

Spatial Knowledge-Infused Hierarchical Learning: An Application in Flood Mapping on Earth Imagery

Zelin Xu
zelin.xu@ufl.edu
Department of CISE
University of Florida
Gainesville, Florida, USA

Tingsong Xiao
xiaotingsong@ufl.edu
Department of CISE
University of Florida
Gainesville, Florida, USA

Wenchong He
whe2@ufl.edu
Department of CISE
University of Florida
Gainesville, Florida, USA

Yu Wang
yuwang1@ufl.edu
Department of MAE
University of Florida
Gainesville, Florida, USA

Zhe Jiang
zhe.jiang@ufl.edu
Department of CISE
University of Florida
Gainesville, Florida, USA

ABSTRACT

Deep learning for Earth imagery plays an increasingly important role in geoscience applications such as agriculture, ecology, and natural disaster management. Still, progress is often hindered by the limited training labels. Given Earth imagery with limited training labels, a base deep neural network model, and a spatial knowledge base with label constraints, our problem is to infer the full labels while training the neural network. The problem is challenging due to the sparse and noisy input labels, spatial uncertainty within the label inference process, and high computational costs associated with a large number of sample locations. Existing works on neuro-symbolic models focus on integrating symbolic logic into neural networks (e.g., loss function, model architecture, and training label augmentation), but these methods do not fully address the challenges of spatial data (e.g., spatial uncertainty, the trade-off between spatial granularity and computational costs). To bridge this gap, we propose a novel Spatial Knowledge-Infused Hierarchical Learning (SKI-HL) framework that iteratively infers sample labels within a multi-resolution hierarchy. Our framework consists of a module to selectively infer labels in different resolutions based on spatial uncertainty and a module to train neural network parameters with uncertainty-aware multi-instance learning. Extensive experiments on real-world flood mapping datasets show that the proposed model outperforms several baseline methods. Codes are available at <https://github.com/ZelinXu2000/SKI-HL>

CCS CONCEPTS

• **Information systems** → *Geographic information systems; Data mining*; • **Computing methodologies** → *Machine learning*; • **Applied computing** → *Earth and atmospheric sciences*.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGSPATIAL '23, November 13–16, 2023, Hamburg, Germany
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0168-9/23/11...\$15.00
<https://doi.org/10.1145/3589132.3625591>

KEYWORDS

knowledge-infused learning, neural-symbolic systems, spatial data mining

ACM Reference Format:

Zelin Xu, Tingsong Xiao, Wenchong He, Yu Wang, and Zhe Jiang. 2023. Spatial Knowledge-Infused Hierarchical Learning: An Application in Flood Mapping on Earth Imagery. In *The 31st ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL '23)*, November 13–16, 2023, Hamburg, Germany. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3589132.3625591>

1 INTRODUCTION

Deep learning for spatial data (e.g., Earth imagery) plays an important role in many applications such as transportation [6, 8], agriculture [25, 29], ecology [10], urban planning [16], and natural hazards management [13, 30]. Unfortunately, one major bottleneck is that deep learning models require a large number of training labels (e.g., ImageNet), which is often unavailable in geoscience domains [18]. This paper studies the integration of spatial domain knowledge and deep learning to overcome limited training labels.

Given Earth imagery with limited training labels, a base deep neural network model, and a spatial knowledge base with label constraints, our problem is to infer the full labels while training the neural network. Figure 1 provides an example of flooding extent mapping. Data samples are Earth imagery pixels in a raster grid, and explanatory feature layers are spectral bands. Initial noisy labels can be from volunteered geographical information (e.g., geo-tagged tweets). These labels are sparse and limited, as collecting complete high-quality labels through manual annotation is impractical (e.g., high time costs, obscured view due to tree canopies near flood boundary). On the other hand, there exists spatial domain knowledge related to topographical constraints on floodwater distribution, e.g., if location A is flooded and location B is at a nearby lower location, then B is flooded. Similar examples exist in crop type classification [29] and tree crown delineation in forest ecology [10], land use classification [16], and ship detection [6].

However, the problem poses unique challenges. First, the input labels are spatially sparse and noisy, making it difficult to directly train a neural network on Earth imagery. For example, in flood mapping, in-situ water sensors are often located at a few locations.

