Method and System for Workload Modeling and Resource Estimation for Systems with Heterogeneous Components Using Meta Models

Disclosed is a method and system for workload modeling and resource estimation of deep learning workloads for system with heterogeneous components using meta models. The method and system performs workload modeling and resource estimation based on component classes in heterogeneous system configurations such as, but not limited to, different types of processing units, different machine learning frameworks, and leverages models of individual components within a class, or components across a class, to improve modeling accuracy.

Component classes include, but need not be limited to, types of processing units, network topologies, interconnect links, machine learning frameworks and libraries, machine learning hyperparameters, and computational algorithms. Models of individual components may be vendor-supplied, or derived analytically or empirically using a set of characterization benchmarks. Further, the estimates/contributions of models of individual components may be refined using all other models in the same component class. An aggregate model is then built for a component class A with respect to another component class B based on the relative functionality and dependencies between component class A and component class B.

Further, the estimates/contributions of models of individual components may be refined using aggregate models of other component classes. Individual models are assigned weights, confidence factors, and prediction range intervals based on relative metrics computed across models.

FIG. 1 illustrates a modeling flow schematic depicting how multiple disparate sets of components are incorporated.

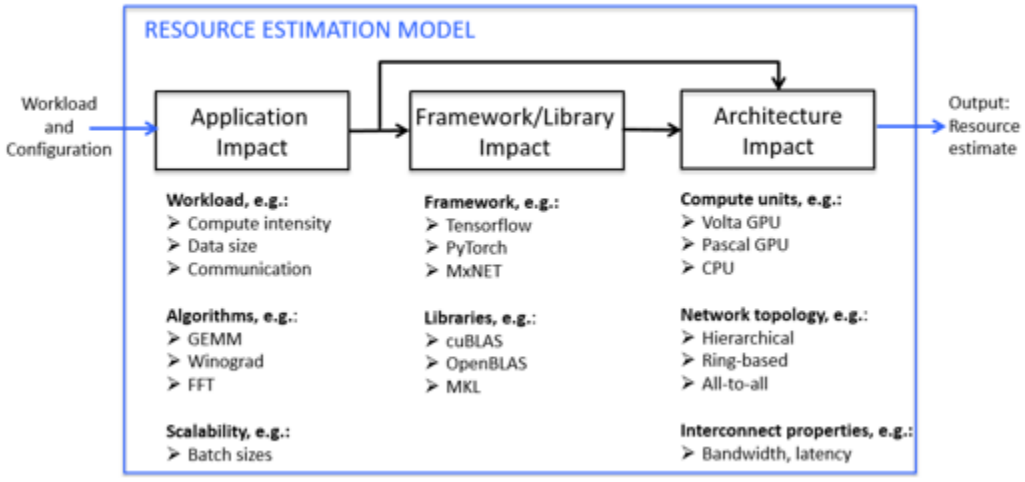

Figure 1

As illustrated in FIG. 1, the method and system enables using information within and across component classes to improve accuracy of modeling complex heterogeneous systems. Components of a heterogeneous system are partitioned into sets of components, each set representing a component class of similar functionality, and different implementations, such as, for example, a class of compute units such as, but not limited to, Volta graphical processing units (GPUs), Pascal GPUs and central processing units (CPUs).

In an embodiment, when modeling one component in a heterogeneous system, accuracy of the model may be improved using information about different components in the same class.

For example, CUDAAdvisor functionality in CUDA® Deep Neural Network library (cuDNN) library predicts performance. This functionality is provided by the system vendor (NVIDIA) and it incorporates deep knowledge of system internals. Comparing the variance in prediction accuracy of CUDAAdvisor on different types of NVIDIA GPUs, provides a measure of the intrinsic complexity and performance predictability of the compute unit which then translates into a confidence factor/prediction-range when modeling the compute unit.

In another embodiment, when modeling one component in a heterogeneous system, accuracy of the model may be improved using related information across components in a different class.

For example, the performance of double precision general matrix multiply (DGEMM) across different compute units may show that the theoretical number of compute operations and peak floating-point operations per second (FLOPS) available in the compute unit are highly correlated. In contrast, for a different algorithm, the correlation may hold for a given compute unit, but not across all compute units.

FIG. 2 illustrates the method and system using modeling of compute units.

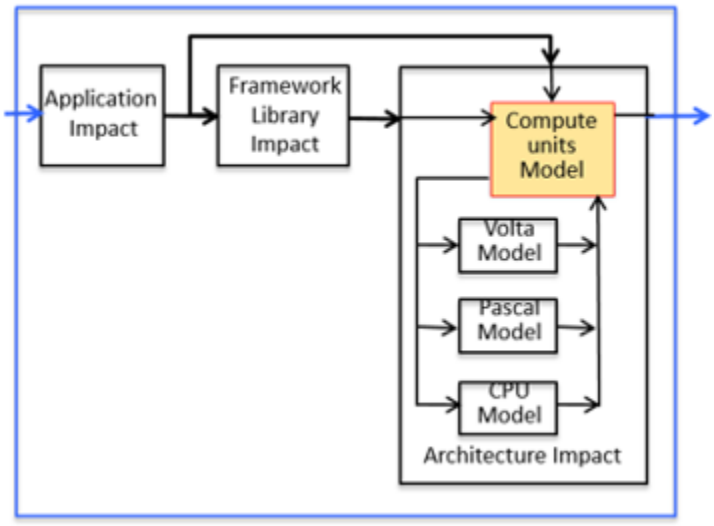

Figure 2

In accordance with an implementation, the method to develop a model for a given system such as, but not limited to, a cloud cluster, is disclosed.

The method differentiates component classes and determines the set of components belonging to each class. The method then builds a model for each individual component based on its functionality and uses a default contribution factor/metric for each individual model.

Subsequently, the method refines an estimate/contribution of models for individual components using other models in the same component class. The method also refines estimate/contribution of models for individual components using models in other component classes.

For each set Z of related component classes and for each class C member of set Z and for each class D member of set Z (except class C), an aggregate model across all components of D with respect to C is built, and an estimate/contribution of models for individual components of C using the aggregate models is refined.

Thereafter, the method combines all individual component models into an overall system model and the aforesaid steps are iterated over multiple times.

Thus, the method and system disclosed herein provides an effective technique for workload modeling and resource estimation using information within and across component classes, to improve accuracy of modeling complex systems.