Automated Feature Validation of Trip Coil Analysis in Condition Monitoring of Circuit Breakers

Michael Hosseini[1], Joseph Helm[2], Bruce Stephen[3], Stephen D. J. McArthur[4]

[1,3,4]*Electronic & Electrical Engineering, University of Strathclyde, Glasgow, G1 1XW, UK*

michael.hosseini@strath.ac.uk
bruce.stephen@strath.ac.uk
s.mcarthur@strath.ac.uk

[2]*Asset Management, Northern Powergrid, Wakefield, WF2 0XQ, UK*

joseph.helm@northernpowergrid.com

ABSTRACT

Datasets of historical performance metrics can offer valuable insight into an asset fleet's health. This is especially so in the context to establishing normal behavior and thresholds of acceptable performance for diagnostic purposes. However, plant performance can often be obscured by data quality issues which introduce artefacts that do not pertain to asset health. This paper utilises a supervised ensemble machine-learning approach to automate the process of filtering maintenance data based on their predicted validity. The results are then presented both in terms of classification performance, and the impact on the distributions directly. This helps to ensure engineers are basing their diagnostic decisions on valid data. The accuracy of the filtration process, and its effect on the final thresholds will be discussed. To illustrate, this paper uses data of varying quality on circuit breaker trip tests obtained from operational medium-voltage circuit-breakers spanning several decades with the aim of providing decision support for switchgear diagnostics.

1. INTRODUCTION

Medium-voltage circuit-breakers are used in the power industry to disconnect portions of the electricity network for reasons regarding safety, reliability, or efficiency. As such, a failure in their operation can lead to increased risks to health and profits, either through direct damage or punitive regulatory repercussions. From a consumer's perspective, it can lead to poor power quality or even complete loss of supply.

Many medium-voltage circuit-breakers deployed in the power network have expected life-spans exceeding several decades and are infrequently activated. This provides the opportunity for faults to develop unnoticed over time in between activations. These faults are then only encountered during an attempted activation, where the activation is either slower than permissible or fails to complete entirely. This can lead to catastrophic cascading failures if the activation was motivated by a time-critical change of state in the network, all whilst having afforded no opportunity for maintenance endeavours.

In order to avoid such eventualities, circuit-breakers are tested routinely under safe conditions. Among these tests, is the analysis of trip-coil currents over an activation. By measuring the trip-coil current, it is possible to non-invasively infer the circuit-breaker's internal mechanism's speed. It is consequently then possible to interpret the results to lead to a probable diagnosis. The theoretical basis for this has been well-documented (Harriezan, & Tiong, 2016), and is used in industry. However, its current application in industry does not utilise the information available in the data fully. This is due to the increased difficulty in correctly extracting and interpreting some of the more nuanced indicators of health.

This paper aims to aid the decision-making process in interpreting the data for arriving to a diagnosis. The simple process of exploring distributions of performance were hampered by pervasive data quality issues, skewing the results. Where datasets are large, it is impractical to manually review each case, necessitating an automated method. However, even establishing a ruleset manually for filtering can be laborious and often complicated. This is especially so when the causes of the data quality issue are not fully understood. This paper utilises a machine-learning ensemble to automate the filtration through the data-driven generation of the ruleset. The next section will outline the prerequisite background knowledge required to appreciate the context of the case study. The details of the data used, and the results are the explained. This automated filtering would be the pre-cursor, and input, to an automated analysis, diagnostic, and prognostic system.

2. BACKGROUND

Illustrated in Figure 1 is an example trace of the current in a trip coil, captured during circuit-breaker trip test. The specifics differ depending on the circuit-breaker mechanism, but for this section, the commonly used spring-operated mechanism is considered. The principle is that the base waveform of the current would resemble a trapezoidal shape, similar to a classic high-inductance circuit due to the coil, and then the deviations from this shape are then interpreted as movement from the plunger resulting in an opposing current (Beattie, 1996). The deviations are thus a function of the velocity and acceleration of the plunger. Knowing the mechanism, it is therefore possible to attribute specific deviations to known events occurring within the circuit-breaker. These knowledge-based features can then be interpreted for diagnostic purposes.

