Leveraging Inductive Logic Programming and Deep Learning for Trustworthy Vision

Zahra Chaghazardi[1], Saber Fallah[2], and Alireza Tamaddoni-Nezhad[3]

University of Surrey, UK
{z.chaghazardi,s.fallah,a.tamaddoni-nezhad}@surrey.ac.uk

Abstract. Deep learning serves as a crucial component in computer vision, enabling accurate predictions from raw data. However, unlike human cognition, deep learning models are vulnerable to adversarial attacks. This paper introduces a new method for traffic sign recognition that employs Inductive Logic Programming (ILP) to generate logical rules from a limited set of examples. These rules are used to assess the logical consistency of predictions, which is then incorporated into the neural network through the loss function. The study investigates the effect of incorporating logical rules into deep learning models on the robustness of vision tasks in autonomous vehicles (AV). The experimental results show that the proposed method significantly improves the accuracy of traffic sign recognition in the presence of adversarial attacks.

Keywords: Inductive Logic Programme · Autonomous Vehicles · Deep learning.

1 Introduction

In recent years, the widespread adoption of deep learning has revolutionized various fields, including vision-based systems, with remarkable performance. However, alongside their achievements, these systems have also revealed susceptibility to adversarial attacks. An adversarial attack is designed to produce adversarial examples to deceive machine learning systems. Despite their subtle alterations from genuine samples, these adversarial examples result in misclassification [13].

The vulnerability of deep learning models to adversarial attacks poses a significant security threat to safety-critical applications. For instance, researchers have shown that stop signs can be manipulated to be misidentified as speed limit signs by autonomous vehicles [8]. Fig. 1 illustrates the various stop signs, each subjected to different attack strategies with the intended misclassification target being a speed limit sign, and the results showed that the altered Stop signs were misclassified as speed limit 45 signs by the image classifier.

In contrast to deep learning, which can be easily fooled, humans leverage prior knowledge, extract high-level attributes as factual data, and apply logical constraints on these facts to make robust and reliable decisions, even when confronted with ambiguity or adversarial inputs. For example, recognizing the octagonal shape of a traffic sign combined with the logical fact that only stop

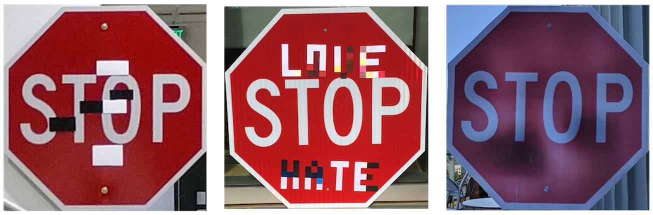


Fig. 1: Different Stop signs manipulated by the RP2 attacks were misclassified as speed limit sign by the image classifier [8]

signs have this shape assists humans in avoiding mistaking it for other signs. We hypothesize that incorporating similar mechanisms into deep learning models can enhance their resilience against adversarial attacks. In line with this idea, some studies, such as [5] and [4] have introduced methods that utilised logical programming for traffic sign recognition, which led to enhanced resilience against adversarial inputs.

