Power System Online Stability Assessment Using Active Learning and Synchrophasor Data

Vuk Malbasa, *Member, IEEE*, Ce Zheng, *Student Member, IEEE*, and Mladen Kezunovic, *Fellow, IEEE*
Department of Electrical and Computer Engineering
Texas A&M University
College Station, TX 77843-3128, U.S.A.
E-mail: vmalbasa@tamu.edu

***Abstract*—Analysis of synchrophasor measurements by means of data mining tools in pursuit of precise stability margin estimates requires the analysis of large amounts of data. The process of learning the underlying power system behavioral patterns introduces a significant computational burden dictating the sophistication of the tool used and the amount of data prepared for and analyzed during learning. Advancements in machine learning may make it possible to reduce the amount of data that need to be analyzed without decreasing accuracy of predictions. Assuming that a probabilistic learning tool is allowed to interactively query a time domain simulation system for exact stability margins, we show that using an active learning methodology significantly reduces the amount of data that needs to be processed. Results show that the advantage of active learning approach is greater on more complicated prediction tasks, those requiring a large amount of data for accurate predictions.**

***Index Terms*—Data mining, phasor measurement units, power system stability**

I. Introduction

Traditional power system stability assessment relies on detailed system modeling and time domain simulations to estimate the stability condition of interest. While this approach is straightforward and accurate as long as a precise system model and adequate measurements are used, it may introduce significant computational complexity, considering the large size of modern power systems.

The recently emphasized importance of real-time stability monitoring has led to applications based on data mining tools such as Classification and Regression Trees [1]. While such tools deal exceptionally well with temporal complexity at the time of prediction, as compared to time domain simulations, the training process involved is still a major obstacle to implementation.

In this paper we will focus on reducing the computational burden of training data mining tools by applying a pool-based active learning methodology. This methodology allows reduction in the number of examples that need to be processed via time domain simulations and considered during learning, while retaining the prediction accuracy by capturing adequate power system stability behavior.

II. Background

Data mining tools have been previously applied in power systems to assess the transient stability [2], system operational security [3], system post-disturbance stability [4], and other areas where the complexity of detailed model computations may be alleviated by creating highly accurate but approximate predictors. In [1] and [5], the authors have used data mining tools in order to quickly estimate the system voltage and small-signal stability margins.

Active learning methodologies have previously been explored in cases where labeled instances are time consuming, difficult or expensive to obtain. Three major scenarios can be differentiated in literature: pool-based, stream-based, and membership query synthesis.

In pool-based active learning a large pool of unlabeled data is available to the data mining tool and the task is to select which examples from the pool need to be labeled for accurate prediction. Pool-based active learning has often been explored in situations where it is necessary to have a human expert provide labels for data, for example natural language processing tasks [6] and classification of networked data, such as links in web pages or social network data [7]. In computer network intrusion detection the data sets may be prohibitively large for direct application of data mining tools and therefore a workable subset needs to be generated [8]. Pool-based active learning has also been considered for classifying software behavior [9].

The active learning methodology has also been applied in a real-time setting. Here the examples are observed as a stream of data and upon observing an example an immediate decision needs to be made as to whether to compute or ask for labels. Recently, the problems related to stream-based active learning have been addressed in [10] and [11].

This work is supported by Power System Engineering Research Center (PSerc) under the project S-44 titled "Data Mining to Characterize Signatures of Impending System Events or Performance Using PMU Measurements", and in part by Texas A&M University.

Membership query synthesis is the active learning scenario where queries may be generated from the entire input space for a data mining algorithm, and the task is not to select an example to label, but to generate an experiment to perform and obtain results from. The idea of query synthesis has been explored for learning the absolute coordinates of a robot hand based on joint angles in [12], and more recently in bioinformatics to discover metabolic pathways [13]. A detailed and recent overview of the literature is given in [14].

