Towards Quality-of-Service Metrics for Symbolic Knowledge Injection

Andrea Agiollo[1,*,†], Andrea Rafanelli[2,3,*,†] and Andrea Omicini[1]

[1]*Dipartimento di Informatica – Scienza e Ingegneria (DISI), ALMA MATER STUDIORUM—Università di Bologna, Italy*

[2]*Dipartimento di Informatica, Università di Pisa, Italy*

[3]*Dipartimento di Informatica – Scienza e Ingegneria e Matematica (DISIM), Università dell' Aquila, Italy*

Abstract

The integration of symbolic knowledge and sub-symbolic predictors represents a recent popular trend in AI. Among the set of integration approaches, Symbolic Knowledge Injection (SKI) proposes the exploitation of human-intelligible knowledge to steer sub-symbolic models towards some desired behaviour. The vast majority of works in the field of SKI aim at increasing the predictive performance of the sub-symbolic model at hand and, therefore, measure SKI strength solely based on performance improvements. However, a variety of artefacts exist that affect this measure, mostly linked to the quality of the injected knowledge and the underlying predictor. Moreover, the use of injection techniques introduces the possibility of producing more efficient sub-symbolic models in terms of computations, energy, and data required. Therefore, novel and reliable Quality-of-Service (QoS) measures for SKI are clearly needed, aiming at robustly identifying the overall quality of an injection mechanism. Accordingly, in this work, we propose and mathematically model the first – up to our knowledge – set of QoS metrics for SKI, focusing on measuring injection robustness and efficiency gain.

Keywords

symbolic knowledge injection, quality of service, efficiency, understandability, robustness

1. Introduction

Recently, the proposal of machine and deep learning (ML & DL) approaches gave rise to the new artificial intelligence (AI) spring. This increased interest in AI solutions is due to groundbreaking performance that data-driven ML approaches show against manually-defined approaches. Here, humanly-designed models semi-automatically learn task-solving procedures from data via some sort of optimisation mechanism. The variety of tasks which can be tackled in a data-driven way is nearly unlimited, depending on the model and optimisation procedure proposed, and ranges from text [1] to speech [2], image recognition [3], and many more.

ML algorithms usually rely on numeric processing of data, aiming at detecting statistically-relevant patterns such as correlation between variables or regularities in the data. Such an

approach proved successful both in terms of obtained performance and flexibility. Indeed, it requires only to identify the data processing mechanism and its optimisation. However, data-driven approaches – and especially DL ones – do not represent "silver bullet" sorts of solutions, as they usually suffer from interpretability issues. Indeed, most popular modern AI solutions exploit *sub-symbolic* predictors – such as neural networks (NNs) – which are black-box models. We here define *interpretability* as the property of a predictor that enables an expert human user to contemplate it and understand its behaviour. This property represents a fundamental requirement in many applications where humans must be in full control of the computational systems supporting their decisions, such as e-health [4] or smart transportation systems [5].

Recently, few works propose to fix this issue through the integration of symbolic knowledge and sub-symbolic predictors [6, 7, 8]. Among them, few popular approaches propose to inject symbolic knowledge – where "symbolic" refers to the way knowledge is represented – into sub-symbolic predictors [9, 10]. We consider as *symbolic* any intelligible language which is naturally interpretable for both human beings and computers. This includes a number of logic formalisms, and excludes the fixed-sized tensors of numbers commonly exploited in sub-symbolic ML. Therefore, Symbolic Knowledge Injection (SKI) mechanisms propose to leverage human comprehensible information and steer the training process of sub-symbolic predictors towards desirable behaviours. Thus, SKI increases the degree of control over a sub-symbolic predictor and its behaviour, constraining it with human-like common-sense.

SKI approaches usually measure the quality of their injection mechanism simply in terms of performance gain over a standard ML counterpart [9, 10]. While being a valid metric, the performance gain is tightly linked with few injection artefacts—analysed in Section 3.1. Therefore, there exist the necessity of identifying a set of Quality-of-Service (QoS) metrics for SKI which analyse thoroughly its improvements over standard ML approaches. In this position paper, we propose the first set of novel performance metrics for evaluating SKI mechanisms. More in detail, we first focus on the injection quality, aiming to identify possible suitable metrics to decouple the injection quality from the artefacts that usually characterise SKI (see Section 3.2). We then highlight possible efficiency gains that SKI may obtain against standard – non SKI – approaches (see Section 3.3), focusing on the definition of different efficiency measures such as energy, memory and data savings. Finally, we analyse the different realms that may benefit from measurements of the proposed metrics in SKI, obtaining a taxonomy of SKI QoS—Section 4.

2. Background & Definitions

Given the black-box nature of sub-symbolic predictors and their fuzzy optimisation procedure, several recent works propose to leverage symbolic knowledge to steer the model towards desired behaviour(s) [9, 10]. The underlying idea is for the sub-symbolic model to keep some symbolic knowledge into account when drawing its predictions, thus making the sub-symbolic mechanism more intelligible to humans. Instead, since it relies on a rather trivial concept, SKI requires the definition of injection mechanisms that may depend on the model at hand and the desired symbolic knowledge. Therefore, it is complex to define SKI in detail; rather, we can describe it broadly as *any algorithmic procedure affecting how sub-symbolic predictors draw their inferences in such a way that predictions are either computed as a function of, or made*

consistent with, some given symbolic knowledge. Formally, given an injection procedure $\mathcal{I}$, a knowledge base (KB) $\mathcal{K}$, and a sub-symbolic model $\mathcal{N}$ aiming at solving task τ, we define the knowledge-aware model as the result of the application of $\mathcal{K}$ through $\mathcal{I}$ over $\mathcal{N}$. Mathematically, we refer to this model as $\mathcal{N}^{ski}(\mathcal{K}, \mathcal{I}, \tau)$, while we identify its *uneducated* counterpart simply as $\mathcal{N}(\tau)$. We use "uneducated" to identify the sub-symbolic model used in the SKI approach, but unaware of the symbolic knowledge.

Depending on the SKI approach at hand, different type of symbolic knowledge, injection mechanism, and targeted model may appear. Indeed, popular SKI mechanisms either rely on *logic formulæ* – adhering to either FOL or some of its subsets – or *expert knowledge* that represent the humanly interpretable notions. Meanwhile, concerning the targeted model, SKI approaches usually focus on NNs and their variations, due to their flexibility and success. Noticeably, the injection mechanism character represents the most variable one, where different SKI proposals rely on their unique characterisation. Indeed, the most popular set of these mechanisms aim at injecting the knowledge at hand via constraining the learning process of the sub-symbolic model, usually modifying the loss function [9, 10, 11]. Meanwhile, other popular approaches aim at constructing sub-symbolic models in such a way that the NN structure reflects the knowledge to inject [12, 13, 6, 14, 15]. Embedding approaches also represent a popular solution in this realm, aiming at converting symbolic knowledge into numeric-array form to be used as extra training data [16, 17, 18].