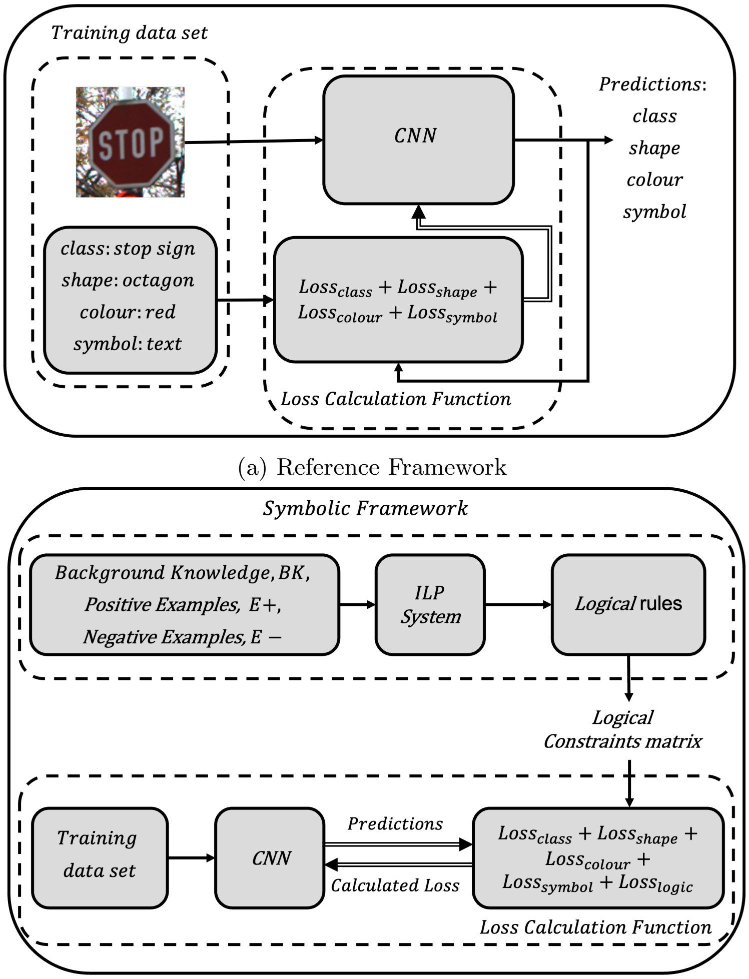
This paper presents a method, to enhance the trustworthiness of Neural Networks (NNs) based traffic sign recognition. Building on our previous work [6], this approach highlights the importance of incorporating Background Knowledge (BK) to bolster robustness by integrating additional features into the learning process. Our approach involves integrating human knowledge to improve the reliability of traffic sign recognition, mirroring human perception. We automatically extract logical rules by leveraging ILP from positive and negative examples. Subsequently, these rules are mapped into propositional logic, forming a logical constraint matrix. We then evaluate the satisfaction of constraints for each prediction, and the resulting constraint dissatisfaction is integrated into the loss function, ensuring compliance with BK.

2 Proposed Framework

In this section, we introduce the proposed framework aimed at improving the robustness of NNs against adversarial attacks by incorporating human knowledge. This framework utilises human-provided attributes extracted from the input images. For each traffic sign category, these attributes are identical.

Furthermore, the proposed approach leverages ILP to systematically derive rules and logical constraints from human-provided knowledge. Integrating these constraints into the neural network's loss function significantly enhances the model's resilience. The architecture of the Reference model and the proposed neuro-symbolic framework is illustrated in Fig. 2.

The Convolutional Neural Network (CNN) serves as a multilabel classifier in the Reference model. During the training phase, the CNN processes an input image of a traffic sign alongside four corresponding labels: traffic sign class, shape, colour, and symbol. Once trained, the model is capable of predicting the class, shape, colour, and symbols of any input traffic sign image.



(a) Reference Framework

(b) Proposed Framework

Fig. 2: a) Reference framework and b) Proposed framework for the traffic sign classifier incorporating logical constraints

The proposed model consists of two main components: a symbolic framework and a modified Reference framework with an adjusted loss function. In the symbolic framework, BK is given to the ILP system, which leverages it to induce rules and logical constraints systematically.

These logical constraints are subsequently used to modify the loss function of the Reference framework. During the training process, the CNN model's predictions are input into the loss calculator, which assesses how well the predicted labels align with the derived logical constraints. A regularisation term is then incorporated into the loss function to enforce compliance with these rules.

In this classification task, the ILP system generates one rule per class. Each traffic sign is thus linked to a specific rule, and during training, the rule that best matches the prediction is used to adjust the loss function.

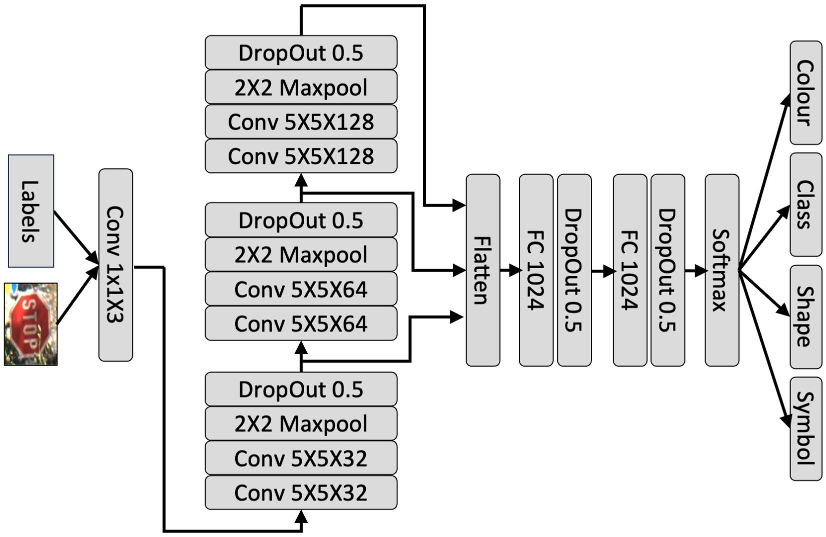


Fig. 3: Architecture of the attributes Convolutional Neural Network (ACNN) model utilised as the CNN Reference model in our study.