III. Methodology

The task of stability margin prediction may be cast as a data mining classification problem [5]. In this case a data mining tool is used to create a mapping from the system measurements, in our case the positive sequence voltage magnitude and angle, and the positive sequence current magnitude and angle, into one of the pre-determined stability states. If we take the small-signal oscillatory stability as an example, the damping ratio (*DR*) of critical oscillation mode may be used as the stability margin indicator, and two basic stability states can be defined as: stable (with critical damping ratio, $DR > 0$) and unstable (with $DR < 0$). Similarly, we can define the voltage stability margin (VS_{margin}) as being stable or unstable in terms of the MW-distance of the current system operating point (OP) from the critical voltage collapse point (saddle-node bifurcation point) on the P-V curve. In this work the voltage stability threshold is set at VS_{margin}=30%. This value can be further adjusted according to the real-time operational needs. A detailed overview of the methods used to obtain VS_{margin} and *DR* are in [5].

For simplicity of notation let us denote the synchrophasor measurements across a power system, including voltage magnitude and angle, and current magnitude and angle, characterizing the system in an operating point i as $\boldsymbol{x}_i = [x_{i1}, x_{i2} \ldots x_{i4P}]$, where P is the number of installed phasor measurement units (PMUs) in the system. Through time domain simulation both the *DR* and VS_{margin} can be obtained. In the case of voltage stability classification for each system OP i let us denote the voltage stability in label $y_i = 1$ if $VS_{margin} > 30\%$ and $y_\mathrm{i} = -1$ otherwise. In the case of oscillatory stability classification let us label with $y_\mathrm{i} = 1$ if the associated oscillatory stability state is stable ($DR > 0$) and −1 otherwise.

Let's gather all measurements and their associated labels into ordered tuples into a labeled data set $\boldsymbol{D}_\mathrm{L} = \{(\boldsymbol{x}_\mathrm{i}, y_\mathrm{i},), i = 1 \ldots N\}$, where N is the number of system operating points. A data set $\boldsymbol{D}_\mathrm{L}$ that may be used to train a data mining tool for either voltage or oscillatory stability margin predictions is produced through extensive time-domain simulations. Let us also introduce the notation $\boldsymbol{D}_\mathrm{U}$ for a pool of unlabeled measurements, consisting of OPs without their associated stability margin labels.

In our previous work [1, 5], we found that among the systematically generated OPs some are redundant and others are spurious. Spurious data can be considered as outliers that should be removed from the training dataset. The important issue of how to intelligently remove the redundant data from the training data set is addressed by assuming all the data points are unlabeled, in $\boldsymbol{D}_\mathrm{U}$, and applying the pool-based active learning methodology presented here to label and include the chosen points into $\boldsymbol{D}_\mathrm{L}$. The procedure can be iterated until a desired accuracy threshold is reached, or the budget of data points that may be learned from is filled. The oracle provides labels, when moving data points from $\boldsymbol{D}_\mathrm{U}$ into $\boldsymbol{D}_\mathrm{L}$, directly from the original data set without additional computation. In this case the presented pool-based methodology presented here reduces only the computational costs associated with learning, and is used to filter out redundant data, since the data is assumed to be labeled a priori.

In the case where labels for all OPs are not provided in advance a substantial reduction in both time domain simulation and learning may be possible. The question of which y_i are necessary to compute via time domain simulation in order to make accurate predictions of stability margins, and which are not, based on the generalization power of the employed predictor, is addressed by the proposed approach. In effect, we will use the probabilistic and generalization properties of Artificial Neural Networks and Support Vector Machines to decide which system states should be labeled and consulted during training and which should not because they contain redundant information.

A. Artificial Neural Networks

Artificial Neural Networks (ANNs) are a biologically inspired mathematical model with significant applications in data mining. Feed-forward neural networks are composed of interconnected processing units, or neurons, each of which compute a simple transfer function, most commonly the logistic sigmoid, based on sum of their inputs and produce an output, which may then be fed as the input into other neurons until the output stage is reached. Therefore a neural network may be characterized by the number and connections between neurons.

In our case the network architecture is a directed acyclic graph having a $4P$ number of input neurons and one output neuron, with a hidden layer of 10 neurons in between. Training is performed by adjusting the weights of the connections between neurons until a close match between the inputs $\boldsymbol{x}_i$ and the desired output, either y_i, is obtained through the network across all training examples i. When making a prediction the input is propagated through the network and a continuous output value is produced at the output neuron.