It is common for works in the SKI realm to measure the strength of their mechanism as the gain in performance that is achieved by the SKI model against its *uneducated* counterpart. Mathematically, the effectiveness of the injection mechanism $\mathcal{I}$ is measured as:

$$\mathcal{E}(\mathcal{I}) = \mathcal{P}(\mathcal{N}^{ski}, \tau) - \mathcal{P}(\mathcal{N}, \tau) \tag{1}$$

where $\mathcal{P}(i, j)$ measures the performance – accuracy, F1-score, MSE, etc. – of a model i over a task j. While being indicative of the quality of the SKI approach, such metric does not grasp every aspect of the injection mechanism, as there exist multiple artefacts that may affect $\mathcal{E}(\mathcal{I})$ (see Section 3.1). Consequently, it is necessary to define reliable measures for measuring the performance of the injection mechanism in SKI. Moreover, due to the sudden rise in research interest towards sustainable AI approaches [19], there exist the opportunity to analyse if and how SKI brings benefit in terms of computations, energy and data required to train and deploy sub-symbolic approaches.

3. SKI Quality-of-Service Metrics Definition

In this section we propose and analyse a novel set of metrics for identifying the quality of SKI systems. The overview, along with a brief classification, of the pinpointed metrics is proposed in Section 3.1, distinguishing between efficiency-related metrics and injection-quality ones. In Section 3.2 we then present a set of qualitative metrics aiming at measuring the quality of the injection process by itself. Finally, we focus on efficiency-related metrics in Section 3.3, presenting a list of suitable measures.

3.1. Overview

Most, if not all, of the proposed SKI mechanisms rely exclusively on measuring predictions quality improvements over an equivalent uneducated counterpart to show their strength. However, predictions quality and the corresponding improvements do not depend solely on the quality of the injection mechanism at hand. Indeed, there exist multiple artefacts that might influence the outcome quality of SKI tools. To name a few:

A1 — *Knowledge quality and coverage.* Highly detailed and target specific a-priori knowledge usually bear higher quality to the overall injection mechanism, resulting in improved performance gains. While being trivial, this aspect attains paramount importance in SKI quality measurements and is usually ignored by SKI proposals. Indeed, different SKI approaches rely on different KBs to tackle similar, if not identical, tasks. Therefore, decoupling the quality measurements of the injection mechanism from the knowledge quality is complex and represents an open issue.

A2 — *Baseline mechanism quality.* Relying on injection, SKI approaches build on top of sub-symbolic models with variable levels of performance. Therefore, performance gains of a SKI mechanism might vary significantly when switching between underlying sub-symbolic models.

A3 — *Task at hand.* Depending on the complexity and nature of the considered task, knowledge injection may bear different levels of benefit. Generally speaking, more complex tasks should profit more than trivial ones.

Taking into account the amount of artefacts that might alter the outcome quality of SKI mechanisms, it is trivial to understand that solely measuring performance improvements represents a dreadful choice and is a no-go. Moreover, the application of injection techniques in sub-symbolic systems might introduce a set of advantages different from sole performance improvements that are usually ignored by previous works. More in detail, we theorise that the use of SKI enables reducing the amount of data and energy required to optimise the model at hand. The underlying idea behind such hypothesis is that part of the thought process that the uneducated system would need to learn from data-driven optimisation is displaced to the a-priori knowledge in SKI frameworks. Indeed, the injected knowledge can be used either as guideline or as foundations for the learning process of the sub-symbolic model, therefore reducing the amount of knowledge load that the system must learn. Furthermore, injection mechanisms may aid in terms of model's explainability. Indeed, SKI relies on a-priori knowledge to steer the sub-symbolic model learning towards a desirable behaviour or behavioural boundaries. We here speculate that such steering process should produce more explainable models—or at least understandable ones. Therefore, we propose to evaluate SKI mechanisms depending on their ability to produce simpler, better explainable models. The underlying assumption is that one of the issues with explaining sub-symbolic models is their unpredictable behaviour, usually expressed as *instability* [20].

To solve the issues concerning SKI quality measurements, we here propose six novel metrics, taking into account different aspects of SKI mechanisms. More in detail, we propose to classify these set of measures into two classes:

Injection quality metrics — This family of metrics represents the theoretical measures aiming at analysing the quality achieved by the injection procedure by itself, decoupling it from the set of artefacts (**A1**-**A3**) that commonly affect SKI.

Efficiency metrics — This family of metrics aims at analysing how much SKI helps achieving more efficient sub-symbolic models. Here, the *efficiency* term bears a wide-ranging meaning, from computational efficiency to data-usage efficiency, and is discussed thoroughly in Section 3.3.

Figure 1 shows the classification of the proposed SKI QoS metrics, listing all of them.
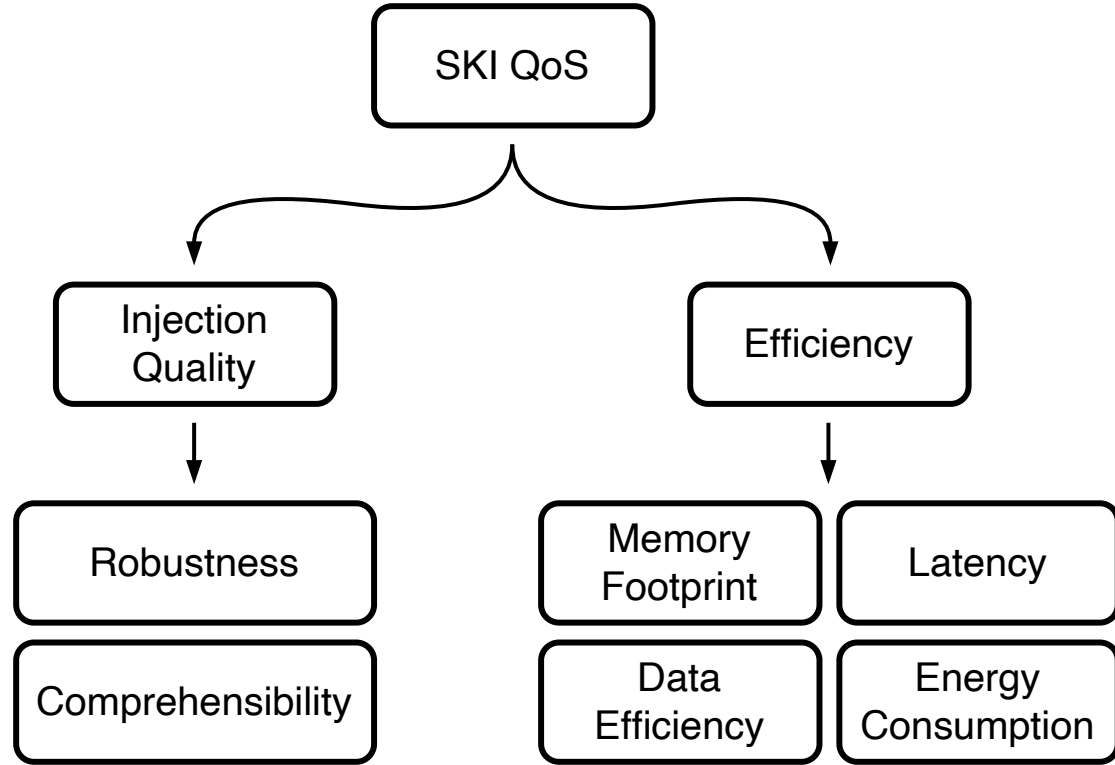
Figure 1: Classification of the proposed QoS metrics.

