Introduction to Machine Learning for TRITON AP-DATA

Machine Learning | TRITON AP-DATA | v8.3.x | 15-Dec-2016

Machine learning is a branch of artificial intelligence, comprising algorithms and techniques that allow computers to learn from examples instead of pre-defined rules. As a user of Forcepoint™ TRITON® AP-DATA, you can provide examples that train the machine learning system to help protect your organization's information. After training, the system creates a classifier that classifies documents based on how similar they are to your examples.

Machine learning offers advantages and disadvantages compared with other Forcepoint data classification methods. It is important to assess whether machine learning is the best solution for your particular circumstances. This article offers a general introduction and looks at the types of data that can be effectively protected using machine learning.

For more information on how to use machine learning, see the following:

- Data Security Manager Help
- Using Machine Learning for Optimal Data Loss Prevention (video)

Machine learning basics

Machine Learning | TRITON AP-DATA | v8.3.x | 15-Dec-2016

There are two main types of machine learning algorithms:

- **Supervised learning algorithms**

  The algorithms are given labeled examples for the various types of data that need to be learned.
- **Unsupervised learning algorithms**

  Data is unlabeled and the algorithms attempt to find patterns within the data or to cluster the data into groups or sets.

Forcepoint machine learning uses both types of algorithms.

Knowing when to use machine learning

Machine Learning | TRITON AP-DATA | v8.3.x | 15-Dec-2016

Forcepoint Machine Learning, like any other decision systems that need to handle complicated data, may generate “false positives” (unintended matches) and “false negatives” (undetected matches). The total fraction of false positives and false negatives is sometimes referred to as the “accuracy” of the system.

Since the accuracy of machine learning is derived from the properties of the data and finding the best data sets can sometimes be challenging, you may want to first determine if other types of classifiers, such as fingerprinting or pre-defined policies, can help you classify and protect your data – before considering using machine learning.

A use case in which machine learning could be effective is if you need to differentiate between proprietary and non-proprietary data, like you might find in source code. It may be hard to fingerprint source code that is under constant development and continually changing, and pre-defined policies cannot distinguish between proprietary and non-proprietary source code.

Forcepoint provides several pre-defined content types that address some common use cases, including source code (in C, C++, Java, Perl, and F#), patents, software design documents, and documents related to financial investments. If you need to protect content that belongs to these content types, consider using machine learning, and select the content type that is pre-defined by the Forcepoint system. Machine learning can also be used to complement and enhance fingerprinting and predefined policies and other TRITON AP-DATA detection and classification methods.

How Forcepoint machine learning works

Supervised machine learning for data protection requires, in general, two types of examples: content that needs to be protected and counterexamples. The former is usually referred to as “positive” and the latter as “negative.” Counterexamples are documents that are thematically related to the positive set yet are not meant to be protected, such as public patents versus drafts of patent applications, or non-proprietary source code versus proprietary source code.

However, since it can be difficult and quite labor intensive to find a sufficient number of documents for the negative set (which includes ensuring that no positive examples are inside this set), Forcepoint has developed methods that allow the system to use a generic ensemble of documents as counterexamples to the positive set. (See *Negative examples consisting of “All documents” examples*, page 4 and *Positive examples*, page 4).

For text-based data, some of the algorithms automatically create an optimal “weighted dictionary” that assigns positive weights to terms and phrases that are more likely to be included in the positive set and negative weights to terms and phrases that are more likely to be included in the negative set. The algorithms also find an optimal threshold. When the weighted sum of the terms that are found in a given document is greater than that threshold, the algorithm decides that the document belongs to the positive set. The assumption is that positive examples are more likely to have common themes.

Most machine learning algorithms are designed to be used with several hundred or several thousand positive and negative examples and require “clean” data, or data that is correctly labeled. Forcepoint machine learning, however, utilizes different algorithms for different data sizes and attempts to automatically match the type of algorithm to the size of the data.

In addition, Forcepoint machine learning algorithms can detect “outliers” among a set of positive examples. These are examples that should probably not be labeled “positive.” Forcepoint algorithms also allow learning to take place even when negative examples are not provided.