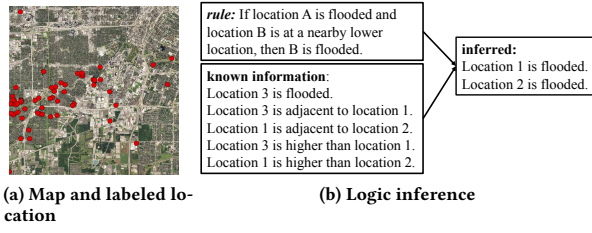


Figure 1: Spatial knowledge-infused problem illustration.

Second, spatial uncertainty is inherent in the knowledge-guided label inference process, which comes from the noise and sparsity of input labels, imperfect knowledge and rules, and grounding spatial rules on a coarse grid. Third, there are high computational costs associated with spatial logic inference on a large number of raster pixels and a trade-off has to be made between computational efficiency and spatial granularity.

The most closely related works are neural-symbolic systems, which integrate symbolic logical reasoning with deep neural networks [9]. Existing methods focus on replace the search process of symbolic reasoning to neural network [24, 26, 37], convert unstructured data, e.g., images into symbols for relational learning [28, 36], representing symbolic knowledge as a loss regularization term [2, 4, 5, 14, 34, 35, 38], or the combination of logic inference of pseudo-labels and neural network training iteratively [3, 15, 21, 22, 31, 32, 40]. However, these methods do not fully address the inherent challenges of spatial data, such as spatial uncertainty and the substantial computational burdens associated with logical inference over a massive number of samples (pixels).

To address the limitations of existing works, we propose **Spatial Knowledge-Infused Hierarchical Learning (SKI-HL)** that integrates deep learning techniques with spatial knowledge-infused label inference [1, 19]. SKI-HL consists of two main modules: the uncertainty-guided hierarchical label inference module and the uncertainty-aware deep learning module. The uncertainty-guided hierarchical label inference module captures spatial relationships and dependencies based on a spatial knowledge base and infers labels with quantified uncertainty. To handle the continuous space issue, we design a multi-resolution hierarchy to iteratively refine labels with a trade-off between granularity and computational efficiency. The uncertainty-aware deep learning module leverages complete but uncertain labels from the label inference module, capturing information from the data features that cannot be obtained through logical reasoning alone. Both modules are trained iteratively to refine inferred labels, reduce uncertainty, and improve deep learning model performance. In summary, the contributions of this paper are as follows:

- We propose SKI-HL, a spatial knowledge-infused framework that integrates deep learning and logical reasoning to leverage both explanatory features and spatial knowledge derived from domain logic rules.
- Our approach is designed to handle uncertainty in both the original labels and the label inference process, making it more robust and reliable.

- We propose a strategy to balance the trade-off between spatial accuracy and computational efficiency when discretizing continuous spatial spaces for constructing logic rules and training deep learning models.
- Taking the flood mapping problem as an example, extensive experiments on real-world datasets demonstrate the superior performance of our model compared to baseline methods.

2 PROBLEM STATEMENT

2.1 Preliminaries

Spatial Raster Framework: A spatial raster framework is a tessellation of a two-dimensional plane into a regular grid of N cells. The framework can consist of m non-spatial explanatory *feature layers* and *one class layer*. We denote the explanatory feature layers by $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ and the class layer by $\mathbf{Y} = \{y_1, y_2, \dots, y_N\}$, where $\mathbf{x}_i \in \mathbb{R}^{m \times 1}$ and y_i are the explanatory features, and class at cell i respectively. Each cell in a raster framework is a spatial data sample, note as $\mathbf{x} = (\mathbf{x}_i, y_i)$, where $i \in \mathbb{N}$, $1 \leq i \leq N$. For example, in the flood mapping problem, the explanatory features are the spectral bands from remote sensing imagery, the target classes are flood and dry categories, and each pixel in the image is a spatial sample.

Spatial Knowledge Base: Given a set of spatial samples, which can represent physical objects, locations, events, or any entity relevant to a spatial task, we can describe the relationships and dependencies between these samples using logical statements. This leads us to the concept of a spatial knowledge base. Before defining it formally, we first introduce some fundamental concepts in the logic framework.

DEFINITION 1. A **predicate** is a relation among objects in the domain or attributes of objects (e.g., *Neighbor*), and an **atom** is a predicate symbol applied to a tuple of terms (e.g., *Neighbor(A, B)*).