3.2. Injection QoS

We here analyse the quality of an injection mechanism $\mathcal{I}$ belonging to a SKI framework, aiming to remove the effect of artefacts **A1**-**A3**. The proposed metrics are:

Robustness — i.e., the capability of the injection mechanism to adapt to variations of input data and knowledge.

Comprehensibility — i.e., the capability of the injection mechanism to produce more intelligible models.

Insights regarding both metrics are also provided, highlighting possible issues, such as the subjective nature of the comprehensibility metric which makes it difficult to formulate mathematically. Therefore, the proposed metrics represent a first step towards a precise and clear-cut formalisation of injection quality.

3.2.1. Robustness

Well-performing models are *generalisable*, as they are capable of producing strong predictions on unseen data. Model generalisability is tightly connected with its robustness as defined in counterfactual explanations approaches [21, 20]. Indeed, counterfactual explanations propose to leverage input variations that cause the system to behave differently to reveal possible interesting behaviour of the analysed model. In essence, the rationale behind counterfactual methods is to use perturbations of the input to identify certain invariant characteristics of the counterfactual analysis, which may have causal relevance to the predicted phenomena—for a overview of these approaches, please refer to [22]. We introduce the concept of statistical robustness, which involves altering the input data in a manner analogous to counterfactual analysis and define a robust estimator as one whose validity is unaffected by the violation of its initial assumptions. Similarly, we define an injection mechanism to be *robust* if its prediction ability is not altered much when a slight perturbation is applied to the injected knowledge or input data. Therefore, we first need to identify possible perturbations to be applied after knowledge injection, namely:

Injected knowledge perturbation $-$ We here aim at determining whether the introduction of a slight perturbation α to the injected knowledge results in a significant change to the model predictions. In other words, we are most interested in understanding the reliability of the injection mechanism. Mathematically, we define a knowledge perturbation as

$$
\begin{aligned}
\mathcal{K}_p &= \mathcal{K} + \alpha \\
\alpha &\sim \mathcal{U}(a, b)
\end{aligned}
\tag{2}
$$

where $\mathcal{K}_p$ represents the perturbed symbolic knowledge, $\mathcal{K}$ represents the original KB, and $\mathcal{U}(a, b)$ is a uniform distribution from which the perturbation is sampled.

Input perturbation $-$ We here assess the robustness of the model $\mathcal{N}^{ski}$ itself by determining whether it is resistant to variations of input. Mathematically, we define the input perturbation as

$$
\begin{aligned}
x_p &= x + \alpha \\
\alpha &\sim \mathcal{U}(a, b)
\end{aligned}
\tag{3}
$$

where x_p represents the perturbed input, and x its original counterpart.

Once the possible perturbations are identified, we can define the SKI robustness measure as:

$$
\begin{aligned}
M_r &= \frac{1}{|\mathcal{P}(\mathcal{N}^{ski}(e), \tau) - \mathcal{P}(\mathcal{N}^{ski}(e_p), \tau)|} \\
&\text{s.t.} \quad e_p = e + \alpha \quad | \quad |\alpha| \leq \epsilon
\end{aligned}
\tag{4}
$$

where e and e_p represent either the input data or the injected knowledge and its perturbed counterpart, and $\mathcal{P}$ represents the performance of the model. Relevantly, ϵ represents the selected maximum amount of perturbation and is used to highlight that model's robustness is measured using small perturbations. By definition, robust SKI approaches are characterised by small values for the denominator – absolute difference of performance when a small perturbation is applied –, thus obtaining high M_r scores.

3.2.2. Comprehensibility

Neural networks are regarded as black-box since the internal processes that lead to specific final decisions are mysterious to humans. Due to the widespread use of these methods, even in relatively-sensitive fields such as medicine and law, the aspect of these models' explainability is becoming increasingly important in modern times. This is exemplified by the emergence of so-called XAI (eXplainable AI) and the GDPR's recommendations regarding the need to provide systems that are explicable and understandable to external users.

The introduction of symbolic knowledge within sub-symbolic models prompts one to consider acquiring more comprehensible models, mostly because symbolic knowledge is expressed in more comprehensible forms to the human eye and thinking. Indeed, symbolic knowledge typically takes the form of logical rules, such as $A \wedge B \Rightarrow C$, knowledge graphs, decision trees, algebraic equations, and so forth. Compared to the numerous complex internal constructs of sub-symbolic models, the form of expression adopted by this type of knowledge is significantly more understandable. Therefore, it is natural to believe that introducing a type of reasoning that is more easily encoded by the human mind would result in a more understandable system in general.

One wonders, how a system's comprehensibility can be measured. First, from a particular perspective, understanding something is itself a subjective construct. In other words, what is comprehensible to one person may be less comprehensible to another, and this is primarily due to how certain concepts are perceived and, consequently, identified and mentally assembled. Etymologically, the Latin word for *"comprehension"* is *"cum capere"*, which literally means *"to take and put together"*. As a result, it is not easy to establish an actual metric of comprehension. What comes to mind is that it might be useful to use extraction mechanisms that allow to extract the internal reasoning processes of black-box systems in a way that is visually or conceptually understandable to humans, such as decision trees.

Once these extractions are obtained, comparisons can be made by assessing, for instance, the obtained model's complexity. Complexity may refer either to the depth of the tree or to visual characteristics, such as the number of nodes present. In [23], for example, a measure of syntactic complexity is defined as:

$$U(n,b) := \alpha \frac{n}{k} + (1-\alpha)\frac{b}{k^2} \tag{5}$$

where $\alpha \in [0,1]$ is a tuning factor that adjusts the weight of n and b, n is the number of nodes, b is the number of branches, and k is a coefficient built by the authors.

We expect the decision tree representing the reasoning processes of the black-box model to be more complex than that obtained from the black-box model with knowledge injection.

