Logic-induced Diagnostic Reasoning for Semi-supervised Semantic Segmentation

Chen Liang, Wenguan Wang, Jiaxu Miao, Yi Yang*
ReLER, CCAI, Zhejiang University
https://github.com/leonnnop/LogicDiag

Abstract

Recent advances in semi-supervised semantic segmentation have been heavily reliant on pseudo labeling to compensate for limited labeled data, disregarding the valuable relational knowledge among semantic concepts. To bridge this gap, we devise LOGIC DIAG, a brand new neural-logic semi-supervised learning framework. Our key insight is that conflicts within pseudo labels, identified through symbolic knowledge, can serve as strong yet commonly ignored learning signals. LOGIC DIAG resolves such conflicts via reasoning with logic-induced diagnoses, enabling the recovery of (potentially) erroneous pseudo labels, ultimately alleviating the notorious error accumulation problem. We showcase the practical application of LOGIC DIAG in the data-hungry segmentation scenario, where we formalize the structured abstraction of semantic concepts as a set of logic rules. Extensive experiments on three standard semi-supervised semantic segmentation benchmarks demonstrate the effectiveness and generality of LOGIC DIAG. Moreover, LOGIC DIAG highlights the promising opportunities arising from the systematic integration of symbolic reasoning into the prevalent statistical, neural learning approaches.

1. Introduction

Deep learning has revolutionized computer vision tasks, yielding remarkable breakthroughs [1–5]. However, such advances are often only possible in the presence of large labeled training datasets, which are challenging to acquire. Semantic segmentation, in particular, poses difficulties due to the need for pixel-level manual labeling [6–13], which is time-consuming and labor-intensive, precluding the application of such methods, especially in domains like medical image analysis. To remedy this issue, there has been a growing interest in semi-supervised semantic segmentation, which aims to train segmentation models using a combination of limited labeled data and a large amount of unlabeled data [14–20]. Pseudo labeling [21, 22] constitutes one such technique, where the unlabeled data is assigned pseudo labels based on model predictions. The model is then iteratively trained using these pseudo labeled data as if they were labeled examples. Typically, this method is implemented within the teacher-student framework [17] (Fig. 1 (a)).

Despite its prevalence, the pseudo labeling paradigm, a statistical learning approach, is commonly perceived as unreliable, due to the accumulation of erroneous predictions [23]. Previous works [22, 24, 25] have attempted to mitigate this issue by rejecting pseudo labels with classification scores below a heuristic threshold, known as confidence thresholding [22]. However, relying solely on this strategy often proves unsatisfactory. First, the confirmation bias that may occur during the early stages of training can be difficult, if not impossible, to be rectified in subsequent learning [26]. Second, the purely data-driven nature of such threshold-based methods makes them challenging to interpret. Third, to achieve optimal performance, the threshold must be manually adjusted for each model and dataset, lim-
iting its practical application. Beyond the confines, symbolic reasoning, as a compelling alternative, offers appealing characteristics: it requires little or no data to generalize systematically. In light of this background, we suggest a promising direction towards an integration of statistical learning and symbolic reasoning to harness the strengths of both paradigms to achieve improved performance.

In this paper, we propose to explicitly compile the rich symbolic knowledge into the prevalent pseudo label based neural training regime (cf. Fig. 1 (b)). Our key insight is that symbolic knowledge can effectively resolve conflicts within the pseudo labels, which reveal potential model errors. For instance, leveraging the prior knowledge on compositionality, we can naturally identify the inconsistencies like classifying a pixel as both a cat and a vehicle (cf. Fig. 1 (c)). Correcting these errors progressively enhances the accuracy of pseudo labels, and thus mitigating the confirmation bias in iterative self-training. Building upon this insight, we introduce LOGIC DIAG, a new SSL framework that employs logical reasoning to identify such conflicts based on the symbolic knowledge expressed in the form of first-order logic. The framework then suggests possible diagnoses to rectify the prediction. To further address the challenge of multiple diagnoses, we model the likelihood of each diagnosis being the actual fault based on a comprehensive measure of both predictive confidence and degree of conflicts according to the fuzzy logic. By doing so, our approach introduces a holistic neural-logic machine, that consolidates the benefits of powerful declarative languages, transparent internal functionality, and enhanced model performance.

LOGIC DIAG is a principled framework that seamlessly integrates with mainstream semi-supervised learning methods. It requires only minor adjustments to the dense classification head. With LOGIC DIAG, we can easily describe diverse symbolic knowledge using first-order logic and inject it into sub-symbolic pipelines. Taking dense segmentation as main battlefield, we capture and evaluate a central cognitive ability of human, i.e., structured abstractions of visual concepts [27], by grounding three logic rules onto LOGIC DIAG: Composition, Decomposition, and Exclusion.

Extensive experiments have validated the effectiveness of our LOGIC DIAG, which exhibit solid performance gains (i.e., 1.21%-4.15% mIoU) on three well-established benchmarks (i.e., PASCAL VOC 2012 [28], Cityscapes [29], and COCO [30]). We particularly observe significant improvements in the label scarce settings, indicating LOGIC DIAG’s superior utilization of unlabeled data. Besides, when employed onto the existing SSL frameworks, e.g., AEL [15], MKD [14], the performance is consistently advanced. The results demonstrate the strong generality and promising performance of LOGIC DIAG, that also evidence the great potential of the integrated neural-symbolic computing in the fundamental large-scale semi-supervised segmentation.

