. Ekanayake. Pattern recognition techniques and their applications for automatic classification of artificial partial discharge sources. *Dielectrics and Electrical Insulation, IEEE Transactions on*, 20(2):468–478, 2013.

[12] H. Ma, J. C. Chan, T. K. Saha, and C. Ekanayake. Pattern recognition techniques and their applications for automatic classification of artificial partial discharge sources. *Dielectrics and Electrical Insulation, IEEE Transactions on*, 20(2):468–478, 2013.

[13] A. A. Mazroua, R. Bartnikas, and M. Salama. Neural network system using the multi-layer perceptron technique for the recognition of pd pulse shapes due to cavities and electrical trees. *Power Delivery, IEEE Transactions on*, 10(1):92–96, 1995.

[14] W. Min, C. Hong, C. Jianneng, H.-L. Nguyen, J. B. Gomes, and S. Krishnaswamy. An overview of state-of-the-art partial discharge analysis techniques for condition monitoring. *IEEE Electrical Insulation Magazine*, to apprear in 2015.

[15] R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[16] D. W. Ruck, S. K. Rogers, M. Kabrisky, M. E. Oxley, and B. W. Suter. The multilayer perceptron as an approximation to a bayes optimal discriminant function. *Neural Networks, IEEE Transactions on*, 1(4):296–298, 1990.

[17] N. Sahoo, M. Salama, and R. Bartnikas. Trends in partial discharge pattern classification: a survey. *Dielectrics and Electrical Insulation, IEEE Transactions on*, 12(2):248–264, 2005.

[18] B. Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52:55–66, 2010.

[19] I. Žliobaitė, A. Bifet, B. Pfahringer, and G. Holmes. Active learning with evolving streaming data. In *Machine Learning and Knowledge Discovery in Databases*, pages 597–612. Springer, 2011.