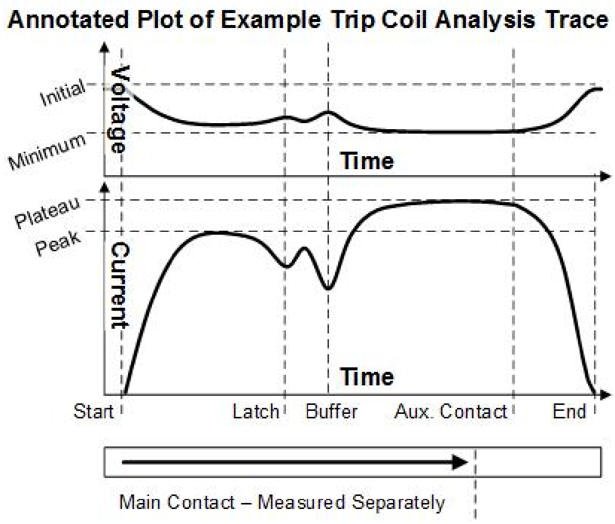

Figure 1. Example of a trip coil analysis trace with the features extracted annotated. There are both voltage readings and current readings. The Main Contact reading is measured separately.

Figure 1 highlights the features captured by the commonly used handheld device when recording the traces. Currently, the Main Contact Time (M_{Con}) is the only feature with guidance values associated with it; the rest of the analysis is down to the discretion of the engineer conducting the trip coil test. This is primarily due to the M_{Con} measurement being directly tied to regulations and thus being a key performance metric for manufacturers and operators alike. It is therefore the most established and standardised.

However, as literature indicates, there is much value to be extracted from the other data to indicate the health of the device, which can be used to preventatively intervene with suitable maintenance prior to a substandard performance. In order to incorporate the features into the guidance notes, it is necessary to know the expected values for each. The challenges associated with obtaining these are the primary motivating factor for this paper. A historical dataset of tests of operational performance of the asset base was made available in order to establish the fleet's expected performance, which can then be interpreted to provide guidance values, similar to the approach in (Strachan, McArthur, Stephen, McDonald, Campbell, 2007). However, it was discovered that the data quality varied significantly between samples, and it was required to remove erroneous results in order to provide a true distribution of performance.

In the presence of excess noise, or faults causing unexpected deviations in current, the feature extraction capabilities of the trip coil current recording device can be compromised (Speed, W. R., 2000). Under such circumstances, the features provided were either rogue values that could be easily identified, or incorrect values that are difficult to programmatically identify. This is due to the difficulty in distinguishing a poorly performing circuit-breaker from a poorly captured feature. These incorrect values skew the distributions of performance and could adversely affect the validity of the guidance values chosen based using said distributions. This is of relevance in both establishing the distributions, and for assuring the validity of new data prior to comparing to said distributions.

It is worth noting that in some cases, even an expert would struggle to identify the correct location for the feature, and that the traces vary significantly between manufacturers, making the task of automating the feature extraction very challenging. Most implementations, such as (Kezunovic, M., Ren, Z., Latisko, G., Sevcik, D. R., Lucey, J. S., Cook, W. E., & Koch, E. A., 2005), are based on expert tuning for each model. Based on this, and the fact that the overall performance of the feature extraction during usual cases is still very high, it was decided to filter out the cases of poor feature extraction instead of attempting to create a new feature extraction system with the intent of outperforming the original.

3. METHODOLOGY

A dataset of 250 historical, operational records were used to test the implementation, each from the same circuit-breaker model. The results are then shown both in traditional machine-learning contexts, as well as its effects on the distributions being used for establishing thresholds of acceptable performance. In practice, this methodology should be repeated for each model type of circuit-breaker.

The full dataset was first manually labelled by a domain expert for validation purposes. In practice, only a subset would be manually labelled for training, and then the algorithm would be applied to the remaining unlabeled samples. To emulate this, a test set of 100 records are withheld from the training process entirely. The remaining 150 records will be used for training using the well-established k-fold cross-validation method, with k set to 5. The labelling consists of accepting or rejecting the features based on the raw traces. Further details regarding the data are tabulated in Table 1.