Obviously, we cannot be certain of this, and one way to assess the comprehensibility of the model is to solicit feedback from external users by asking simple questions about how much more comprehensible the tree of the black-box model and the model with knowledge injection is. This approach is also utilised in [24], and introduces the concept of *human-in-the-loop*, in which the human acts to correct, refine, monitor, and evaluate the models. In this context, pertaining to the interpretability and comprehensibility of ML models, human involvement is crucial and decisive. This is due to the fact that, as stated previously, interpretation and comprehension of models depend not only on aspects attributable to the model's construction in technical terms, but also on a subjective aspect attributable solely to human experience. Incorporating such feedback and considering it during the development and refinement of models could be persuasive in this regard.

3.3. Efficiency-related QoS

While being mainly proposed for steering sub-symbolic model towards desirable behaviours, SKI can, in principle, bear many advantages related to the overall predictor efficiency. Indeed, injection techniques are intrinsically substituting part of the data-drivenly learnt concepts with possibly complex a-priori knowledge. Thus, we here build upon the idea of measuring the efficiency improvements that SKI might provide against traditional sub-symbolic predictors.

The *efficiency* term, however, is foggy by itself and needs a more fine-grained definition, as it requires to identify the target of the measurement. More in detail, throughout this work we consider measuring characteristics related to *computational efficiency* of a predictor, due to the paramount importance that AI sustainability is gaining recently. Indeed, a multitude of works stress the necessity of identifying resource-friendly AI solutions [25]. Therefore, we focus on computational complexity, analysing the following interesting features:

Memory footprint M_m — i.e., the size of the sub-symbolic predictor under examination.

Energy consumption M_e — i.e., the amount of energy required to build and run the model.

Latency M_l — i.e., the time required to run a predictor.

Data efficiency M_d — i.e., the amount of data required to optimise the sub-symbolic model.

We then identify their efficiency as the effective gain that the injection mechanism obtains over its uneducated counterpart. Therefore, we define efficiency metrics as the variation between $\mathcal{N}$'s M_i and $\mathcal{N}^{ski}$'s M_i.

3.3.1. Memory Footprint

We first consider the ability of SKI mechanisms to produce lightweight sub-symbolic models. The injected knowledge lifts part of the learning burden from the predictor at hand, aiming at avoiding complex or unfeasible data-driven notions learning procedure. Indeed, the a-priori concepts which are injected should not be learnt from the training samples anymore. As a consequence, the amount of notions that the sub-symbolic models must learn data-drivenly might reduce significantly. Fewer notions to be learnt are generally linked with the possibility

of shrinking the sub-symbolic model at hand. Here, we consider model shrinking as a simple reduction on the amount of parameters that characterises the sub-symbolic model. Enabling model shrinkage, SKI mechanisms are capable of producing memory efficient predictors. Therefore, we consider to be stronger SKI approaches the mechanisms that can produce smaller sub-symbolic models, as they maximise the utility of the injected notions.

The analysis of sub-symbolic models efficiency – especially NN – is growing in popularity recently, due to the ever-increasing need for sustainable intelligent models. Therefore, there exist a set of well-established memory footprint measures of sub-symbolic models in current literature. We consider leveraging such measures to analyse the efficiency gain of SKI approaches, avoiding cumbersome definition of ad-hoc – possibly faulty – SKI metrics. More in detail, we propose to measure models footprint by counting the number of parameters composing the underlying NN model. Alternatively, we can also leverage metrics such as Floating Point OPerations (FLOPs) or Multiplication Addition Computations (MACs), which measures the amount of total operations or multiplications and additions required to perform a single inference respectively. MACs consider solely multiplications and summations as they represent the most common computations in NNs. These measures are indicative of the amount of memory required either to fit the whole sub-symbolic model – total number of parameters – or to run it—FLOPs and MACs. Albeit simple, these metrics are effective for measuring models complexity and overall computational memory efficiency [26, 27].

However, measuring FLOPs or MACs by themselves is not sufficient to measure the effectiveness of SKI in increasing memory efficiency. Indeed, we need to properly define the memory efficiency improvements obtained leveraging SKI against its uneducated counterpart. Therefore, we define the memory efficiency improvement as the amount of memory gain that injection attains while achieving at least the same prediction performance of the uneducated sub-symbolic model it works upon. Mathematically:

$$M_m = \Psi \left(\mathcal{N}^{ski} \left(\mathcal{K}, \mathcal{I}, \tau \right) \right) - \Psi(\mathcal{N}(\tau))$$
$$\text{s.t.} \quad \mathcal{P}(\mathcal{N}^{ski}, \tau) \geq \mathcal{P}(\mathcal{N}, \tau) \tag{6}$$

where $\mathcal{N}$ represents the sub-symbolic model working on task τ, and $\mathcal{N}^{ski}$ identifies its SKI counterparts trained with the aid of a-priori knowledge $\mathcal{K}$ injected through the injection mechanism $\mathcal{I}$. Ψ represents the memory efficiency metric chosen for the single model – e.g., FLOPs, MACs, etc. –, while $\mathcal{P}(x, y)$ identifies the performance obtained by the model x over the task y.

From Equation (6) it is possible to notice that the proposed memory efficiency gain metric suffers from the same artefacts **A1**-**A3** dependency discussed in Section 3.1. However, the aim of the proposed metric is to identify efficiency gains of SKI w.r.t. to its uneducated counterpart. Therefore, the **A2** and **A3** artefacts dependency assume smaller relevance. Meanwhile, dependency on knowledge quality – i.e., **A1** – can be eliminated by standardising the KB used to compare different SKIs.

3.3.2. Energy Consumption

Recently, research groups scattered around the globe are starting to focus their attention toward the development of sustainable intelligent systems. Indeed, popular AI solutions commonly rely on complex and resource hungry sub-symbolic systems such as NNs aiming at attaining general intelligence—i.e., AI systems capable of solving multiple heterogeneous tasks. However, the rampaging evolution of NNs resource hungriness is not sustainable both economically and environmentally.

Multiple research proposals focus on the analysis of energy consumption of AI solutions throughout their life cycle [28, 29]. Most of such approaches rely on ad-hoc strategy to compress or optimise sub-symbolic models, missing the massive opportunity given by SKI approaches. Indeed, the introduction of injection mechanisms in the data-driven pipeline of sub-symbolic training mechanisms allows to reduce the amount of computations required to train and run sub-symbolic predictors. Knowledge injection reduces the complexity of the learning process, removing the burden of learning the a-priori KB from the training samples available. Thus, it is reasonable to assume for SKI mechanisms to allow reducing the amount of computations that characterise a model life-cycle. At least it is reasonable to identify as stronger SKI approaches the set of frameworks that majorly reduce the energy consumption of a sub-symbolic model.