DEFINITION 2. A **rule** in logic is a clause recursively constructed from atoms using logical connectives and quantifiers. An example would be: *Neighbor(A, B) ∧ Neighbor(B, C) → Neighbor(A, C)*.

DEFINITION 3. A **ground atom** and a **ground rule** are specific variable instantiations of an atom and rule, respectively. A **grounding** of an atom or rule is a replacement of all of its arguments by constants.

With these preliminaries, we are now equipped to formally define a spatial knowledge base, $\mathcal{KB}$:

DEFINITION 4. A **spatial knowledge base** $\mathcal{KB}$ is a set of logic rules: $\mathcal{KB} = \{r_1, r_2, \dots, r_{|\mathcal{KB}|}\}$. Here, each r_i is a rule that represents a spatial relationship, dependency, or constraint between entities in the set of spatial samples $\mathcal{S}$. The quantity $|\mathcal{KB}|$ represents the number of rules in the spatial knowledge base $\mathcal{KB}$.

Table 1 provides an example of a spatial knowledge base used for a flood mapping on earth imagery problem. Here the variable s_i, s_j stands for a location in the study area or a pixel of earth imagery. It is important to clarify that these rules are probabilistic in nature, reflecting the likelihood of a flood occurrence under certain conditions, rather than providing an absolute certainty. In addition, although the rules are represented here in first-order logic, they can be converted into a Markov Logic Network [27] or Probabilistic

Table 1: An example of a spatial knowledge base in flood mapping.

Spatial Rules
$\forall s_i, s_j (Flood(s_i) \wedge Adjacent(s_i, s_j)) \rightarrow Flood(s_j)$
$\forall s_i, s_j (River(s_i) \wedge Distance(s_i, s_j) \leq d) \rightarrow Flood(s_j)$
$\forall s_i, s_j (Flood(s_i) \wedge River(s_i) \wedge Downstream(s_i, s_j)) \rightarrow Flood(s_j)$
$\forall s_i (LandCover(s_i, Wetlands) \wedge HeavyRain(s_i)) \rightarrow Flood(s_i)$
$\forall s_i (Slope(s_i) > s) \rightarrow \neg Flood(s_i)$
$\forall s_i (Elevation(s_i) > e) \rightarrow \neg Flood(s_i)$

Soft Logic framework [1, 19], which are statistical relational models that combines probabilistic graphical models and first-order logic.

2.2 Problem Definition

Formally, we define our problem as follows:

Input:

- A large-scale spatial raster framework with spatial samples $S = \{s_1, s_2, \dots, s_N\}$
- A set of explanatory feature layers X in S
- A limited set of labels $Y_l = \{y_1, y_2, \dots, y_l\}$, usually $l \ll N$, each label associated with a sample in S
- A spatial knowledge base $\mathcal{KB}$
- A base neural network model (e.g., U-Net)

Output:

- Inferred labels $\hat{Y}$ with quantified uncertainty U
- A deep learning model $DL : X \rightarrow Y$

Objective:

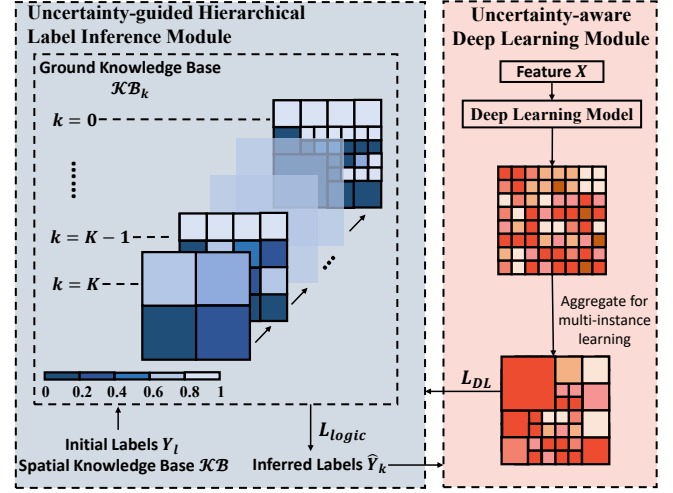
- Maximize the consistency between inferred labels $\hat{Y}$ and $\mathcal{KB}$
- Maximize the prediction accuracy of deep learning model