2.1 Material

We adapted a CNN architecture implementation [18], originally designed for traffic sign recognition, to create a multilabel classifier. The adopted model is illustrated in Fig. 3 as our Reference model. The training configuration included the Adam optimiser with a learning rate of 0.001, 20 epochs, a batch size of 32, and the Cross-Entropy loss function.

This multilable classifier was trained on the German Traffic Sign Recognition Benchmark (GTSRB) [17] For our training, we focused on a subset of eleven traffic sign categories, with each category represented by 150 training images. Fig. 4 illustrates the traffic signs employed in our study, organized by their specific attributes such as colour, shape, and symbol.

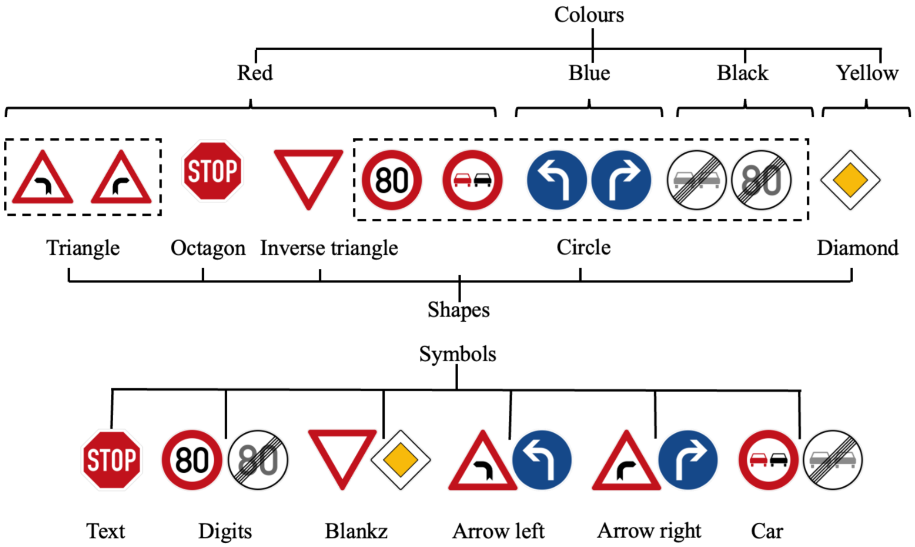


Fig. 4: A set of eleven traffic signs utilised in this study, each associated with corresponding shape, colour and symbol labels.

We selected 50 images from each class of the GTSRB dataset to create a normal test dataset. Furthermore, In this study, the proposed classifier undergoes testing across various adversarial attack datasets, including GRATS (Dirty) [2], Dart [16], and Shadow [22] attack datasets. These datasets introduce various challenges, enabling the assessment of the classifier's robustness across different adversarial scenarios.

2.2 Logical Constraint Extraction

We employ an ILP system to extract comprehensive and optimal logical constraints. ILP is an inductive reasoning technique at the intersection of Machine Learning and Logic Programming, specifically designed to derive logical rules from a few sets of examples [15]. For this framework, we have utilised the ILP system Aleph [3].

The Aleph system operates through a procedure that begins with selecting a positive example to generalise. The system then constructs a specific clause, called the bottom clause, based on the selected example and corresponding BK. next, the system searches for a more general clause than the bottom clause. The best general clause is then added to the hypothesis set, redundant examples are removed and the process returns to the first step. This process continues until no more examples are left. The outcome is a set of induced hypotheses that should cover as many positive and as few negative examples as possible.

After the ILP system induces interpretable logical rules, the next stage involves mapping these rules into a constraint matrix denoted as "C" which is depicted as follows:

$$C = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1L} \\ c_{21} & c_{22} & \dots & c_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ c_{N1} & c_{N2} & \dots & c_{NL} \end{bmatrix}$$

Here, N represents the number of rules or constraints, where each traffic sign category is associated with a specific rule. L denotes the number of labels, which is four in our case, corresponding to the attributes of colour, shape, symbol, and class.