2. Related Work

Semi-Supervised Semantic Segmentation. Recent years have witnessed remarkable progress in image-level semi-supervised learning, driven primarily by two paradigms: self-training [21, 31, 32] and consistency regularization [33, 34]. Despite the impressive outcomes they produce, both paradigms heavily rely on the quality of pseudo-labels generated by the network itself, which makes them susceptible to confirmation bias [23] and leads to error accumulation throughout training. Considerable efforts have been devoted to addressing this issue through confidence thresholding [22, 35–37], curriculum scheduling [25, 38], sample ensembling across training iterations [17, 39], multiple augmented views [24], and/or neighboring examples [40–42]. Some others introduce perturbations in data [22, 36, 37], feature [43], and/or network [34, 44]. Similarly, in the context of semi-supervised semantic segmentation, which provides a pixel-level interpretation of visual semantics, prevalent approaches also follow the self-training paradigm [45–48] and consistency regularization paradigm [49–52]. Very recent endeavors have further pushed forward the frontier through incorporating dense perturbations [14, 47, 51, 55, 56], contrastive supervisions [57–60], or coping with imbalanced and long-tailed nature of pixel samples [15, 61]. Although significant advancements have been made, the problem of error accumulation remains far from being fully resolved. Existing methods heavily rely on an empirical threshold to justify pseudo labels, lacking explicit manipulation of rich symbolic knowledge. One common oversight in existing methods is the neglect of informative structures between semantic concepts. As a consequence, these approaches often yield barely satisfactory results in terms of both model performance and generality. One notable exception is [62]. While a label hierarchy is incorporated, it is only used for reducing labeling costs of fine-grained classes, resulting in limited improvement. This work pursues an integrated neural-logic framework that addresses these fundamental limitations, providing a refreshing viewpoint on semi-supervised semantic segmentation.

Hierarchical Classification. Class-wise hierarchical dependencies have been studied in supervised tasks across several machine learning domains, e.g., functional genomics [63–65], text categorization [66], object recognition [67–69], image classification [70, 71]. In the computer vision field, the class taxonomy is mainly explored through: i) semantic-aligned label embedding [72–74]; ii) hierarchy-coherent loss constraint [71, 75, 76]; or iii) structured network architecture [77–79]. Among the previous efforts, only a few attempts [80–84] towards label hierarchy-aware semantic segmentation have been made. Whereas, they all heavily rely on labeled hierarchical data and/or specialized neural architectural design, making them impractical for use when merely a handful of labeled data is available, as in SSL. In
contrast, our algorithm exploits class taxonomy as the symbolic knowledge for diagnosing conflicts in model predictions, enabling comprehensive exploration of hierarchical relations in both labeled and unlabeled data. The general design also makes our algorithm versatile and easily implementable to standard hierarchy-agnostic SSL architectures.

**Neural-Symbolic Computing.** Building preferable computational methods for integrated statistical learning and symbolic reasoning is a long-standing challenge [85]. This active line of research, namely neural-symbolic computing (NSC), draws soaring attention in recent years [86–89]. NSC shows great potential to reconcile the robust learning capabilities of neural networks with the interpretability and reasoning abilities of symbolic representations [90, 91], and thereby gains widespread recognition as a catalyst for the next generation of AI [92, 93]. NSC has demonstrated its superiority across a wide range of domains, including mathematical reasoning [94–96], robotics control [97–99], as well as scientific discovery [100–103]. Its virtue of data efficiency has also attracted researchers from SSL fields [104–107], where the symbolic knowledge, usually expressed in logic, is mostly incorporated as a form of regularization through loss constraints applied to output space. Though being exciting, the advances have been primarily limited to “toy” tasks. The full potential and challenges of NSC for large-scale realistic problems remain largely unexplored. Our method is partly motivated by, but also distinct from, previous efforts that merely encourage valid output structures in a soft manner, thereby allowing errors to accumulate, especially during the initial stages of training.

To our best knowledge, this is the first work that promotes and implements an integrated neural-symbolic framework in large-scale vision-oriented SSL. Previous studies have primarily concentrated on the neural aspect, i.e., sub-symbolic methods. In contrast, our method explicitly compiles symbolic knowledge into training regime of the neural network. This integration allows us to leverage the advantages of powerful declarative languages and transparent internal functionality. The encouraging results we have obtained provide compelling empirical evidence of the significant potential of NSC in the large-scale vision domain.

### 3. Methodology

We commence by formalizing modern semi-supervised semantic segmentation approaches, situating them within a sub-symbolic framework and highlighting their inherent limitations (§3.1). We then present LOGICDIAG, a general logic-induced diagnosis framework, that complements the current sub-symbolic approaches with a principled infusion of symbolic knowledge in the form of logic rules (§3.2). Finally, we showcase its practical application in the realm of visual semantic interpretation (§3.3).