In traditional applications to classification tasks the output of ANNs is compared to a threshold in order to obtain a discrete prediction. For active learning, however, we will use the raw output as is typically seen in regression tasks because it can be used to provide a measure of uncertainty.

A specific property of feed-forward artificial neural networks using a logistic sigmoid transfer function is that due to the properties of the transfer function this tool generalizes the entire possible input space even if only a few examples are available for training, and therefore may falsely provide highly confident predictions for unseen examples which are very dissimilar to any observed data points.

B. Support Vector Machines

Support Vector Machines (SVMs) are mathematical models which in their simplest form solve a linearly separable classification problem by finding the maximum-margin hyper-plane separating the two sets of points. Predictions on unseen examples are made by computing their distance to data selected to be part of the support vector set during training. Predictions on unseen examples then assigned labels based on the closest examples in the support vector set. By employing an implicit high dimensional representation, through calculating distance based on metrics other than Euclidean distance, often called kernel distance, it is possible to solve non-linear problems. Through the introduction of slack variables SVMs may be applied to data which is not separable or contains noise.

For the pool-based active learning methodology presented here the SVM is used in regression mode, as an implicit probabilistic classifier (see Active Learning Methodology), which may be considered to provide the probabilities of an example belonging to each class. There are several variants of SVMs distinguished by the kernel function that is employed to compute distance between observed data. These include the linear, polynomial and logistic kernels; however for our work we have obtained most accurate results using the radial basis function (RBF) kernel which resembles a Gaussian probability distribution function. Unlike the logistic sigmoid used in neural networks a properly trained SVM using the RBF kernel does not provide confident predictions for points which are dissimilar to examples observed during training. For the following experiments the SVM is used as implemented in LibSVM library [15].

C. Active Learning Methodology

In active learning a probabilistic data mining tool is used to interactively query a source of information (or oracle) that is assumed to always be correct, but is expensive to use. In our work the oracle is time domain simulation of a power system. With pool-based active learning we assumed a large number of unlabeled measurements $\boldsymbol{x}_i \in \boldsymbol{D}_{\mathrm{U}}$ are available without their associated labels y_i. In this work we have explored an active learning methodology based on uncertainty sampling by choosing to label those examples whose class probability is closest to 0.5. To obtain the uncertainty a predictor has about an unseen example, based on the output of a trained ANN or SVM, requires the scalar continuous output of the applied predictor $f(\boldsymbol{x}_{\mathrm{i}})$ to be transformed into the probability of that example belonging to the positive class $p(y_{\mathrm{i}} = 1|f(\boldsymbol{x}_{\mathrm{i}}))$. This can be accomplished by the transformation [16]

$$p(y_{\mathrm{i}} = 1 \mid f(x_{\mathrm{i}}) = \frac{1}{1 + \exp(Af(\mathbf{x}_{\mathrm{i}}) + B)} \tag{1}$$

This function is monotonous and increasing for any value of B and of $A < 0$. Therefore we may conclude that the output of ANNs and SVMs can be implicitly interpreted as the class probability and used directly in active learning by considering predictions $f(\boldsymbol{x}_{\mathrm{i}})$ closer to 0 in absolute terms as more uncertain, or having $p(y_{\mathrm{i}} = 1|f(\boldsymbol{x}_{\mathrm{i}}))$ closer to 0.5, than those farther away from 0.

The proposed active learning procedure is initialized by asking the oracle to provide the labels for a small number of examples from $\boldsymbol{D}_{\mathrm{U}}$, removing them from $\boldsymbol{D}_{\mathrm{U}}$ and including them in $\boldsymbol{D}_{\mathrm{L}}$. After learning on $\boldsymbol{D}_{\mathrm{L}}$ the tool makes a prediction on all the examples for which labels have not yet been computed, $\boldsymbol{D}_{\mathrm{U}}$, and finds those which have predictions closest to 0 in absolute terms. In other terms the unlabelled examples are sorted according to certainty the tool has about their label and those with highest uncertainty are used to query the oracle again.

