Reviewer 1:

The only addition I would recommend, given that this was suggested by another reviewer as well, would be to have a table summarising features that are/are not present in this overall approach compared to other approaches discussed in the SOTA. I agree that the narrative form is capturing nouances, but it might leave the reader the impression that some aspects are not as yes/no which makes it hard to compare at a glance.
This would be also beneficial for papers citation and to make the point re. the added value of the approach by having a table with approaches and features with a yes/no.
It would also somehow force the authors to carefully select what are the key selling point of the approach and make sure they are directly visible to others/readers.

We thank the reviewer for this suggestion. To make the key selling points of our Neurosymbolic C‑XAI approach immediately visible, we have added Table 1 in Section 2 (“Related Work”) that compares our method against representative SOTA XAI techniques on a common set of features. This table highlights, for each method:

- whether it is white‑box or black‑box,
- whether it uses systematic (ontology‑driven) concept extraction,
- whether it provides quantitative precision/recall error‑margins,
- whether it is model‑agnostic,
- whether it supports end‑to‑end automation, and
- whether it relies on large background knowledge.

We believe this tabular summary complements the narrative and makes our contributions—and their advantages over prior work—clear at a glance.

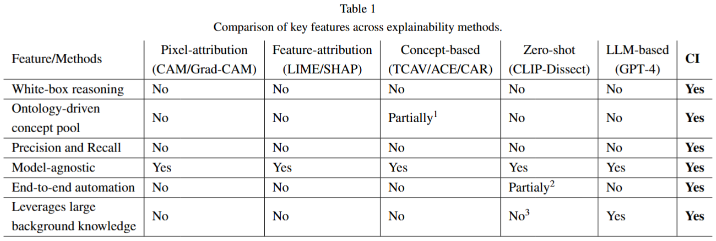


Table 1

Comparison of key features across explainability methods.

Feature/Methods	Pixel-attribution (CAM/Grad-CAM)	Feature-attribution (LIME/SHAP)	Concept-based (TCAV/ACE/CAR)	Zero-shot (CLIP-Dissect)	LLM-based (GPT-4)	**CI**
White-box reasoning	No	No	No	No	No	**Yes**
Ontology-driven concept pool	No	No	Partially[1]	No	No	**Yes**
Precision and Recall	No	No	No	No	No	**Yes**
Model-agnostic	Yes	Yes	Yes	Yes	Yes	**Yes**
End-to-end automation	No	No	No	Partialy[2]	No	**Yes**
Leverages large background knowledge	No	No	No	No[3]	Yes	**Yes**

Reviewer 3:

A1: I would still strongly advise for a table that sums up the literature review in a tabular format. If you believe there are such nuances, you could point them out in any cell that needs it. What I envision would be something along the lines of Table 1 of this work: "Improving rule-based classifiers by Bayes point aggregation" (Bergamin et al., 2025). As you mentioned, there are different nuances (degree of supervision, concept pools, neural vs symbolic, etc.), that can become a new column for each table.

We thank the reviewer again for this suggestion. We have now incorporated Table 1 into the main text of Section 2. This table compares representative XAI methods (including neural, symbolic, supervision degree, concept‑pool dynamics, and other key axes) alongside our Concept Induction approach. Nuances are noted in footnotes where necessary. We believe this addition directly addresses your recommendation by providing an at‑a‑glance taxonomy while preserving the detailed narrative discussion.

<table>
<tr>
<td>Personally, I would prefer the table to be in the main text.</td>
<td>