Table 1. Table of the data used for training and testing.

Feature	Training Set		Testing Set	
	Valid	Invalid	Valid	Invalid
Latch Time	103	43	69	31
Buffer Time	108	42	74	26
A_{Con} Time	139	11	88	12
End Time	146	4	98	2
M_{Con} Time	123	27	82	18
Peak Current	123	27	84	16
Plateau Current	143	7	94	6
Initial Voltage	135	15	73	27
Min. Voltage	148	2	100	0

3.1. Features

Features represent the data being input into the machine-learning algorithm to improve its performance. The primary features are simply the features that were extracted by the commonly-used handheld device shown in Figure 1 that are being validated. However, in addition to this, standard statistical metrics were included regarding the raw current data. These metrics are tabulated in Table 2. They were captured in various levels of granularity. The first is using the entire trace, the rest segmented the trace and recorded the metrics from each section. Three segmentation approaches were applied:

1. Every trace is divided into windows of fixed length. The length was chosen by dividing the 98^{th} percentile longest trace into a predetermined number of windows.
2. Every trace is divided into a fixed number of windows.
3. Every trace is divided using the Start, Latch, Buffer, Auxiliary Contact, and End Times. These are shown in Figure 1. Where the values for these features were missing, the mid-point from the next adjacent feature was used. For example, if Latch Time was missing, the mid-point between the Start Time and the Buffer Time was selected in its stead.

It is known that noise or current-based faults can cause the feature extraction process of the handheld device to go awry. Segmenting the traces is motivated by attempts to localise the regions of instability, thus helping identify which features are most prone to miscapture.

Table 2. Table of the statistical metrics used as features.

Metrics
Minimum
0.05 Quantile
0.50 Quantile
0.95 Quantile
Maximum
Interquartile Range
Mean (Arthimetic)
Mode
Standard Deviation
Sum
Length
Max. / Mean

3.2. Machine Learning Methodology

An ensemble approach, using various well-established machine learning algorithms and associated hyperparameters was used. This was motivated by the difficulty in predicting the performance of a given machine learning algorithm prior to testing. This is especially the case in contexts such as the intended application of this paper, where multiple datasets are to be trained. This could mean the highest performing algorithm will vary depending on the dataset being analysed. Furthermore, it has been shown that an ensemble approach leveraging the differences in each algorithm's capabilities to arrive to a consensus can improve overall performance (Opitz, Maclin, 1999).

The ensemble's methodology is illustrated in Figure 2. The training stage involves dynamic selection of candidate representatives for the final ensemble. The final ensemble is to consist of a candidate from each of the three classifier types. For each classifier type, three hyperparameter settings are explored. Each of the three hyperparameter settings are tested five times, and the one with the highest mean performance is selected. This stratified selection process ensures diversity in the chosen members of the ensemble by forcing a representative from each algorithm. The outputs from each representative was then input into a final classifier to interpret the results.

The three classifier types were: decision trees, ensemble decision trees, and support vector machines. The decision trees had the maximum number of splits varied as their hyperparameter. The ensembled decision trees varied their ensembling mechanism as the hyperparameter. Ada-boosting, bagging, and RUS-Boosting were explored.

Finally, the support vector machines had their kernel functions varied. The three trialled were: linear, polynomial, and Gaussian. The final decision was taken by a Subspace Discrimant.

The emphasis placed on robustness for reliability in performance motivated the extensive reliance on ensembles, and the insistence of high levels of representation in its members. The balance may be shifted to further emphasise peak performance if desired.