PSEUDO-CODE FOR POOL-BASED ACTIVE LEARNING
1. Label and remove a small subset of examples from $\boldsymbol{D}_{\mathrm{U}}$ and place into $\boldsymbol{D}_{\mathrm{L}}$ 2. While stopping criteria is not met: a) Train classifier on $\boldsymbol{D}_{\mathrm{L}}$ b) Make predictions on $\boldsymbol{D}_{\mathrm{U}}$ c) Choose a small subset of $\boldsymbol{D}_{\mathrm{U}}$ based on maximum uncertainty, remove them from $\boldsymbol{D}_{\mathrm{U}}$, acquire labels for chosen examples from oracle and include them in $\boldsymbol{D}_{\mathrm{L}}$

IV. EXPERIMENTS

In our experiments two test systems, the IEEE 3-machine 9-bus system and IEEE 10-machine 39-bus system, are used to illustrate the performance gain from the proposed scheme. These two systems are known for their realistic configurations and robustness in testing stability-related applications. The OPs generated for these two systems are summarized in Table I.

TABLE I
OPERATING POINTS GENERATED FOR TRAINING OF DATA MINING TOOLS

System	OPs Generated for Oscillatory Stability Estimation		OPs Generated for Voltage Stability Estimation	
	Stable OPs	Unstable OPs	Stable OPs	Unstable OPs
9-Bus	**1021**	**50**	**404**	**21**
39-Bus	**4950**	**126**	**1843**	**59**

For the following experiments the pool-based active learning methodology was used to train SVMs and ANNs. We first performed experiments in batch-mode using 5-fold cross-validation to obtain the optimal parameters for SVM and ANN training, and the used these parameters to test the active learning approach.

We compared the performance of training on OPs chosen by active learning with training on random subsets of equal size. In the following figures each horizontal axis represents the number of OPs that were used for training, chosen either through active learning (full blue line) or random sampling (dashed red line), while the vertical axis represents the ratio of correctly classified examples to total examples. Because of the class imbalance we also present the results of the mean predictor (green dotted line) which always predicts the majority class, in our case the positive or stable class.

At each step of the proposed method we chose to label a single example from $\boldsymbol{D}_{\mathrm{U}}$ and include it in $\boldsymbol{D}_{\mathrm{L}}$. Testing is then performed across the entire set of generated OPs in order to illustrate the generalization power of the proposed approach, however this step is not necessary in real applications. In each experiment four initial OPs were labeled by the oracle in order to start the procedure.

A. Support Vector Machine Experiments

Let us first consider the 9-bus system and the problem of oscillatory stability classification. From Fig. 1 we note that from the start of the procedure active learning outperforms random sampling. Random sampling starts to outperform the mean predictor only after 50 examples have been labeled.

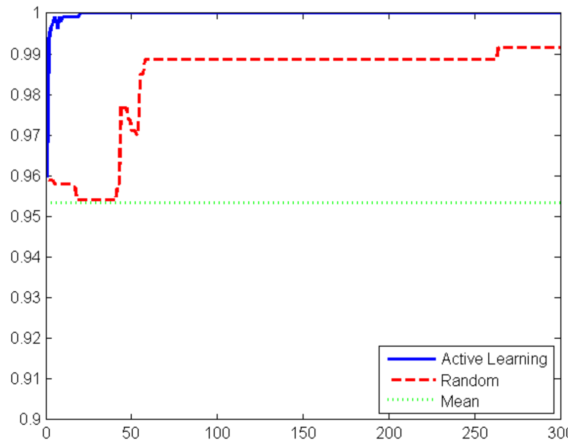

Figure 1. Comparison of active learning and random sampling on the 9-bus system for the oscilatory stability classification task using SVM

In Fig. 2 we show the comparison results for the the 9-bus voltage stability estimation performance comparison between active learning and random sampling. From Fig. 2 it can be seen that active learning outperforms random sampling more than in the case of OSM prediction.