Selecting examples for training

Positive examples

For effective machine learning to occur, it is most important to select the best positive examples. These are textual examples for the data that you want to protect. The documents in this set should be related to a certain theme or share some other commonalities – otherwise the learning algorithm will not be able to find a way to categorize the data.

The required number of examples depends on the level of commonality. If the positive examples share many common terms that are very rare, in general, a small number suffices. On the other hand, if the differences between the positive and the negative set are more subtle, more examples will be required. A positive set typically consists of 100-200 textual documents.

Negative examples

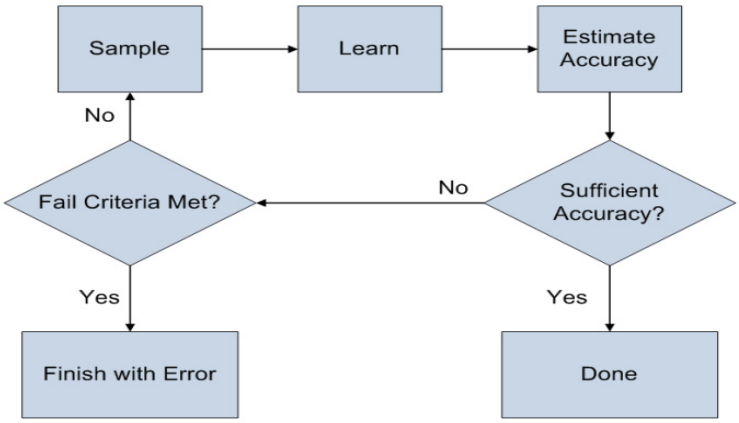
Negative examples refer to samples of data that are semantically or thematically similar to the set of positive samples but that should not be protected, such as public patents versus drafts of patent applications, or non- proprietary source code versus proprietary source code. The size of this set of negative examples can be similar to the size of the positive set, although a larger set is preferable.

Negative examples consisting of “All documents” examples

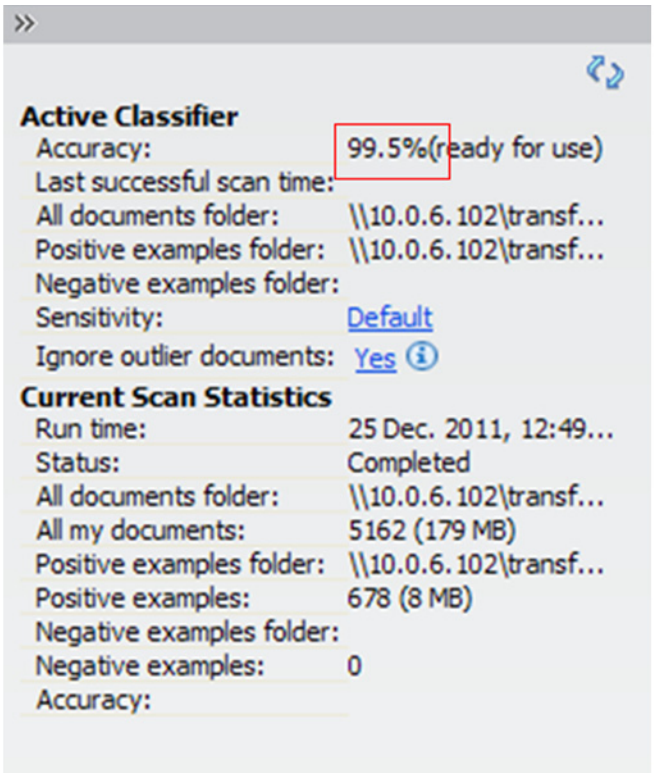
To create a generic ensemble of documents that Forcepoint machine learning can use as negative examples or counterexamples to the positive set, select the path to a large folder with a representative sample of documents from your organization. This folder can contain both positive and negative examples, but the underlying assumption is that substantially more negative examples exist. The size of this set of counterexamples can be similar to the size of the positive set, although a larger set is recommended.

