Journal Title
XX(X):1–28

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Neuro-LENS: a neuro-symbolic framework integrating incomplete background knowledge and deep learning

Giulia Murtas[1,2], Veselka Boeva[2] and Elena Tsiporkova[1]

Abstract

In this study, we propose Neuro-LENS, a Neuro-Symbolic Evidence-based Logic and Symbolic Reasoning framework, that combines incomplete symbolic knowledge with neural learning to address ambiguity and improve the accuracy and interpretability of the results. We explore three strategies for integrating symbolic reasoning with deep learning and evaluate their effectiveness in practical settings: (i) applying the symbolic component to the neural output; (ii) generating additional neural input features through symbolic rules; (iii) creating an ensemble reasoning model. The potential of the proposed Neuro-LENS framework is demonstrated through real-world use cases, specifically image scene classification with abandoned object detection and prognostic health monitoring with vehicle failure prediction.

Keywords

Evidence measures, Modal logic, Multi-valued mapping, Deep learning

[1]EluciDATA Lab, Sirris, Ravensteinstraat 4, Brussels, Belgium
[2]Department of Computer Science, Blekinge Institute of Technology, Sweden

Corresponding author:
Giulia Murtas, Sirris, Brussels, Belgium and Blekinge Institute of Technology, Sweden.
Email: giulia.murtas@sirris.be, giulia.murtas@bth.se

Prepared using sagej.cls *[Version: 2017/01/17 v1.20]*

Introduction

Deep learning has achieved remarkable results in perception-driven tasks such as image recognition, natural language processing, and fault detection in industrial systems. However, deep learning methods still suffer from the lack of robustness, interpretability, and the difficulty of directly incorporating structured background knowledge [Mar18]. Symbolic logic, on the other hand, is apt for representing structured thought and explainable reasoning, although it struggles with scalability and perception tasks. This long-standing trade-off has motivated the development of neuro-symbolic integration, which aims to unify the learning capacity of neural networks with the structured reasoning power of symbolic systems [Bes+17].

Injecting reasoning abilities in artificial intelligence remains one of the central challenges in the field, as it would allow to enhance generalization and adaptability and produce explainable AI models which can perform logical inference, make decisions based on knowledge, and tackle structured problem solving [LWT25; BL04]. This hybrid approach has shown advantages over purely symbolic or purely neural systems, especially in real-world settings with noisy, unstructured data, as its flexibility makes it robust and well-suited for real-world AI applications [Bes+17].

Specifically in real-world industrial applications, neuro-symbolic approaches can help when dealing with noisy data and incomplete background knowledge. Traditionally, probabilistic models, such as Bayesian networks and Markov decision processes, are used to capture uncertainty and randomness in reasoning processes, while logic-based systems are exploited to model high-level reasoning and decision making. Evidence theory provides a bridge between the two paradigms, allowing to achieve high-level reasoning while dealing with uncertainty and incomplete knowledge, making its combination with deep learning approaches suited for real-world use cases.

In the current study, we propose a neuro-symbolic framework, Neuro-LENS (modal Logic and EvideNce-based Symbolic reasoning), based on evidence fusion, which integrates incomplete symbolic knowledge with neural learning in order to improve both accuracy and interpretability of the obtained results. Three strategies for integrating symbolic reasoning with deep learning in practical settings are explored:

(i) **Symbolic reasoning on neural outputs:** Applying symbolic rules to attributes extracted by neural networks to perform classification tasks;
(ii) **Feature augmentation via symbolic rules:** Using symbolic reasoning to create new features that extend the neural input space, enabling more robust predictions and the integration of background knowledge/context;
(iii) **Neuro-symbolic ensembles:** Combining decision rules derived from both neural and symbolic components into a hybrid, rule-based classifier, providing improved interpretability.

This work builds upon and develops further the methodology presented in [MBT25]. In the latter, a novel neuro-symbolic approach was introduced, integrating modal logic, evidence theory, and deep learning, for the purpose of reasoning and decision making under ambiguity. The potential of the proposed hybrid method was validated on a

real-world use case, more concretely, on image scene classification for surveillance applications. In the current study, besides further enhancement and refinement of the theoretical framework, two new additional alternative mechanisms for the integration of modal logic, evidence theory, and deep learning are also considered. Further, the aim of the current work is to demonstrate that the applicability of the proposed neuro-symbolic approach is not limited to image data scenarios and can be generalized to completely different use cases dealing with data types of very different nature, e.g., specification records or time series sensor measurements.

Our neuro-symbolic framework, Neuro-LENS, is intended to advance the broader goal of neuro-symbolic AI: *building intelligent systems that can learn from data while reasoning with structured knowledge in uncertain and dynamic environments [Fen+25].*

The current work makes the following additional contribution with respect to the paper [MBT25], which it extends:

- The approach presented in the original paper is inserted within a framework integrating deep learning and symbolic reasoning
- The theoretical background of the presented approach is extended
- Two new strategies for the integration of a neural and a symbolic component are introduced
- A completely new use case is studied for the validation of the two novel strategies

Background

Multi-valued mapping

In this section, we introduce some basic concepts from the theory of multi-valued mappings [AF90; Ber77]. A *multivalued mapping* $\mathcal{F}$ from a universe X into a universe Y associates to each element x of X a subset $\mathcal{F}(x)$ of Y. The *domain* of $\mathcal{F}$, denoted $\text{dom}(\mathcal{F})$, is defined as

$$\text{dom}(\mathcal{F}) = \{x \mid x \in X \ \wedge \ \mathcal{F}(x) \neq \emptyset\}.$$

$\mathcal{F}$ is called *non-void* if $(\forall x \in X)(\mathcal{F}(x) \neq \emptyset)$, i.e., if $\text{dom}(\mathcal{F}) = X$.

Consider a subset A of X and a subset B of Y. The following direct and inverse images can be defined under multi-valued mapping $\mathcal{F}$:

(i) The *direct image* of A under $\mathcal{F}$ is the subset $\mathcal{F}(A)$ of Y, defined as

$$\mathcal{F}(A) = \bigcup_{x \in A} \mathcal{F}(x).$$

(ii) The *inverse image* of B under $\mathcal{F}$ is the subset $\mathcal{F}^{-}(B)$ of X, defined as

$$\mathcal{F}^{-}(B) = \{x \mid x \in X \ \wedge \ \mathcal{F}(x) \cap B \neq \emptyset\}. \tag{1}$$

(iii) The *superinverse image* of B under $\mathcal{F}$ is the subset $\mathcal{F}^{+}(B)$ of X, defined as

$$\mathcal{F}^{+}(B) = \{x \mid x \in \text{dom}(\mathcal{F}) \ \wedge \ \mathcal{F}(x) \subseteq B\}. \tag{2}$$

(iv) The *subinverse image* of B under $\mathcal{F}$ is the subset $\mathcal{F}^{\sim}(B)$ of X, defined as

$$\mathcal{F}^{\sim}(B) = \{x \mid x \in X \ \wedge \ B \subseteq \mathcal{F}(x)\}.$$

(v) The *pure inverse image* of B under $\mathcal{F}$ is the subset $\mathcal{F}^{-1}(B)$ of X, defined as

$$\mathcal{F}^{-1}(B) = \{x \mid x \in X \ \wedge \ \mathcal{F}(x) = B\}.$$

A visualization of the inverse and superinverse images, used in this work, is shown in Fig. 1.

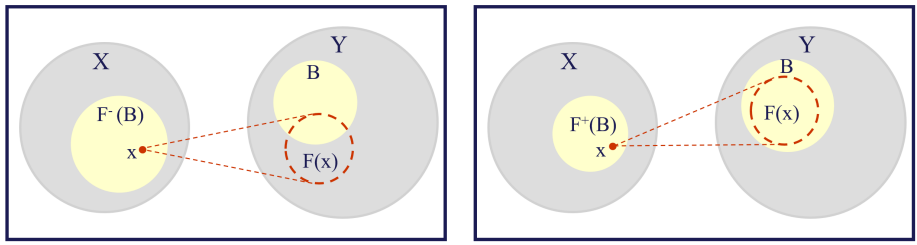

Figure 1. A visual illustration of inverse $F^{-}(B)$ (left) and superinverse $F^{+}(B)$ (right) images of a set B under a multi-valued mapping F from a set X into a set Y, which associates to each element x of X a subset $F(x)$ of Y. The figure is adapted from [MBT25].

Evidence measures

Evidence theory, also known as Dempster-Shafer theory, was initiated by Dempster with his study of upper and lower probabilities [Dem08]. He showed that if P is a probability measure on $\mathcal{P}(X)$, then a multi-valued mapping $\mathcal{F}$ from X into Y induces *upper* P^* and *lower* P_* *probabilities* on $\mathcal{P}(Y)$, as follows:

$$\begin{aligned} P^*(B) &= P(\mathcal{F}^{-}(B) \mid \mathrm{dom}(\mathcal{F})) \\ P_*(B) &= P(\mathcal{F}^{+}(B) \mid \mathrm{dom}(\mathcal{F})). \end{aligned} \tag{3}$$

It is clear that P^* and P_* are only well defined if $P(\mathrm{dom}(\mathcal{F})) > 0$. Note that P^* and P_* are dual, i.e., $P^*(B) = 1 - P_*(\mathrm{co}\, B)$.

