Alive on Back-feed Culprit Identification via Machine Learning

Bert Huang, Ansaf Salleb-Aouissi, Phil Gross
Center for Computational Learning Systems
Columbia University
New York, NY 10115
{bert@cs, ansaf@ccls, phil@ccls}.columbia.edu

Abstract

We describe an application of machine learning techniques toward the problem of predicting which network protector switch is the cause of an Alive on Back-Feed (ABF) event in the New York City power distribution system. When an electrical feeder is shut down, all network protector switches connected to the feeder should open to isolate the feeder. When a switch malfunctions and does not open, electrical current flows into the feeder, which remains energized. This causes the feeder to be "alive" on back-feed current, and maintenance cannot proceed. Our goal is to provide a ranking of network protector switches according to their susceptibility to such malfunction. Such a ranking can assist prioritization of which switches to repair when an ABF event occurs. We compare three methods for computing a ranking: an SVM classification approach, a maximum entropy density estimation approach and an SVM-ranking approach.

1 Introduction

This article describes an application of machine learning techniques toward the problem of fault detection in the power distribution system of New York City. Specifically, we address the problem of identifying the cause of *Alive on Back-Feed* (*ABF*) events. These events occur when a *feeder* (a large distribution cable for electricity at medium voltage) is shut down, either automatically due to a fault or manually for maintenance, but one of its isolation mechanisms (called *network protector switches*) malfunctions, and electrical current continues to flow. The task is to identify which switch is malfunctioning. Our goal is to apply machine learning to produce a ranking of network protector switches according to their likelihood of experiencing such a malfunction. Such a ranking can help engineers prioritize inspections and repair of these important switches.

Consolidated Edison, the power management company of New York City typically experiences over 700 ABF events a year. Each event requires approximately nine hours of work by a two-person crew to identify the culprit. Thus, reductions in the search time lead directly to substantially reduced costs. Furthermore, these events occur when feeders are shut off for service or repair, which means these activities cannot begin until the ABF event is cleared. Further, when the feeder is shut down, adjacent feeders must bear the load of the disabled feeder. Faster return of the feeder to service will reduce the amount of time the network is overloaded.

1.1 Previous work

Previous collaborations between Consolidated Edison and CCLS include predicting feeder failures using machine learning [3, 4]. Our work uses algorithms detailed in previous work on support vector machines [1] used for classification and ranking [6], as well as density estimation using the maximum entropy approach [2] with missing data [5].

1.2 Outline

The remainder of this article is organized as follows. In Section 2, we describe the alive-on-back-feed problem in detail. In Section 3, we describe the data which we are using to train and describes some of the practical challenges we face in applying machine learning for this problem. In Section 4, we discuss the algorithmic approaches we compare, their assumptions and implementation details. In Section 5, we illustrate the results of applying these approaches and describe some evaluation methods. In Section 6, we discuss the current status of this project, including how exactly Consolidated Edison is using our results in practice, as well as future directions.

2 Alive on back-feed

Topologically, a feeder cable is a tree-like structure, rooted at a substation, and, in New York City, an average of 25 distribution transformers at the leaves, where the electricity is stepped down to household voltages for delivery to customers. Feeder length ranges from a few hundred meters to ten kilometers or more, and there are roughly 1,000 underground network feeders in the city, which were the focus of our study.

When a feeder experiences any type of fault or failure, or when it is de-energized for maintenance work, a relay opens at the substation, disconnecting it from the transmission sources. Network protector switches (NWPs), located at each distribution transformer, detect the absence of upstream current and automatically disconnect the transformer from the secondary (household voltage) distribution grid. When all NWPs have opened, the feeder is de-energized and safe to work on.

Due to the high energies involved, NWPs are substantial physical devices with heavy-duty automatic switches. On occasion, they malfunction. When a NWP fails to open, electricity "back-feeds" from the secondary grid into the feeder. Maintenance or repair on the feeder cannot commence until the malfunctioning NWP is found and manually opened.

Unfortunately, telemetry from the system is transmitted over the feeder cable itself to the substation. When the feeder's relay opens at the substation, this data stream goes mostly silent. Thus the only way to locate the malfunctioning NWP is to perform what is, in effect, a linear search on the set of transformers belonging to the feeder, sending a crew to visit vault after vault looking for the still-closed NWP.

3 Data description

The data used for the analysis comes from a variety of sources. Two of the most significant are a capital equipment database, which has good information about precisely what was purchased and when, and an inspection-record database with observations from periodic inspections of transformer/NWP vaults.

