VoI for complex AI based solutions in coalition environments

Dinesh Verma[a], Geeth de Mel[b], and Gavin Pearson[c]

[a]IBM T. J. Watson Research Center, PO Box 218, Yorktown Heights, NY 10598, U.S.A.
[b]IBM Research UK, Hartree Center, Warrington WA4 4AD, UK
[c]Defence Science & Technology Laboratory, Ministry of Defence, Porton Down, Salisbury SP4 0JQ, U.K

ABSTRACT

Real-life AI based solutions usually consist of a complex chain of processing elements, which may include a mixture of machine learning based approaches and traditional programmed knowledge. The solution uses this chain of processing elements to convert input information into an output decision. When information is provided for a specific solution, the impact of the information on the decision can be measured quantitatively as a Value of Information (VoI) metric. In prior work, we have considered how the VoI metric can be defined for a single AI-based processing element. To be useful in real-life solution instances, the VoI metric needs to be enhanced to handle a complex chain of processors, and be extended to AI-based solutions, as well as supporting elements that may not necessarily use AI. In this paper, we propose a definition of VoI that can be used across AI-based processing, as well as non AI based processing, and show how the construct can be used to analyze and understand the impact of a piece of information on a chain of processing elements.

keywordsAI, Coalition Operations, Multi Domain Operations, Value of Information, Quality of Information

1. INTRODUCTION

Artificial Intelligence (AI) is an important aspect of multi-domain operations: multi-domain operations[1,2] and AI based operations are driven by means of data. This data may manifest itself as training data or as data used for inference and decision. In the context of multi-domain operations, this data is shared across many different domains. In the context of coalition operations, information is frequently exchanged among different partners. Such information may be very useful in the decision making tasks encountered by any single partner. However, all information is not equally useful, and we need a mechanism to understand what the value of a piece of received information is.

Due to the massive amount of data that is generated by a variety of sensors, it is not feasible, nor would it be efficient if it were feasible, to store all of the data that is being generated, and even more difficult to transport and use all of the data that is being generated in real-time. Having techniques which allow the association of a value metric with a piece of information enables filtering of the information so that the most valuable information is stored, transported or processed, and less valuable information is discarded. Similarly, when filtering the information to determine which needs to be processed immediately, and what can be deferred, having a value metric is useful.

A definition of Value of Information (VoI) has been proposed for the context of machine learning in coalition operations, but VoI was shown only for a single step of decision making.[3] Practical applications tend to be a complex network of several steps, so it is important to extend that definition to the case of networks of components. We discuss that extension in this paper.

In addition to VoI, Information can also be associated with a Quality metric (Quality of Information or QoI).[4] In this paper, our primary focus is on VoI, and we assume that all the information that is available for decision making is of good quality and can be trusted. When using information, in the context of any real applications, both QoI and VoI need to be considered, and the results of this paper ought to be used with those of QoI.[4]

Further author information: (Send correspondence to Dinesh Verma)
Dinesh Verma: E-mail: dverma@us.ibm.com

We begin this paper with an abstract model to characterize complex decision making. Then, we briefly recap the definition of VoI, which is followed by a discussion on how the VoI concept can be applied to the complex decision making step. Finally, we discuss a few sample application use-cases for VoI.