**Problem Statement.** In the standard SSL setting, given an unknown distribution over visual space $\mathcal{X}$ (e.g., pixel space for segmentation) and category label space $\mathcal{Y} = \{1, \cdots, C\}$ with $C$ semantic categories, the goal is to find a predictor $h: \mathcal{X} \rightarrow \mathcal{Y}$, such that the generalization error is minimized, based on the observed data $\mathcal{D}$, consisting of a labeled subset $\mathcal{D}^l = \{(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}\}_{i=1}^{N_l}$ and an unlabeled subset $\mathcal{D}^u = \{u_j \in \mathcal{X}\}_{j=1}^{N_u}$. Typically, $N_u$ is rather small, i.e., $N_u \gg N_l$, resulting in the issue of insufficient learning signal.

#### 3.1. Sub-symbolic Semi-supervised Learning

Mainstream solutions predominantly revolve around a neural pipeline, known as sub-symbolic methods [108]. They rely on the consistency regularization paradigm [22, 109], which comprises four key components, as shown in Fig. 2 (a).

- **Data augmentors $A(\cdot)\alpha(\cdot)$**, that transform examples into strongly-/weakly-augmented views, respectively.
- **Base encoder $f(\cdot)$**, that maps augmented views into $D$-dimensional representations, i.e., $x = f \circ \alpha(x) \in \mathbb{R}^D$.
- **Prediction head $g(\cdot)$**, that gives the predictive probability distribution from the representation, i.e., $o = g(x) \in \Delta^C$, and $\Delta^C$ is the $C$-way probability simplex.
- **Pseudo label processor $\psi(\cdot)$**, that converts the raw predictions into pseudo labels guided by heuristic priors or assumptions, e.g., confidence thresholding [22, 24, 25, 35].

The predictor $h$, consisting of the base encoder and prediction head (i.e., $h = g \circ f$), is jointly optimized on the complete observed set $\mathcal{D}$, i.e., $\{\mathcal{D}^l, \mathcal{D}^u\}$. For the labeled ones, the standard cross entropy, denoted by $H(\cdot, \cdot)$, can be directly applied on the weakly augmented examples:

$$L^l = \frac{1}{N_l} \sum_{i=1}^{N_l} H((h \circ \alpha)(x_i), y_i).$$  \hspace{1cm} (1)$$

For the unlabeled data, the pseudo labels are generated from the weakly augmented views of given examples. As a prevalent choice [109], the pseudo label processor $\psi(\cdot)$ filters out unreliable pseudo labels with a confidence threshold $\tau$, i.e., $\psi(o) = \mathbb{I}(\max(o) \geq \tau) \cdot \arg\max o$. Here $\mathbb{I}$ defines the identify function. Then, the predictor $h$ is expected to yield consistent predictions on the strongly augmented views:

$$L^u = \frac{1}{N_u} \sum_{j=1}^{N_u} H((h \circ A)(u_j), (\psi \circ h \circ \alpha)(u_j)),$$  \hspace{1cm} (2)$$

The entire network is supervised with both the two losses:

$$L = L^l + \lambda L^u,$$  \hspace{1cm} (3)$$

where the scalar hyperparameter $\lambda$ traded off the two terms. Despite the prevalence it achieved, there remains three issues in mainstream SSL methods: First, they simply treat all categories equally in a flat view, overlooking the structured relationships between the visual concepts. The transferable knowledge residing in the label hierarchy is also discarded, which is of particular essential in the label scarce scenarios. Second, self-training with pseudo labels in turn...
(Eq. 2) inevitably accumulates errors and causes confirmation bias [23], which severely affects the model performance. Although low-confidence filtering (i.e., ψ) can alleviate the problem, errors accumulated in the early training stage are difficult to be corrected in subsequent training, especially for the poorly-behaved categories [26]. Third, the label filtering process relies on the empirical adjustment of a threshold τ, which requires specialization for each model and dataset to promote the performance. This significantly limits the flexibility and generality of current methods.

Accordingly, we suggest that it is now imperative to rethink prevailing sub-symbolic pipeline, in which the filtering process ψ(·) has been more detrimental than beneficial.

3.2. Logic-induced Diagnostic Reasoning

A well-performed SSL model not only generates accurate sample labels but also aligns with our background knowledge of the world, e.g., bird is an animal, sky is above the grass. By resolving conflicting predictions through reasoning based on the knowledge, we can potentially correct erroneous pseudo-labels, which leads to a more effective learning based on the knowledge, we can potentially correct errors accumulated in the early training process of the neural aspect (i.e., ω(·)), which reasons about malfunctions, dubbed conflict, in the model’s output (i.e., pseudo label) according to the symbolic knowledge, and offers possible solutions, dubbed diagnosis [110], to resolve the conflicts within predictions.

3.2.1 Diagnosis Computation

We formalize our target with a triple \( S = (\mathcal{D}_x, h, \mathcal{K}) \):

- \( \mathcal{D}_x = \{x_i\}_{i=1}^{N_1+N_n} \) defines the collection of pixel samples from both labeled and unlabeled datasets;
- \( h \) is the neural predictor, that generates a set of binary pseudo labels, denoted as \( \mathcal{O} = \{o_j\}_{j=1}^{\vert \mathcal{O} \vert} \), with one label for each semantic concept such as bird, animal, etc. Each \( o \) determines whether a given pixel \( x \) belongs to a specific concept or not, which can be achieved by applying a binarizing operation \( b \), to the output of \( h \), i.e., \( \mathcal{O} = b \circ h(x) \).