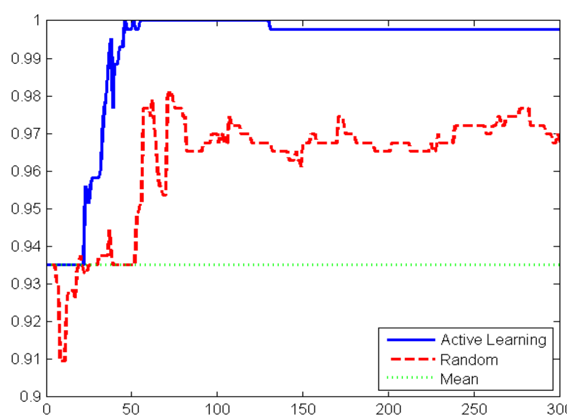

Figure 2. Comparison of active learning and random sampling on the 9-bus system for the voltage stability classification task using SVM

We hypothesize that this is due to the drastic difference between the sizes of the positive and negative classes. The difference in class sizes means that a greater variance may be expected when randomly sampling points because the addition of a few unstable OPs in $\boldsymbol{D}_{\mathrm{L}}$ may drastically change the decision boundary.

Next we will illustrate how the active learning approach performs on the 39-bus system oscillatory stability assessment using SVMs. From Fig. 3 the active learning approach significantly starts to outperform random sampling after 100 examples are labeled.

In Fig. 4 similarly to Fig. 2 the simpler task of voltage stability margin estimation results in a smaller but still significant performance gain from using active learning.

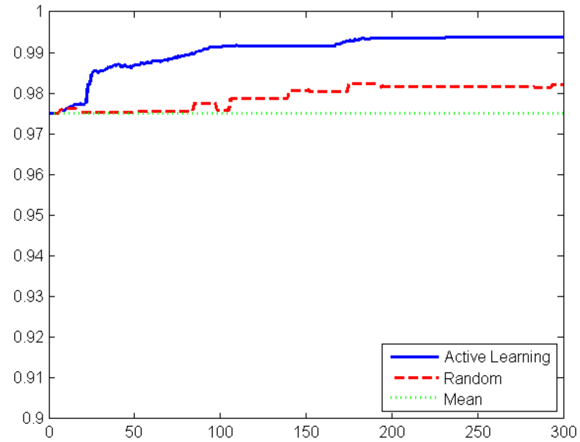

Figure 3. Comparison of active learning and random sampling on the 39-bus system for the oscilatory stability classification task using SVM

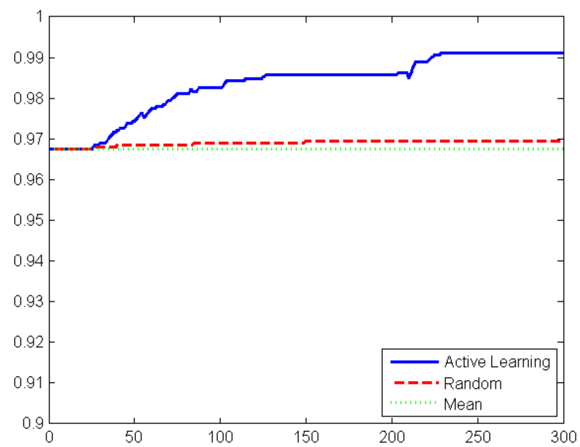

Figure 4. Comparison of active learning and random sampling on the 39-bus system for the voltage stability classification task using SVM

B. Artificial Neural Network Experiments

Unlike the SVM, in many cases the ANN using a logistic sigmoid transfer function may provide very confident predictions for data points dissimilar to those observed during training. Because of the imbalance of classes the four points used to initialize the active learning training will often of be in the positive, or stable, class. These two causes force the ANN to behave like a mean predictor, classifying the entire input space as the positive class with high confidence, until a negative example is included in $\boldsymbol{D}_{\mathrm{L}}$. To overcome this issue we included three positive and one negative point in the initialized $\boldsymbol{D}_{\mathrm{L}}$. In the resulting figures this is reflected as poor performance when very few examples are included in $\boldsymbol{D}_{\mathrm{L}}$. However, once enough points are included in DL the performance of ANN becomes closer to that of SVMs.