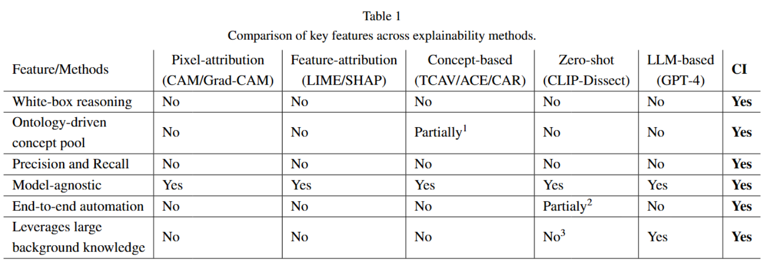
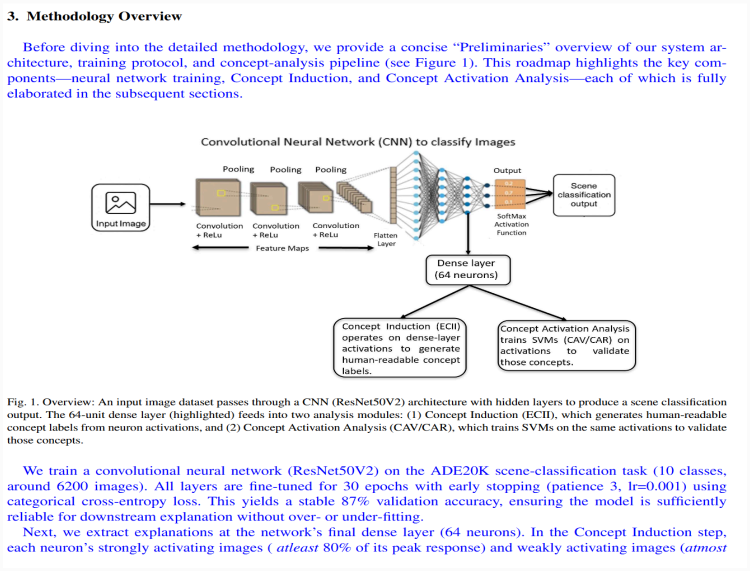



Table 1
Comparison of key features across explainability methods.
<table>
<tr><th>Feature/Methods</th><th>Pixel-attribution (CAM/Grad-CAM)</th><th>Feature-attribution (LIME/SHAP)</th><th>Concept-based (TCAV/ACE/CAR)</th><th>Zero-shot (CLIP-Dissect)</th><th>LLM-based (GPT-4)</th><th><b>CI</b></th></tr>
<tr><td>White-box reasoning</td><td>No</td><td>No</td><td>No</td><td>No</td><td>No</td><td><b>Yes</b></td></tr>
<tr><td>Ontology-driven concept pool</td><td>No</td><td>No</td><td>Partially[1]</td><td>No</td><td>No</td><td><b>Yes</b></td></tr>
<tr><td>Precision and Recall</td><td>No</td><td>No</td><td>No</td><td>No</td><td>No</td><td><b>Yes</b></td></tr>
<tr><td>Model-agnostic</td><td>Yes</td><td>Yes</td><td>Yes</td><td>Yes</td><td>Yes</td><td><b>Yes</b></td></tr>
<tr><td>End-to-end automation</td><td>No</td><td>No</td><td>No</td><td>Partialy[2]</td><td>No</td><td><b>Yes</b></td></tr>
<tr><td>Leverages large background knowledge</td><td>No</td><td>No</td><td>No</td><td>No[3]</td><td>Yes</td><td><b>Yes</b></td></tr>
</table>
</td>
</tr>
<tr>
<td>Q2: While I understand the utility of having the notions related to each section structured to give the background needed at the beginning of each section, some common preliminary notions could be moved to a background section before entering Section 3. This section could also help provide a visual example to help understand all the inputs/outputs involved in the system. In my opinion, this would help to make the paper less of a collection of existing published papers and more of a comprehensive work on the Topic.
<b>A2: I think this is a very good idea that should be incorporated to make the paper more accessible to reader. I advice the authors to incorporate this point.</b></td>
<td>
Thank you for this valuable suggestion. We have added a new Section 3: Pipeline Overview which gathers all common preliminaries in one place and includes a high-level diagram (Figure 1) of our end-to-end system. This section briefly describes:

● Neural network training (input images ⟶ CNN ⟶ dense layer ⟶ output),
● Concept Induction and LLM labeling (ECII and GPT-4 on dense-layer activations),
● Hypothesis confirmation (statistical testing of neuron-concept associations),
● Concept Activation Analysis (CAV/CAR SVMs on activations), and
● Error-margin computation (precision/recall bounds for each explanation).

