WHITE PAPER

superna

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

The Impact of Machine Learning on Cyber Storage Solutions

Andrew MacKay

Chief Technology & Strategy Officer, Superna

OVERVIEW

In this project, we set out to apply a machine learning approach for predicting complex applications performance and assist with root cause analysis. This is a summary of an internal project, along with some best practices we learned in the process.

ABSTRACT

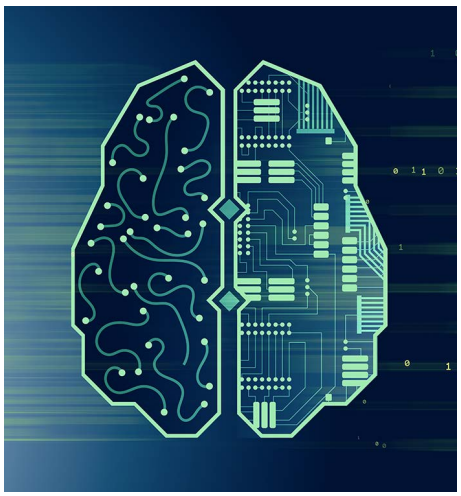
- We apply Machine Learning techniques to examine file data access patterns of users and applications.
- We explore how data access patterns can provide predictive analytics to assess root cause application performance before it impacts your application.
- We discuss how this can be applied to other data management challenges including emerging cyberthreats.

METADATA AND MACHINE LEARNING

The wealth of data generated by storage devices that support access auditing provides a continuous stream of data access patterns. This is perfect for machine learning because it's generated as file systems are being accessed and it provides a continuous summary of data transactions over time. A typical machine learning project will often utilize time series data.

Modern storage devices support auditing for compliance and security purposes and provide per-user, per-file transaction logs that include read, write, delete, rename, and ACL change, along with timestamps. This data can be used to build a data access behavior that functions as a digital "fingerprint" or unique identifier.

MACHINE LEARNING: THE TRAINING PHASE



This is the process of building model files with a specific ML algorithm. This learning process creates model files that can be used in prediction or anomaly detection. In this instance, time series data is generated by Superna Performance Auditor, and can be used directly in a training data set. This is a key step in any Machine Learning project as the data itself needs to be prepared and validated to ensure that it's representative of the problem statement for the ML project. Additional data preparation steps may be required to remove certain types of data that might skew the training process. This detail is beyond the scope of this paper.

Many different Machine Learning algorithms and libraries exist from which to choose. A key phase of the project was evaluating the statistical nature of the input data and selecting from various algorithms that are tuned for different use cases. This is a key decision point in any ML project; this choice is often iterated several times based on testing.

In this project, the goal was to predict application performance degradation and identify the root cause application hosts that contributed to performance degradation. In other words, which host(s) shows signs that it will negatively impact application performance. If administrators had a predictive root cause *before* actual performance was impacted, they could proactively address the root cause to mitigate the negative impact..

The application performance scenario that's explored in this project is based on a multi-host application that uses an SMB or NFS file system to process data. Each host had a different IO profile unique to the host. Each host accessed common data on an SMB share. Superna Performance Auditor was installed and Dell PowerScale file system auditing was enabled.

DATA NORMALIZATION PRIOR TO TRAINING

Testing was required to generate training data that was representative of the normal performance characteristics of the application. Superna Performance Auditor uses audit data to monitor application performance. The solution creates real-time summarized data that captures all "views" of performance from various perspectives. This includes a view of performance by host, by path in the file system, by application server, by subnet, and by Storage device nodes. All "views" are combined into a summary of reads, writes, IOPS, and are time-stamped into a single record. This allows training models to train on any of the 5 views of performance. By testing different training approaches and data views, we are able to land on the best option for a given scenario.

TRAINING MODULES

By entering training data into a Machine Learning framework, we are able to create training model files. In this project, the training data records are stored in Apache Kafka Topics within Virtual Machines hosting Superna Performance Auditor. The Kafka platform is a good choice for streaming data into or out of topics, to create a machine learning data pipeline. The structure of the data was also ideal in that it contained 5 different timestamp-aligned views of performance data that could all be used to create training models. This data format allows 5 different time series datasets of the same application IO profile.