The element c_{ij} denotes the value of the index related to a specific attribute j that appeared in a specific rule i. Each attribute l includes S_l categories, for instance, the 'shape' attribute includes different geometrical categories such as circles, triangles, and rectangles. This indexing system systematically captures the relationships between rules and attributes.

In this configuration, the matrix will have dimensions of $N \times L$, which is considerably smaller compared to the constraint matrix proposed by [11], whose size is $N \times (S_1 + \ldots + S_L)$. In our case study with 11 rules, 4 colours, 5 shapes, 6 symbols and 11 classes, the constraint dimension for our proposed method is 11×4. This more compact representation is advantageous, particularly for larger problems.

2.3 Logic-Integrated Loss

We introduced a logic-based regularisation term into the multi-label classification loss to enhance robustness. This term quantifies the level of satisfaction of the CNN model's predictions with respect to the applied logical constraints. Eq. 1 represents the total loss L_{total}, where the first term consists of a combination of cross-entropy losses for each attribute, and the second term is an additional logical loss, which ensures that the model adheres to logical constraints.

$$\mathrm{L}_{total} = \sum_{l=1}^{L} L_{attribute} + L_{logic} \tag{1}$$

The process of logical loss computation begins by considering each rule (c), represented as a row in the logical constraint matrix (C), which specifies the corresponding indices for each attribute (class, shape, colour, and symbol) for that rule. The predicted values associated with these indices are selected and evaluated according to the chosen type of logic operation, either "product" or "Gödel".

If the logical operation is set to "product," the satisfaction score is computed as the product of these predicted values. Alternatively, if the logical operation follows the Gödel t-norm, the satisfaction is determined by selecting the minimum value from the predicted labels including class, shape, colour and symbol probabilities. The rule with the highest satisfaction score is then selected as the best match, and the loss is computed based on this rule as 1 - satisfaction.

The rule matrix C of size $N \times L$, contains elements c_{nl}, representing the index of the attribute l associated with the respective rule n. For each attribute l, the deep learning model generates a prediction vector P_l, containing n_l predictions. Here n_l represents the number of categories for that attribute.

Given C and P, our objective is to determine the satisfaction level of each constraint for the corresponding prediction. To achieve this, we define a matrix G of dimensions $N \times L$, each element G_{nl} is determined by the probability from prediction vectors $P_l(c_{nl})$. This relationship is expressed as:

$$G_{nl} = P_l(c_{nl}), \quad \text{for } n = 1, \ldots, N \text{ and } l = 1, \ldots, L \tag{2}$$

The satisfaction level G' is computed using a t-norm (conjunction) operation along the second dimension (attributes) followed by a t-conorm (disjunction) across the first dimension (rules):

$$G' = \text{t-conorm}(\text{t-norm}(G, dim = 2), dim = 1) \tag{3}$$

For evaluating the satisfaction of each rule, we consider the Gödel and Product t-norms. The logical conjunction operator is substituted with the product t-norm ($x \wedge_p y \equiv xy$) or the Gödel t-norm ($x \wedge_g y \equiv \min(x, y)$). Additionally, product t-conorm ($x \vee_p y \equiv 1 - (1 - x)(1 - y)$) or the Gödel t-conorm ($x \vee_g y \equiv \max(x, y)$) is employed as a disjunction.

Eq. 3 first applies a t-norm operation to compute the satisfaction level for each rule across the attributes. For example, if we use the Gödel t-norm, the satisfaction score for each rule is obtained by taking the minimum value across all attributes for that rule ($t\text{-norm}(G, \text{dim} = 2) = \min(G_{n,1}, \ldots, G_{n,l})$). Once the satisfaction level for each rule is computed, a t-conorm operation is applied to aggregate the satisfaction levels across the rules. In the case of Gödel t-conorm, this involves taking the maximum value among the computed satisfaction scores. This operation helps in selecting the rule with the highest overall satisfaction. The logic loss is then defined as:

$$L_{logic} = 1 - G' \tag{4}$$

The resulting dissatisfaction is then incorporated into the loss function as a regularization term. Let's consider an example with this logical constraint for an input image:

```
Shape: octagon
Color: red
Symbol: text
Class: stop sign
```

The CNN provides output probabilities for various attributes. For example, consider the shape prediction vector:

$$\text{P\_shape} : [0.1, 0.2, 0.0, 0.7, 0.0]$$

This vector indicates that the probability of the shape being an octagon is 0.7. Similarly, from P_colour, P_symbol and P_class, the deep learning model obtains probabilities of 0.9 for the colour being red, 0.6 for the symbol being text and 0.8 for the sign being a stop sign.