In Fig. 5 we compare active learning to random sampling and the mean predictor when using ANNs on the oscillatory stability task using 9-bus system data. From Fig. 5 the active learning provides significant improvement when few examples are observed. Interestingly, random sampling provides better results when using ANN than SVM on this task after 250 points are included in $\boldsymbol{D}_{\mathrm{L}}$.

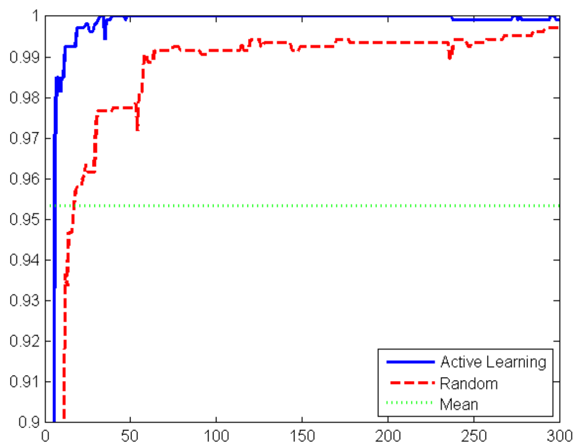

Figure 5. Comparison of active learning and random sampling on the 9-bus system for the oscillatory stability classification task using ANN

The next result, in Fig. 6, shows the accuracy comparisons of using ANNs on the voltage stability task for the 9-bus system data set. Again after many labeled examples are included in DL the performance of random sampling becomes close to that of active learning.

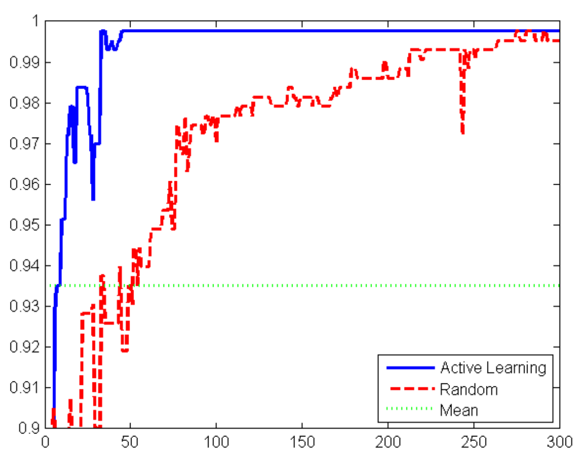

Figure 6. Comparison of active learning and random sampling on the 9-bus system for the voltage stability classification task using ANN

In Fig. 7 we show the 39-bus system oscillatory stability experiment results. Here random sampling struggles to become more accurate than the mean classifier even when 300 points are included in $\boldsymbol{D}_{\mathrm{L}}$. The ANN trained using active learning provides higher accuracy than random sampling in this case as well.

Finally, in Fig. 8 we show the results of ANN using active learning and random sampling on the 39-bus system voltage stability classification task. Although initially in this case random sampling outperforms active learning, after 20 examples are included in DL the active learning trained ANN starts to outperform random sampling. Again, random sampling struggles to outperform the mean predictor.

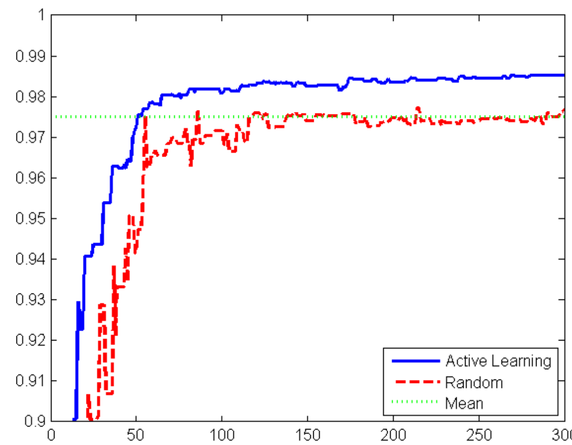

Figure 7. Comparison of active learning and random sampling on the 9-bus system for the voltage stability classification task using ANN