PICKING A TRAINING MODEL

Picking the best model for your project can require trial and error to validate the fit for your problem statement. It's always a good idea to try multiple models and compare the results before committing to a final choice. Since application performance that's considered "normal" can span a long period of time, a training model is needed that can accommodate this type of learning. In this project, the Long Short Term Memory model was selected (LSTM). This is a deep learning model that can accommodate patterns that span long periods of time. More information on this recurrent neural network model is available here.

THE PROBLEM STATEMENT

Every machine learning project needs a clear problem statement before even starting any modeling or learning steps. The problem statement above was targeting a prediction of performance degradation before it occurred. Another common objective is anomaly detection, which predicts an anomaly that is inconsistent with training models that were trained on "normal" host performance. In this project, training models were built on each of the 3 hosts, resulting in 3 separate host models, since each host performed different tasks. A 4th training model used the authenticated user or service account used by the application, to provide an additional input to the learning.

MACHINE LEARNING PROJECT BEST PRACTICES

Any successful Machine Learning project needs team members with diverse skill sets, and familiarity with the project problem statement. Below is a summary of the team functions that are needed.

ML Visionary. This role should encompass:

- The business value and technical fit of solving the problem with machine learning. Note that not *all* problems are ideal candidates for Machine Learning.
- A system-level understanding of all components involved in the problem statement solution. This would include example applications, hosts, network devices, storage devices, performance attributes of an application, etc.
- The customer weight of this problem versus other capabilities
- An understanding of Machine Learning processes, tools, data validation, normalization, results validation aligned to the project objective, i.e., did the prediction make sense? Would acting on these results address the predicted performance problem?
- Problem statement definition, including how the model will need to train, how the model needs to be validated, and how the results are sanitized.

ML Data Science and Algorithm Expert. This role requires skills with the various tools, and the ability to configure and set-up data pipelines and result validations on training and prediction testing.

- Validate the results as useful, validate the results of one algorithm against a different algorithm.
- A background in Machine Learning algorithm selection processes for mapping algorithms to a specific objective.
- Suggest algorithms and compare/contrast results of training models and predictions against business objectives.

Systems Expert Role. This role is someone that has hands-on experience to development tools needed to execute machine learning model training and predictions

- Hands-on to each component, with the necessary skills for creating application IO profiles. Without real-world training data, the Machine Learning process is basically “garbage in, garbage out”, and projects can easily fail at this step. This is a critical skill, and is required to help ensure that no time is wasted training on bad system-level data.
- The ability to execute training sessions, code pipeline tools to automate the training and prediction steps, and validating the results as usable.
- Interacting with the tools, systems, and iterations required to bring a project to a successful completion.

PUTTING IT ALL TOGETHER

The accompanying diagram below shows data flows through the system and the various tools of the Machine Learning pipeline.

- At the top are the Kafka Topics that contain the raw records captured by Superna Performance Auditor.
- The second step consumes records from Kafka for each host and runs 3 parallel LSTM autoencoder training models.
- This produces a Machine Learning Training Model file for each host. These files are used in the anomaly detection and prediction step.
- To allow a data pipeline of processing, a new Kafka Topic was used to store the prediction algorithms results.
- The next step also required testing to use the training data and resultant model files in an algorithm tuned for the problem statement. This is where data science comes into play and expertise on which algorithm to choose and validation

of the results is perhaps the single most important step in any machine learning project. If done incorrectly, the predictions could simply be wrong... "garbage in, garbage out."