Specifically, we assume the raster framework S contains a large number of pixel samples but only with a limited set of labels Y_l . The main objective is to predict the class layer Y for all spatial samples. To illustrate, consider the case of flood mapping on earth imagery. In this scenario, the set of spatial samples S corresponds to Earth imagery pixels. The explanatory feature layers X are the spectral bands. The label set Y corresponds to the flood status of each pixel (i.e., flooded or not). The spatial knowledge contains domain constraints on flood locations (e.g., terrains and topography), which is used to infer flood labels $\hat{Y}$. Considering the errors in the inference process and the imperfect logic rules, uncertainty U naturally exists in the inferred label.

3 THE PROPOSED APPROACH

3.1 Overview

Our task is to train the deep learning model and infer sample labels based on the spatial knowledge base. The task is non-trivial for several reasons. First, spatial knowledge inference on labels is computationally expensive due to the immense volume of spatial samples in high-resolution imagery and complex spatial dependencies and interactions. This process quickly becomes computationally infeasible due to the exponential increase in the number of ground atoms and rules. Therefore, scalable grounding strategies are required that can effectively handle these issues by balancing computational efficiency and grounding granularity. In addition,

**Figure 2: Framework of SKI-HL.**

the label inference is complicated due to incomplete and sparse initial labels compared with the large study area. Such a low proportion of known data make logic inference difficult. Furthermore, the labels inferred are not deterministic; instead, they come with uncertainty at different granularity levels, which is non-trivial for the training of a deep learning model.

To address these challenges, we propose a **Spatial Knowledge-Infused Hierarchical Learning (SKI-HL)** framework. Our SKI-HL framework, illustrated in Figure 2, consists of two interdependent modules: a hierarchical label inference module and an uncertainty-aware deep learning module. The former infers sample labels in the raster framework with a trade-off between computational efficiency and spatial granularity. We formulate the inference process as an optimization problem with an objective based on the distance loss from Probabilistic Soft Logic (PSL) [1, 19], a probabilistic logic framework to capture spatial relationships and dependencies, and the spatial grounding configuration in a multi-resolution hierarchical grid structure. We design a greedy heuristic to iteratively refine the inferred labels based on inferred spatial uncertainty. The uncertainty-aware deep learning module trains neural network parameters from uncertain labels in multiple resolutions by an uncertainty-aware loss function and multi-instance learning. The two modules run in iterations: the outputs of the deep learning model will serve as the initialization of the hierarchical label inference module in the next iteration.

3.2 Hierarchical label inference with spatial knowledge

The hierarchical label inference module is designed to leverage spatial knowledge within a hierarchical framework to drive the label inference process. Given a raster framework, a spatial knowledge base with logic rules on sample labels, and a subset of initial labels, our goal is to infer the labels of all samples in the framework. We need to find a spatial grounding of the rules within the raster framework (e.g., on sample pixels or pixel blocks) and infer the optimal sample class probabilities according to the grounded rules.

For large-scale spatial data (e.g., high-resolution Earth imagery), we need to strike a balance between granularity, computational efficiency, and inference accuracy. If we ground the rules on all high-resolution pixels will lead to too many atoms, making the logic inference computationally expensive. On the other hand, if we ground rules only in low-resolution pixel blocks, the inferred labels are too coarse to train effective deep-learning models.

We formulate this sub-problem as an optimization problem. There are two specific objectives: the first is to generate accurate sample labels that are consistent with spatial knowledge (expressed by logic rules); the second is to balance computational efficiency and inference granularity in the spatial grounding process. We next discuss the specifics of our optimization objective and the proposed greedy strategy.