Shafer reinterpreted upper and lower probabilities as degrees of *plausibility* Pl and *belief* Bel, abandoning Dempster's idea that they emerge as upper and lower bounds of Bayesian probabilities [Sha76]. Furthermore, in case of a finite universe Y, Shafer introduced the concepts of a basic probability assignment and its focal elements. Formally, a $\mathcal{P}(Y) \to [0,1]$ mapping m is called a *basic probability assignment* on $\mathcal{P}(Y)$ if $m(\emptyset) = 0$ and

$$\sum_{B \in \mathcal{P}(Y)} m(B) = 1.$$

A subset F of Y for which $m(F) > 0$ is called a *focal element* of m. The belief Bel and plausibility Pl measures can be defined in terms of basic probability assignment as follows:

$$\text{Bel}(B) = \sum_{C \subseteq B} m(C) \qquad \text{Pl}(B) = \sum_{C \cap B \neq \emptyset} m(C),$$

where, the corresponding basic probability assignment m is given by [Dem67]:

$$m(B) = P(\mathcal{F}^{-1}(B) \mid \text{dom}(\mathcal{F})). \tag{4}$$

Modal logic

Modal logic is an extension of classical propositional logic. It has been developed to formalize arguments that involve the notions of necessity and possibility [Che80]. These notions are often expressed using the concept of *possible worlds*: necessary propositions are those that are true in all possible worlds, whereas possible propositions are those that are true in at least one possible world. Possible worlds are abstract concepts, and it is difficult to provide a precise definition of them. Intuitively, however, we can view them as possible states of affairs, situations or scenarios.

The language of modal logic consists of a set of atomic propositions, logical connectives $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$, and modal operators of *possibility* $\Diamond$ and *necessity* $\Box$. The propositions of the language can be the atomic propositions, and if p and q are propositions, then are so $\neg p$, $p \wedge q$, $p \vee q$, $p \rightarrow q$, $p \leftrightarrow q$, $\Box p$, $\Diamond p$.

The interpretations of the Dempster-Shafer theory [TBD99; TBB00] used in this study are based on the semantics of modal logic using the concept of a standard model. A *standard model* of modal logic is a triplet $M = \langle W, R, V \rangle$, where W denotes a set of possible worlds, R is a binary relation on W called *accessibility relation*, and V is the *value assignment function* by which truth T or falsity F of each atomic proposition p in each world w is assigned. A proposition p may have different truth-values in different worlds. Therefore V assigns the truth-values not to proposition constants alone, but to pairs consisting of a possible world and a proposition constant, i.e., the value $V(w, p)$ is to be thought of as the truth-value of p in w. The value assignment function is inductively extended to all propositions in the usual way. The extension to possibilitations, i.e., propositions of the type $\Diamond p$, and necessitations, i.e., propositions of the type $\Box p$, are defined for any proposition p and any world $w \in W$ as follows:

$$V(w, \Diamond p) = T \Leftrightarrow (\exists v \in W)(wRv \wedge V(v, p) = T)$$
$$V(w, \Box p) = T \Leftrightarrow (\forall v \in W)(wRv \Rightarrow V(v, p) = T).$$

Modal logic interpretations of evidence measures

Dempster-Shafer theory is closely related to the theory of multi-valued mappings as discussed above. In several studies [BTB98; TBB00; TBD99], set-valued interpretations of plausibility and belief measures in modal logic have been proposed. The authors consider a model $M = \langle W, R, V, P \rangle$, where P is a probability measure on $\mathcal{P}(W)$.

Furthermore, the propositions have the form e_A ="*a given incompletely characterized element* ϵ *is classified in set* A", where $\epsilon \in X$ and $A \in \mathcal{P}(X)$. As atomic propositions, they consider the propositions $e_{\{x\}}$, for all $x \in X$. In addition, it is assumed that exactly one $e_{\{x\}}$ is true in each world. This implies that e_X and also $e_A \leftrightarrow \neg e_{\mathrm{co}\,A}$ are always true in M. In this context it is shown that a plausibility measure and a belief measure can be expressed in terms of conditional probabilities of truth sets of possibilities and necessities, i.e.

$$\mathrm{Pl}(A) = P(\|\Diamond e_A\|^M \mid \|\Diamond e_X\|^M)$$
$$\mathrm{Bel}(A) = P(\|\Box e_A\|^M \mid \|\Diamond e_X\|^M).$$

Related work

Neuro-symbolic approaches in literature have been leveraged to obtain interpretable systems that are robust to uncertainty while still being accurate. The integration of symbolic components alleviates the downsides of deep learning-based methods, improving their performance on reasoning tasks and providing them with explainability.

On image data, deep learning approaches have dominated the literature. However, recent advancements have shown the potential of neuro-symbolic methods in various applications, even outperforming traditional neural models in tasks like question answering and image classification [Fit25].

Neuro-symbolic approaches have shown great value specifically in safety-critical fields such as surveillance, medical imaging, or autonomous systems, where reasoning is paramount. In [Lu+25], Logical Neural Networks (LNNs) are used to combine learnable parameters with logical operators. The networks incorporate first-order logic and is able to learn rule thresholds and weight from the training data.

In [Wan+23], the issue of lack of annotated image data is tackled. The work combines a pre-trained computer vision model which extracts features from the unlabeled images, and an inductive logic learner module inferring logic-based rules that can be exploited for the annotation. A human in the loop is queried to confirm the labeling of uncertain samples and to improve the derived logic-based rules. The study delivers promising results, but the reached accuracy is not yet on par with the labeling of human experts, on which it still relies for feedback in the active learning portion of the method pipeline.

Evidence theory is usually leveraged in the symbolic component of integrated systems to deal with uncertainty in the data. In [Zha+23], it is used to re-label the training set, assigning ambiguous images to a meta-category, i.e., a subset of all possible categories, by selecting the meta-category with the highest degree of belief for each selected image. Ambiguous images are defined as samples showing features of multiple classes. The model is re-trained on the dataset updated with meta-categories, so that the model can learn without overfitting to incorrect labels or misclassified examples.

The application of neuro-symbolic approaches to time series is also a challenging task that is being extensively researched. Time-series data are central to applications ranging from finance and healthcare to manufacturing, autonomous driving, and traffic management. In safety critical domains such as medicine and public security, interpretability in models is fundamental and only trustworthy approaches are likely to be

adopted. Thus, black-box deep learning models need to be enhanced with explainability features. Post-hoc methods such as SHAP ([LL17]) can provide an explanation of the model's output based on the input features that were most influential in a prediction, but do not really aid in understanding the underlying model mechanism.

Neuro-symbolic frameworks can reach intrinsic interpretability while balancing an accuracy trade-off in the final results. Neuro-symbolic rule-based approaches, such as [Wan+25], have been investigated in this regard. In [Wan+25], a model called TemporalRule is proposed to automatically learn Signal Temporal Logic rules for interpretable time series classification. The work aims at solving the discrepancy between discrete logical rules and continuous neural networks, which might make generated logical rules inconsistent with the decision process that needs to be carried out, while having an approach that takes the temporal properties of the data into account. Here, the input time series is represented in three views: raw data, frequency-domain features, and derivative (rate of change between subsequent points), each capturing different temporal properties. After having been binarized, the inputs are passed to a Temporal Logical Layer, where temporal operators (Always, Eventually, Until, and their combinations) are simulated using small neural networks. A Logical Layer combines temporal predicates using logical connectives (AND/OR), and a final Linear Layer assigns weights to the learned rules and generates the classification output. So far, the method has only been tested on univariate time series.

Dhont et al. ([DMT25]), again put an emphasis on interpretability, employing a hybrid modelling framework for traffic dynamics forecast in terms of humanly interpretable traffic states. The work proposes three different workflows: a purely neural approach leveraging CNNs or RNNs, a neural-to-symbolic one where a deep learning model predicts current traffic state probabilities, which are then fed into Markov chains for the forecast, and a symbolic-to-neural one, where the raw signals are predicted into traffic state sequences, which form the input for a deep neural predictor performing the forecast. The purely neural model achieved the highest accuracy; the neuro-symbolic models, while performing slightly worse in accuracy, provide interpretability, computational efficiency, and easier adaptability. In addition to [Wan+25], the workflows are applied to multi-variate time series. The sequential nature of the proposed neuro-symbolic approaches makes them subject to a possibly compounding error; in the symbolic-to-neural models in particular, the final performance is highly dependent on the quality of the initial state detection step.