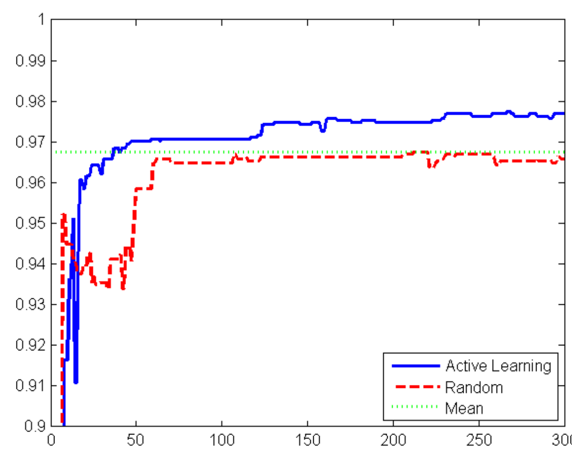

Figure 8. Comparison of active learning and random sampling on the 9-bus system for the voltage stability classification task using ANN

TABLE II
ACCURACY RESULTS ON OSCILLATORY STABILITY TASK

Data set	ANN		SVM	
	Active Learning	Random	Active Learning	Random
9-Bus	**99.9%**	**99.7%**	**100%**	**99.2%**
39-Bus	**98.5%**	**97.7%**	**99.4%**	**98.2%**

TABLE III
ACCURACY RESULTS ON VOLTAGE STABILITY TASK

Data set	ANN		SVM	
	Active Learning	Random	Active Learning	Random
9-Bus	**99.8%**	**99.5%**	**99.8%**	**96.8%**
39-Bus	**97.6%**	**96.6%**	**99.2%**	**96.9%**

In Table II we summarize the accuracy of predictors on the oscillatory stability tasks and in Table III we include accuracy on the voltage stability tasks after 300 points have been included in $\boldsymbol{D}_{\mathrm{L}}$.

CONCLUSION

This paper shows that significant improvement in accuracy can be obtained by using active learning to select a subset of data to learn from. In the case of an existing labeled data set the presented methodology can be used to filter out redundant data in order to reduce computational burden of training data mining tools.

Additionally, when only a set of power system OPs is available without the related stability indicators, and precise values of *DR* and VS_{margin} must be obtained through time domain simulation, the proposed method may be used to select which OPs to process in order to create the most accurate data set to learn from. This may significantly reduce the complexity involved in time domain simulation.

When comparing accuracy of trained classifiers, those using the proposed active learning approach to choosing training examples outperform those using random sampling in all experiments.

The performance improvement observed on more complex power system tasks is greater than on simpler tasks. The experiments also show that for simpler tasks the used ANNs are less sensitive to data set selection than SVMs. On more complex tasks higher accuracy can be obtained using SVMs.

We conclude that in the examined cases using active learning to pick which system OPs are simulated in the time domain, and afterwards used for training will lead to a more accurate classifier, decrease the computational complexity, or both.

REFERENCES

[1] C. Zheng, V. Malbasa, and M. Kezunovic, "Regression Tree for Stability Margin Prediction Using Synchrophasor Measurements," *IEEE Trans. Power Syst.*, Accepted, In Press.

[2] L. Wehenkel, T. V. Cutsem, and M. Ribbens-Pavella, "An artificial intelligence framework for on-line transient stability assessment of power systems," *IEEE Trans. Power Syst.*, Vol. 4 No. 2, pp 789-800, May 1989.

[3] T.V. Cutsem, L. Wehenkel, M. Pavella, B. Heilbronn, and M. Goubin, "Decision tree approaches to voltage security assessment," *Proc. Inst. Elect. Eng.*, Vol. 140, No. 3, pp 189-198, May 1993.