2. STRUCTURE OF A COMPLEX AI BASED SOLUTION

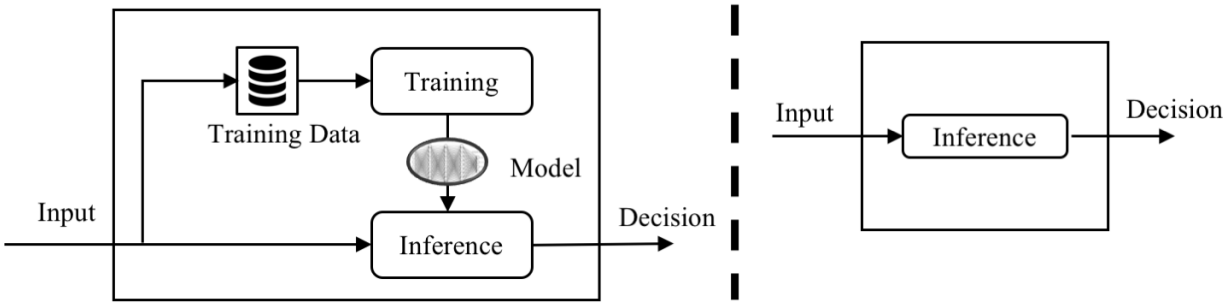

Figure 1. Generic Structure of Solution Components. We assume that each component has an input and an output. The standard component shown on the left maps an input into a decision. The AI based components performs this mapping using an AI model, which may have been derived using machine learning on some training data. The arrow from the input to the training data shows that part of the input can also be used to train/retrain the model.

An AI or ML (machine learning) based solution consists of many components, each of which may be an AI/ML based component, or a standard software component. The assumption we make about any component is that it maps an input to an output decision. The standard software component would perform this mapping using a software algorithm, or by looking up a table of input-output pairs, or any other approach which is programmed in. The AI/ML based component performs the same task, but it does so using an AI model. The AI model could be defined manually (e.g. using a set of rules in an expert system), or it could have been developed using a learning process that extracts the model from a set of training data. In some systems, the training data may be collected by sampling the input system itself. In other systems, the training data may be collected from some other sources. The structure is shown in Figure 1

Table 1. Typical Decisions for some common categories of AI/ML tasks

Task	Output
Classification	The class label
Mapping	An output class (transformed from input)
Regression	A function description
Clustering	A set of groups and parameters
Anomaly Detection	A boolean (normal or not)
Optimization	A parameter value

It is worth noting that the input and output can be fairly complex. An input signal may be an image, an audio signal, a text document, a relational database, any software object, a structured XML document, etc. The output decision itself could be a number, a text document, a single string, a complex array etc. In some systems, there could be multiple inputs and multiple outputs. However, for the purpose of this paper, we can consider each component to have a single input and a single output, without loss of generality.

In many cases, an AI enabled component and an non-AI enabled component can provide different implementations of the same function. As a black box, they will be indistinguishable. The difference will be in the non-functional aspects of the component, how long it takes to develop the component, how flexible and adaptable that component would be, and how easy/difficult it would be to update that component. AI/ML converts the problem of creating the component into a problem of defining an AI model instead of developing a software component. The former can be done by a model builder, instead of a software engineer. Machine Learning converts the problem of creating a component to the task of finding and providing a good training data set for the system. Instead of an AI model developer, a data engineer can provide the training data, and create a component. AI/ML make the task of creating the software components easier, which has many associated benefits in deployment and life-cycle management.

Table 1 shows how the model above describes the typical category of applications that are usually associated with AI/ML based capability.

To become more specific than the general categories, we also enumerate some of the common AI/ML based tasks and show how they correspond to the model described in Figure 1.

Table 2. Some Examples of AI/ML Components

Component	Input	Output Decision	Category
Speech to text	An Audio Signal	A set of words	Classification
Text to Speech	A set of words	An Audio Signal	Mapping
Face Recognition	An Image	A name	Classification
Network Intrusion Detection	Network traffic	Normal/Under Attack Indication	Anomaly Detection

Both Table 1 and Table 2 are illustrative in nature and are provided to demonstrate that the abstract representation of an AI/ML based component shown in Figure 1 is applicable widely, and are not intended as a comprehensive set of all categories and tasks associated with AI/ML.

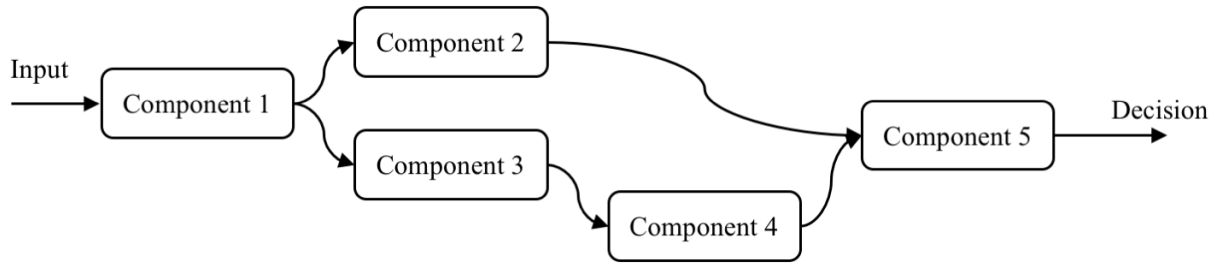

Figure 2. The components that make up a complex AI Solution. Any component may be either an AI/ML enabled component or a component developed using programmed software or a table lookup.