Additionally, there is a database of information on the vaults themselves, including their own maintenance records, history of flooding, and so forth; information on how many different transformers have been installed to each vault over the years, which may show evidence of a problematic vault; and historical information on transformer loads and feeder failures. All of this data is joined together in a single table for training.

Once the various sources of data are aggregated, the result is a data set with a number of less desirable properties for easy application of machine learning techniques. This section describes some of the challenges that we attempt to address.

3.1 Interpretation of labels

The label information available for training is not exactly the same form as most standard ranking, classification, or density estimation problems. Each ABF event is localized to one specific feeder. After the incident has been fully resolved, one of the network protectors on this feeder will have been identified as the culprit NWP. No new information is available about the network protectors on any other feeder. Furthermore, the purpose of our output is to guide engineers during their search for the culprit network protector. This search is local to the single feeder suffering the ABF, and predictions about the likelihood of network protectors being culprits on other feeders are not relevant.

If we consider the likelihood of causing an ABF event to be dependent on the feeder, and not only on the qualities of a network protector, then we must consider the problem to be many independent ranking problems, each corresponding to a feeder. On the other hand, if we are only interested in the conditional likelihood of a network protector causing an ABF event given its feeder, we can consider the problem a single ranking for all the network protectors in the city, taking into account the interpretation of the labels described previously.

3.2 Missing data

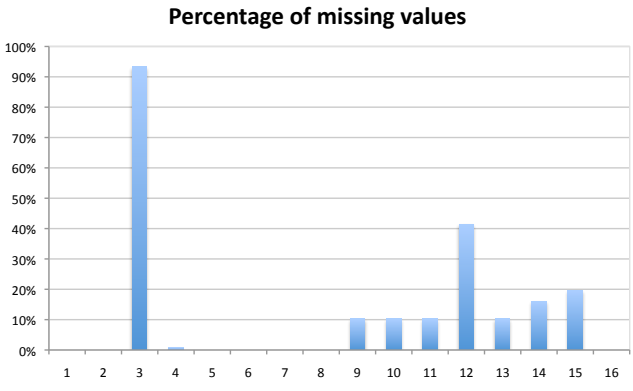

Figure 1. Percentage of missing values per attribute. Some attributes are collected manually during inspections or by equipment installed on only part of the population.

As in many real-life applications, our application suffers from the problem of missing data. Figure 1 shows the ex-

tent of missing data on each attribute. Depending on the approaches we adopted, we either (1) discard all the attributes that are missing for more than 50% of the examples, and perform a mean imputation to fill in the missing values of the other attributes or (2) mark the missing values and employ a special algorithm designed to handle incomplete data.

3.3 Sparsity

Finally, the number of ABF events is disproportionately low compared to the number of network protectors in the system. This is quite good news for the engineers and customers of the distribution system, however not so for statistical analysis. This presents issues when using models with high complexity or high dimensionality, because we may not have enough data to populate the model of unhealthy network protectors. In the data set, there are $28,605$ network protectors in total, and only 713 labeled culprits (2.5% of the total population).

4 Approaches

We apply three approaches to learn from ABF data. Each approach attempts to handle the challenges discussed in the previous section with different ideas.

For all three methods, we prepare the data by first converting categorical feature values to numerical values using the standard method of splitting each categorical feature into individual binary features, each representing a possible setting of the categorical feature. We also scale all numerical features to lie in the range $[0, 1]$. We set parameters according to the best performance on 4-fold cross-validation, and finally evaluate on a held-out test set of the 100 most recent ABF culprits. Since we want to simulate the testing scenario that will occur in the future, after we have produced our ranking, we "hold-out" an example as a blind test case by labeling it as "healthy" in the training data.

4.1 SVM Classification

The first approach is to treat the problem as a straightforward classification algorithm. We thus ignore the feeder structure of the training data. We use a support vector machine with a linear kernel and mean imputation for the missing data. In order to allow the algorithm to behave reasonably with our rather imbalanced data set, we weight the hinge loss penalty proportionally to the size of the classes. In other words, misclassifying each culprit, of which there are few, as healthy is penalized more than misclassifying non-culprits, of which there are many. The cost function is thus a slightly modified version of the classical SVM cost:

$$\begin{aligned} \min_{w,\xi} \quad & \frac{1}{2}||w||^2 + C\sum_{y_i=-1}\xi_i + RC\sum_{y_j=+1}\xi_j \\ \text{s.t.} \quad & y_k[w^\top x_k + b] \geq 1 - \xi_k, \quad \forall k, \end{aligned} \tag{1}$$

where x's are the NWP feature vectors, y's are their labels, w is learned weight vector, the ξ values are slack variables used during optimization, C is the regularization parameter, and R is

$$R = \frac{|\{y_i|y_i = +1\}|}{|\{y_j|y_j = -1\}|}. \tag{2}$$

In other words, R scales the hinge loss penalty from the cost function proportionally to the size of each class.