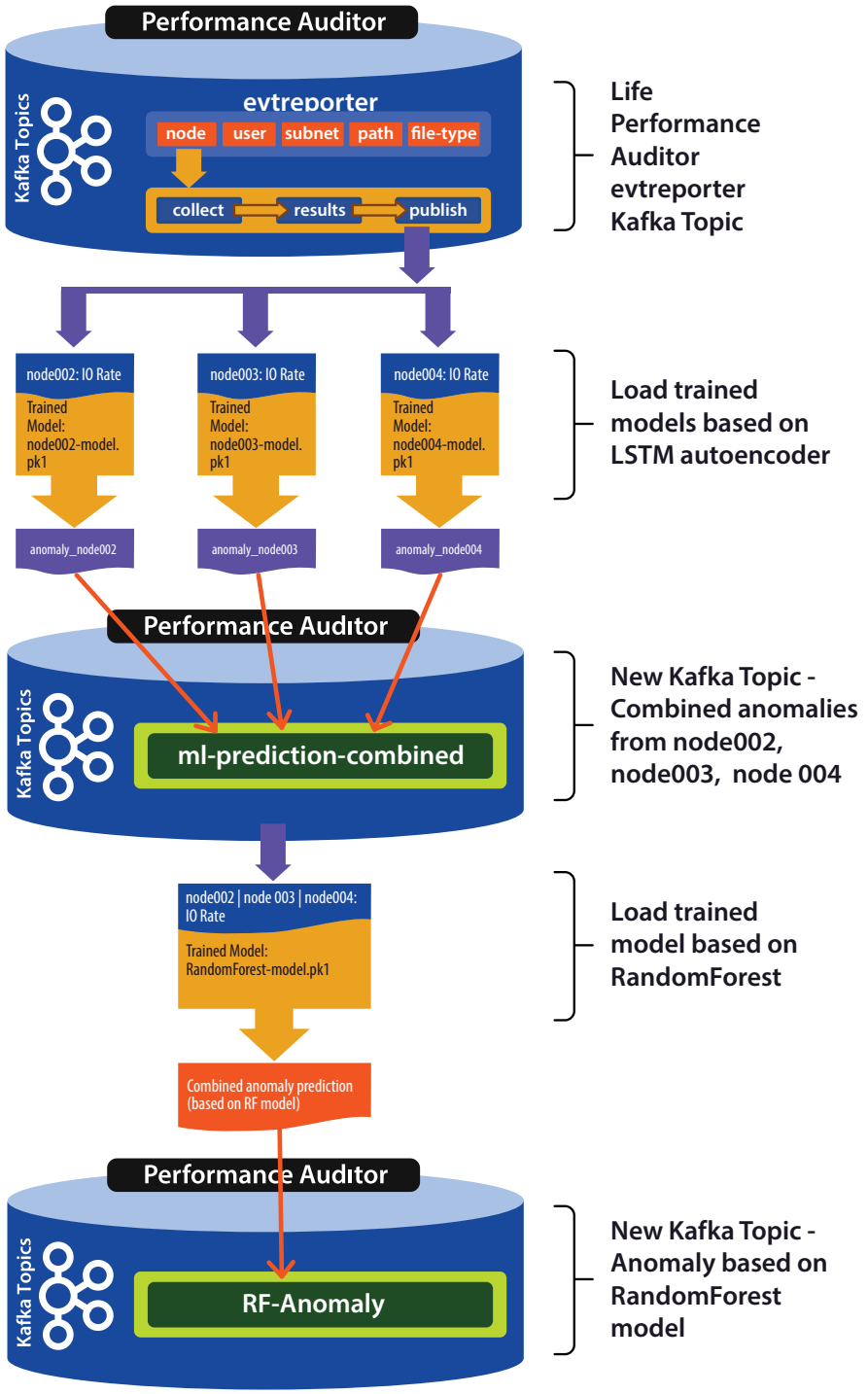
- Performance anomaly prediction validation. Once all the training and validation are completed, using the model in a prediction requires showing the model different "live" data sets to validate whether or not the prediction algorithm can make usable predictions along with the root cause. The results in this phase may require new iterations focused on re-training the models or selecting a different analysis algorithm.

For more insight, watch this deep dive video on Superna's Machine Learning framework which demonstrates our real-time Machine Learning prototype connected to Superna Performance Auditor. It covers multiple performance degradation scenarios that were detected by Performance Auditor, along with the root cause for each

CONCLUSION

This paper showcases how Superna's next generation technology augmented by Machine Learning can help solve complex data management problems at scale.

Cyber storage products will need to harness the power of Machine Learning to help combat unknown attack vectors, as well as automating the creation of protection policies based on data access patterns and workflows.



A follow-on paper will cover the use of Machine Learning to accelerate and simplify security at the storage layer.

For more insight into how Superna® can help solve your organization's unstructured data security challenges, visit us at superna.io.