The proposed energy consumption metric is tightly related with memory efficiency (Section 3.3.1). Indeed, it is usually the case for smaller models to require fewer amounts of energy to train and run. However, there might exist memory efficient models requiring a higher amount of energy to train and run, such as sparse models. Indeed, sparsity induces a lower amount of operations, but is not usually effectively implemented at hardware level, increasing power consumption [26]. Therefore, energy efficiency represents a metric that is worth to be analysed by itself.

To analyse energy consumption and the possible improvements that SKI might introduce, we need to first define the life-cycle of AI models, analysing each component resource hungriness. In order to build and deploy a data-driven AI solution, there exist several steps to complete, namely:

1. *Model definition.* This phase represents the process of analysing the task at hand and selecting the best suitable sub-symbolic model.

2. *Model training.* During this phase the sub-symbolic model is trained using the set of samples extracted from a specific dataset. The amount of training samples may differ depending on the task and model at hand, impacting the resource requirements. Indeed, during training, the predictor runs and is updated multiple times to achieve its optimal setup.

3. *Model testing.* The obtained optimal model is tested against a – limited – set of testing samples to check if the performance are satisfactory.

4. *Model deployment.* Once the model is trained, reaching satisfactory performance, it is deployed in its real-world application. Here, the model runs multiple times, depending on the specific application.

From the definition of the data-driven AI life-cycle, it is possible to highlight that the *training* and *deployment* phases are the most resource hungry. Indeed, training requires a huge amount of model runs and updates, while the model deployment might be very costly – w.r.t. energy – depending on predictions frequency and its life expectation. Therefore, we need to define an application specific trade-off parameter α balancing the cost of training and deployment.

We can now define the energy consumption efficiency of a SKI mechanism as the amount of energy saving of its life-cycle against an uneducated counterpart. Mathematically:

$$M_e = \Upsilon_t \left(\mathcal{N}^{ski} \left(\mathcal{K}, \mathcal{I}, \tau \right) \right) + \alpha \Upsilon_d \left(\mathcal{N}^{ski} \left(\mathcal{K}, \mathcal{I}, \tau \right) \right) - \left[\Upsilon_t(\mathcal{N}(\tau)) + \alpha \Upsilon_d(\mathcal{N}(\tau)) \right]$$
$$\text{s.t.} \quad \mathcal{P}(\mathcal{N}^{ski}, \tau) \geq \mathcal{P}(\mathcal{N}, \tau) \tag{7}$$

where Υ_t and Υ_d represent the energy spent by the model during training and deployment respectively.

3.3.3. Latency

The efficiency level of a sub-symbolic model can be intended as the amount of time that it takes to operate—i.e., compute its prediction. This time is usually called *latency* and refers to the amount of time necessary to produce an output from a single sample fed into a trained NN. Latency is a good indicator of the ability of a model to produce real-time outputs in a real-world application. Such metric represents a crucial characteristic of an AI model in various time-costly scenarios. Few examples might come from scenarios where human lives depend on the AI model deployed, such as intelligent transportation [5] and e-health [4]. Moreover, latency assumes a relevant role in multi-agent scenarios, where collaboration between multiple intelligent entities is required, and there can not exist lag between them due to lengthy computations [30]. Therefore, research efforts in the field of sub-symbolic models recently focused on identifying time-sensitive models.

Given a SKI mechanism, we now propose to measure its capability of increasing the time efficiency of the sub-symbolic model upon which it works. Indeed, injection systems may bring several timing benefits, removing few unnecessary computations. On the other hand, SKI systems might also introduce delayed computations linked with the analysis of the given KB, such as grounding issues [31]. Therefore, measuring if and how SKI influence model latency represents a relevant research aspect for defining SKI QoS. We here define the latency gain M_l as the difference between the inference time of the SKI model and its uneducated counterpart:

$$M_l = \mathcal{T}(\mathcal{N}^{ski}(\mathcal{K}, \mathcal{I}, \tau)) - \mathcal{T}(\mathcal{N}(\tau))$$
$$\text{s.t.} \quad \mathcal{P}(\mathcal{N}^{ski}, \tau) \geq \mathcal{P}(\mathcal{N}, \tau) \tag{8}$$

where $\mathcal{T}$ represents inference time.

Similarly to the energy measurement, the latency metric is tightly related to the complexity of the obtained sub-symbolic models and therefore with M_m. However, like energy consumption, latency is not always directly proportional to the amount of operations that construct the model at hand. Sparsely-structured operations might slow down the inference process due to their inefficient computation at hardware level. Moreover, input data complexity and quality might alter the latency achieved by the predictor. Indeed, inference over different samples

characterised by the same structure, may take vastly different timings, as shown in the attack proposed in [32].

3.3.4. Data Efficiency

Sub-symbolic models rely on data-driven training approaches to optimise their performance over a given task. While representing the key of their groundbreaking performance, this data-driven procedure is not a silver bullet solution. Indeed, it requires collecting significant amount of data samples for each task to be tackled. The data collection process is time costly and depending on the application might be cumbersome and foggy—e.g., emotion recognition [33]. Given this costly drawback, recent research efforts have focused on proposing data-frugal models [34]. Among them, knowledge injection mechanisms play a significant role [10]. Indeed, leveraging a-priori knowledge, SKI removes part of the learning process burden. Several concepts that an uneducated model would need to learn from a set of data are injected automatically into the educated model. Therefore, some portions of the training data are not required for the model to attain an acceptable performance level. Here, the leveraged effect of SKI is to decouple the learning process from part of the data, obtaining frugal models.

We here define the data efficiency gain M_d of a SKI mechanism as the difference between the *data footprint* required to train an uneducated sub-symbolic model and its SKI counterpart. Given a dataset D defined as

$$D = \{(x_i^{\{d_1,\ldots,d_n\}}, y_i^{\{h_1,\ldots,h_m\}})\}, \ x_i \in \mathcal{X}, \ y_i \in \mathcal{Y}, \ \forall i \in [0, N] \tag{9}$$

we define the *data footprint* $\mathcal{D}$ as the amount of samples used during training multiplied by the memory cost of a single sample, which can be roughly computed as:

$$\mathcal{D} = N \times \prod_{i=1}^{n} d_i \times m_i \tag{10}$$

where N represents the amount of samples in the dataset, d_i represents the ith dimension of a training sample and m_i represents the memory used to store the value corresponding to i^{th} dimension of the training sample. Therefore, the data efficiency gain of a SKI mechanism can be written as:

$$\begin{aligned} M_d = &\mathcal{D}(\mathcal{N}(\tau)) - \mathcal{D}(\mathcal{N}^{ski}(\mathcal{K}, \mathcal{I}, \tau)) \\ &\text{s.t.} \quad \mathcal{P}(\mathcal{N}^{ski}, \tau) \geq \mathcal{P}(\mathcal{N}, \tau) \end{aligned} \tag{11}$$

The simplest approach to improve data efficiency in SKI mechanisms is to reduce the amount of samples that compose the training dataset—N in Equation (10). However, it may also be interesting to consider increasing data efficiency via reduction of the dimensionality of the dataset, either by reducing n or d_i. Indeed, injecting some a-priori knowledge describing a subset of the dataset features, might allow to remove the corresponding dimension from the dataset samples. Moreover, it would also be possible to reduce data complexity by compressing data representations—i.e., reduce memory representation m. However, SKI techniques usually do not impact m directly, as this is usually hardware defined.