Hogea et al. ([Hog+24]) integrated logical rules into recurrent neural networks to improve interpretability and accuracy in fault diagnosis of gearboxes. The authors introduced LogicLSTM, which adds an Explainability Layer and a Logic Tensor Network (LTN) on top of a pre-trained LSTM model. The Explainability Layer reweights the features based on feature importance, forcing the model to focus more on signals which are relevant for the task; in the LT, logical rules derived from domain knowledge are introduced, and the network is further trained to maximize both predictive accuracy and logical consistency with the provided constraints. The method is best suited for scenarios where there is prior knowledge about the relationship between classes or numerical values. LogicLSTM performed better than the presented purely neural

baselines, confirming the effectiveness of the addition of symbolic constraints to enhance model robustness in noisy environments. However, the method's performance is critically impacted by the number of available samples within each considered sequence of data; moreover, manual intervention seems to be required to define the leveraged logical rules.

Method: Neuro-LENS Framework

In this section, we provide a detailed explanation of the two main components (symbolic and neural) of our neural-symbolic approach, Neuro-LENS, which was initially proposed in [MBT25]. We also explain how these components can be integrated into a neuro-symbolic learning framework to tackle different use case scenarios.

Neural-symbolic systems are characterized by *modularity* and *hierarchical organization*. Modularity relates to the construction of a neural-symbolic network as an ensemble of neural networks. As stated in [Bes+17], modularity greatly contributes to the comprehensibility and maintenance of a framework, as it allows to work with relatively simple components which are combined into an expressive method. Moreover, each module can be modified or substituted on a use case basis, depending on the type of data and task configuration, resulting in greater flexibility and generalization potential. Hierarchical organization means that each subsequent network level uses the output of the preceding level as input, thus increasing the abstraction level of the model [GLG09].

The proposed Neuro-LENS approach is also characterized by modularity, achieved through the combination of neural and symbolic modules. Hierarchical organization can also be considered in terms of how the two components (neural and symbolic) are integrated. The two components are discussed in Sections **??** and .

Three different variations of the hierarchical organization of the two modules are proposed. The first strategy was first presented in [MBT25]. The two additional alternative strategies presented in the current study are novel extensions of [MBT25]. A high-level schematics of the three approaches can be seen in Fig. 2

Symbolic component

The symbolic component exploits modal logic and evidence theory in order to extract measures to quantify the uncertainty embedded in the raw data itself or in the available background knowledge. In brief, binary attributes of the considered samples are extracted to construct logical constraints that need to be satisfied by a sample to belong to a certain class, with a degree of uncertainty specified by its plausibility and belief measures. These measures can be used as such or combined into a single score, leveraged directly for interpretable classification, or fed to a neural network for further processing. A schematic view of the steps followed within this component can be seen in Fig. 3.

More concretely, in this component, multi-valued interpretations of upper and lower probabilities in modal logic are employed in order to reason within ambiguous scenarios. Consider a set of entities (objects) Y described by a set of attributes (properties) X. Each entity may have multiple properties, and a property may be associated with multiple entities. In addition, the entities in Y are distributed across c different categories (classes),

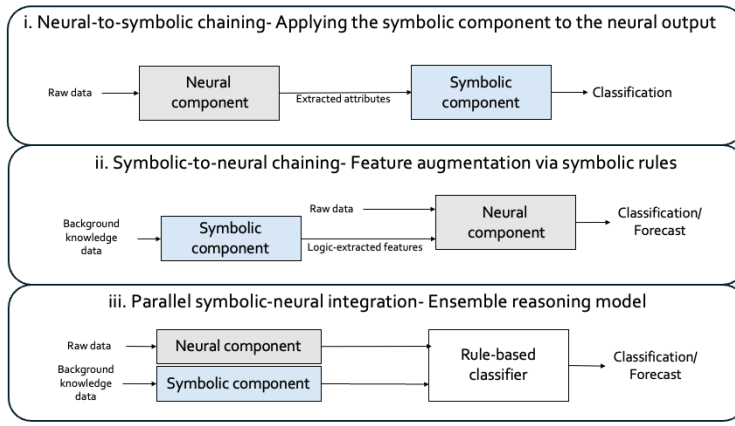

Figure 2. Three strategies for integrating neural and evidence-based (symbolic) components: (i) The neural component extracts attributes for use by the symbolic component; (ii) The symbolic component generates additional input features for the neural component; (iii) Both the symbolic and the neural components are used to extract inputs for a rule-based classifier.

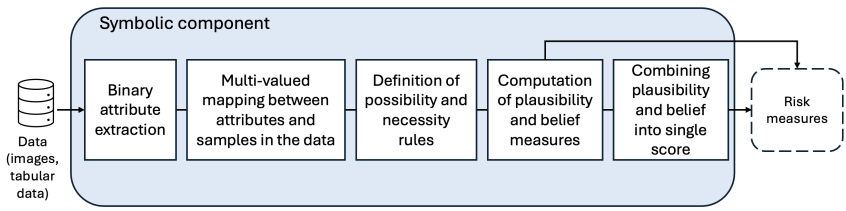

Figure 3. The symbolic component calculation pipeline.

i.e., $Y = \bigcup_{i=1}^{c} Y_i$, where $Y_i \subset Y$ and $Y_i \cap Y_j = \emptyset$, for $i \neq j$. In this scenario, our aim is to interpret each class in terms of its associated properties in such a way as to enable automatic recognition of the most probable class of a new, unseen entity described by its properties.

In the above context, we can define a multi-valued mapping $\mathcal{F}$ from the set of properties X to the set of entities Y. This mapping associates each property $x \in X$ with a set of entities $\mathcal{F}(x) \subseteq Y$ that possess it. The properties are defined as binary attributes that an entity can satisfy or cannot satisfy.

In the general case of multi-class classification, the mapping $\mathcal{F}$ is exploited to characterize each class Y_i, for $i = 1, 2, \ldots, c$, in terms of its possibility and necessity conditions, by constructing inverse and superinverse images of the class as defined in (1) and (2). Formally, the necessity and possibility conditions referring to class Y_i can be described by the following two expressions:

$$\Box Y_i = \bigvee_{x_j \in \mathcal{F}^+(Y_i)} x_j \quad \text{and} \quad \Diamond Y_i = \bigvee_{x_j \in \mathcal{F}^-(Y_i)} x_j. \tag{5}$$

Intuitively, a property x_j contributes to the possibility condition of a class Y_i, if at least one entity in its direct image $\mathcal{F}(x_j)$ satisfies this property of class Y_i. Similarly, a property x_j contributes to the necessity condition of class Y_i if all entities in its direct image under the function $\mathcal{F}$ satisfy this property of class Y_i. This reasoning is repeated for all defined properties and the final possibility and necessity conditions for class Y_i are defined as the disjunction of all single properties contributing to each of them.

The inferred possibility and necessity conditions of the classes defined in (5) can be used to reason about, and eventually predict, the most probable class of unseen entities, based on their properties.

In addition, we can compute the plausibility and belief that each new unseen entity belongs to each class Y_i, for $i = 1, 2, \ldots, c$. The plausibility (Pl_i) and belief (Bel_i) that an entity presented by a set of properties X_j belongs to class Y_i are computed as the ratio of instances that satisfy the possibility and necessity conditions of the class, as follows:

$$\text{Pl}_i(X_j) = \mid \Diamond Y_i(X_j) \mid / \mid \Diamond Y_i \mid \quad \text{and} \quad \text{Bel}_i(X_j) = \mid \Box Y_i(X_j) \mid / \mid \Box Y_i \mid. \tag{6}$$

The calculated plausibility and belief values can be used to extend the feature set in the proposed integration strategy (ii) discussed in Section .

These values can also be combined to calculate a single score for each entity-class pair. Namely, a scoring function S can be defined which combines the plausibility and belief measures for all classes, producing a value in the interval $[0, 1]$ that can be interpreted as the likelihood of an entity X_j to belong to a certain class.

Two alternative approaches for constructing S are proposed in [MBT25], see (7) and (8) below, where they are applied to a binary classification task. Specifically, the calculated beliefs and plausibilities for a given entity X_j with respect to two classes, positive ($+$) and negative ($-$), can form two intervals, $[\text{Bel}_+(X_j), \text{Pl}_+(X_j)]$ and $[\text{Bel}_-(X_j), \text{Pl}_-(X_j)]$. The width of these intervals is correlated with the uncertainty associated with X_j. Inspired by the work of [BD04], S can be based on the degree to which the two intervals overlap.

$$S(X_j) = \begin{cases} 1 & \textit{if } \text{Bel}_+(X_j) \geq \text{Pl}_-(X_j) \\ 0 & \textit{if } \text{Bel}_-(X_j) \geq \text{Pl}_+(X_j) \\ \dfrac{\text{Pl}_+(X_j) - \text{Bel}_-(X_j)}{\big(\text{Pl}_+(X_j) - \text{Bel}_-(X_j)\big) + \big(\text{Pl}_-(X_j) - \text{Bel}_+(X_j)\big)} & \textit{otherwise}\ . \end{cases} \tag{7}$$

Alternatively, S can defined as ratio of available evidence supporting the positive class:

$$S(X_j) = \frac{\text{Pl}_+(X_j) + \text{Bel}_+(X_j)}{\big(\text{Pl}_+(X_j) + \text{Bel}_+(X_j)\big) + \big(\text{Pl}_-(X_j) + \text{Bel}_-(X_j)\big)}. \tag{8}$$

The definition of the scoring function of (8) is used in the new applications presented in the current paper.