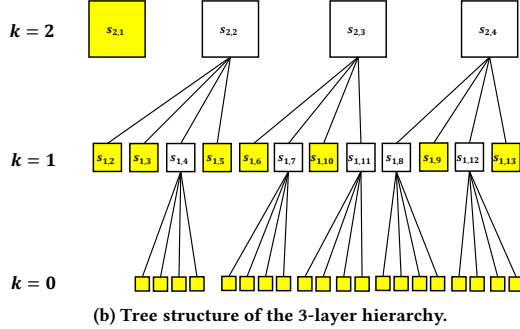
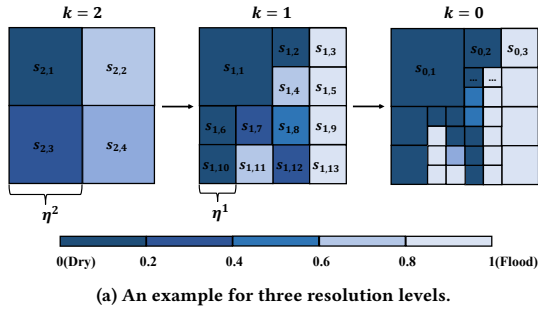


Figure 3: Illustration of hierarchical structure.

3.2.1 Optimization objective. We now formulate the spatial logic inference of sample labels in a raster framework as an optimization problem. First, we need to define the candidate feasible solution of spatial grounding. The process of spatial grounding refers to substituting the variables in the knowledge base rules with specific, concrete instances, which in our case are spatial samples such as pixels in earth imagery. Given a set of spatial samples and rules from a spatial knowledge base, we substitute each possible sample into the rules to generate candidate feasible solutions for spatial grounding. Let's illustrate this with an example. Suppose we have a spatial raster comprising 1 million pixels (size of 1000 by 1000) and a spatial logic rule that involves two variables. The process of grounding would mean substituting these variables with every possible pair of pixels. Thus, if we exhaust all possible substitutions,

we would end up with a whopping 1 trillion ground rules, derived from each possible combination of the 1 million pixels.

Spatial hierarchical structure: Based on the observation that spatial relationships often exhibit hierarchical characteristics and fractal patterns, we exploit a hierarchical framework to address these challenges. Given a large-scale spatial raster framework composed of pixels, we can represent the raster at multiple resolutions. At the coarse level, we can treat each cell as a condensed representation of many pixels. For example, in Figure 3a, if the original raster is 8×8 pixels (the rightmost grid), a coarser level representation (the leftmost grid) could be a 2×2 cell grid, where each cell represents a 2×2 pixel area of the original raster.

In the hierarchical structure of the spatial raster, the coarsest level grid represents the root of the hierarchy, and each subsequent refinement to a finer resolution represents a branching in the hierarchy. As shown in Figure 3b, in this tree-like structure, each node represents a region with different sizes of the original spatial raster at the finest resolution. With the hierarchical structure, we can start the logic inference from coarse resolution, which decreases the proportion of unlabeled data and make the inference practical. In the following, we provide a more formal description of the architecture.

Let S_0 be an original large-scale spatial raster framework at the finest resolution. We define a constant η where $\eta \in \mathbb{N}$ and $\eta > 1$, then we can get a set of resolutions $\{1, \eta, \eta^2, \dots, \eta^K\}$ which denotes the grid sizes at $K+1$ levels. We use the power number to represent the indexes of the layers k in the hierarchical structure, i.e., $k = 0, 1, 2, \dots, K$. The set of samples in each layer S_k , for $k \in \{1, \dots, K\}$, represents the spatial raster at a specific resolution. The highest layer K corresponds to the coarsest resolution and the lowest layer 0 corresponds to the finest resolution, usually the original resolution. Each spatial sample $s_{k,i}$ at layer k corresponds to a group of cells at the next finer layer $k-1$. To exemplify the hierarchy, consider Figure 3 representing 3 varying resolution levels. In this example, the grid size constant $\eta = 2$ and we have 3 layers in the hierarchy, i.e., $k = 0, 1, 2$. We select a subset of cells from layer 2 to "zoom in" further into smaller cells. This partitioning process results in layer 1, which contains both the original cells and the divided ones. The process is then repeated on the second grid, creating layer 0, which symbolizes the finest resolution layer. Through this hierarchical approach, each cell $s_{k,i}$ in layer k corresponds to a group of cells in the next finer layer $k-1$, e.g., cell $s_{2,2}$ corresponds to cell $s_{1,2}, s_{1,3}, s_{1,4}$, and $s_{1,5}$ in Figure 3. At a certain level, we only need to ground the "leaf node", as shown in Figure 3b, which significantly decreases the number of ground atoms. Each layer in the hierarchy form a ground knowledge base $\mathcal{KB}_k = \{r_{k,1}, r_{k,2}, \dots, r_{k,|\mathcal{KB}_k|}\}$ and then generates a set of inferred labels $\hat{Y}_k = \{\hat{y}_{k,1}, \hat{y}_{k,2}, \dots, \hat{y}_{k,N_k}\}$ and their corresponding uncertainties $U_k = \{u_{k,1}, u_{k,2}, \dots, u_{k,N_k}\}$, where N_k is the number of samples, i.e., grid cells at the k -th resolution level.