- \( \mathcal{K} \) is a finite set of rules expressed in first-order logic (FOL) that captures the world knowledge on normality. We provide a gentle introduction on FOL, which comprises four parts: i) constants representing specific pixel instances \( x_i \) or truth (i.e., \( \top: \text{true, } \bot: \text{false} \)); ii) variables ranging over these constants, denoted by \( x \); iii) predicates evaluating the semantics of variables to be true or false (e.g., \( \text{bird}(x) \) is true states the fact that pixel \( x \) belongs to a category \( \text{bird} \)); iv) connectives (e.g., \( \land: \text{and, } \lor: \text{or, } \neg: \text{not, } \rightarrow: \text{implication} \)) and quantifiers (i.e., \( \forall: \text{for all, } \exists: \text{exist} \)) over finite predicates.

To simplify subsequent formulations without altering the meaning, we may omit the explicit mention of the pixel variable \( x \) in certain predicates (e.g., \( o(x) \) to \( o \)).

A conflict arises when the assumption that all outputs are normal is inconsistent with our symbolic knowledge \( \mathcal{K} \):

\[
\mathcal{K} \land \mathcal{O} \land \bigwedge_{o \in \mathcal{O}} t(o) \models \bot. \tag{4}
\]

Here \( \models \) is logical entailment. We define the unary predicate \( t(\cdot) \) over the output \( o \) such that \( t(o) \) is true when \( o \) is normal in terms of consistency, while \( \neg t(o) \) is true when \( o \) is faulty.

To explain the inconsistency above, a diagnosis is proposed by assuming that some outputs in a set \( \omega \subseteq \mathcal{O} \) are faulty:

\[
\mathcal{K} \land \mathcal{O} \land \bigwedge_{o \in \omega} \neg t(o) \land \bigwedge_{o \in \mathcal{O}\setminus \omega} t(o) \not\models \bot. \tag{5}
\]

For all possible diagnoses, we restrict our focus solely to the ones that contain only false elements, in the world being modeled, known as minimal diagnosis [110]. Formally, a diagnosis \( \omega \) is minimal iff. any strict subset \( \omega' \subset \omega \) is not a diagnosis. Thus far, the problem has been reduced to a Boolean satisfiability problem. We present to solve it with greedy algorithms. In practice, considering the rather small search space (i.e., \( \mathcal{K} \) and \( \mathcal{O} \)), we keep our pipeline straightforward without employing sequential approximation (e.g., MCMC). At this point, a diagnosis has already

\[\text{Fig. 2: Illustrations of (a) sub-symbolic SSL pipeline (§3.1); (b) LOGIC DIAG, logic-induced diagnostic reasoning framework (§3.2).}\]
been computed and the pseudo label \( O \) can be revised to the logically consistent \( O' \) according to the following equation:

\[
O' = \{ o \}_{o \in \omega} \cup \{ o \}_{o \in O \setminus \omega}.
\]

However, there might be multiple diagnoses that are consistent with our target, it is not immediately clear which one is the 'correct' revision to make. A naïve approach would be to uniformly sample from all possible minimal diagnoses and revise them one at a time, which is inefficient and ineffective, as demonstrated by the empirical results (c.f. §4.3). To address this issue, we take one step further to model the likelihood of the diagnosis being the actual faulty.

### 3.2.2 Resolution with Fuzzy Diagnosis Likelihood

We can derive the likelihood of a correct diagnosis given the triple system \( S = (D_x, h, K) \) under the common assumption of independent failure [111], using the multiplication rule:

\[
P(\omega|S) = \prod_{o \in \omega} P(t(o)|S) \prod_{o \in \omega} (1 - P(t(o)|S)),
\]

where \( \omega \) is the diagnosis; \( \omega^+ \) and \( \omega^- \) are the sets of outputs assigned normal and abnormal behavior modes, respectively. To comprehensively estimate the probability of an output \( o \) being normal, denoted as \( P(t(o)|S) \), we consider both the degree of conflict and predictive distance:

\[
P(t(o)|S) = \begin{cases} P(o|S) \cdot (1 - c(o; S)) & \text{if } o \models \top \\ (1 - P(o|S)) \cdot (1 - c(o; S)) & \text{if } o \models \bot \end{cases}
\]

Here, function \( c \) maps each output \( o \in O \) to its conflict degree \( c(o; S) \in [0, 1] \); and \( P(o|S) = P(o|x) = h(x) \in [0, 1] \) represents the predictive probability. Intuitively, if an output violates the established knowledge but has a high predictive confidence, it should be penalized more than the others (i.e., a small valued \( c(o; S) \)), increasing the likelihood of correct diagnosis. Conversely, high confidence indicates low probability in abnormality, thereby balancing the likelihood.