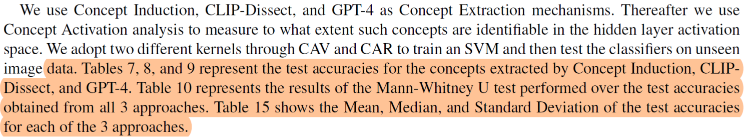

<b>3. Methodology Overview</b>

Before diving into the detailed methodology, we provide a concise “Preliminaries” overview of our system architecture, training protocol, and concept-analysis pipeline (see Figure 1). This roadmap highlights the key components—neural network training, Concept Induction, and Concept Activation Analysis—each of which is fully elaborated in the subsequent sections.

Convolutional Neural Network (CNN) to classify Images
Input Image
Pooling
Pooling
Pooling
Convolution + ReLu
Convolution + ReLu
Convolution + ReLu
Feature Maps
Flatten Layer
Output
SoftMax Activation Function
Scene classification output
Dense layer (64 neurons)
Concept Induction (ECII) operates on dense-layer activations to generate human-readable concept labels.
Concept Activation Analysis trains SVMs (CAV/CAR) on activations to validate those concepts.

Fig. 1. Overview: An input image dataset passes through a CNN (ResNet50V2) architecture with hidden layers to produce a scene classification output. The 64-unit dense layer (highlighted) feeds into two analysis modules: (1) Concept Induction (ECII), which generates human-readable concept labels from neuron activations, and (2) Concept Activation Analysis (CAV/CAR), which trains SVMs on the same activations to validate those concepts.

We train a convolutional neural network (ResNet50V2) on the ADE20K scene-classification task (10 classes, around 6200 images). All layers are fine-tuned for 30 epochs with early stopping (patience 3, lr=0.001) using categorical cross-entropy loss. This yields a stable 87% validation accuracy, ensuring the model is sufficiently reliable for downstream explanation without over- or under-fitting.
Next, we extract explanations at the network’s final dense layer (64 neurons). In the Concept Induction step, each neuron’s strongly activating images (<i>atleast</i> 80% of its peak response) and weakly activating images (<i>atmost</i>
</td>
</tr>
<tr>
<td>Q3: It is quite strange that only the</td>
<td>We appreciate your continued attention to model-performance</td>
</tr>
</table>

Resnet50V2 achieved high validation accuracy scores, while other architectures show a big gap with the training accuracy, especially when using early stopping. Do other metrics highlight this issue (e.g., top-k accuracy) as well? Could you compare the confusion matrices? Also, is patience=3 / learning rate=0.001 sufficient/necessary to fine-tune this task? Usually, you could get better results in fine-tuning with lower learning rates and/or providing more epochs. While I understand the argument of the low need for high accuracy, the explanations should be made on a sufficiently reliable/performant model, and I can't see how Resnet50v2 has such a wide margin compared to the classic Resnet50. **A3: I still have my doubts on the soundness of this part of the experimental setting, due to the lack of systematic hyperparameter tuning, where hyperparams were set ad-hoc, and the lack of additional data regarding other metrics, confusion matrices (or even just training losses plots, etc.). As you are very well aware, this could lead to unwanted under/overfitting, and other uncontrolled model behavior. I still believe this is a weaker side of this paper, but I agree this was not the focus to begin with. Therefore, I do not have explicit requests for this point (but, still, the authors are welcome to improve it if they deem it necessary.)**	details. We performed basic sweeps over learning rates (1e-2, 1e-3, 1e-4), patience (3, 5, 10), and up to 50 epochs. ResNet50V2 (lr = 0.001, patience = 3) gave stable ~87% validation accuracy with no over- or under-fitting. Since our focus is on explanation fidelity rather than peak classification accuracy, we believe these settings are sufficient and have retained the original text.
Q4: I am not sure of the usefulness of Table 6-7-8. In particular, they show the raw performance in both training and test settings. Wouldn't a chart be more informative, especially while comparing the results of GPT/CLIP/Concept Induction? Those tables could be moved to an Appendix if possible. Also, I am unsure of the utility of having the training accuracy reported as well, if not discussed in the paper. **A4: Thank you for your response to my comment regarding Tables 6, 7, and 8. I understand that you were unable to devise a meaningful way to visualize the data without adding redundancy or length to the paper. One option could be to craft a bar chart for each row. These bar charts could be sorted by a target metric (e.g., either CAR or CAV test accuracy). To improve readability, they could be split across multiple columns to reduce length.**	We appreciate the suggestion, and moved the full per-concept Tables 7–9 into Appendix A. Our current Table 12 already provides exactly that "at-a-glance" comparison: it reports, for each method and kernel (CAV/CAR), the mean, median, and standard deviation of test accuracies, as well as the count of concepts in high (≥ 90%), medium (80–89%), and low (< 80%) accuracy bins. Likewise, Table 7 and Table 11 gives the Mann–Whitney U test results that quantify statistical significance across methods. Both tables are discussed in detail in the Results section, where we call out their key take-aways. We therefore believe these existing summary tables fully address your concern.  We use Concept Induction, CLIP-Dissect, and GPT-4 as Concept Extraction mechanisms. Thereafter we use Concept Activation analysis to measure to what extent such concepts are identifiable in the hidden layer activation space. We adopt two different kernels through CAV and CAR to train an SVM and then test the classifiers on unseen image data. Tables 7, 8, and 9 represent the test accuracies for the concepts extracted by Concept Induction, CLIP-Dissect, and GPT-4. Table 10 represents the results of the Mann-Whitney U test performed over the test accuracies obtained from all 3 approaches. Table 15 shows the Mean, Median, and Standard Deviation of the test accuracies for each of the 3 approaches.