In the product t-norm, rule satisfaction is computed by multiplying the probabilities of the conditions, while in the Gödel t-norm, it is determined by selecting the minimum value among them.

$$\text{Product t-norm rule satisfaction} = 0.7 \times 0.9 \times 0.6 \times 0.8 = 0.3 \tag{5}$$

$$\text{Gödel t-norm rule satisfaction} = \min(0.7, 0.9, 0.6, 0.8) = 0.6 \tag{6}$$

This process is repeated for all rules. Once the satisfaction for each rule is computed, a t-conorm operation is used to determine the overall satisfaction across all rules. The complement of this overall satisfaction value is then incorporated into the loss function as a regularization term.

3 Results and Discussion

In this section, we evaluate the robustness of the proposed model by comparing it with the Reference model[1]. Each model was independently trained over ten iterations, and the average accuracy was calculated.

Figure 5 illustrates the accuracy of basic and various 4-attribute models including Reference, Non_Logic, Logic_Product, and Logic_Gödel models when tested on targeted stop signs. The evaluation was conducted across several datasets, including Normal, Dart, Dirty, and Shadow. Results show that while the Logic_Gödel closely matches the Reference model's performance on the Normal dataset, it significantly outperforms other models on the attack test datasets, demonstrating the enhanced robustness of the proposed logic-based approach.

[1] The code required to reproduce the experiments detailed in this paper can be accessed at https://github.com/Chaghazardi/Leveraging-Inductive-Logic-Programming-and-Deep-Learning-for-Trustworthy-Vision

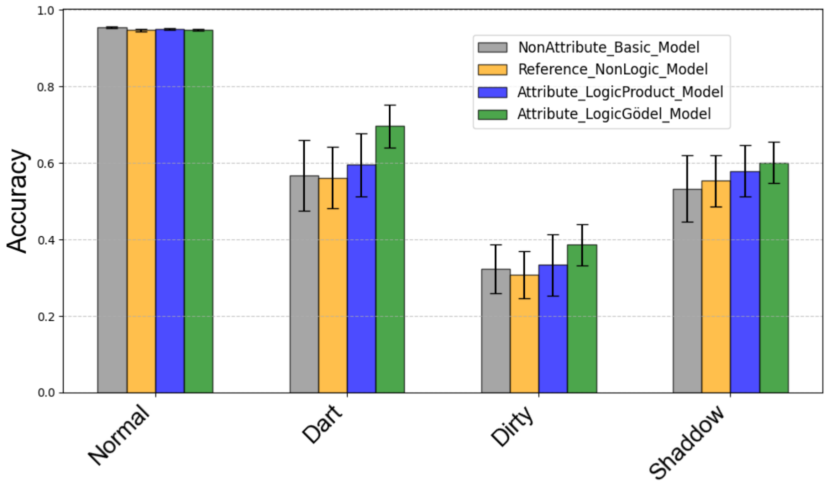


Fig. 5: Comparison of accuracy among Non_Logic (described in Fig. 3), Logic_Product, and Logic_Gödel on normal and targeted stop signs by various attack datasets including Dart, Dirty, Shadow, Subtle, and LoveHate.

Logic_Product also surpasses the Reference and Non_Logic models in all attack methods, though it consistently underperforms when compared to Logic_G ödel, which underscores the latter's superior robustness against adversarial attacks.

To investigate the impact of incorporating additional related attributes on the robustness of logic_based models against adversarial attacks, we compared a 3-attribute model (including shape, colour, and class attributes) with a 4-attribute model (including shape, colour, symbols, and class attributes) across various test datasets. Figure 6 illustrates the accuracy of both the 3-attribute and 4-attribute models for Logic_Product and Logic_Gödel on the Normal, Dart, Dirty, and Shadow datasets. The results indicate that the 4-attribute Logic_based models outperform the 3-attribute Logic_based models, highlighting the significance of incorporating additional background knowledge.

Fig. 7 presents the predictions from both the Non_Logic and proposed Logic_G"odel 4-attribute models for targeted stop signs across various attack scenarios, including Dart, Dirty, and Shadow attacks. The results demonstrate that the Logic_Gödel model produces more robust and confident predictions compared to the Non_Logic model predictions.