To measure this degree of conflict, we resort to the fuzzy logic [112], a form of soft probabilistic logic, specifically, the Goguen fuzzy logic [113] and Gödel fuzzy logic [114]. Fuzzy logic generalizes FOL to uncertain inputs, where the variables have truth values in \([0, 1] \), e.g., the predictive probability with respect to each visual concept in our case. The logical connectives (e.g., \( \land, \lor, \neg \)) are approximated with fuzzy operators (i.e., \( t\text{-norm}, \ t\text{-conorm}, \ fuzzy \negation \)):

\[
\phi \land \varphi := \phi \cdot \varphi, \ \phi \lor \varphi := \max(\phi, \varphi), \ \neg \phi := 1 - \phi.
\]

Besides, the existential and universal quantifiers (i.e., \( \exists, \forall \)) are approximated in a form of generalized mean [115, 116]:

\[
\exists \phi(x) := \left( \frac{1}{|D_x|} \sum_{x \in D_x} \phi(x)^q \right)^{\frac{1}{q}},
\]

\[
\forall \phi(x) := 1 - \left( \frac{1}{|D_x|} \sum_{x \in D_x} (1 - \phi(x)^q) \right)^{\frac{1}{q}},
\]

where \( q \in \mathbb{Z} \). Instead of being strictly true or false, fuzzy logic offers a soft measure on how much a logic rule is fired, i.e., truth degree, which can be interpreted as the complement to the desired conflict degree:

\[
c(o; S) = 1 - \frac{1}{|K|} \sum_{k \in K} g(o, k; S),
\]

where we define \( g(\cdot) \) as the fuzzy truth measurement.

Following the derivation, we are now ready to assess the extent to which a series of outputs violate the rules in our knowledge \( K \) (c.f. Eq. 11), so as to estimate the likelihood of actual diagnoses (c.f. Eq. 7-8). Then, the naïve utilization is to select the diagnosis with the highest probability concerning pixel \( x_i \), i.e., \( \arg \max_{o \in O} P(o|x_i, h, K) \). However, such local estimation is prone to get stuck in spurious corrections, as evidenced by our experiments (c.f. §4.3). Instead, we resort to Monte Carlo estimation to calculate the posterior distribution. Our empirical findings suggest that optimizing with just one sample per datapoint is sufficient.

Overall, as shown in Fig. 2 (b), the reasoning and learning aspects of LOGICDIAG work iteratively. The neural model \( h(\cdot) \) first poses assumptions about current observations, then the symbolic model \( \psi(\cdot) \) reasons over the symbolic knowledge \( K \) to determine the diagnoses, which in turn facilitates the learning process of the neural model.

### 3.3. Diagnosis with Visual Semantics: An Example

LOGICDIAG is model-agnostic to the neural aspect of SSL pipelines, and is also compatible with general symbolic knowledge described in FOL. As a result, LOGICDIAG can further be advanced by embracing the development of new architectures or incorporating more knowledge. To showcase the practical deployment of LOGICDIAG, we examine its application in segmentation scenario, with particular interest in structured visual semantics [80] that are commonly overlooked in the mainstream. Specifically, the semantic concepts and their relations are formed as a tree-shaped label hierarchy \( T = (O, E) \) (c.f. Fig. 3 (a)). The node set \( O \) is the union of nodes from \( L \) levels of abstraction, denoted as \( O = \bigcup_{l=1}^{L} O_l \). The leaf nodes, \( O_1 \), represent the most specific concepts, namely category labels, such as bird, cat, where \( O_1 = \mathcal{V}, |O_1| = C \). The internal nodes represent higher-level concepts such as vehicle, animal, and the root nodes \( O_L \) represent the most abstract concepts, such as object. Besides, the edge set \( E \) encodes relational knowledge among all these concepts, with directed edges \( u \rightarrow v \in E \) denoting a part-of relation between two concepts \( u, v \in O \) in adjacent levels, e.g., animal→bird.

Recall that we define the \( \langle \text{data, model, knowledge} \rangle \) triple \( (D_x, h, K) \) for LOGICDIAG (c.f. §3.2.1). In the context of structured semantic concept exploration, the model \( h \) yields \( |O| \) binary pseudo labels, denoting nodes within the label hierarchy. Besides, \( K_T \) contains FOL rules, describing the
structured symbolic knowledge according to the label hierarchy $\mathcal{T}$. Inspired by previous efforts [80, 117, 118] in hierarchical classification, we define $K_\mathcal{T}$ with three types of rules, i.e., composition, decomposition, and exclusion.

- **Composition Rule ($K_C$).** If one class is labeled true, its parent (i.e., superclass) is labeled true (Fig. 3 (b)):
  \[ \forall x (o(x) \rightarrow p_0(x)), \]  
  (12)

  where $p_0$ is the parent node of $o$ in $\mathcal{T}$, i.e., $p_0 \rightarrow o \in \mathcal{E}$. For example, “bird is a subclass of animal” shall be interpreted as: $\forall x (\text{bird}(x) \rightarrow \text{animal}(x))$.

- **Decomposition Rule ($K_D$).** If one class is labeled true, at least one of its children (i.e., subclasses) is true (Fig. 3 (c)):
  \[ \forall x (o(x) \rightarrow \bigvee_{r_o \in R_o} r_o(x)), \]  
  (13)

  where $R_o$ is the set of children node(s) of $o$ in $\mathcal{T}$, i.e., $o \rightarrow r_o \in \mathcal{E}$. For example, “animal subsumes (is the superclass of) bird, dog, · · · , cat” shall be interpreted as: $\forall x (\text{animal}(x) \rightarrow \text{bird}(x) \lor \text{dog}(x) \lor \cdots \lor \text{cat}(x))$.