Another option could be to show a summary table instead, where you report mean accuracy and std scores for each category, and move the table to the supplementary materials. In essence, in order to be useful, the tables need to visually convey what you want to compare. If you take the tables in isolation, and let them be read by an external reader, this table shows that sometimes CAR and CAV work better under the test accuracy metric, sometimes not. I'm not sure if this should be the purpose of these tables. Could you briefly comment on what do you believe their purpose is? In this way, I could provide a more precise advice on their presentation.

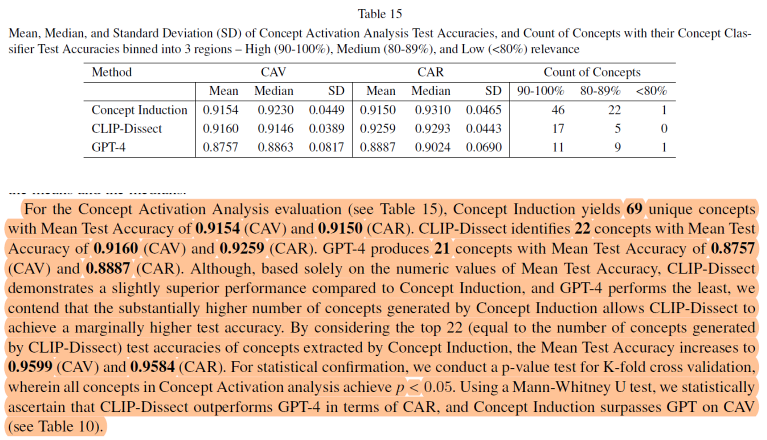


Table 15

Mean, Median, and Standard Deviation (SD) of Concept Activation Analysis Test Accuracies, and Count of Concepts with their Concept Classifier Test Accuracies binned into 3 regions – High (90-100%), Medium (80-89%), and Low (<80%) relevance

Method	CAV			CAR			Count of Concepts		
	Mean	Median	SD	Mean	Median	SD	90-100%	80-89%	<80%
Concept Induction	0.9154	0.9230	0.0449	0.9150	0.9310	0.0465	46	22	1
CLIP-Dissect	0.9160	0.9146	0.0389	0.9259	0.9293	0.0443	17	5	0
GPT-4	0.8757	0.8863	0.0817	0.8887	0.9024	0.0690	11	9	1