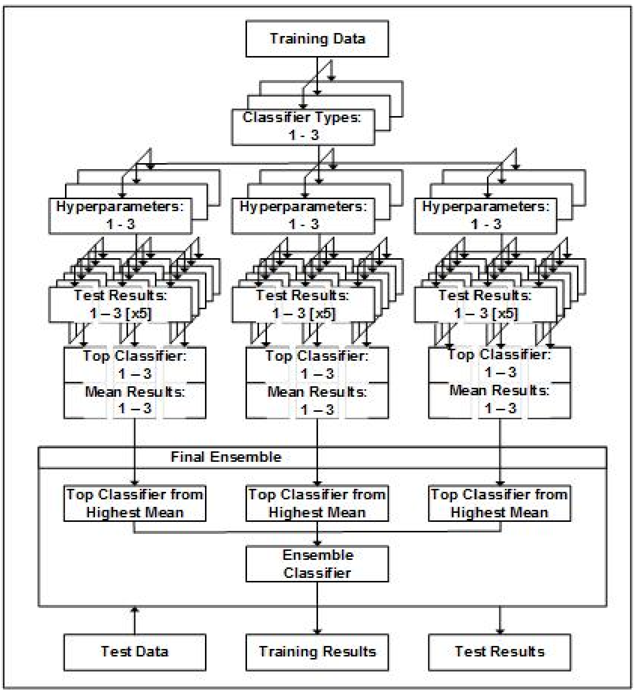

Figure 2. Figure of the ensemble methodology. The training stage includes a process for selecting the representatives.

4. RESULTS

The results of the classification system are shown in Table 3 using the F-Score, which is the harmonic mean of the precision and recall. Precision being the measure of true positives versus false positives, and recall being a measure of true positives versus false negatives. These classification results are most pertinent in the context of using the filter to validate new incoming features prior to benchmarking. A confusion matrix of a sample feature, Buffer Time, is shown in Table 4. For context, the results of the training showed that the highest performing representative inside the ensemble changed depending on the feature being filtered. Both the algorithm, and the hyperparameters varied; this somewhat validates the ensemble approach used to increase robustness in performance. It is worth noting that the features did not include information regarding the raw voltage traces, this is a likely factor in the reduced performance for the feature Initial Voltage.

Table 3. Table of F-Scores for each feature being validated.

Feature	F-Score
Latch Time	1.00
Buffer Time	0.99
A_{Con} Time	0.99
End Time	0.99
M_{Con} Time	0.98
Peak Current	0.98
Plateau Current	0.96
Initial Voltage	0.84
Min. Voltage	1.00

Table 4. Confusion matrix of the filtration of the Buffer Time feature.

N = 100	Predicted	
Actual	Invalid	Valid
Invalid	24	2
Valid	0	74

For the use-case of establishing the true distribution of performance, the filter's impact is better appreciated by mapping it to the distributions directly. A kernel density estimate function was used to predict the distributions of the populations. From this, the traditionally-used 5th and 95th percentiles were obtained. A sample distribution set for the feature Buffer Time is shown in Figure 3. From the top, the figures show the distribution when data is unfiltered (3a), when filtered using the labels (3b), and when filtered when using the machine-learned filter (3c). The parameters used for the smoothing function will greatly impact the distributions. The parameters, and thresholds should be tuned by an engineer. They are included in this context solely for indicative purposes. Table 5 tabulates the deviations in the thresholds caused by using the automated process versus the unfiltered case. The bandwidth parameter for the kernel density estimate function was increased for the current and voltage features, compared to the time features. This is due to their spread being less, necessitating the greater sensitivity.

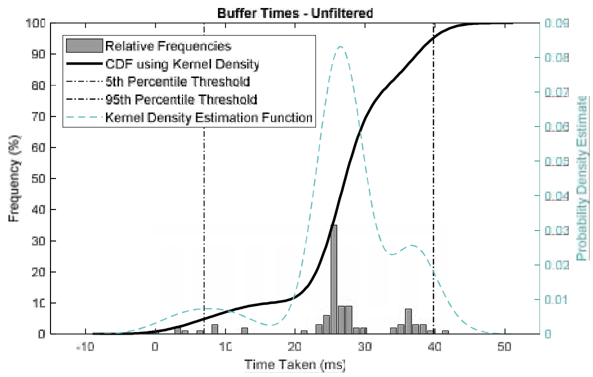

Figure 3a) Plot of unfiltered buffer times.