Second, we need to define the loss function based on the spatial grounding and inferred label probabilities. To make inferences that are consistent with spatial knowledge, we adopt t-norm fuzzy logic to define the extent of a rule as satisfied, which relaxes binary truth values to a continuous value between $[0, 1]$. These relaxations are exact at the extremes but provide a consistent mapping for values in between. The logical conjunction ($\wedge$), disjunction ($\vee$) and negation

($\neg$) are as follows:

$$\begin{aligned} I(a_1 \wedge a_2) &= \max\{I(a_1) + I(a_2) - 1, 0\} \\ I(a_1 \vee a_2) &= \min\{I(a_1) + I(a_2), 1\} \\ I(\neg a_1) &= 1 - I(a_1) \end{aligned} \quad (1)$$

where I is the soft truth value function that can map an atom a or a rule r to an interval between $[0, 1]$, indicating the probability that the atoms or rule holds. Then, following the structure of Probabilistic Soft Logic (PSL), we can induce the distance $d_r(I)$ to satisfaction for a rule $r : r_{body} \rightarrow r_{head}$:

$$d_r(I) = \max\{0, I(r_{body}) - I(r_{head})\} \quad (2)$$

where r is composed of atoms or negative atoms. PSL determines a rule r as satisfied when the truth value of $I(r_{body}) - I(r_{head}) \geq 0$. To this end, we can convert logical sentences into convex combinations of individual differentiable loss functions, which not only improves the training robustness but also ensures monotonicity with respect to logical entailment, *i.e.*, the smaller the loss, the higher the satisfaction. Therefore, given a set of ground rules in the ground knowledge base $\mathcal{KB}$, we can obtain the truth value for all ground atoms, which can serve as inferred labels for the spatial samples.

$$\hat{Y} = \arg \min_I \sum_{r \in \mathcal{KB}} \omega_r d_r(I) \quad (3)$$

It is noted that here the inferred labels $\hat{Y}$ are not binary values but soft truth values between $[0, 1]$.

Therefore, in our hierarchical framework, our optimization problem can be summarized as searching for an optimal grounding strategy and minimizing the overall distance to satisfaction for the ground atoms. we formally define the objective to minimize in the hierarchical label inference module as:

$$L_{logic} = \sum_{k=1}^K \left(\sum_{r \in \mathcal{KB}_k} \omega_r d_r(I_k) + \lambda |\mathcal{KB}_k| \right) \quad (4)$$

where $\mathcal{KB}_k$ stands for the ground knowledge base in the k -th layer, λ is a balancing coefficient. The summation over k stands for the overall objective of all layers in the hierarchical structure. The first term is the loss defined by PSL distance, which can drive accurate inference. The second term is used to decrease the ground atoms in each layer.

3.2.2 A greedy algorithm. To make a balance between inference accuracy, efficiency, and granularity, we proposed a greedy heuristic grounding strategy. Intuitively, uncertain atom inference always causes a higher distance to the satisfaction of a rule, so here we choose uncertain cells in a coarse layer to refine. The quantified uncertainty $u_{k,i}$ for each cell i at the k -th resolution level can be calculated using the entropy of the inferred label $\hat{y}_{k,i}$ as follows:

$$u_{k,i} = -\hat{y}_{k,i} \log \hat{y}_{k,i} - (1 - \hat{y}_{k,i}) \log(1 - \hat{y}_{k,i}) \quad (5)$$

We select a subset of cells with the highest uncertainty at each resolution level to refine the spatial partitioning. Let T_k be a threshold for selecting high-uncertainty cells at the k -th resolution level. We define a set of cells $\{s_{k,i} \mid u_{k,i} \geq T_k\}$ that will be refined to the next finer resolution level ($k-1$). Taking layer 2 in Figure 3 as an example, each cell in this grid is color-coded to denote the probability to dry (dark) and water (light). We view the cell with the