What happens during training

After submitting your examples, the crawler starts going over the files and providing them to the learning algorithms. If the number of files in a folder is very large, a sampling algorithm samples the folder several times and checks for convergence:



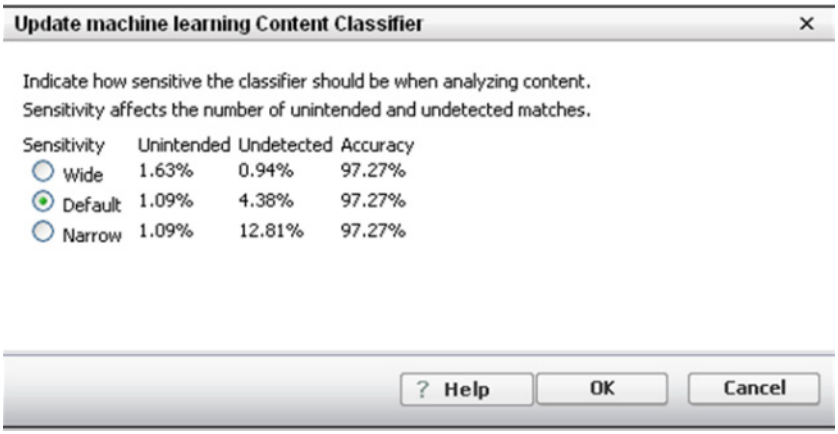
If learning is successful (i.e., the data is "learnable"), the following window appears:



By default, the sensitivity level is set to“Default” (an optimal trade-off between false positives (unintended matches) and false negatives (undetected matches)). The training is performed, by default, ignoring outliers, or examples that could be labeled “positive,” but that don’t seem to belong to the positive set.

You may choose not to ignore the outliers by clicking on “Yes” and changing that to “No.”

You may also change the sensitivity level by clicking on the “Default” link, which brings you to this window:



Note that it is important to consider the percentage of unintended and undetected matches before deciding about the sensitivity level. For example, choosing the “Narrow” level in the window shown above will only increase the expected level of undetected matches, without reducing the expected level of unintended matches, and is, therefore, highly undesirable.

Accuracy of machine learning

Machine Learning | TRITON AP-DATA | v8.3.x | 15-Dec-2016

The ability of the system to accurately classify data depends to a large extent on the examples that you provide. If the system fails to find enough common elements, the results from machine learning may not be accurate. Should this happen, the system performs another stage of validation to assess the level of false positives (unintended matches) and false negatives (undetected matches) on new data that is not used during the training phase, sometimes referred to as “zero-day documents.”

If the “recall” level of the classifier (i.e., the total number of “true positives” divided by the sum of false positives and false negatives in the new data) is below 70 percent,

the system returns a FAIL message that includes the likely reason the attempt to accurately classify data failed. Examples of these error messages follow:

Error Code	Error Message
DSCV_ERR_-420_CODE	There are not enough examples in your positive examples folder. X were provided and at least Y are required. Please add more examples then restart the machine learning process.
DSCV_ERR_-421_CODE	There are not enough examples in your negative examples folder. X were provided and at least Y are required. Please add more examples then restart the machine learning process.
DSCV_ERR_-422_CODE	The files in your positive examples folder don't contain enough text. Of X files provided, only Y have enough text. At least Z are required. Please update the files or point to another folder, then restart the machine learning process.
DSCV_ERR_-423_CODE	The files in your negative examples folder don't contain enough text. Of X files provided, only Y have enough text. At least Z are required. Please update the files or point to another folder, then restart the machine learning process.
DSCV_ERR_-424_CODE	Your positive and negative examples are too similar. No significant difference in words distribution was found. Please provide new examples.
DSCV_ERR_-425_CODE	Your positive and negative examples are too similar, or your positive examples may not be consistent enough to draw conclusions. There were bad error rates on both training X and validation Y. Use different example folders in the classifier.
DSCV_ERR_-426_CODE	The examples you provided were not sufficient for accurate training. Though the accuracy of the training set is good X, the machine learning process cannot make accurate conclusions on unseen data X. Your positive examples may not be homogeneous enough. Please provide more consistent examples then restart the machine learning process.
DSCV_ERR_-427_CODE	Your examples don't fit the content type you specified. You provided X positive examples, but only {2} of them fit the type.
DSCV_ERR_-428_CODE	The files in your example folders don't contain enough meaningful text (only X words). Please add files with more meaningful content or point to other folders, then restart the machine learning process.
DSCV_ERR_-429_CODE	More than one file in your examples folders doesn't contain enough text (only X words). Please update the files or point to other folders, then restart the machine learning process.