Neural component

The symbolic component, which is based on modal logic interpretations of evidence theory, can be combined with a deep learning model following one of the integration strategies depicted in Fig. 1. The neural paradigm to be used needs to be selected based on the type of data to be processed, the scope, and requirements of the considered use case. Pre-trained, off-the-shelves models or customized models can be employed.

The first integration strategy (see Section **??** and [MBT25] for more details), employs the neural component to extract attributes from the raw data. The strategy was validated on image data, for the purpose of abandoned object detection. Subsequently, the chosen neural components are: 1) a pre-trained OneFormer model [Jai+23], returning the classes of the detected objects together with the coordinates of the bounding boxes indicating where each object can be found in the image; 2) the Depth Anything V2 model [Yan+24], used to estimate the depth of the detected objects.

In the second integration strategy, the neural component is used to process the raw data and at the same time is fed with the output of the symbolic component, containing information about the available partial background knowledge. Again, the specific neural model can be chosen based on the data type and use case at hand, as the approaches are made to be modular. As the use case presented in the current paper deals with time series data, a Long Short-Term Memory (LSTM) network is used, able to effectively learn long-term dependencies in sequential data.

The third integration strategy leverages both symbolic and neural components to extract features on which a rule-based classifier will pose constraints to obtain the final results. In the presented use case, again the deep learning model is applied to time series data. Thus, an autoencoder LSTM is applied to the data, in order to obtain a reconstruction error for each sample, indicating how anomalous the recorded sensor data are at each given point in time. In our specific application, this also allows to overcome the discrepancy between the discrete labels provided in the ground truth, and the continuous nature of the observed phenomenon.

Neuro-LENS: Neuro-symbolic integration

In the current section, we discuss the different ways in which background knowledge can be expressed in an evidence-based language and integrated into a neural / deep learning (DL) component to improve model performance. The proposed Neuro-LENS framework considers three different ways of combining the two components (symbolic and neural) as described in the foregoing subsection and further formalized below:

(i) *Neural-to-symbolic chaining: Applying the symbolic component to the neural output*. This approach is relevant when the type of data available cannot be immediately used by the symbolic component as such (e.g., images), and needs to be transformed into a suitable representation first. The neural component can be used for this purpose. Practically, the two components are chained one after the other. A DL model suited for the use case in question is initially applied to the raw data, with the aim of extracting features that can be used as input for the symbolic

model described in Subsection . The symbolic model produces both logical rule and a score, that can be used to perform either rule- or score-based classification, as demonstrated in [MBT25].

(ii) *Symbolic-to-neural chaining: Generating additional neural input features through symbolic rules*. This strategy is valuable when the use case requires the integration of information coming from different types of data, e.g., time series sensor measurements complemented with some background knowledge as configuration specifications or log events. In a real-world industrial setting, background knowledge datasets contain very relevant information about the phenomenon of interest, but unfortunately their actual usage is often compromised by the difficulty of integration or by the high degree of ambiguity typically present in such datasets. Our symbolic model, being based on the interpretation of evidence theory in modal logic, can be of use for the integration, while efficiently dealing with uncertainty and ambiguity. Here, the symbolic component is initially applied to background knowledge datasets in order to derive relevant features, which can be subsequently used to enhance the already existing features to be fed to a suitable DL model for training. Our validation study on the Scania use case supported the potential of this integration scheme.

(iii) *Parallel symbolic-neural integration: Creating an ensemble reasoning model*. In this integration scheme, both components extract features from the data in parallel. The features are subsequently used to construct decision rules. This allows to handle the different data types in a differential fashion by employing the most suitable modeling paradigm, to obtain a composite model which still benefits from the interpretability of the symbolic component.

Experiments and evaluation

In our experiments, we have used real-world datasets to simulate two use case scenarios: image scene classification with abandoned object detection, and prognostic health monitoring with vehicle failure prediction.

Datasets

Image scene classification use case

The first integration strategy, presented in [MBT25] and evaluated on image data, is validated on the datasets PETS2006 and AVS2007, both containing videos depicting abandoned luggage scenarios.

The PETS2006 dataset contains videos with multi-sensor sequences depicting scenes of a luggage being abandoned inside a train station. Static frames are extracted from the videos in order to apply the proposed approach. Ground truth is not available, neither for the object detection task or the abandoned bag scene classification task. Labels indicating whether the represented scene contains an abandoned bag have been manually identified and created. The dataset consists of 1325 images, of which 95% do not depict an abandoned object, while in the remaining 5% an abandoned bag can be detected.

The AVS2007 dataset (Advanced Video and Signal Based Surveillance) provides benchmark datasets for testing and evaluating detection and tracking algorithms. The i-LIDS bag subset of AVS2007 is considered, as it consists of abandoned luggage scenarios. The dataset comprises of 161 images, 14% of which shows an abandoned object. Again, labels indicating whether an abandoned bag is present in the image have been manually added to the data.

Truck failure prediction use case

The remaining two integration strategies are applied on time series data and validated on the Scania dataset [Kha+25]. The Scania dataset is a real-world, multivariate dataset of time series collected from a single engine component across a fleet of SCANIA trucks. All data is anonymized. The dataset contains: operational data collected by onboard sensors; repair records, which include information about maintenance, repairs, and servicing performed on the vehicles; specifications of the analyzed component, collected with the production system, such as engine type, weight capacities, dimensions, and other technical details. The operational data are stored as multi-variate time series where the time steps are chronologically sequential but do not have a specified duration, and the amount of time they encompass can vary from one truck to another. In the data collection process, 14 attributes were selected and anonymized. The variables are organized into single numerical counters and histograms with several bins, each bin representing certain conditions linked to the values observed within the measured features. The dataset is highly unbalanced, as most featured trucks do not experience a fault. The table below shows the percentage of trucks that did or did not require maintenance in the training, validation, and test set. The high imbalance of the set constitutes a challenge when training a model on the data.

Dataset	Healthy trucks (%)	Faulty trucks (%)
Train set	90.4	9.6
Validation set	97.3	2.7
Test set	97.2	2.8

Table 1. Percentage of healthy and faulty trucks.

The measures taken to anonymize the dataset make working with the data cumbersome at times, a challenge that is added to the high unbalance between failing and non-failing trucks, and to the intrinsic complexity and uncertainty present in all real-world data from industrial contexts. A schematic example of the characterization of a truck with anonymized specifications and categories is shown in Fig. 4.

The class labels provided for the dataset distinguish 5 classes, depending on how much time is left until failure:

- Class 0: more than 48 hours left until failure
- Class 1: between 48 and 24 hours until failure
- Class 2: between 24 and 12 hours until failure
- Class 3: between 12 and 6 hours until failure
- Class 4: less than 6 hours until failure

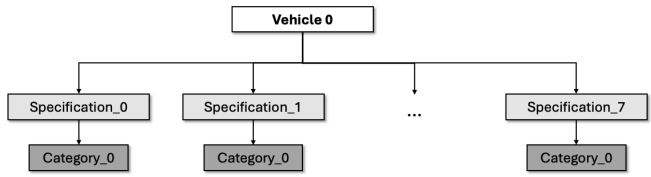

Figure 4. Specifications characterizing a single vehicle.

Below, the distribution of the last readouts of the trucks in the training set is shown. Again, the the dataset appears to be highly imbalanced.

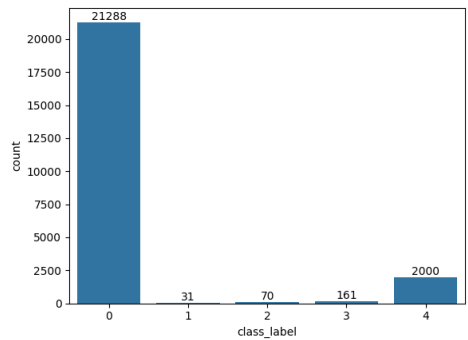

Figure 5. Distribution among classes of the last readouts of the trucks in the training set.

Image scene classification: Neural-to-symbolic chaining

The current section describes the application of a framework exploiting the neural-symbolic chaining strategy, which was evaluated on the two presented image datasets, in the context of an abandoned luggage detection use case [MBT25]. We consider a binary image scene classification task, aimed at understanding whether a frame taken from surveillance videos contains an abandoned luggage. Thus, a set of positively- and negatively-labeled images is given, representing, respectively, images in which an abandoned luggage is not depicted, or images in which it is. The proposed framework first characterizes the two classes by extracting attributes (also called instances in the cited paper) through the neural component; then, the symbolic component builds a classification model by learning a mapping from the extracted attributes to the set of possible labels. The process can be summarized in three stages, as seen in Fig. 6:

1. The given set of labeled images is initially fed to an object detection and a depth estimation models, both pre-trained; this constitutes the neural component. The object detection model returns the classes and bounding boxes of the objects of interest that were detected in the images; the depth estimation model provides a

pixel-wise measure for the depth of the object in the images. From these outputs, we can derive attributes meaningful for the use case at hand, which characterize the input images and form a set of instances.