- **Exclusion Rule ($K_E$).** If one class is labeled true, all its sibling classes are labeled false (Fig. 3 (d)):
  \[ \forall x (o(x) \rightarrow \bigwedge_{s_o \in S_o} \neg s_o(x)), \]  
  (14)

  where $S_o$ is the set of sibling node(s) of $o$ in $\mathcal{T}$. For example, “bird cannot be train, · · · , nor cat” shall be interpreted as: $\forall x (\text{bird}(x) \rightarrow \neg \text{train}(x) \land \cdots \land \neg \text{cat}(x))$.

According to the fuzzy measure defined in Eq. 9-10, we have the truth degrees as follows (Note, $\phi \rightarrow \varphi \leftrightarrow \neg \phi \lor \varphi$):

- Composition Rule (Eq. 12): $g(o, K_C\|S) = \frac{1}{|D_o|} \sum_{x \in D_o} (P(o|x) - P(o|x) \cdot P(p_0|x))^\frac{3}{2}$.
  (15)

- Decomposition Rule (Eq. 13): $g(o, K_D\|S) = 1 - \frac{1}{|D_o|} \sum_{x \in D_o} (P(o|x) - P(o|x) \cdot \max_{r_o \in R_o} P(r_o|x))^\frac{3}{2}$.
  (16)

- Exclusion Rule (Eq. 14): $g(o, K_E\|S) = 1 - \frac{1}{|S_o|} \sum_{s_o \in S_o} \left[ \frac{1}{|D_o|} \sum_{x \in D_o} (P(o|x) \cdot P(s_o|x))^\frac{3}{2} \right]$.
  (17)

where we translate one-vs-multiple exclusion (Eq. 14) to the equivalent one-vs-one form to avoid sorites paradox [119].

We are now prepared to proceed with the calculations (c.f. Eq. 7, 8, 11) and sample from the diagnosis likelihood $P(\omega|S)$, where the pseudo labels can be revised accordingly (c.f.Eq. 6). Overall LOGIC DIAG is supervised with Eq. 3.

**Implementation Detail.** In practice, computing the full semantics of the universal quantification $\forall$ is infeasible due to the large learning corpora, where we use batch-training as sampling based approximation [115]. To facilitate efficient distributed training, we implement diagnostic reasoning steps using matrix multiplications. Besides, we employ point-wise supervision [121] to further optimize efficiency while maintaining high performance standards.

**4. Experiment**

**4.1. Experimental Setup**

**Datasets.** We evaluate LOGIC DIAG on standard datasets:

- **PASCAL VOC2012** [28] is a famous semantic segmentation dataset, consisting ~4k samples in original dataset, that are split into 1,464/1,449/1,456 images for train/val/test, respectively. It provides annotations for 21 categories including the background, which are grouped into 4 superclasses. Following the conventions [57, 109], 9,118 coarsely labeled images in SBD [122] are adopted to complement train data, namely the augmented set. For PASCAL VOC/Cityscapes, we sample 1/2, 1/4, 1/8, and 1/16 of the whole training set as labeled data. For COCO, we use smaller ratios, i.e., 1/32, 1/64, 1/128, 1/256, 1/512, considering the larger size of the dataset. We adopt the same sampled data to the state-of-the-arts [51, 57] to enable meaningful comparisons.

**Partition Protocol.** We conduct evaluation under standard partition protocols [57, 59]. For PASCAL VOC/Cityscapes, we sample 1/2, 1/4, 1/8, and 1/16 of the whole training set as labeled data. For COCO, we use smaller ratios, i.e., 1/32, 1/64, 1/128, 1/256, 1/512, considering the larger size of the dataset. We adopt the same sampled data to the state-of-the-arts [51, 57] to enable meaningful comparisons.

**Base Network Architecture.** We take DeepLabV3+ [120] as base segmentation architecture (i.e., $h(\cdot)$, c.f. §3), where ResNet101 [1] and Xception65 [123] pretrained on ImageNet.
Table 1: Quantitative results (§4.2) on PASCAL VOC 2012 [28] \textit{val}. All methods are built upon DeepLabV3+ [120]-ResNet101 [1].