We model the general solution as a graph of many individual components. The input and output of the individual components are linked together, as shown in Figure 2. As in the abstract representation of a single component, we assume that there is a single input and a single output. Many of these components can be linked together into an overall system. The output of some component can feed into the other components, with the last component in the graph producing the final decision. For some of the internal components, we show multiple inputs or multiple outputs to reflect the fact that the graph can be composed in many different ways.

This structure of the solution is found in many real world applications. As an illustrative example, we are showing the representative console of a system used to detect IoT devices using AI/ML based components from

network traffic, as described in .[5] The console of the real-word application matches the abstract model specified in Figure 2 reasonably well. Therefore, the approach described in the figure would be a reasonable approach to address the requirements of a broad class of applications.

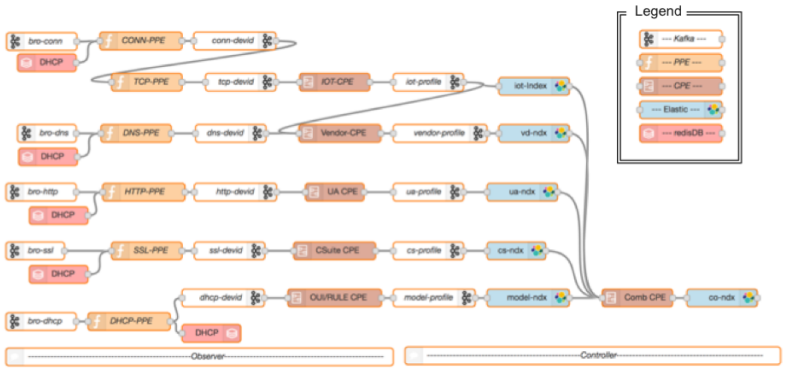

Figure 3. The components of an AI/ML based solution used for Traffic Analysis.

3. A DEFINITION OF VOI

A definition of Value of Information (VoI) that conforms to the NATO Code of Best Practice for Command and Control Assessment[6] for assessing measures of merit has been proposed in previous work.[3] This definition puts VoI as a measure of command and control effectivenes (MoCE) in the framework. To provide a brief recap of the NATO Framework, it classifies different measures of metrics into:

- **Measures of Policy Effectiveness** (*MoPE*), which focus on policy and societal outcomes;
- **Measures of Force Effectiveness** (*MoFE*), which focus on how a force performs its mission or the degree to which it meets its objectives;
- **Measures of C2 Effectiveness** (*MoCE*), which focus on the impact of C2 systems within the operational context;
- **Measures of Performance** (*Mo*P), which focus on internal system structure, characteristics and behaviour; and
- **Dimensional Parameters** (*DP*), which focus on the properties or characteristics inherent in the physical command and control systems.

For a component as described in Figure 1, the formulation of VoI [3] considers input as a point in an information space (***IS***), and the output of a component as a point in a decision space (***DS***). The information space covers all the possible inputs that may come into a component, and may include time as one of the attributes. The formulation assume that a distance metric Δ is defined over the information space, so that $\Delta(f,g)$ measures the distance between any two pieces of information. Similarly, a distance metric δ is defined in the decision space to measure the distance between two decisions.

Each component provides a mapping from the information space to the decision space. Denoting the decision corresponding to existing information I as $a(I)$, the change in decision for new information J for an analysis task denoted as A which is already using a given information I, would be given by

$$Change(J \mid I,A) = \delta(a(J+I), a(I))$$

If we assume that a goodness metric $g(D)$ is associated with any decision D, where the goodness metric maps any decision to a numeric value, then we can define the VoI of new information J when the current information is I as:

$$VoI(J \mid I,A)=g(a(J+I))-g(a(I))$$

For any machine learning component, there are two types of input information that have an impact on the quality of its performance. Looking at the abstract model shown in Figure 1, one of the input information pieces is the training data, and the other is the input used to make inferences.