4 Related Work

This section overviews current research efforts aimed at integrating prior knowledge into neural networks, either by network architecture adjustment or loss function modification or data transformation. These strategies aim to improve

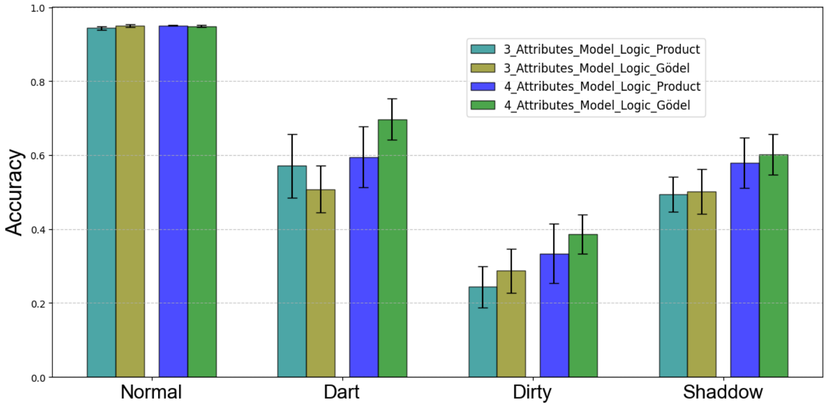


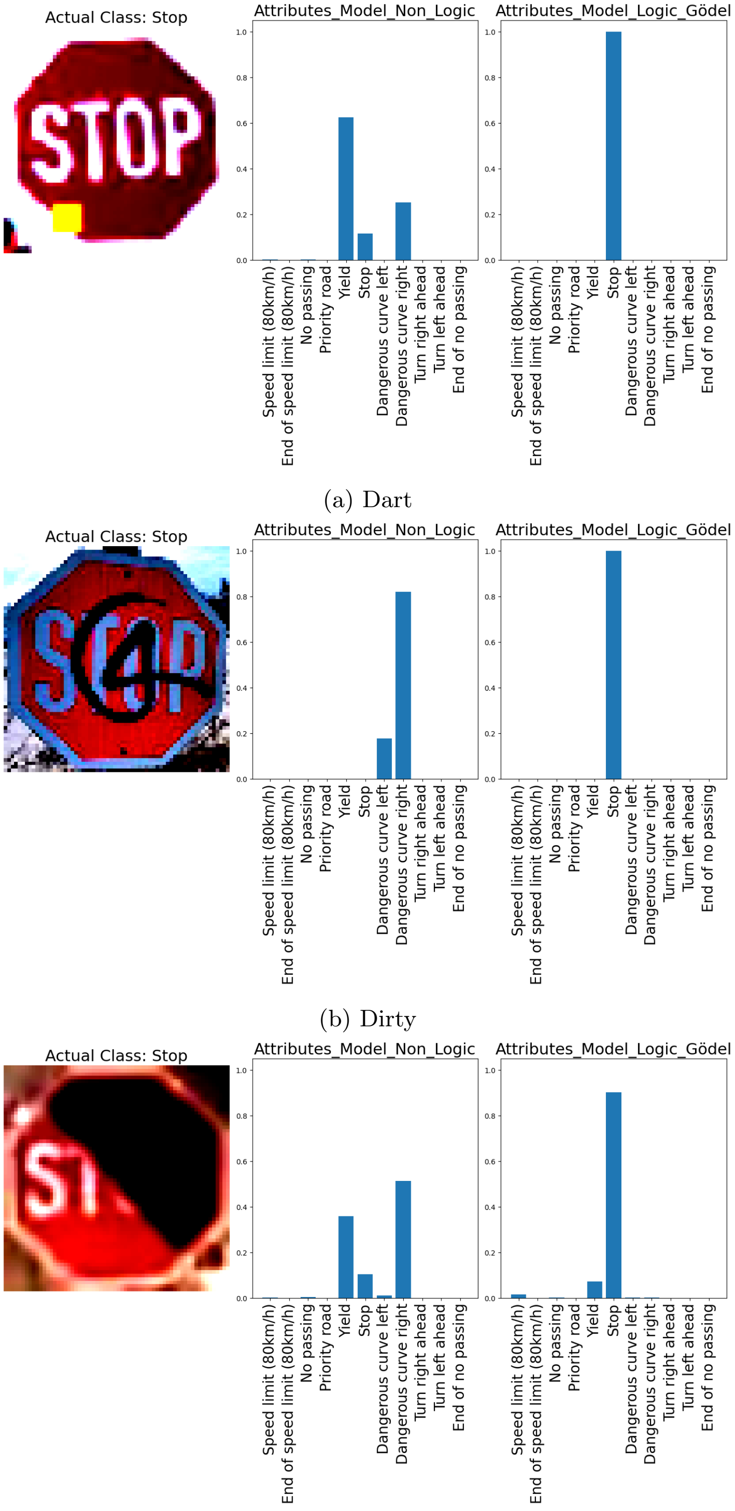
Fig. 6: Accuracy comparison between 3-attribute and 4-attribute logic-based models on the Normal dataset and various adversarial datasets, including Dart, Dirty, and Shadow.

performance, support learning from limited data, and ensure that the model adheres to existing knowledge [12].