<table>
<thead>
<tr>
<th>Method</th>
<th>PASCAL VOC 2012 original</th>
<th>PASCAL VOC 2012 augmented</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1/2 (732)</td>
<td>1/4 (366)</td>
</tr>
<tr>
<td>MT [17]</td>
<td>69.16</td>
<td>63.01</td>
</tr>
<tr>
<td>GCT [50]</td>
<td>70.67</td>
<td>64.71</td>
</tr>
<tr>
<td>CutMixSeg [49]</td>
<td>69.84</td>
<td>68.36</td>
</tr>
<tr>
<td>PseudoSeg [51]</td>
<td>72.41</td>
<td>69.14</td>
</tr>
<tr>
<td>AEL [15]</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>CPS [109]</td>
<td>75.88</td>
<td>71.71</td>
</tr>
<tr>
<td>PC2Seg [59]</td>
<td>73.05</td>
<td>69.78</td>
</tr>
<tr>
<td>ST++ [47]</td>
<td>77.30</td>
<td>74.60</td>
</tr>
<tr>
<td>U2PL [57]</td>
<td>76.16</td>
<td>73.66</td>
</tr>
<tr>
<td>GTA-Seg [16]</td>
<td>78.37</td>
<td>75.57</td>
</tr>
<tr>
<td>MKD [14]</td>
<td>78.66</td>
<td>76.76</td>
</tr>
<tr>
<td>\textbf{Ours}</td>
<td>\textbf{79.39} \uparrow 0.73</td>
<td>\textbf{77.93} \uparrow 1.17</td>
</tr>
<tr>
<td>\textbf{Ours} + AEL [15]</td>
<td>\textbf{79.56} \uparrow 0.17</td>
<td>\textbf{78.10} \uparrow 1.23</td>
</tr>
<tr>
<td>\textbf{Ours} + MKD [14]</td>
<td>\textbf{80.06} \uparrow 0.67</td>
<td>\textbf{78.43} \uparrow 0.50</td>
</tr>
</tbody>
</table>

Table 2: Quantitative results (§4.2) on Cityscapes [29] \textit{val}.

<table>
<thead>
<tr>
<th>Method</th>
<th>1/2 (1488)</th>
<th>1/4 (744)</th>
<th>1/8 (372)</th>
<th>1/16 (186)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MT [17]</td>
<td>78.59</td>
<td>76.53</td>
<td>73.71</td>
<td>68.08</td>
</tr>
<tr>
<td>C2CT [52]</td>
<td>78.29</td>
<td>76.35</td>
<td>74.48</td>
<td>69.64</td>
</tr>
<tr>
<td>GCT [50]</td>
<td>78.58</td>
<td>76.45</td>
<td>72.96</td>
<td>66.90</td>
</tr>
<tr>
<td>CutMixSeg [49]</td>
<td>78.95</td>
<td>77.24</td>
<td>75.83</td>
<td>72.13</td>
</tr>
<tr>
<td>AEL [15]</td>
<td>80.28</td>
<td>79.01</td>
<td>77.90</td>
<td>75.83</td>
</tr>
<tr>
<td>CPS [109]</td>
<td>76.81</td>
<td>74.58</td>
<td>74.31</td>
<td>69.78</td>
</tr>
<tr>
<td>S-Baseline [46]</td>
<td>78.70</td>
<td>77.80</td>
<td>74.10</td>
<td>-</td>
</tr>
<tr>
<td>U2PL [57]</td>
<td>79.05</td>
<td>76.47</td>
<td>74.37</td>
<td>70.30</td>
</tr>
<tr>
<td>GTA-Seg [16]</td>
<td>76.08</td>
<td>72.02</td>
<td>69.38</td>
<td>62.95</td>
</tr>
<tr>
<td>MKD [14]</td>
<td>80.74</td>
<td>78.28</td>
<td>75.98</td>
<td>75.31</td>
</tr>
<tr>
<td>\textbf{Ours}</td>
<td>\textbf{80.95}</td>
<td>\textbf{80.21}</td>
<td>\textbf{78.90}</td>
<td>\textbf{76.83}</td>
</tr>
</tbody>
</table>


Reproducibility. Our models are implemented in PyTorch. All experiments are conducted on four Tesla A100 GPUs.

4.2. Quantitative Comparison Result

PASCAL VOC 2012 original. Table 1 (left) summarizes the quantitative comparisons under varying label amounts, from which we take three major observations: First, Our method indeed surpasses the SOTAs across all settings and establishes the new state-of-the-arts of \textbf{79.39\%}/\textbf{77.93\%}/\textbf{76.66\%}/\textbf{73.25\%} under 1/2-1/16 partitions, indicating the effectiveness of our neural-logic framework. Second, The superiority of our method is more significant on fewer labeled data, with the largest margin achieved when only 1/16 labeled data are available, \textit{i.e.}, \textbf{4.15\%} over MKD, demonstrating its potential in extremely label-scarce scenarios. Third, Our framework is fully compatible with mainstream SSL methods [14, 57]. The performance can be consistently lifted when equipped with additional perturbation techniques.

PASCAL VOC 2012 augmented. Table 1 (right) demonstrates the comparison results on the PASCAL VOC 2012 \textit{val} using \textit{augmented} set for training, where our method again sets new state-of-the-arts across all partition protocols, yielding an average gain of \textbf{0.80\%} upon the previous SOTA [14]. When only provided with scarce labeled data, \textit{i.e.}, 92 labeled images, our method still performs impressive owing to the compactly incorporated symbolic logic.
against the competitors on Cityscapes val. In spite of the presence of complex street scenes, our method still delivers a solid overtaking trend across different partitions, with an averaged advancement of **1.65%** in terms of mIoU.

**COCO.** Table 3 presents the model performance on COCO val. As observed, by incorporating the large semantic hierarchy in COCO, LOGIC DIAG provides even greater performance gains against the leading method (i.e., MKD [14]) across all partitions by **2.94%** mIoU on average. The experimental results confirm again the efficacy of LOGIC DIAG.