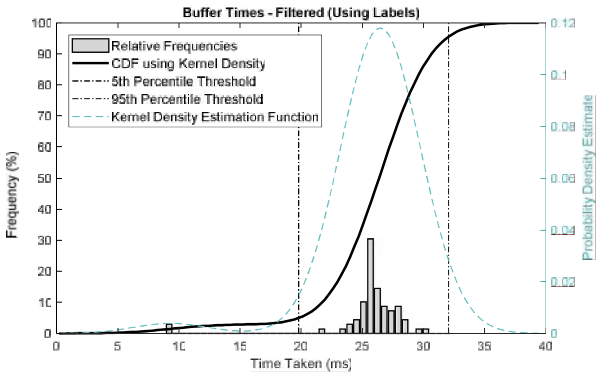

Figure 3b) Plot of filtered buffer times using labels.

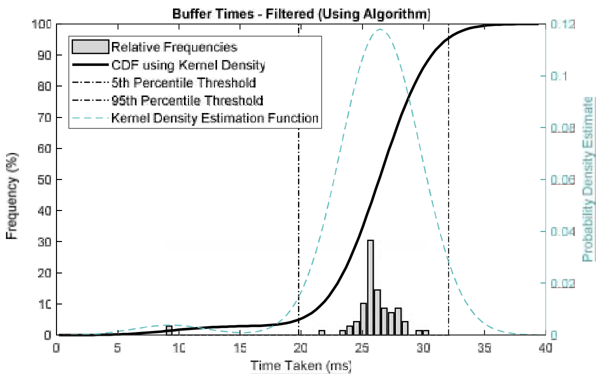

Figure 3c) Plot of filtered buffer times using algorithm.

Figure 3. Plots of the Buffer Time for the cases: Unfiltered (a), filtered using labels (b), and filtered using algorithm (c). The bars are the relative frequencies of times (left axis). The thresholds are based on the cumulative frequency distributions (left axis) from probability density estimates (right axis).

5. DISCUSSION

The impact of filtering on the thresholds based on the 5th and 95th percentiles are shown to be significant. It is reiterated that these threshold values are arbitrary and are based on a kernel density smoothing function, which itself is subject to arbitrary parameters. However, for indicative purposes, they still clearly demonstrate the potential of this methodology. With the guidance of an engineer for tuning, post-filtering, the results can be a useful decision support tool for establishing expected performances of existing devices for diagnostic purposes. It is worth noting that the values being filtered are not simply those at the tails of these distributions. There are cases where the values filtered, taken out of context, may seem entirely plausible and probable for a normally performing circuit-breaker. It is these values in particular, that may mislead an engineer, where this filtering becomes most impactful.

Despite the strong performance of the approach, it is important to note the potential limitations and underlying assumptions present. The first issue is regarding the training of the ensemble, or any similar machine-learning algorithm. The traditional method of independent and identically distributed random sampling used to maintain classifier performance post-training may neglect the importance of maintaining representation of rarer events within the sampled set. In the context of condition-monitoring, data representing healthy samples can be expected to greatly outnumber unhealthy sample. As an example of poor representation, the test set for Minimum Voltage had no examples of poor features. This may lead to unhealthy samples being conflated with healthy samples of poor data quality. Ideally, the sample size should be increased, but this means increased time taken for manual labelling. Failing this, it is important to validate the overall procedure for a given dataset. As such, where the data shows particularly poor representation of certain classes, this method should not be used.

Further work should include testing with data from different circuit-breaker models and incorporating features from the raw voltage trace for cross-validation purposes. An additional improvement may be to include the relative time taken to transition from one event to another. For example, the time taken from Latch Time to Buffer Time, as opposed to having each referenced from Start Time. This would require a check to ensure the previous event time is valid, but should isolate potential issues better.

6. CONCLUSION

Historical datasets can be used to establish expected performance measures. However, often there is the issue of data quality necessitating a filtering process. This can be costly and time-consuming. However, through the use of machine-learning, it is possible to automate the generation of the filter. This required manually filtering only a sample of the dataset to provide labels for training the machine-learning algorithm.