4. Taxonomy of SKI QoS

We here propose a taxonomy (Figure 2) that classify the various proposed metrics according to two relevant research realms such as sustainability and explainability. The *latency* and *energy consumption* metrics are clearly related to model sustainability and in disjunction from explainability. Indeed, while it is obvious to consider more energy-friendly mechanisms to be characterised by higher level of sustainability, it is not possible to identify them as more explainable. On the opposite side of the spectrum we find the *robustness* and *comprehensibility* metrics, which are clearly related to the XAI realm, while being disconnected from suistainability. Indeed, there exist no proofs that an explainable model tends to be more suistainable, but rather the opposite, as the explanation mechanism might be very complex a energy hunger. A more interesting analysis can be done concerning the *memory footprint* and *data efficiency* metrics, for which we believe there exist strong connections to both the model's sustainability and explainability. The relationship between the memory footprint metric and model sustainability is immediate: what we measure is the number of parameters that comprise a model and therefore its computational complexity and power hungriness. More complex models are less sustainable, and less complex models are more sustainable. In general, we anticipate that models with knowledge injection will lead to a restriction of the sub-symbolic model with a noticeable reduction in parameters, resulting in increased sustainability. Also with regard to the model's explainability relationship, it can be argued that a simpler models with fewer parameters are more interpretable [35]. Integration of symbolic and sub-symbolic systems is thought to improve the explainability of black-box models in part because the sub-symbolic model is simplified and requires fewer parameters. Focusing on data efficiency, in terms of sustainability we can state that utilising a smaller amount of data to train a model as opposed to large amounts of data is considered a sustainable solution—i.e., less data equals greater sustainability and viceversa. In terms of explainability, the process of learning models trained with fewer data is simpler and less ambiguous. Again, it is believed that knowledge injection improves the sustainability and explainability of models by requiring less training data. As described in Section 3.3.4, the large amount of data required by the models is compensated for by the knowledge injected into them.
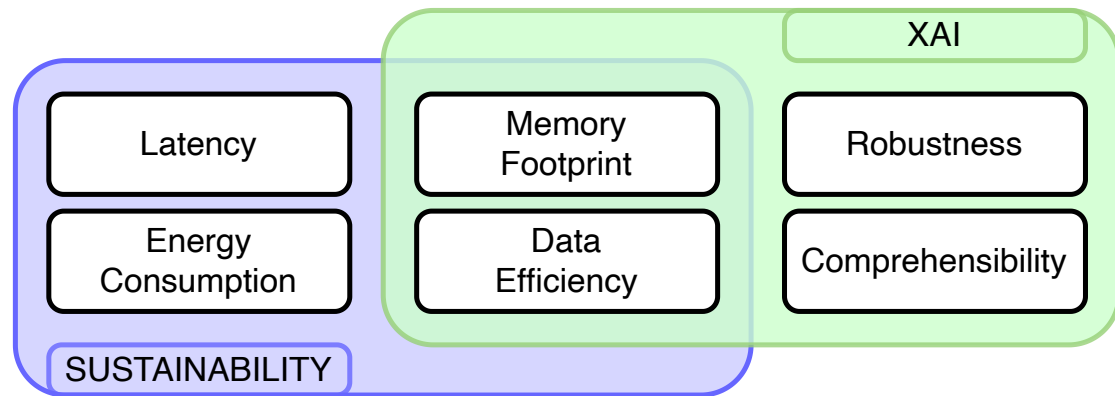

Figure 2: Taxonomy of the proposed QoS metrics.

5. SKI QoS Impacts on Agents

In this section we briefly discuss the possible impact of SKI and its related QoS metrics on the field of agents and Multi-Agents Systems (MAS). Here, it is surely relevant to highlight the impact of sub-symbolic – especially ML and DL – systems efficiency in the agents life-cycle. Indeed, popular works and products propose to leverage intelligent agents to solve complex tasks such as identification and understanding of the surrounding environment. Such intelligent agents rely on heavily integrated sub-symbolic mechanisms which might be even learnt on the flight. Thus, advanced agents must be capable of training and running sub-symbolic mechanisms which might be arbitrarily complex, depending on the task at hand. As a consequence, the possible inefficiency of sub-symbolic models integrated into agents impacts their functioning in several ways, namely:

- Energy and computational inefficiency of sub-symbolic models affects the overall independence of agents, limiting their life-span [29]. Indeed, agent systems usually rely on limited amount of energy to enable agents freedom and thus require energy-efficient solutions. Therefore, measuring the improvements under energy consumption obtained via SKI mechanisms positively affects agents systems, enabling the identification of deployable solutions.

- Data inefficiency of sub-symbolic models hinders the ability of agents to learn on the flight [36]. Indeed, depending on the considered scenario, agents might gather only small amount of viable data from the surrounding environment. Thus, it is relevant to measure the data efficiency gains obtained through SKI, as it alleviates part of the data collection burden from intelligent agents.

- Computational inefficiency of sub-symbolic models hinders the deployability of intelligent agents solutions in heavily constrained environments [37]. Indeed, there exist several scenarios requiring agents to be as rawboned as possible—e.g., emergency systems. In this context, heavy and computationally inefficient sub-symbolic mechanisms could not be embedded into agents.

QoS aspects of sub-symbolic models come into play heavily when considering MAS as well. Indeed, such systems present the efficiency issues related with the deployment of intelligence into agents, as well as requiring hefty coordination aspects. The coordination requirements characteristics of MAS introduce the need for efficient interaction between the entities of the systems. Here, interaction efficiency is tightly linked with the latency of the internal intelligence engine – i.e., sub-symbolic model – of communicating agents. Indeed, interacting agents can not be delayed by lengthy predictions of sub-symbolic models. Moreover, the agents interaction might require exchanging information concerning internal sub-symbolic systems to enable common intelligence or community reasoning. Therefore, there exist the need for the underlying sub-symbolic systems to be comprehensible between agents. The comprehensibility aspect is fundamental, especially when dealing with heterogeneous agents which incorporate their own model and knowledge base and tackle different highly-specific tasks. Therefore, it is surely relevant to analyse possible improvements or degradation in comprehensibility aspects of multiple sub-symbolic models, in order to allow MAS to converge towards global intelligence.