In order to keep our analysis simple, we assume that all the inputs and outputs in the machine learning system are continuous functions that vary in range from 0 to 1. Furthermore, we also assume that the goodness metric is a numeric value that varies from 0 to 1, with 1 being perfectly good and 0 being perfectly bad for the decision being considered. This assumption allows us to study and examine the VoI in a simple manner. Moreover, the analysis can be extended for other environments by defining a mapping function from the corresponding input or decision to a space between 0 and 1.

3.1 VoI for input of the ML Component

The VoI of the input component can be described by its impact on the output decision. If we associate a goodness metric with the output decision, then the value of information of the component would be defined by the change in the goodness metric given a unit change in the input. Suppose an input value of x produces and output decision y, which is associated with a goodness of $g(y)$, and an new value of $x+\delta x$ results in the new decision value of $y+\delta y$ with the associated goodness metric of $g(y+\delta y)$. The VoI of an incremental piece of information given an input x would then be defined by the derivative of the goodness function around x, which can be defined to be $(dg/dy)(dy/dx)$.

If we consider many of the cascaded chains of a complex solution, e.g. a chain as shown in Figure 2, the same applies to the final decision outputted by the chain. If the final decision of the chain is y, and the intermediate stages are y_1, y_2, y_3, then the chain rule of derivatives will apply to the entire chain.

3.2 VoI for ML training data

The other input provided to a ML component is its training data. The training data, followed by the training process, results in the determination of the function that maps the input x to the output y. The VoI of a piece of training data is the difference that is caused in the function as a result of incorporating additional training data.

As an example, consider the machine learning task which is involved in the classification problem of machine learning. In the classification problem, the training data consists of several records, each record consisting of one or more feature vectors and an output category. The resulting model is a system which would take a record containing the features only (without the output category) and map it to one of the categories used for the training process. The resulting model can be viewed as a mapping from the different regions in a feature space (where each feature is one dimension in the virtual space) to one or more of the categories. The value of an incremental piece of training data is the difference in the mapping space that results due to the addition of the new piece of training data.

Another high-value application area for machine learning is the estimation of relationship among several input parameters (features) and one or more output parameters. The task of the machine learning process is to build the best estimate for the function which predicts the output parameter from the input parameters. In such problems, the parameters can usually be considered to be numeric. Thus, the machine learning process over some N input parameters $(x_1,\ldots x_N)$ results in a function which predicts an output parameter $y=f(x_1,\ldots x_N)$.

Suppose f_0 is the resulting estimated function resulting from the original training data, and f_i is the resulting estimated function resulting after adding the incremental training data. The $\int(f_i-f_0)$ can be viewed as the change in VoI, where the integral is taken over the entire space defined by the different parameters.

3.3 VoI for the Complex Solution

In a complex solution, an input may process through many stages of intermediary processing steps before it produces a decision. The VoI of any piece of the input ought to be measured by means of the final output it produces.

In effect, we would be mapping the complex chain show in Figure 3, and consider a black box equivalence of the solution with the complex chain reduced to the equivalent output. The VoI for the input of the ML component will be considered as the impact on the output decision. If each of the components in the chain can be viewed as a function whose derivative exists, then the net VoI can be derived using the chain rule of derivatives.

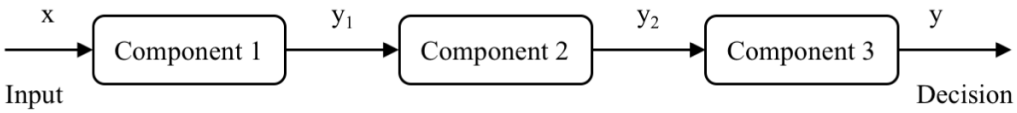

Figure 4. VoI Composition for a chain consisting of three components

As a specific example, if we consider the three chains (each with a single input) resulting in the final decision, as shown in Figure 4, then the net VoI of the complex procedure can be viewed as $\frac{dy}{dx}$, which can be computed as $\frac{dy}{dy_2} \cdot \frac{dy_2}{dy_1} \cdot \frac{dy_1}{dx}$.