Finally, we rank the resulting output by the numerical prediction value, or the signed distance to the discriminating hyperplane.

This approach has the advantage of casting the problem into the popular and well-studied classification framework. On the other hand, this casting may oversimplify for various reasons. As discussed in Section 3, casting non-culprit examples as a "healthy" class cannot be correct because our goal is essentially to rank these supposedly-healthy NWPs in order of which will be most likely to malfunction.

4.2 Maximum Entropy Density Estimation

The second approach is to treat the problem as *presence-only*. The idea of casting the problem as presence-only means that there are no negative examples, which may be a better representation of reality than considering network protectors that have never been culprits in an event as healthy, negative examples. Choosing a presence-only framework also has the advantage that recent work demonstrates a principled method of handling presence-only problems with missing data using maximum entropy density estimation (maxent) [5]. The algorithm thus assumes that all ABF culprits are drawn from some distribution of unhealthy network protectors. We then find the maximum entropy distribution over all network protectors subject to the constraints that the feature averages of the empirical, unhealthy network proctor distribution and the estimated distribution are similar. Formally, we consider the set of all NWPs to be the state space χ, and the m culprits to be a sample from χ. We write μ_j to represent the empirical average of the culprits' j'th feature values:

$$\mu_j = \frac{1}{m}\sum_{y_i=+1} x_i(j).$$

Then we find a probability distribution over all entries in χ according to the following objective:

$$\max_p \quad H(p) \tag{3}$$

$$\text{s.t.} \quad \sum_{x_i \in \chi} p(x_i) = 1$$
$$\left| \sum_i p(x_i) x_i(j) - \mu_j \right| \leq \beta \sigma_j, \quad \forall j$$

where $p(x_i)$ refers (using a slight overload of the symbol x_i) to the probability that the NWP corresponding to feature vector x_i is a culprit, $H(p)$ is the Shannon entropy of the distribution, and β is a regularization parameter, which, with the standard deviation σ_j of feature j, determines how much the estimated expectations can deviate from the empirical averages. In other words, we maximize entropy subject to constraints that the empirical averages of all features are within β standard deviations of the expectation of the predicted distribution, where β is a regularization term selected via cross-validation.[1] We optimize this objective using axis-parallel gradient descent in the dual [5]. We rank the network protectors by the estimated likelihood of being drawn as the next unhealthy unit.

4.3 SVM-Ranking

The third approach is to treat each ABF event as ranking information. Since each ABF event identifies a culprit and a number of non-culprits, each such event can be treated as a ranking constraint that the culprit must be ranked higher than all the non-culprits along the feeder. We apply mean imputation on the missing data and SVM-ranking [6] to extract an overall ranking from the data. Formally, we optimize the objective

$$\begin{aligned} \min_{w,\xi} \quad & \tfrac{1}{2}||w||^2 + C \textstyle\sum_i \xi_i \\ \text{s.t.} \quad & w^\top x_i \geq w^\top x_j - \xi_i, \quad \text{for all ABF culprits } x_i \\ & \qquad\qquad\qquad\qquad\quad \text{and } x_j \text{ on same feeder.} \end{aligned} \tag{4}$$

Here, the x_j's (where $i \neq j$) on the same feeder as the culprit are the innocent "suspect" NWPs for the ABF events, the ξ values are slack variables used during optimization and C is once again a regularization parameter. In other words, each ABF event forms a set of hinge-loss constraints that enforce that the prediction value of the culprit is greater than that of all non-culprits (subject to slack).

This approach seems to most accurately represent the problem, albeit without as principled a method as maxent for handling missing data. Nevertheless, it seems that given the nature of ABF events, it is more accurate to represent the labels as these ranking constraints.

[1] We omit the details of how to handle missing values for readability and refer the reader to [5] for more. In essence, the empirical averages and expectations can be taken only over samples whose values are known, with proper bookkeeping.

Table 1. Quantitative evaluation of learned rankings on held-out test set. AUC is the area under the Receiver Order Characteristic (ROC) curve, and a higher AUC is better. Avg. Position represents the average ranking of the culprits amongst other network protectors on their feeder. Interestingly, while SVM-rank scores a higher AUC on its overall ranking, Maxent yields a lower average position of culprits along their feeders.