2. A multi-valued mapping between the set of instances and the images to be categorized is constructed, by associating each instance with the set of images in which it appears.
3. Each class is described in terms of its necessity and possibility conditions or its plausibility and belief values, computed through the multi-valued mapping. To exemplify, the necessity conditions for the positive class describe the attributes an image has when it depicts a scene *necessarily* containing an abandoned luggage. The possibility conditions for the positive class specify the attributes an image has if it depicts a scene *possibly* containing an abandoned luggage. The plausibility and belief measures are computed as indicated in (6). Necessity and possibility conditions are exploited to define decision rules in a rule-base classifier, while plausibility and belief are used to construct a scoring function and perform a score-based classification.

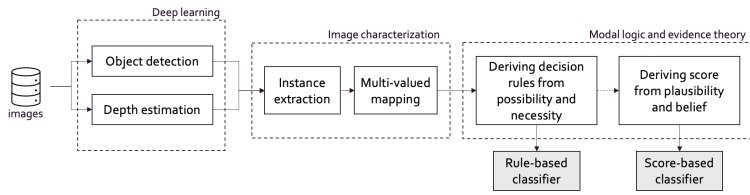

Figure 6. A schematic illustration of the first integration strategy as applied for image scene classification.

A schematic illustration of the strategy workflow is provided in Fig. 6. More concretely, in the first phase of the framework, the **neural component** is applied to the raw input data (images). Within this component, two pre-trained DL models are used to detect the objects of interest (in this case, people and bags) and obtain an estimation of their depth in the image. A pre-trained OneFormer model [Jai+23] returns the classes all objects detected in each image, together with the coordinates of the bounding boxes indicating the location of each object; a pre-trained Depth Anything V2 model [Yan+24] then estimates the depth of each pixel in the image, allowing to more accurately place objects in a 2D image. Fig. 7 shows the output of the two deep learning models on an example image.

The outputs of the neural components are used to derive attributes characterizing the images in a meaningful manner for the use case of abandoned luggage detection. Information about the people and luggage depicted in the image and the relationship between them need to be extracted to be fed to the symbolic component for reasoning. The overlap between bounding boxes and the distance between a bag and the person closest to it are computed. Here, the distance between two objects is intended as the

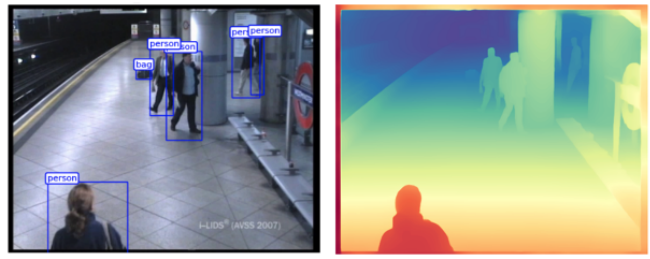

Figure 7. Example of object detection (left) and depth estimation (right) results, taken from [MBT25]

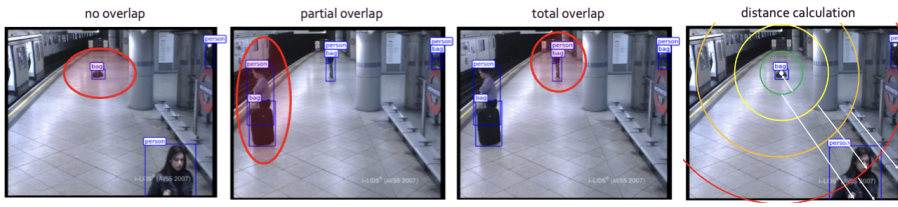

Figure 8. Overlap types and distance calculation for a selected bag, taken from [MBT25]

.

distance between the centers of their bounding boxes while considering the estimated depth of each object, i.e., a 3-dimensional Euclidean distance is calculated. The calculated distances are binned into five overlapping ranges formed by increasing the radius of concentric circles with the bag of interest in their center. A visualization of the distance calculation and the attributes indicating the possible overlap options between a detected bag and person is provided in Fig. 8.

All extracted attributes are binary, to be suited for usage by the symbolic component.

Within the **symbolic component**, a multi-valued mapping $\mathcal{F}$ is constructed between each attribute, or instance x in X and a subset of images in Y which satisfy that instance. For instance, $\mathcal{F}$ maps the instance "has_no_overlap" with all the images in the dataset where no overlap is present between the bounding box of a person and the bounding box of a bag. Then, the inverse and superinverse images of the positive and negative classes under $\mathcal{F}$ are constructed to define their necessity and possibility conditions. The obtained conditions can be seen in Table 2.

As shown in Table 2, eight instances contribute to the discrimination between the two classes in the PETS2006 dataset (one fewer in the AVS2007 dataset). The ambiguity aspect is captured by the instances which are common to the two classes, indicated below by their index: $ambiguous\_evidence = \Diamond\{abandoned\} \cap \Diamond\{non\text{-}abandoned\} = \{0, 1, 5, 7, 8\}$. Thus, we can represent the possibilities of the two classes as shown below.

$$\Diamond\{abandoned\} = \Box\{abandoned\} \vee ambiguous\_evidence$$

Table 2. Inverse ($poss_+$ and $poss_-$) and superinverse (nec_+ and nec_-) images for the two classes.

attributes	$poss_+$	$poss_-$	nec_+	nec_-
0: contains_bag	**True**	**True**	False	False
1: contains_person	**True**	**True**	False	False
2: contains_person_but_no_bag	False	**True**	False	**True**
3: has_partial_overlap	**True*** | $False$	**True**	False	False* | **True**
4: has_total_overlap	False	**True**	False	**True**
5: has_no_overlap	**True**	**True**	False	False
6: min_distance_below_0.1	False	**True**	False	**True**
7: min_distance_above_0.1	**True**	**True**	False	False
8: min_distance_above_0.25	**True**	**True**	False	False
9: min_distance_above_0.5	**True**	False	**True**	False
10: min_distance_above_0.75	**True**	False	**True**	False

* These are the values for the AVS2007 data set. All other values are the same for both datasets.

$$\Diamond\{\textit{non-abandoned}\} = \Box\{\textit{non-abandoned}\} \vee ambiguous\_evidence.$$

Then, the decision rules exploited by the rule-based classifier are defined. Below, in (9), the decision rule for the positive class is shown. An image is assigned to the positive (negative) class if the instances representing it satisfy the necessity conditions for the positive (negative) class.

$$\begin{array}{l} \text{IF } \Box X_+(X_i) \text{ THEN } X_i \in \text{positive class} \\ \text{IF } \Box X_-(X_i) \text{ THEN } X_i \in \text{negative class,} \end{array} \tag{9}$$

where $\Box X_+$ and $\Box X_-$ are the possibility and necessity conditions of the two classes, respectively. Consequently, in the context of our use case, Table 2 can be used to define the decision rules for the two classes as follows:

$$\begin{array}{l} \text{IF } (9 \vee 10) \text{ THEN } x \in \{\textit{abandoned}\} \\ \text{IF } (2 \vee 3 \vee 4 \vee 6) \text{ THEN } x \in \{\textit{non-abandoned}\}. \end{array}$$

If an image does not satisfy either decision rule, it is assigned to a "none of known" class, in order to avoid misclassifications.

The necessity and possibility are further exploited to compute the plausibility and belief values for the two classes, using (6). These values are then combined into a single score using either (7) or (8). The computed scores focus on the positive class, indicating the likelyhood of an image to contain an abandoned luggage.

Truck failure prediction

In this section, the applications of symbolic-to-neural and the parallel neural-symbolic integration strategies are presented. Both are evaluated on the SCANIA dataset, with the aim of predicting failures in heavy vehicles.

The same symbolic component is used in the two strategies; thus, it is only presented once, in the paragraphs that follow.

The **symbolic component** aims at deriving the predisposition to failure of a vehicle from its technical characteristics. Each vehicle is described by 8 specifications; for each specifications, a category is indicated. An example can be seen in Fig. 4. Within the symbolic component, the performed classification is seen as a binary task: the negative class includes all vehicles which did not encounter a failure during the observed time; the positive class includes vehicles which encountered a failure. The difference between the 5 classes listed when presenting the dataset is not considered here. The predisposition to failure is computed for the test set based on the training and validation sets of the SCANIA dataset. The results obtained by the component are further validated by comparing the obtained score with the risk analysis bias reported in [FTB24]. The paper performs survival risk analysis on the dataset; in particular, it extracts a risk score based on specification data which computes the failure predisposition of a vehicle using Cox Proportional Hazard analysis and survival trees. As the authors do not compute the bias on the test data, this comparison was only possible on the training set. Similarly to the already seen neural-to-symbolic strategy, the steps carried out within the symbolic component are those seen in Fig. 3.

The binary attributes needed for the approach are extracted by listing all possible combinations of specification and category. Each combination represents a technical characteristic that a vehicle can either have or not have. In order to take into account interactions between different specifications, all possible pairs and triplets of specifications are also included in the attributes. The process can theoretically be extended to bigger groups of specifications, but the needed computational time quickly explodes. Examples of attributes satisfied by a vehicle for single and pairs of specifications are shown in Fig. 9.