[4] I. Kamwa, S. R. Samantaray, and G. Joos, "Development of rule based classifiers for rapid stability assessment of wide-area post disturbance records," *IEEE Trans. Power Syst.*, Vol. 24, No. 1, pp. 258-270, Feb. 2009.

[5] C. Zheng, V. Malbasa, and M. Kezunovic, "A Fast Stability Assessment Scheme based on Classification and Regression Tree," IEEE PES International Conference on Power Systems Technology (POWERCON), Oct. 2012.

[6] D. D. Lewis, W. A. Gale, "A sequential algorithm for training text classifiers," Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 3-12, 1994.

[7] L. Shi, Y. Zhao, J. Tang, "Batch mode active learning for networked data," *ACM Transactions on Intelligent Systems and Technology*, Vol 3, No 2, pp. 33, 2012.

[8] N. Seliya, T.M. Khoshgoftaar, "Active learning with neural networks for intrusion detection," IEEE International Conference on Information Reuse and Integration, pp. 49-54, 2010.

[9] J. F. Bowring, J. M. Rehg, M. J. Harrold, "Active learning for automatic classification of software behavior," *ACM SIGSOFT Software Engineering Notes*, Vol. 29, No. 4, pp. 195-205, 2004.

[10] C. Mesterharm, M. J. Pazzani, "Active learning using on-line algorithms," Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 850-858, 2011

[11] W. Chu, M. Zinkevich, L. Li, A. Thomas, B. Tseng, "Unbiased online active learning in data streams," Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 195-203, 2011

[12] D. Cohn, Z. Ghahramani, M. I. Jordan, "Active learning with statistical models," *Journal of Artificial Intelligence Research*, Vol 4. pp. 129-145, 1996

[13] R. D. King, J Rowland, S. G. Oliver, M. Young, W. Aubrey, E. Byrne, M. Liakata, M. Markham, P. Pir, L. N. Soldatova, A. Sparkes, K. E. Whelan, A. Clare, "The automation of science. Science," Vol. 324, No. 5923, pp. 85-89, 2009.

[14] B. Settles, "Active learning literature survey," Computer Sciences Technical Report 1648, University of Wisconsin-Madison, 2009.

[15] C. Chang, and C. Lin, "LIBSVM: a library for support vector machines", *ACM Transactions on Intelligent Systems and Technology*, Vol. 2, No 3, pp. 27:1-27:27, 2011, Available at http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[16] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in Large Margin Classifiers,* Vol. 10, No. 3, pp. 61-74.

Vuk Malbasa (M'12) received his B.S. from PMF, University of Novi Sad, Serbia, in 2006, and his Ph.D. from Temple University, Philadelphia, PA, in 2012. He is now with Texas A&M University as a post doctoral researcher. His research interests include regualrized spatial models, intelligent sensor applications, graphical models, and cognitive communications in networking.

Ce Zheng (S'07) received his B.S. and M.S. degrees from North China Electric Power University, Beijing, China, in 2005 and 2007 respectively, all in electrical engineering. He is now with Texas A&M University pursuing his Ph.D. degree. His research interests include application of data mining techniques to power system monitoring and protection, power system stability analysis based on PMU measurements, and the impact analysis of grid integration of renewable energy.

Mladen Kezunovic (S'77-M'80–SM'85–F'99) received the Dipl. Ing. , M.S. and Ph.D. degrees in electrical engineering in 1974, 1977 and 1980, respectively. Currently, he is the Eugene E. Webb Professor, Director of the Smart Grid Center, Site Director of NSF I/UCRC "Power Engineering Research Center, PSerc", and Deputy Director of another NSF I/UCRC "Electrical Vehicles: Transportation and Electricity Convergence, EV-TEC". His main research interests are digital simulators and simulation methods for relay testing, as well as application of intelligent methods to power system monitoring, control, and protection. He has published over 450 papers, given over 100 seminars, invited lectures and short courses, and consulted for over 50 companies worldwide. He is the Principal of XpertPower™ Associates, a consulting firm specializing in power systems data analytics. Dr. Kezunovic is a Fellow of the IEEE, a member of CIGRE and Registered Professional Engineer in Texas.