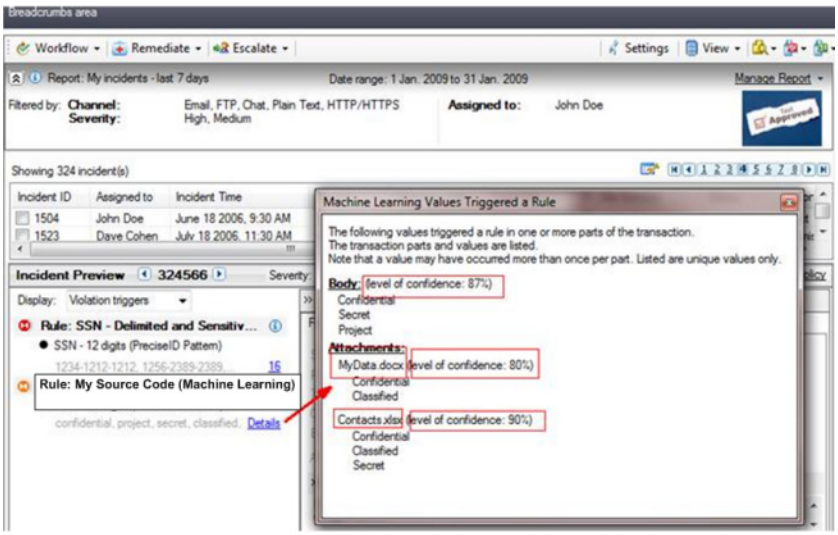
By adjusting the sensitivity level of the classifier, you can reduce the number of false negatives (unintended matches) while accepting a higher level of false positives (undetected matches) or accept some false negatives to reduce the rate of false

positives (or find an acceptable balance in between). Factors influencing your choice include the level of commonality in your positive set of examples (a low level tends to decrease accuracy); the business implications of false positives; and the resources that you have available to deal with false positives.

Using the classifier

Machine Learning | TRITON AP-DATA | v8.3.x | 15-Dec-2016

After successful training, the machine learning classifier can be used to create rules and policies. An incident that resulted from a match with a classifier might look like this:



Tuning the classifiers

Machine Learning | TRITON AP-DATA | v8.3.x | 15-Dec-2016

In some cases, you may wish to tune the classifiers. For example, if too many false positives occur, start by setting the sensitivity level to “Narrow.” You can also combine the classifier with other classifiers, such as looking at certain file-types like both Microsoft Office files with PDF files.

If the overall accuracy level is too low, check to see if all your positive examples are related to the same subject. If you have a small number of subjects and enough samples for each of them, you can create a different classifier for each subject by assigning a folder to each subject, locating documents that are related to the subject in the corresponding folder, and then training the system separately on each folder.

In many cases, several small specific classifiers can provide better accuracy than one general classifier.

Comparison with other types of classifiers

The following table summarizes the advantages and disadvantages of the various classifier types:

	Machine Learning	**Fingerprint-ing**	**Pre-Defined Policies**	**User-Defined Dictionaries and Regular Expressions**
Coverage	High: Covers any document with semantic similarities to the learned data	Medium: Detects only derivatives of fingerprinted documents	Limited to the existing pre-defined types	Unlimited, providing that the user has properly defined the dictionaries and the regular expressions
Accuracy	Depends on the data	Very High	High for data types that are common enough	Medium
"Zero-Day" Protection	High	Very Low	High	High
Size/Footprint	Medium	High	Low	Low
Deployment and Configuration Effort	Medium (may require some tuning)	Medium	Low	High - requires careful setting and tuning

For more information on how to use machine learning, see the following:

- Data Security Manager Help
- Using Machine Learning for Optimal Data Loss Prevention (video)