6. Conclusions & Future Work

In this work we propose and mathematically model the first set of Quality-of-Service (QoS) metrics for SKI mechanisms, aiming at overcoming the open issues that affect SKI strength evaluation. We define robustness and comprehensibility metrics to analyse effectively the efficacy of the injection mechanism by itself, decoupling it from possible artefacts that characterise SKI . We then focus on the efficiency gains achievable through SKI and identify four qualitative metrics, namely: *(i)* memory footprint efficiency – i.e., gain in terms of model's complexity; *(ii)* energy efficiency – i.e., gain in terms of total energy required to train and deploy a sub-symbolic model; *(iii)* latency efficiency – i.e., improvements in terms of time required for inference; and *(iv)* data efficiency – i.e., improvement in terms of amount of data required to optimise a sub-symbolic model. We also propose to categorise the QoS metrics into a taxonomy that highlights the different realms that may benefit from measurements of the the proposed metrics.

The QoS metrics proposed in this work are modelled theoretically, and we aim at testing their efficacy in the future. In particular, we aim at benchmarking a set of state-of-the-art SKI mechanisms through our QoS metrics. This would allow to identify the effective benefits that injection mechanisms bring to the realm of symbolic and sub-symbolic integration. Finally, in the future, we would like to develop a tool capable of automatically benchmarking a given SKI mechanism leveraging our QoS. To do so, we aim at developing a SKI-QoS library, which will allow the research community to thoroughly test novel injection mechanisms.

References

[1] D. W. Otter, J. R. Medina, J. K. Kalita, A survey of the usages of deep learning for natural language processing, IEEE Transactions on Neural Networks and Learning Systems 32 (2021) 604–624. doi:10.1109/TNNLS.2020.2979670.

[2] A. B. Nassif, I. Shahin, I. B. Attili, M. Azzeh, K. Shaalan, Speech recognition using deep neural networks: A systematic review, IEEE Access 7 (2019) 19143–19165. doi:10.1109/ACCESS.2019.2896880.

[3] Z.-Q. Zhao, P. Zheng, S.-t. Xu, X. Wu, Object detection with deep learning: A review, IEEE Transactions on Neural Networks and Learning Systems 30 (2019) 3212–3232. doi:10.1109/TNNLS.2018.2876865.

[4] A. Esteva, K. Chou, S. Yeung, N. Naik, A. Madani, A. Mottaghi, Y. Liu, E. Topol, J. Dean, R. Socher, Deep learning-enabled medical computer vision, NPJ Digital Medicine 4 (2021) 1–9. URL: https://www.nature.com/articles/s41746-020-00376-2.

[5] S. M. Grigorescu, B. Trasnea, T. T. Cocias, G. Macesanu, A survey of deep learning techniques for autonomous driving, Journal of Field Robotics 37 (2020) 362–386. doi:10.1002/rob.21918.

[6] M. V. M. França, G. Zaverucha, A. S. d. Garcez, Fast relational learning using bottom clause propositionalization with artificial neural networks, Machine Learning 94 (2014) 81–104. doi:10.1007/s10994-013-5392-1.

[7] A. S. d'Avila Garcez, M. Gori, L. C. Lamb, L. Serafini, M. Spranger, S. N. Tran, Neural-symbolic computing: An effective methodology for principled integration of machine

learning and reasoning, Journal of Applied Logics 6 (2019) 611–632. URL: https://collegepublications.co.uk/ifcolog/?00033.

[8] A. Agiollo, G. Ciatto, A. Omicini, Graph neural networks as the copula mundi between logic and machine learning: A roadmap, in: R. Calegari, G. Ciatto, E. Denti, A. Omicini, G. Sartor (Eds.), WOA 2021 – 22nd Workshop "From Objects to Agents", volume 2963 of *CEUR Workshop Proceedings*, Sun SITE Central Europe, RWTH Aachen University, 2021, pp. 98–115. URL: http://ceur-ws.org/Vol-2963/paper18.pdf, 22nd Workshop "From Objects to Agents" (WOA 2021), Bologna, Italy, 1–3 September 2021. Proceedings.

[9] M. Diligenti, S. Roychowdhury, M. Gori, Integrating prior knowledge into deep learning, in: X. Chen, B. Luo, F. Luo, V. Palade, M. A. Wani (Eds.), 16th IEEE International Conference on Machine Learning and Applications, ICMLA 2017, Cancun, Mexico, December 18-21, 2017, IEEE, 2017, pp. 920–923. doi:10.1109/ICMLA.2017.00-37.

[10] J. Xu, Z. Zhang, T. Friedman, Y. Liang, G. Van den Broeck, A semantic loss function for deep learning with symbolic knowledge, in: J. G. Dy, A. Krause (Eds.), Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, volume 80 of *Proceedings of Machine Learning Research*, PMLR, 2018, pp. 5498–5507. URL: http://proceedings.mlr.press/v80/xu18h.html.

[11] G. Marra, F. Giannini, M. Diligenti, M. Gori, LYRICS: A general interface layer to integrate logic inference and deep learning, in: U. Brefeld, É. Fromont, A. Hotho, A. J. Knobbe, M. H. Maathuis, C. Robardet (Eds.), Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2019, Würzburg, Germany, September 16-20, 2019, Proceedings, Part II, volume 11907 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 283–298. doi:10.1007/978-3-030-46147-8_17.

[12] V. Tresp, J. Hollatz, S. Ahmad, Network structuring and training using rule-based knowledge, in: S. J. Hanson, J. D. Cowan, C. L. Giles (Eds.), Advances in Neural Information Processing Systems 5, Morgan Kaufmann, 1992, pp. 871–878. URL: http://papers.nips.cc/paper/638-network-structuring-and-training-using-rule-based-knowledge, NIPS Conference, Denver, Colorado, USA, November 30–December 3, 1992.

[13] A. S. d. Garcez, D. M. Gabbay, Fibring neural networks, in: D. L. McGuinness, G. Ferguson (Eds.), Proceedings of the Nineteenth National Conference on Artificial Intelligence, Sixteenth Conference on Innovative Applications of Artificial Intelligence, July 25-29, 2004, San Jose, California, USA, AAAI Press / The MIT Press, 2004, pp. 342–347. URL: http://www.aaai.org/Library/AAAI/2004/aaai04-055.php.

[14] R. Evans, E. Grefenstette, Learning explanatory rules from noisy data, Jounal of Artificial Intelligence Research 61 (2018) 1–64. doi:10.1613/jair.5714.

[15] R. Manhaeve, S. Dumancic, A. Kimmig, T. Demeester, L. De Raedt, Neural probabilistic logic programming in DeepProbLog, Artificial Intelligence 298 (2021) 103504. doi:10.1016/j.artint.2021.103504.