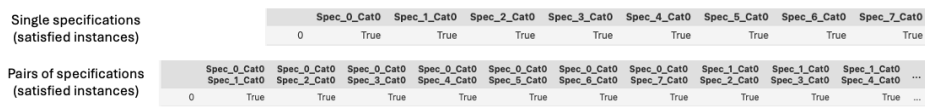

Figure 9. Example of binary instances leveraging a single specification or pairs of specifications

Subsequently, the multi-valued mapping described in **??** is constructed between the set of extracted attributes and the set of vehicles in the training dataset. The inverse and superinverse images of the constructed mapping specify which attributes need to be satisfied by a vehicle for it to be possibly or necessarily predisposed for failure.

Based on the ratio of discovered possibility and necessity rules that are satisfied by a sample, plausibility and belief measures, respectively, can be computed. These measures convey the information provided by the possibility and necessity rules, in a more granular manner.

The plausibility and belief measures can directly be taken as output of the symbolic component and exploited in the following steps of the pipeline; alternatively, they can be combined into a single score, using (8).

Experiments are performed using either plausibility and belief for the positive class (trucks presenting a failure) or using the combined score, indicating a vehicle's likelihood to have a failure based on its characteristics. The discriminatory potential of the measures is first verified by predicting failing and non failing trucks in both training and test set by using only the information from the specification data; then, their contribution to the overall performance of the approach in which they are used.

Symbolic-to-neural chaining

This section presents the second integration strategy in Fig.2. Within this strategy, the output(s) of the symbolic component is used as additional feature(s) for the neural component. The additional features aim to provide the neural component with the background information contained in the data which are not directly fed to the neural network. As anticipated, the strategy is applied to the SCANIA dataset, with the goal of predicting failures in trucks. In the use case at hand, the features are extracted from the specifications dataset, where the technical characteristics of the observed vehicles are indicated. The output of the symbolic component, thus, provides the deep learning model with information about the predisposition of a truck to failure, based on its technical characteristics. The characteristics of a vehicle are not the only factor at play when leading to a possible failure, thus introducing uncertainty and ambiguity in the provided information, dealt with by the usage of evidence theory. The sensor data, in the form of time series, are processed directly by the neural component. The extended set of features including the raw time series data and the features conveying background knowledge information are fed to the neural component in order to obtain a more accurate final prediction.

In the considered use case, the **neural component** exploits an LSTM model, suited for handling sequential data. The model takes windows of 12 time steps as input and predicts one of the 5 class labels for each of the 12 future time steps. The missing data in the training set are handled by performing forward filling, and the training and test set are defined in the same way as for the symbolic component. In order to validate the contribution of the symbolic component's output, the model is first trained on the sensor data only, including 105 features per timestep. In Fig. 13 in the Results section, this model is indicated with the label "Sensor data".

The experiment is then repeated by adding the features produced by the symbolic component, describing the predisposition of a vehicle to failure through its plausibility, belief, and combined score measures. The usage of plausibility and belief or the combined score is investigated. The addition of the features is done in two alternative ways:

- The features are simply concatenated to the input of the model; the same value is repeated at each time steps, as the outputs of the symbolic model represent a static property of the vehicles. The models where this approach was used are indicated with "concatenated" in Fig. 13;
- The features are used to set the LSTM initial state, allowing to treat them as a context rather than features replicated across all time steps. Additionally, a gate is added to the static features in order to avoid their influence to dominate over the

time-series signal. In this manner, the network can learn how much weight to give the static features. The models indicated with "gated" in Fig. 13 have been trained in this way.

Parallel symbolic-neural integration

In the third explored strategy for the integration of a neural and a symbolic, evidence-based component, both components are used to generate features, which are then combined in a rule-based model exploiting logical rules to detected failures in the observed vehicles. The symbolic component is the same as the previous strategy. Within the neural component, an LSTM-autoencoder is employed to individuate anomalous trucks. The model is trained on entire data sequences from normal vehicles, i.e. vehicles that do not experience a failure during the monitored time period. Data from 80% of normal vehicles in the training set are used for training set; the data from the remaining 20% of non-failing trucks is joined with the data from trucks which underwent maintenance at the end of the monitored period, to form the validation set. The autoencoder is trained to reconstruct sequences of sensor data. It takes windows of 48 time steps as input. In Fig. 10 the average evolution of the reconstruction error on the thus defined validation set, is shown. The plot distinguishes between failing and non failing vehicles; moreover, the 5 classes defined in the dataset labels can be seen in the graph. It can be noted how the difference between failing and non failing trucks can be seen already at about 60 hours from a failure. Moreover, the reconstruction error of non-failing trucks also tends to increase in time. This confirms the inconsistency between the used discrete class labels and the phenomenon of health degradation in a vehicle. However, it must be noted that although the difference between failing and non-failing vehicles can be clearly seen when looking at the average values per time step, there is a significant overlap between the two classes when looking at the values for single trucks.

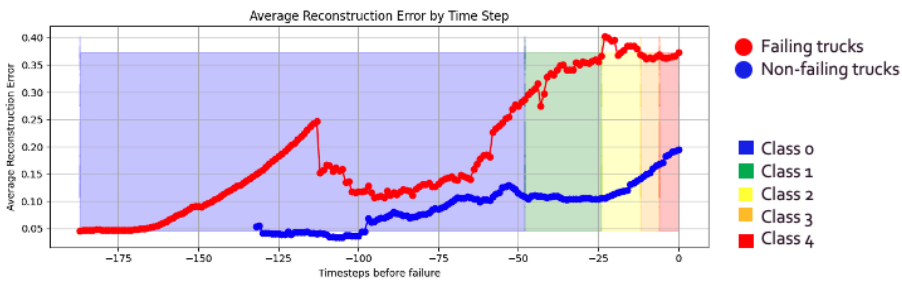

Figure 10. Evolution of the LSTM-autoencoder reconstruction error over time.

Once the reconstruction error has been obtained, together with the evidence metrics returned by the symbolic component, the following logic rule is leveraged by the classifier:

IF $reconstruction\_error > x$ *and* $(Pl_+ > y$ *or* $Bel_+ > 0)$ THEN $failure\_detected$,

where x and y are chosen based on the validation data distribution.

Results and discussion

Image scene classification

The strategy is applied to the two presented image datasets: AVS2007 and PETS2006. The datasets are split into training and test sets with proportion 80/20, with the ratio of images belonging to each class kept fixed in the division. The extracted decision rules are leverages to construct a rule-based classifier which assigns every image to its class according to the decision rule it satisfies. A ”none of known” class is also created for ambiguous images which do not satisfy either decision rule. In Table 3, the averaged results of the classifier over 20 iterations are shown. No sample is misclassified; thus, we report as metric the percentage of samples which are considered ambiguous by the model.

Table 3. Performance of the rule-based classifier on the two datasets.

Metric (%)	AVS2007	PETS2006
Detected positives	50	76.7
Detected negatives	60.2	97.7
Positives in ”none of known”	50	23.3
Negatives in ”none of known”	39.8	2.3
Overall in ”none of known”	41	3.5

The scores obtained by combining plausibility and belief using Eq. 7 and Eq 8 are exploited by a score-based classifier. The scores are named Interval-based Abandonment Risk (IAR) and Abandonment Risk (AR) in the shown results. The computed scores quantify the likelihood of an image to depict abandoned luggage. In Fig. 11 the ROC curves of the classifiers on the two datasets are shown, illustrating the variation in the model's performance when varying the threshold.

As it can be seen, both IAR and AR perform well on the PETS2006 dataset. However, AR performs better on the AVS2007, appearing to be more robust to data scarcity and complexity. In fact, the AVS2007 dataset is much smaller than the PETS2006 dataset and contains more complex scenarios.

In both rule-based and score-based classifiers, the proposed approach shows to have discrimination potential in distinguishing between scenes depicting abandoned luggage or not. The rule-based classifier allows to avoid misclassifications and signals uncertain scenarios which cannot be assigned to a class due to lack of evidence. The score-based classifier quantifies the risk of an abandoned object being present in a scene, providing a more granular expression of the level of risk of a depicted situation and the possibility of setting a threshold to trigger alarms for high-risk scenarios. The neural-to-symbolic chaining in the framework allows to leverage pre-trained DL models to extract attributes from images, used to obtain robust logical rules through the usage of modal logic and evidence theory. The resulting approach is significantly less sensitive to data scarcity and imbalance than fully DL-based methods, validating the value of neural and symbolic integration. As the performance of the object detection DL model poses an upper bound to the performance of the overall framework, it is important to ensure that the models

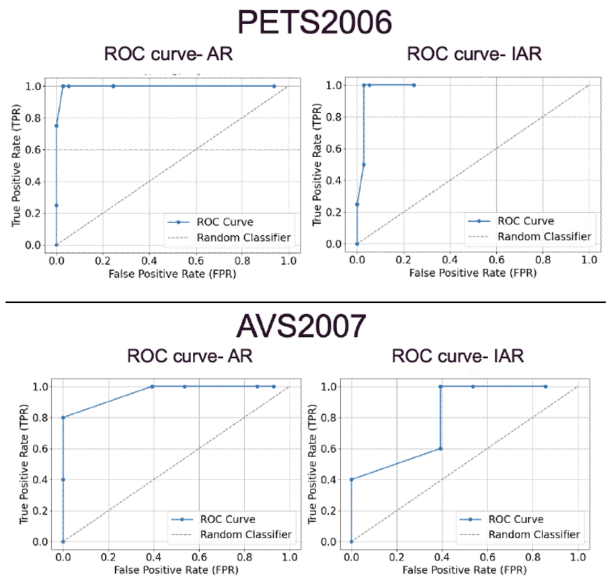

Figure 11. ROC curve of the score-based classifier for the two datasets (IAR and AR scores).

perform sufficiently well on the considered dataset. If labels for object detection are available in the training set, the ground truth can be exploited when extracting attributes, to avoid adding uncertainty due to incorrect or missed detections to the framework's results.