In the former category, methods construct a constrained layer on top of a neural network ensuring full constraint satisfaction. For instance, MultiplexNet [14], Coherent-by-Construction Network (CCN) [10] and Semantic Probabilistic Layer (SPL) [1] approaches exemplify this strategy. MultiplexNet augments the prediction layer with transformations that allow the output layer to function like a logical circuit multiplexor. The CCN approach employs a dual methodology, introducing a supplementary top layer that modifies the output to meet specified constraints and integrates these constraints into a collaborative loss function. While SPL integrates a compiled logic circuit layer into the network to enforce constraints directly.

Our approach aligns with the second category, where domain knowledge is integrated through modifications to the loss function, which is optimized during the deep network's training process. In this method, a penalty is introduced into the loss function to reflect the degree to which the model's predictions satisfy the constraints imposed by the domain knowledge. This ensures that the network not only learns from the data but also adheres to the specific requirements dictated by the background knowledge.

Several studies have explored the integration of domain-specific knowledge into neural network training through modifications to the loss function. For example, Xu et al. [21] introduced a syntax-independent semantic loss function that remains effective regardless of how constraints are expressed. Logic Embedding Network with Semantic regularisation (LENSR) [20] integrates logical constraints into the loss function by using a Graph Convolutional Network (GCN) to project logical formulas onto a manifold. The model then learns to minimize the distance to satisfying assignments. DL2 [9] expanded on this by translat-



(a) Dart

(b) Dirty

(c) Shadow

Fig. 7: Exemplifying predictions generated by both the Non_Logic and the proposed Logic_Gödel models for targeted stop signs subjected to various attacks.

ing constraints into convex sets and adding to the loss function, allowing for additional features such as querying model decisions. Wang and Pan [19] employed a parallel neuro-reasoning engine, generating an output consistent with the neural process and incorporating the disparity between the two outputs into the loss function. This innovative method aims to leverage both neural networks and reasoning engines to enhance model performance and adherence to logical constraints.

Another approach, categorized as the third method, integrates domain knowledge into deep networks by modifying inputs to include symbolic domain-specific attributes. For example, By transforming input data into graph representations, GNNs can effectively incorporate relational domain knowledge into learning.

For a comprehensive review of the integration of logical constraints in deep learning, please refer to [7]

5 Conclusion

In summary, our experiments highlight significant advancements achieved by the proposed Logic-based model, particularly in accurately detecting adversarial traffic signs. Despite training on a relatively limited dataset of 1650 images, the proposed model exhibits substantial enhancements in accuracy, underscoring its robustness in handling challenging scenarios.

One critical aspect of our approach is the integration of logical rules into the neural network architecture. This fusion of logic into deep learning not only enhances the model's performance but also fortifies its reliability. By incorporating domain-specific knowledge through logical rules, the proposed model can better generalize from limited data and correctly recognise traffic signs with adversarial manipulations that might deceive traditional neural networks.

These findings underscore the potential of the logic-based approach as a valuable strategy for bolstering the trustworthiness of neural networks in real-world applications, particularly in the domain of autonomous vehicles. By combining the strengths of deep learning and symbolic logic, the logic-based model presents a robust framework for enhancing the accuracy and reliability of vision systems in autonomous driving. Future work could explore expanding the dataset and refining the logical rules to further improve the model's performance and applicability to a broader range of scenarios.

Acknowledgments. The first author would like to acknowledge her PhD grant funding from the Breaking Barriers Studentship Award at the University of Surrey. The third author would like to acknowledge the EPSRC Network Plus grant on Human-Like Computing (HLC) and the EPSRC grant on human-machine learning of ambiguities.