This paper used an ensemble of various well-established machine-learning algorithms consisting of decision trees and support vector machines of various hyperparameters in order to automatically filter an example dataset where the ground-truth is known. Depending on the particular feature being filtered, the F-Score ranged between 0.84 and 1.00. This was then mapped to the distributions and subsequently to the thresholds representing boundaries of acceptable performance. In this example, the 5th and 95th percentiles were used. It was shown that the deviation in accuracy caused by automating the process were minor. Though the results are promising, further work should include cross-validation through the inclusion of the raw data from the voltage trace, as well as further testing on different circuit-breaker models. Additionally, providing times relative to previous event may be more useful than times relative to the Start Time.

ACKNOWLEDGEMENTS

This work was supported by the EPSRC (Engineering and Physical Sciences Research Council).

REFERENCES

Harriezan, A., & Tiong, S. K., (2017). Trip coil signature measurement and analysis techniques for circuit breakers. *2016 7th International Conference on Intelligent Systems, Modelling and Simulations (ISMS)*. January 25-27, Bangkok, Thailand. doi: 11.1109/ISMS.2016.28

Beattie, S., (1996). Circuit breaker condition assessment by vibration and trip coil analysis. *IEE Colloquium on Monitors and Condition Assessment Equipment*, December 5, Leatherhead, UK.

Strachan, S. M., McArthur, S. D. J., Stephen, B., McDonald, J. R., & Campbell, A., (2007). Providing Decision Support for the Condition-Based Maintenance of Circuit Breakers Through Data Mining of Trip Coil Current Signatures. *IEEE Transactions on Power Delivery*, vol. 22, pp. 178-186. doi: 11.1109/TPWRD.2006.883001

Speed, W. R., (2000). Circuit Breaker Operator Signature Analysis. *Proceedings of the International Conference of Doble Clients*, Sec 4-4.

Kezunovic, M., Ren, Z., Latisko, G., Sevcik, D. R., Lucey, J. S., Cook, W. E., & Koch, E. A., (2005). Automated monitoring and analysis of circuit breaker operation. *IEEE Transactions on Power Delivery*, vol. 20, issue. 3, pp. 1910-1918. doi: 11.1109/TPWRD.2005.848466.

Opitz, D., Maclin, R., (1999). Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, vol. 11, pp. 169-198. doi: 10.1613/jair.614

Table 5. Caption of the table.

Features	Unfiltered values Vs. known filtered values						Predicted filtered values Vs. known filtered values					
	5th %		50th %		95th %		5th %		50th %		95th %	
	Value	Error	Value	Error	Value	Error	Value	Error	Value	Error	Value	Error
Latch Time (ms)	6.98	12.82	27.23	-0.78	39.79	-7.74	19.80	0.00	26.45	0.00	32.05	0.00
Buffer Time (ms)	28.47	0.71	39.01	-1.25	95.55	-50.93	27.47	1.71	37.76	0.00	44.62	0.00
A_{Con} Time (ms)	96.48	22.10	161.05	1.93	178.27	0.95	118.58	0.00	162.98	0.00	178.14	1.08
End Time (ms)	132.36	2.57	172.86	-0.34	193.63	0.37	132.79	2.14	172.52	0.00	194.00	0.00
M_{Con} Time (ms)	-2.39	95.06	111.13	2.38	128.76	4.49	96.91	-4.24	113.77	-0.26	133.99	-0.74
Peak Current (amp)	-0.56	3.26	4.74	0.40	6.77	0.00	2.89	-0.19	5.34	-0.20	7.18	-0.41
Plateau Current (amp)	3.72	0.75	6.95	-0.05	8.75	0.18	4.46	0.01	6.99	-0.09	9.03	-0.10
Initial Voltage (V)	30.25	2.87	34.72	2.74	37.28	-1.85	30.25	2.87	37.28	0.18	34.72	0.71
Min. Voltage (V)	30.49	0.00	33.18	0.00	35.53	0.00	30.49	0.00	33.18	0.00	35.54	-0.01