Truck failure prediction

Symbolic component

The ROC curves in Fig. 12 visualize the discriminatory power of the metrics calculated in individual vehicles in which a failure was recorded during the observed time interval. The performance of the metrics is computed when considering single specifications only, or when adding pairs or triplets of specifications. A comparison is performed with a Random Classifier as a baseline, and with the values from the work in [FTB24].

The plausibility and the single score computed considering pairs and triplets of specifications, perform better than all other methods on the datasets. The belief measures, the performance of which is not included in the figure, do not seem to contribute much in distinguishing between the classes. This can be explained by considering that the technical characteristic of a truck only contain limited information about its future failure, and that it is highly unlikely that one (or a combination of) specific technical characteristic alone will necessarily lead to failure. In other words, decisions taken when considering the specification data by itself are too uncertain in order to have significant

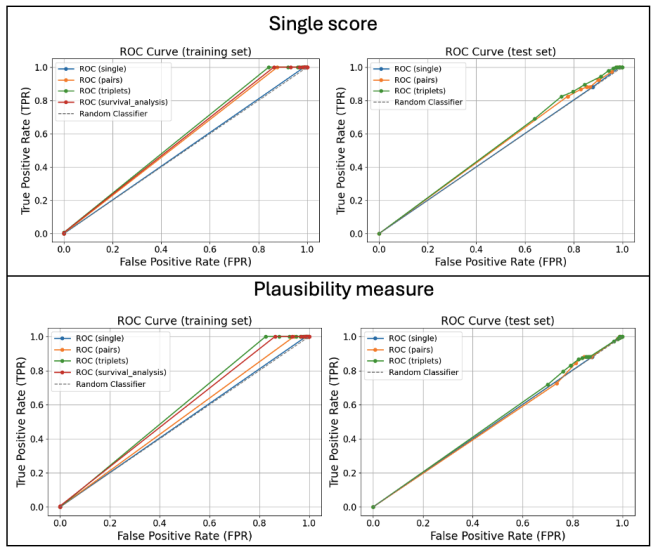

Figure 12. Discriminatory power of the single score and plausibility measure on the training and test set.

values for the belief score. This also justify the limited overall discriminatory potential of the measures, as they only contain partial knowledge about the use case.

Symbolic-to-neural chaining

The results of the experiments conducted with the 5 presented models is shown in Fig. 13. First, the LSTM model trained on sensor data only is applied to the test set. As explained, the input of the model consists of 12 consecutive time steps, while the model predict a class label for 12 time steps into the future. Subsequently, the plausibility and belief of the positive class are added as extra features to the model, either by simply concatenating them to the input, or providing them as initial context and letting the model learn how much weight they should have in the final prediction, through the usage of a gate placed before the first layer of the model. Finally, the single score obtained by all the evidence measures computed by the symbolic component is added to the sensor data as an extra features, fed to the model in two different ways, as with plausibility and belief.

All tested models obtained a very high accuracy (over 95%); however, as the dataset is highly imbalanced, and predicting correctly the higher classes is more important for the use case at hand than correctly individuating normal conditions, the F1-score and precision metrics are used to compare the performance of the models.

In Fig. 13, the evolution of the two metrics across all time steps is shown, for class 0 (normal operation) and class 4 (imminent failure). The difference in performance between class 0 and class 4 is striking, highlighting once again the challenge of dealing with an imbalanced dataset. All models perform strongly when predicting normal operation, maintaining high F1-score and precision across all time steps. In this case,

augmenting the LSTM's input with the symbolic measures doesn't provide measurable improvement, although this is not surprising as class 0 constitutes the vast majority of samples in the dataset.

The performance of all models is substantially lower for class 4, reflecting the difficulty in identifying failing trucks. Here, the model trained with the addition of the single score (gated) outperforms all the others. The outperforming of the model trained with plausibility and belief of the positive class shows that useful information might also be contained in the plausibility and belief of the negative class, which is included in the formula for the computation of the single score, see (8). It might also reflect the added difficulty of the model to deal with multiple static features, which could create redundancy and noise. The fact that feeding the single score as a separate gated input to the network performs better than simply concatenating the score to the sensor data, supports the idea that a more context-aware integration yields better results than naive concatenation.

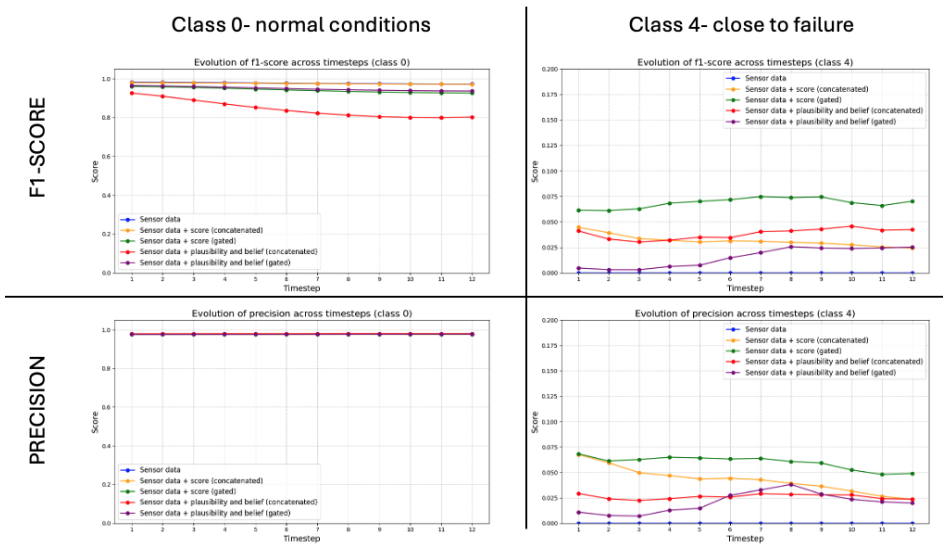

Figure 13. Comparison of F1-score and precision evolution across timesteps for the tested models.

Parallel symbolic-neural integration

In this strategy, the neural and symbolic components work in parallel to extract features from the data. The extracted features are then passed to a rule classifier exploiting logical rules to individuate failing trucks. The symbolic component is the same as in the previous strategy, thus its results have already been presented.

where y is a threshold on the plausibility score set as the minimum plausibility score of a positive sample from the training set, and x is a threshold set on the reconstruction

error. The performance of the rule-based classifier was evaluated with several thresholds, as seen in Fig 14.

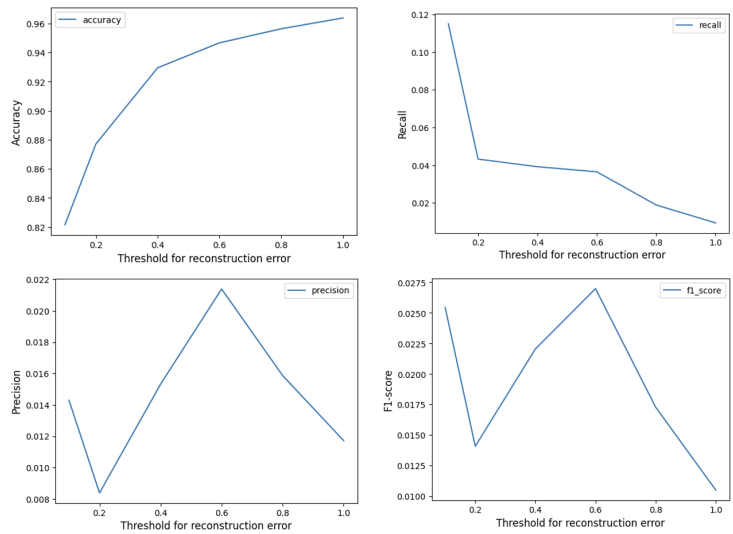

Figure 14. Performance of the rule-based classifier when varying the threshold on the reconstruction error.