[16] A. Bordes, N. Usunier, A. García-Durán, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: C. J. C. Burges, L. Bottou, Z. Ghahramani, K. Q. Weinberger (Eds.), Proceedings of 27th Annual Conference on Neural Information Processing Systems (NeurIPS), Lake Tahoe, Nevada, United States, December 5-8, 2013, 2013, pp. 2787–2795. URL: https://proceedings.neurips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html.

[17] Q. Wang, B. Wang, L. Guo, Knowledge base completion using embeddings and rules, in: Q. Yang, M. J. Wooldridge (Eds.), Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI), Buenos Aires, Argentina, July 25-31, 2015, AAAI Press, 2015, pp. 1859–1866. URL: http://ijcai.org/Abstract/15/264.

[18] Q. Liu, H. Jiang, Z. Ling, S. Wei, Y. Hu, Probabilistic reasoning via deep learning: Neural association models, CoRR abs/1603.07704 (2016). arXiv:1603.07704.

[19] T. Ahmad, D. Zhang, C. Huang, H. Zhang, N. Dai, Y. Song, H. Chen, Artificial intelligence in sustainable energy industry: Status quo, challenges and opportunities, Journal of Cleaner Production 289 (2021) 125834. doi:10.1016/j.jclepro.2021.125834.

[20] D. Alvarez-Melis, T. S. Jaakkola, On the robustness of interpretability methods, CoRR abs/1806.08049 (2018). arXiv:1806.08049.

[21] S. Verma, J. P. Dickerson, K. Hines, Counterfactual explanations for machine learning: A review, CoRR abs/2010.10596 (2020). arXiv:2010.10596.

[22] F. Bodria, F. Giannotti, R. Guidotti, F. Naretto, D. Pedreschi, S. Rinzivillo, Benchmarking and Survey of Explanation Methods for Black Box Models, CoRR abs/2102.13076 (2021). arXiv:2102.13076.

[23] R. Confalonieri, T. Weyde, T. R. Besold, F. M. d. P. Martín, Using ontologies to enhance human understandability of global post-hoc explanations of black-box models, Artificial Intelligence 296 (2021) 103471. doi:10.1016/j.artint.2021.103471.

[24] R. Piltaver, M. Luštrek, M. Gams, S. Martinčić-Ipšić, What makes classification trees comprehensible?, Expert Systems with Applications 62 (2016) 333–346. doi:10.1016/j.eswa.2016.06.009.

[25] A. Canziani, E. Culurciello, A. Paszke, Evaluation of neural network architectures for embedded systems, in: IEEE International Symposium on Circuits and Systems, ISCAS 2017, IEEE, Baltimore, MD, USA, 2017, pp. 1–4. doi:10.1109/ISCAS.2017.8050276.

[26] G. Huang, S. Liu, L. van der Maaten, K. Q. Weinberger, CondenseNet: An efficient DenseNet using learned group convolutions, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, Computer Vision Foundation / IEEE Computer Society, 2018, pp. 2752–2761. doi:10.1109/CVPR.2018.00291.

[27] H. Cheng, T. Zhang, Y. Yang, F. Yan, H. Teague, Y. Chen, H. Li, MSNet: Structural wired neural architecture search for internet of things, in: 2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019, IEEE, 2019, pp. 2033–2036. doi:10.1109/ICCVW.2019.00254.

[28] Y. Wang, B. Li, R. Luo, Y. Chen, N. Xu, H. Yang, Energy efficient neural networks for big data analytics, in: G. P. Fettweis, W. Nebel (Eds.), Design, Automation & Test in Europe Conference & Exhibition, DATE 2014, Dresden, Germany, March 24-28, 2014, European Design and Automation Association, 2014, pp. 1–2. doi:10.7873/DATE.2014.358.

[29] E. H. Lee, D. Miyashita, E. Chai, B. Murmann, S. S. Wong, LogNet: Energy-efficient neural networks using logarithmic computation, in: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, IEEE, New Orleans, LA, USA, 2017, pp. 5900–5904. doi:10.1109/ICASSP.2017.7953288.

[30] W. Hou, M. Fu, H. Zhang, Z. Wu, Consensus conditions for general second-order multi-agent systems with communication delay, Automatica 75 (2017) 293–298. doi:10.1016/j.

automatica.2016.09.042.

[31] E. Tsamoura, V. Gutiérrez-Basulto, A. Kimmig, Beyond the grounding bottleneck: Datalog techniques for inference in probabilistic logic programs, in: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, AAAI Press, 2020, pp. 10284–10291. URL: https://ojs.aaai.org/index.php/AAAI/article/view/6591.

[32] I. Shumailov, Y. Zhao, D. Bates, N. Papernot, R. D. Mullins, R. Anderson, Sponge examples: Energy-latency attacks on neural networks, in: IEEE European Symposium on Security and Privacy, EuroS&P 2021, Vienna, Austria, September 6-10, 2021, IEEE, 2021, pp. 212–231. doi:10.1109/EuroSP51992.2021.00024.

[33] J. Deng, F. Ren, A survey of textual emotion recognition and its challenges, IEEE Transactions on Affective Computing (2021). doi:10.1109/TAFFC.2021.3053275.

[34] R. Sanchez-Iborra, A. F. Skarmeta, TinyML-enabled frugal smart objects: Challenges and opportunities, IEEE Circuits and Systems Magazine 20 (2020) 4–18. doi:10.1109/MCAS.2020.3005467.

[35] A. Agiollo, G. Ciatto, A. Omicini, *Shallow2Deep*: Restraining neural networks opacity through neural architecture search, in: D. Calvaresi, A. Najjar, M. Winikoff, K. Främling (Eds.), Explainable and Transparent AI and Multi-Agent Systems. Third International Workshop, EXTRAAMAS 2021, Virtual Event, May 3–7, 2021, Revised Selected Papers, volume 12688 of *Lecture Notes in Computer Science*, Springer, Cham, Switzerland, 2021, pp. 63–82. doi:10.1007/978-3-030-82017-6_5.

[36] S. Kamthe, M. P. Deisenroth, Data-efficient reinforcement learning with probabilistic model predictive control, in: A. J. Storkey, F. Pérez-Cruz (Eds.), International Conference on Artificial Intelligence and Statistics, AISTATS 2018, volume 84 of *Proceedings of Machine Learning Research*, PMLR, Playa Blanca, Lanzarote, Canary Islands, Spain, 2018, pp. 1701–1710. URL: http://proceedings.mlr.press/v84/kamthe18a.html.

[37] A. Agiollo, A. Omicini, Load classification: A case study for applying neural networks in hyper-constrained embedded devices, Applied Sciences 11 (2021). doi:10.3390/app112411957, Special Issue "Artificial Intelligence and Data Engineering in Engineering Applications".