Method	AUC	Avg. Position
SVM	0.82338	6.49
Maxent	0.82701	5.07
SVM-rank	0.84169	6.41

5 Results

We evaluate the three methods on the held-out test set by measuring the quality of the overall ranking as well as individual sub-rankings corresponding to feeder groupings. To measure overall ranking quality, we compute Receiver Operator Characteristic (ROC) curves and the area under the curve (AUC). Figure 2 contains ROC curves and histograms of the three methods on the test set. The results indicate that the three methods perform comparably on the 100 most recent ABF events, and that all three seem to rank rather well.

Perhaps more importantly, Figure 3 contains histogram plots of the culprit ranks within each feeder group. This metric is closer in spirit to the proposed usage of our ranking, in which one feeder is experiencing back-feed and engineers must search for the culprit amongst the network protectors along the feeder. The top row histograms represent the raw rank of each culprit along its feeder. In other words, if we use the ranking exactly, the total number of network protectors that are checked before finding the culprit. Since different feeders have different numbers of network protectors, we also plot in the second row the percentile of the ranked network protectors along the feeder that the culprit appears. Qualitatively, it is difficult to tell which ranking is better according to these histograms, but the three methods seem to place most of the blind-test ABF culprits early in their feeder rankings. Quantitatively, the AUC values of the overall ranking and the average position along each feeder are listed in Table 1. Interestingly, while SVM-rank scores a higher AUC on its overall ranking, Maxent yields a lower average position of culprits along their feeders.

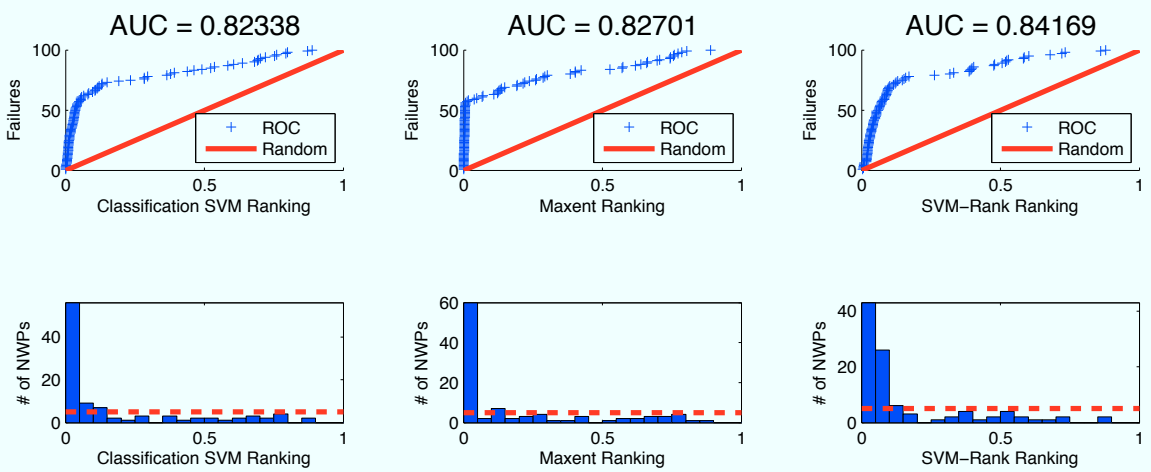

Figure 2. ROC curves and histograms of the three methods on the test set. ROC curves show the true positives over the false positives. The diagonal is equivalent to expected performance if the ranking is random. The bottom row histograms depict the number of culprits in each bin of the overall ranking. The dotted red line represents the expected bin sizes if the ranking is random. SVM classification results are in the left column, maximum entropy results are in the middle column, and SVM ranking results are in the right column.