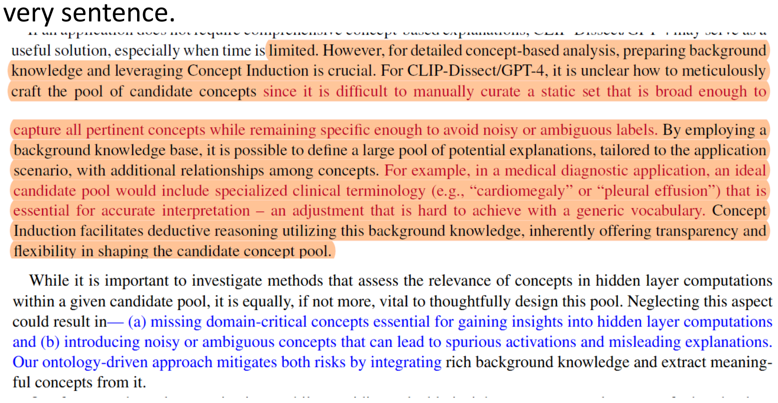
the means and the medians.

For the Concept Activation Analysis evaluation (see Table 15), Concept Induction yields **69** unique concepts with Mean Test Accuracy of **0.9154** (CAV) and **0.9150** (CAR). CLIP-Dissect identifies **22** concepts with Mean Test Accuracy of **0.9160** (CAV) and **0.9259** (CAR). GPT-4 produces **21** concepts with Mean Test Accuracy of **0.8757** (CAV) and **0.8887** (CAR). Although, based solely on the numeric values of Mean Test Accuracy, CLIP-Dissect demonstrates a slightly superior performance compared to Concept Induction, and GPT-4 performs the least, we contend that the substantially higher number of concepts generated by Concept Induction allows CLIP-Dissect to achieve a marginally higher test accuracy. By considering the top 22 (equal to the number of concepts generated by CLIP-Dissect) test accuracies of concepts extracted by Concept Induction, the Mean Test Accuracy increases to **0.9599** (CAV) and **0.9584** (CAR). For statistical confirmation, we conduct a p-value test for K-fold cross validation, wherein all concepts in Concept Activation analysis achieve $p < 0.05$. Using a Mann-Whitney U test, we statistically ascertain that CLIP-Dissect outperforms GPT-4 in terms of CAR, and Concept Induction surpasses GPT on CAV (see Table 10).

Q5: 9c. P27,r9: "it is equally vital to thoughtfully design this pool"; could you better explain what are the risks of a poorly designed pool?

A5: As the manuscript says, "neglecting this aspect results in overlooking crucial concepts essential for gaining insights into hidden layer computations." From an external reader, this sentence seems fuzzy and not precise enough; my request was simply to expand this explanation to make it more intuitive to an external reader, by adding additional context.

While the preceding paragraph already illustrates via a medical-diagnosis example why careful pool curation matters, we have now also explicitly named the two concrete risks in the very sentence.



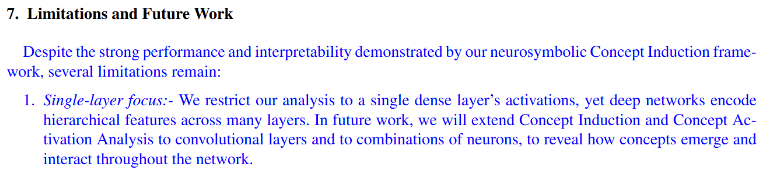
If an application does not require comprehensive concept-based explanations, CLIP-Dissect/GPT-4 may serve as a useful solution, especially when time is limited. However, for detailed concept-based analysis, preparing background knowledge and leveraging Concept Induction is crucial. For CLIP-Dissect/GPT-4, it is unclear how to meticulously craft the pool of candidate concepts since it is difficult to manually curate a static set that is broad enough to capture all pertinent concepts while remaining specific enough to avoid noisy or ambiguous labels. By employing a background knowledge base, it is possible to define a large pool of potential explanations, tailored to the application scenario, with additional relationships among concepts. For example, in a medical diagnostic application, an ideal candidate pool would include specialized clinical terminology (e.g., "cardiomegaly" or "pleural effusion") that is essential for accurate interpretation – an adjustment that is hard to achieve with a generic vocabulary. Concept Induction facilitates deductive reasoning utilizing this background knowledge, inherently offering transparency and flexibility in shaping the candidate concept pool.