When the chain is more complex, with branches, the partial derivatives need to be calculated in order to determine the net VoI.

4. USE-CASES FOR VOI

The concept of VoI can be used in many different ways. In this section, we look at the different use-cases of VoI.

4.1 Policy Generation for Sharing Training Data

The quantification of VoI is useful for determining whether or not to accept the information offered by a coalition partner. The decision would depend on the mixture of QoI and VoI of the offered information. These can be used to automatically generate policies for acceptance or rejection of data provided across domains.

In general, data may be obtained across domains for training purposes in a machine learning context. When the QoI and VoI are both high, it would make sense to use the information and thus set the policy as accept the information. If the VoI is low, then the cost of a policy of rejection is also low; and it may be deemed safer to set the policy to be reject due to less than perfect trust in partner provided information, or data that comes from a different domain. If the VoI is high, but the quality is low, it may again be best to reject the data: indeed it has been shown that for several applications (e.g. tracking) using low quality data can reduce the overall confidence in the decision (e.g. resultant track). However, depending on the context of operation, the data may be accepted in some cases.

4.2 Size of Training Data

The VoI of training data available can be used to determine whether or not one has sufficient training data for a machine learning purpose. The effectiveness of training process depends significantly on the amount of training data that is available. However, acquisition of training data remains the most time-consuming process in creating machine learning based solutions for specialized domains, and is a major impediment in creating AI enabled solutions for special industrial or military domains.

In general, having a large but varying and domain-specific training data set is more beneficial for machine learning applications. The improvement in accuracy of the training data can be captured as learning curves, and theoretical estimations of the learning curves are available when statistical assumptions about the type of data being seen can be made. However, in real-life, data rarely conforms to any stable statistical properties. Thus, it is still something of an art to determine whether one has obtained sufficient training data.

A quantitative estimation of VoI provides a way to determine whether one has obtained sufficient data in order to train the model, making data acquisition more of a science than an art. For each bundle of additional training data that is obtained, one can estimate the VoI by seeing how much the resulting function is changing. When the change is not significant, one may determine that sufficient training data has been obtained.

Note that the stopping criteria must be coupled with the QoI of the data as well. If bad quality data is being delivered, then VoI metrics could swing widely. Thus, using VoI for determining when enough data has been collected should ensure a minimum QoI.

4.3 Estimating Trustworthiness

The QoI and VoI of information received from external sources can be used as a proxy to measure the trustworthiness of the source. A trustworthy source must provide information with good QoI and good VoI. Untrustworthy sources may provide data with poor VoI or poor QoI.

When acquisition of data from different domains or partners is automated, estimation of the trustworthiness of the source of data can provide for useful safeguards against an attempt to subvert the automated system.

There are likely to be several other applications of VoI beyond the ones mentioned in this section.

ACKNOWLEDGMENTS

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF-16-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

DSTL/CP114480

REFERENCES

[1] Reilly, J. M., "Multidomain operations: a subtle but significant transition in military thought," tech. rep., Air Force Research Institute Maxwell AFB United States (2016).

[2] Perkins, D. G., "Multi-domain battle: driving change to win in the future," *Military Review* **97**(4), 6 (2017).

[3] Pearson, G. and Verma, Dinesh andde Mel, G., "Value of information: Quantification and application to coalition machine learning," in [*Proc. of Policies for Autonomous Data Governance*], Springer Lecture Notes in Computer Science (2018).

[4] Bisdikian, C., Kaplan, L., Srivastava, M., Thornley, M., Verma, D., and Young, R., "Building principles for a quality of information specification for sensor information," in [*IEEE International Conference on Information Fusion*], 1370–1377 (2009).

[5] Le, F., Ortiz, J., Verma, D., and Kandlur, D., "Policy based identification of iot devices' vendor and type by dns traffic analysis," in [*Policies for Autonomic Data Governance at ESORICS*], (2018).

[6] "Nato code of best practice for c2 assessment," *Command Control Research Program (CCRP)* (2002).