References

1. Ahmed, K., Teso, S., Chang, K.W., Van den Broeck, G., Vergari, A.: Semantic probabilistic layers for neuro-symbolic learning. Advances in Neural Information Processing Systems **35**, 29944–29959 (2022)
2. Apostolidis, K.D., Gkouvrikos, E.V., Vrochidou, E., Papakostas, G.A.: Traffic sign recognition robustness in autonomous vehicles under physical adversarial attacks. In: Cutting Edge Applications of Computational Intelligence Tools and Techniques, pp. 287–304. Springer (2023)
3. Ashwin Srinivasan: The aleph manual. `https://www.cs.ox.ac.uk/activities/programinduction/Aleph/aleph.html` (2001)
4. Chaghazardi, Z., Fallah, S., Tamaddoni-Nezhad, A.: Explainable and trustworthy traffic sign detection for safe autonomous driving: An inductive logic programming approach. In: Proceedings of the 39th International Conference on Logic Programming. Electronic Proceedings in Theoretical Computer Science, vol. 385, pp. 201–212. Open Publishing Association, Imperial College London, UK (July 2023). https://doi.org/10.4204/EPTCS.385.21
5. Chaghazardi, Z., Fallah, S., Tamaddoni-Nezhad, A.: A logic-based compositional generalisation approach for robust traffic sign detection. In: International Joint Conference on Artificial Intelligence 2023 Workshop on Knowledge-Based Compositional Generalization (2023), `https://openreview.net/forum?id=jfU2Xv84_O`
6. Chaghazardi, Z., Fallah, S., Tamaddoni-Nezhad, A.: Trustworthy vision for autonomous vehicles: A robust logic-infused deep learning approach. In: 27th IEEE International Conference on Intelligent Transportation Systems ITSC (2024), in press
7. Dash, T., Chitlangia, S., Ahuja, A., Srinivasan, A.: A review of some techniques for inclusion of domain-knowledge into deep neural networks. Scientific Reports **12**(1), 1040 (2022)
8. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., Song, D.: Robust physical-world attacks on deep learning visual classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1625–1634 (2018)
9. Fischer, M., Balunovic, M., Drachsler-Cohen, D., Gehr, T., Zhang, C., Vechev, M.: Dl2: training and querying neural networks with logic. In: International Conference on Machine Learning. pp. 1931–1941. PMLR (2019)
10. Giunchiglia, E., Lukasiewicz, T.: Multi-label classification neural networks with hard logical constraints. Journal of Artificial Intelligence Research **72**, 759–818 (2021)
11. Giunchiglia, E., Stoian, M.C., Khan, S., Cuzzolin, F., Lukasiewicz, T.: Road-r: The autonomous driving dataset with logical requirements. Machine Learning pp. 1–31 (2023)
12. Giunchiglia, E., Stoian, M.C., Lukasiewicz, T.: Deep learning with logical constraints. arXiv preprint arXiv:2205.00523 (2022)
13. Gui, J., Sun, Z., Wen, Y., Tao, D., Ye, J.: A review on generative adversarial networks: Algorithms, theory, and applications. IEEE transactions on knowledge and data engineering (2021)
14. Hoernle, N., Karampatsis, R.M., Belle, V., Gal, K.: Multiplexnet: Towards fully satisfied logical constraints in neural networks. In: Proceedings of the AAAI conference on artificial intelligence. vol. 36, pp. 5700–5709 (2022)

15. Muggleton, S.: Learning from positive data. In: International conference on inductive logic programming. pp. 358–376. Springer (1996)
16. Sitawarin, C., Bhagoji, A.N., Mosenia, A., Chiang, M., Mittal, P.: Darts: Deceiving autonomous cars with toxic signs. arXiv preprint arXiv:1802.06430 (2018)
17. Stallkamp, J., Schlipsing, M., Salmen, J., Igel, C.: Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. Neural networks **32**, 323–332 (2012)
18. Vivek Yadav: German sign classification using deep learning neural networks. `https://github.com/vxy10/p2-TrafficSigns` (2016)
19. Wang, W., Pan, S.J.: Integrating deep learning with logic fusion for information extraction. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 9225–9232 (2020)
20. Xie, Y., Xu, Z., Kankanhalli, M.S., Meel, K.S., Soh, H.: Embedding symbolic knowledge into deep networks. Advances in neural information processing systems **32** (2019)
21. Xu, J., Zhang, Z., Friedman, T., Liang, Y., Broeck, G.: A semantic loss function for deep learning with symbolic knowledge. In: International conference on machine learning. pp. 5502–5511. PMLR (2018)
22. Zhong, Y., Liu, X., Zhai, D., Jiang, J., Ji, X.: Shadows can be dangerous: Stealthy and effective physical-world adversarial attack by natural phenomenon. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15345–15354 (2022)