### 4.3. Diagnostic Experiment

For in-depth analysis, we perform a set of ablative studies on PASCAL VOC 2012 [28] val with 1/16 augmented set. Please refer to the supplementary for more experiments. **Key Component Analysis.** In Table 4a, we first validate the importance of our proposed components by attaching them one at a time. The 1st row reports the result of a bare baseline model - DeepLabV3+ with plain consistency regularization [22]. Next, in the 2nd row, we convert the prediction mode from flat to hierarchical, which already boosts performance and supports our claim that hierarchical semantics can provide additional training signals implicitly. Moreover, the 3rd row gives the score when the minimal diagnosis set is further computed, and we uniformly sample one conflict from it to resolve. As seen, this leads to moderate improvement caused by the explicit introduction of symbolic knowledge that potentially resolves conflicts. Finally, as shown in the 4th row, through sampling from the diagnosis likelihood, the biggest improvement is achieved, demonstrating the necessity of the fuzzy measurement that guides the resolution.

**Semantic Logic Rules.** Then, we investigate the effectiveness of logic-induced hierarchy rules (§3.3) in Table 4b. Starting from the baseline (1st row), we individually add Composition (cf. Eq. 12), Decomposition (cf. Eq. 13), and Exclusion (cf. Eq. 14) rules, denoted as $\mathcal{K}_C$, $\mathcal{K}_D$, $\mathcal{K}_E$, resp., into the proposed framework, resulting in the scores listed in the 2nd to 4th rows. The last row exhibits the outcome achieved with our full training regime. Upon examining the table, three observations can be made. **First,** incorporating each of the logic rule results in consistent performance gains, indicating that the set of rules captures diverse facets of visual semantics, and indeed benefits SSL models within our symbolic resolution framework. **Second,** the best performance is attained by combining all three logic rules, highlighting the significance of comprehensive interpretation of structured semantic concepts. **Third,** this also implies that integrating extra symbolic knowledge has great potential to further enhance current sub-symbolic SSL.

**Resolution Strategy.** Table 4c reveals the impact of conflict resolution strategies (§3.2.2). The default strategy, ‘Sampling’, utilizes the Monte Carlo method to estimate the actual posterior distribution $P(\omega|S)$ (cf. Eq. 7). Here we investigate three alternatives: ‘Uniform’, which assigns equal weight to all valid diagnoses; ‘Predictive’, which samples solely based on predictive probability $P(o|S)$ without considering the degree of conflicts; ‘Greedy’, which resolves conflicts by prioritizing the highest probability according to $P(\omega|S)$. The results show that ‘Predictive’, ‘Greedy’, and ‘Sampling’ all outperform ‘Uniform’, indicating the necessity of likelihood modeling. Of these strategies, ‘Sampling’ stands out as the most effective, providing strong evidence for its capacity to escape from the spurious corrections.

**Training Speed.** For completeness, we include the training time of 10K iterations in the last column of Table 4a. Our training regime (the last row) incurs a trivial delay of ~4.7%.

**Inference Speed.** During inference, the ancestral classes in the hierarchical classification head can be safely disregarded without introducing additional computational burden.

### 5. Conclusion

This paper introduces LOGIC DIAG, a neural-logic SSL framework that consolidates the benefits from both symbolic reasoning and sub-symbolic learning. Through resolving conflicts within pseudo labels using logic-induced diagnoses, LOGIC DIAG systematically compiles rich symbolic knowledge into the neural training pipeline. Experimental findings illustrate that LOGIC DIAG outperforms existing SSL frameworks, especially in label scarce settings. The enhanced flexibility and generality of LOGIC DIAG showcase the immense potential of this holistic neural-logic paradigm in pixel-wise semi-supervised learning. We believe this paper opens a new avenue for future exploration in the field.

**Acknowledgement.** This work was partly supported by the Fundamental Research Funds for the Central Universities (No. 226-2022-00051), and the National Natural Science Foundation of China (No. 62293554).
References


[17] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In NeurIPS, 2017. 1, 2, 7


[29] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In CVPR, 2016. 2, 6, 7

[37] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In CVPR, 2020. 2
[38] Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Yu-Feng Li, Baigui Sun, Hao Li, and Rong Jin. Dash: Semi-supervised deep learning with dynamic thresholding. In ICML, 2021. 2

[65] Yu Li, Sheng Wang, Ramzan Umarow, Bingqing Xie, Ming Fan, Lihua Li, and Xin Gao. Deepre: sequence-based enzyme ec number prediction by deep learning. Bioinformatics, 2018. 2


[71] Jia Deng, Alexander C Berg, Kai Li, and Li Fei-Fei. What does classifying more than 10,000 image categories tell us? In ECCV, 2010. 2


[74] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In CVPR, 2015. 2


[76] Bin Zhao, Fei Li, and Eric Xing. Large-scale category structure aware image categorization. In NeurIPS, 2011. 2


[95] Qing Li, Siyuan Huang, Yining Hong, Yixin Chen, Ying Nian Wu, and Song-Chun Zhu. Closed loop neural-symbolic learning via integrating neural perception, grammar parsing, and symbolic reasoning. In ICLR, 2020. 3