While it is important to investigate methods that assess the relevance of concepts in hidden layer computations within a given candidate pool, it is equally, if not more, vital to thoughtfully design this pool. Neglecting this aspect could result in— (a) missing domain-critical concepts essential for gaining insights into hidden layer computations and (b) introducing noisy or ambiguous concepts that can lead to spurious activations and misleading explanations. Our ontology-driven approach mitigates both risks by integrating rich background knowledge and extract meaningful concepts from it.

Q6: The limitations of the work could be summed up in a specific section at the end of the paper (e.g.: activation patterns involving more than one neuron, requirement of labeled data, single dataset analysis, concept formation across multiple layers). Mitigations and/or suggestions for implementing these improvements could be reported as well.

A6: I believe it would be helpful to have such a section, at the very bottom of the paper (before conclusions), to sum up concisely all the limitations of the methods presented. They should encompass all the previous sections presented.

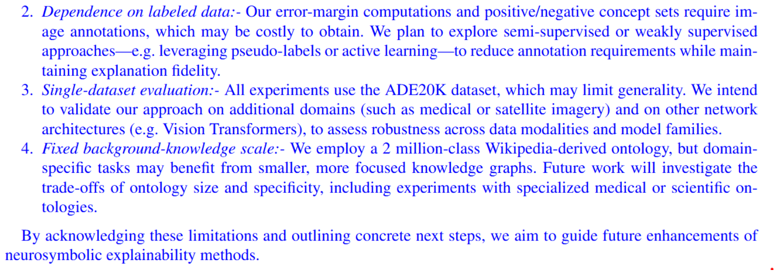
Thank you for this suggestion. We agree that an explicit, consolidated "Limitations and Future Work" section will help readers quickly see the boundaries of our current study and our plans to address them. Accordingly, we have added a new Section 7 just before the Conclusion.



7. Limitations and Future Work

Despite the strong performance and interpretability demonstrated by our neurosymbolic Concept Induction framework, several limitations remain:

1. *Single-layer focus:-* We restrict our analysis to a single dense layer's activations, yet deep networks encode hierarchical features across many layers. In future work, we will extend Concept Induction and Concept Activation Analysis to convolutional layers and to combinations of neurons, to reveal how concepts emerge and interact throughout the network.

	 2. *Dependence on labeled data:-* Our error-margin computations and positive/negative concept sets require image annotations, which may be costly to obtain. We plan to explore semi-supervised or weakly supervised approaches—e.g. leveraging pseudo-labels or active learning—to reduce annotation requirements while maintaining explanation fidelity. 3. *Single-dataset evaluation:-* All experiments use the ADE20K dataset, which may limit generality. We intend to validate our approach on additional domains (such as medical or satellite imagery) and on other network architectures (e.g. Vision Transformers), to assess robustness across data modalities and model families. 4. *Fixed background-knowledge scale:-* We employ a 2 million-class Wikipedia-derived ontology, but domain-specific tasks may benefit from smaller, more focused knowledge graphs. Future work will investigate the trade-offs of ontology size and specificity, including experiments with specialized medical or scientific ontologies. By acknowledging these limitations and outlining concrete next steps, we aim to guide future enhancements of neurosymbolic explainability methods.
Q7: 1. Regarding the CAR non-linear kernel, some details (e.g., the value chosen for the bandwidth of the RBF kernel) are missing. **A7: I could not find an updated reference into the paper (I could have missed it since it was not pointed out by the authors in their answer). I advise the authors to fully disclose the hyperparameters of their kernel methods to enhance the reproducibility of their work.**	CAR classifiers efficiency does not largely depend on Kernel width. For kernel width tests by using Bayesian optimization and a validation concept — it does not vary the results in any significant way. We have found that Gaussian RBF kernels indeed gives the best result over linear/polynomial. Number of examples to train a concept classifier - we have seen that anything above 200 results in diminishing rate of return.