As seen with the symbolic-to-neural chaining application, obtaining a high accuracy on the dataset is not a challenge, as the data is heavily skewed toward samples operating in normal conditions (class 0). However, what matters the most in the use case at hand is the number of failing trucks correctly identified. Thus,a threshold of 0.2 is selected to optimize this metric (best represented by the recall), while the overall accuracy stays over 80%. In Fig. 15, the cumulative percentage of failing vehicles correctly detected is plotted. Three temporal horizons are considered: classifications made 1 time step in advance, 6 time steps in advance, or 12 time steps in advance. As expected, the performance of the model decreases for longer time horizons.

Conclusion

The study presented Neuro-LENS, a neuro-symbolic framework exploring different integrations of a neural model with a symbolic reasoning component. The symbolic component is based on modal logic and evidence theory. The Neuro-LENS framework is modular, allowing great flexibility and generalizability to various use cases and data types. The usage of evidence-based logic provides interpretability and robustness to the uncertainty and ambiguity intrinsic in the data, making the approach suitable to be applied in real-world scenarios where only incomplete knowledge is available.

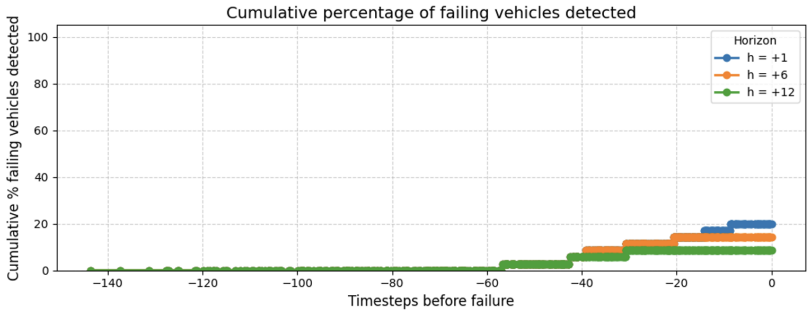

Figure 15. Cumulative percentage of failing vehicles correctly detected.

Three complementary strategies integrating symbolic reasoning and deep learning were investigated:

- **Neural-to-symbolic chaining**, validated on an image scene classification use case aimed at recognizing abandoned luggage scenarios. The approach demonstrated interpretability and robustness to uncertainty in the data, additionally to being able to deal with scarce and imbalanced datasets.
- **Symbolic-to-neural chaining**, applied to truck failure prediction. The results showed how symbolic reasoning can produce features to augment neural inputs with incomplete background knowledge, while embedding the uncertainty contained in the limited available information. The integration of the two paradigms yielded improvements in the predictive performance and specifically in spotting failing vehicles in a highly imbalanced dataset.
- **Parallel integration**, again validated to truck failure prediction. By combining the neural and symbolic components in a parallel manner and applying logical rules to their joined outputs, the approach is able to provide a high interpretability and distinguish between the contribution of the single outputs to a failure.

Validating the Neuro-LENS framework across different use cases and data types (e.g., images and tabular data) highlights its generalizability and practical relevance in real-world use cases and in industrial settings, where explainability, robustness, and trust are fundamental.

Acknowledgments

This research was partially funded by the Flemish Government through the AI Research Program.

Veselka Boeva's research was funded partly by the Knowledge Foundation, Sweden, through the Human-Centered Intelligent Realities (HINTS) Profile Project (contract 20220068).

References

[Ber77] Claude Berge. *Topological spaces: Including a treatment of multi-valued functions, vector spaces and convexity*. Oliver & Boyd, 1877.

[Dem67] A. Dempster. "Upper and lower probabilities induced by a multivalued mapping". In: *The Annals of Mathematical Statistics* 38 (1967), pp. 325–339.

[Sha76] G. Shafer. "A Mathematical Theory of Evidence". In: Princeton University Press, Princeton, 1976.

[Che80] B. Chellas. "Modal Logic, an Introduction". In: Cambridge University Press, Cambridge, 1980.

[AF90] J.-P. Aubin and H. Frankowska. "Set-Valued Analysis". In: Birkhäuser, Boston–Basel–Berlin, 1990.

[BTB98] V. Boeva, E. Tsiporkova, and B. De Baets. "Modelling uncertainty with kripke's semantics". In: *Artificial Intelligence: Methodology, Systems, and Applications. AIMSA 1998. LNCS*. Ed. by F. Giunchiglia. Vol. 1480. Springer, Berlin, Heidelberg, 1998.

[TBD99] Elena Tsiporkova, Veselka Boeva, and Bernard De Baets. "Dempster–Shafer theory framed in modal logic". In: *International journal of approximate reasoning* 21.2 (1999), pp. 157–175.

[TBB00] E. Tsiporkova, B. De Baets, and V. Boeva. "Evidence theory in multivalued models of modal logic". In: *Journal of Applied Non-Classical Logics* 10.1 (2000), pp. 55–81.

[BD04] V. Boeva and B. De Baets. "A new approach to admissible alternatives in interval decision making". In: *2nd Int. IEEE Conference on 'Intelligent Systems'. Proc. (IEEE Cat. No.04EX791)*. Vol. 1. 2004, pp. 110–115.

[BL04] Ronald Brachman and Hector Levesque. *Knowledge representation and reasoning*. Elsevier, 2004.

[Dem08] Arthur P. Dempster. "Upper and Lower Probabilities Induced by a Multivalued Mapping". In: *Classic Works of the Dempster-Shafer Theory of Belief Functions*. Ed. by Roland R. Yager and Liping Liu. Springer Berlin Heidelberg, 2008, pp. 57–72.

[GLG09] Artur S d'Avila Garcez, Luis C Lamb, and Dov M Gabbay. *Neural-symbolic cognitive reasoning*. Springer, 2009.

[Bes+17] Tarek R. Besold et al. "Neural-Symbolic Learning and Reasoning: A Survey and Interpretation". In: *ArXiv* abs/1711.03902 (2017). URL: https://api.semanticscholar.org/CorpusID:1755720.

[LL17] Scott M Lundberg and Su-In Lee. "A unified approach to interpreting model predictions". In: *Advances in neural information processing systems* 30 (2017).

[Mar18] Gary Marcus. "Deep learning: A critical appraisal". In: *arXiv preprint arXiv:1801.00631* (2018).

[Jai+23] Jitesh Jain et al. “Oneformer: One transformer to rule universal image segmentation”. In: *Proc. of IEEE/CVF Conf. on Comp. Vision and Pattern Recogn.* 2023, pp. 2989–2998.

[Wan+23] Yifeng Wang et al. “Rapid Image Labeling via Neuro-Symbolic Learning”. In: *Proc. of the 29th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*. 2023, pp. 2467–2477.

[Zha+23] Zuowei Zhang et al. “Representation of imprecision in deep neural networks for image classification”. In: *IEEE Transactions on NN and Learning Systems* (2023).

[FTB24] Fabian Fingerhut, Elena Tsiporkova, and Veselka Boeva. “Interpretable Data-Driven Risk Assessment in Support of Predictive Maintenance of a Large Portfolio of Industrial Vehicles”. In: *2024 IEEE International Conference on Big Data (BigData)*. IEEE. 2024, pp. 2870–2879.

[Hog+24] Eduard Hogea et al. “LogicLSTM: Logically-driven long short-term memory model for fault diagnosis in gearboxes”. In: *Journal of Manufacturing Systems* 77 (2024), pp. 892–902.

[Yan+24] Lihe Yang et al. “Depth Anything V2”. In: *arXiv preprint arXiv:2406.09414* (2024).

[DMT25] Michiel Dhont, Adrian Munteanu, and Elena Tsiporkova. “Forecasting Traffic Progression in Terms of Semantically Interpretable States by Exploring Multiple Data Representations”. In: *IEEE Transactions on Intelligent Transportation Systems* (2025).

[Fen+25] Siling Feng et al. “Integrating D–S evidence theory and multiple deep learning frameworks for time series prediction of air quality”. In: *Scientific Reports* 15.1 (2025), p. 5971.

[Fit25] Ricardo Fitas. “Neuro-Symbolic AI for Advanced Signal and Image Processing: A Review of Recent Trends and Future Directions”. In: *IEEE Access* (2025).

[Kha+25] Zahra Kharazian et al. “Scania component x dataset: A real-world multivariate time series dataset for predictive maintenance”. In: *Scientific Data* 12.1 (2025), p. 493.

[LWT25] Baoyu Liang, Yuchen Wang, and Chao Tong. “AI Reasoning in Deep Learning Era: From Symbolic AI to Neural–Symbolic AI”. In: *Mathematics* 13.11 (2025), p. 1707.

[Lu+25] Qiuhao Lu et al. “Explainable diagnosis prediction through neuro-symbolic integration”. In: *AMIA Summits on Translational Science Proceedings* 2025 (2025), p. 332.

[MBT25] Giulia Murtas, Veselka Boeva, and Elena Tsiporkova. “An evidence-based neuro-symbolic framework for ambiguous image scene classification”. In: *19th International Conference on Neurosymbolic Learning and Reasoning*. 2025. URL: https://openreview.net/forum?id=6UnuZcQ2zY.

[Wan+25] Yang Wang et al. “Learning Reliable and Intuitive Temporal Logic Rules for Interpretable Time Series Classification”. In: *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*. 2025, pp. 3067–3078.