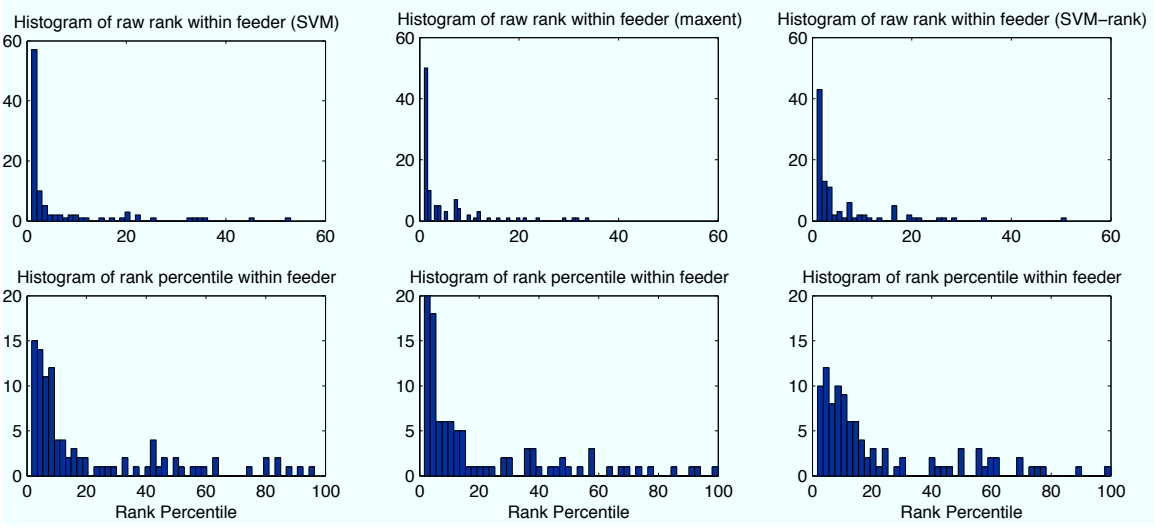

Figure 3. Histogram plots of the culprit ranks within each feeder group. The top row histograms represent the raw rank of each culprit along its feeder. In other words, if we use the ranking exactly to search for the culprit, each data point is the total number of network protectors that are checked before finding the culprit. A more effective ranking would rank the culprit early in the order. The second row histograms represent the percentile of the ranked network protectors along the feeder that the culprit appears, which normalizes for the fact that feeders have different numbers of NWPs. SVM classification results are in the left column, maximum entropy results are in the middle column, and SVM ranking results are in the right column.

6 Discussion

We have described our experience applying machine learning to help guide maintenance decisions for the electrical distribution system of New York City. We compare three approaches, including a classical method (SVM) and a new method (Maxent).

While our experiments compared three methods, we have only delivered one ranking to our collaborators at Consolidated Edison. At the time these experiments were first run, which was in February 2009, the SVM-rank approach was still in its early stages. Even though the maximum entropy approach performed slightly better according to the AUC metric and used a more principled approach, the paper describing the algorithm had not yet been published. In addition, our group had employed SVM in previous collaborations with Consolidated Edison so it was a method with which both parties were comfortable. Thus, we delivered the results from the simpler SVM classification approach. This ranking is currently in the process of being integrated into Consolidated Edison's ABF-management application to optimize the search for ABF culprits.

Furthermore, we limited ourselves to considering the model in static settings. We are exploring methods to take advantage of time-series data to more accurately model the probabilities, which likely change with seasons and generally drift over time. In future iterations, we will include results from the newer approaches described in this article as we have had time now to validate our methods and results, as well as possible extensions using more dynamic data and models. Our rankings do not take into account many practical factors such as driving distance, which may be more important than our estimated rank in terms of scheduling the search for the ABF culprit. The ranking we provide is currently used as a tiebreaker when the engineers planning the ABF maintenance have narrowed down their search via traditional engineering knowledge.

Acknowledgements

We would like to thank Nathanael Gay, Jim Gaughan, and Maggie Chow at Consolidated Edison for their support with data aggregation.

References

[1] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[2] M. Dudik, S. Phillips, and R. Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *J. Mach. Learn. Res.*, 8:1217–1260, 2007.

[3] P. Gross, A. Boulanger, M. Arias, D. L. Waltz, P. M. Long, C. Lawson, R. Anderson, M. Koenig, M. Mastrocinque, W. Fairechio, J. A. Johnson, S. Lee, F. Doherty, and A. Kressner. Predicting electricity distribution feeder failures using machine learning susceptibility analysis. In *AAAI*. AAAI Press, 2006.

[4] P. Gross, A. Salleb-Aouissi, H. Dutta, and A. Boulanger. Susceptibility ranking of electrical feeders: A case study. *CCLS Technical Reports*, April 2008.

[5] B. Huang and A. Salleb-Aouissi. Maximum entropy density estimation with incomplete presence-only data. In D. van Dyk and M. Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume Volume 5 of JMLR: W&CP, pages 240–247, April 2009.

[6] T. Joachims. Optimizing search engines using clickthrough data. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 133–142, 2002.

[7] T. A. Short. *Electric Power Distribution Handbook*. CRC Press, 2003.