Algorithm 1 SKI-HL algorithm

Input: Spatial knowledge base $\mathcal{KB}$, Maximum resolution level K , Resolution constant η , Existing sparse label Y_1 , rule weights ω , Deep learning model DL , Explanatory features X

Output: Fine resolution predicted labels $\hat{Y}$, trained deep learning model DL

- 1: Initialize resolution level $k \leftarrow K$
 - 2: **while** $k \neq 0$ **do**
 - 3: Grounding rules in current resolution level
 - 4: Infer the uncertain labels $\hat{Y}_k$ based on Equation 3
 - 5: Calculate the uncertainty U_k based on Equation 5
 - 6: Train the deep learning model DL using $\hat{Y}_k$ based on Equation 8
 - 7: $k \leftarrow k - 1$
 - 8: **return** $DL, \hat{Y}_k$
-

leftmost and rightmost color (certain Dry and Flood) in the color bar as certain cells, and others as uncertain cells, *i.e.*, only the $s_{2,1}$ can be view as a certain cell in layer 2.

For the selected cells in S_k , we construct a new spatial partitioning with a smaller cell size and update the grounding atoms set accordingly. As shown in Figure 3, the uncertain coarser cells in layer 2 are split into 2×2 finer cells, respectively. We then perform PSL inference using the hierarchical label inference module at the $(k-1)$ -th resolution level with the leaf node in the hierarchy (yellow cells) only. Since the distance-based loss is convex, we can use gradient descent to optimize it. To initialize I at different resolutions, in the first iteration, *i.e.*, we pre-train the deep learning model with limited labels and use the output probabilities as the initialization. In the following iterations, the predicted probabilities of the corresponding deep learning model are regarded as the initial soft truth value of the ground atom in each r .

Starting from the coarsest resolution ($k = K$), the process continues iteratively until the finest resolution ($k = 0$) is reached. By employing this iterative hierarchical approach, our framework effectively improves the resolution of sparse labels from coarse to fine while efficiently handling the computational challenges associated with high-resolution spatial analysis and decision-making problems. Moreover, the method focuses computational resources on high-uncertainty regions, ensuring accurate label inference in these critical areas. Then, these inferred labels will serve as the training label in the uncertainty-aware deep learning module, which we will introduce in the next section. The overall process is described in Algorithm 1.

By observing the hierarchical structure and the figures, it becomes evident that this approach provides a computationally efficient way to handle large-scale spatial data, making a trade-off between granularity and efficiency. This is especially crucial in real-world applications like flood mapping, where both precision and computational resources are of prime importance.

As we mentioned before, there is a fundamental trade-off between computational efficiency and inference granularity in the grounding process. Therefore, given this hierarchical structure, the problem is how to select cells at each level of the hierarchy for refinement to the next finer resolution. We need to select cells for

refinement in such a way that balances the need for detailed information (i.e., higher resolution) against the computational cost of refining cells (i.e., efficiency). This problem also involves determining which cells to refine based on the potential benefit in terms of inference accuracy. The challenge lies in making these decisions efficiently and effectively, given the large scale of the spatial raster and the complexity of the spatial relationships represented in the spatial knowledge base.

3.3 Uncertainty-aware deep learning

The uncertainty-aware deep learning module, another component of SKI-HL, is capable of capturing information from the explanatory features, which is not accessible via logic inference alone. In addition, it plays a significant role in handling the uncertainty of inferred labels and variations in resolution. In traditional deep learning models, the model makes a prediction for each sample, but it doesn't utilize any information about how confident the model is about that prediction [11]. This could lead to overconfident predictions in regions with scarce or noisy labels. The module employs a modified version of the Binary Cross Entropy loss, offering a method to manage the uncertainty from the spatial knowledge and inferred label. In this module, all the deep learning predictions are at the finest resolution, i.e., under the original spatial raster framework and using original explanatory features.

3.3.1 Incorporating Uncertainty in Labels. The initial phase of training employs the Binary Cross Entropy loss function, defined as:

$$L_{DL} = - \sum_{i=1}^N y_i \log p_i + (1 - y_i) \log(1 - p_i) \quad (6)$$

where y_i is the ground truth label, p_i signifies the predicted probability output by the deep learning model, and N represents the number of spatial samples in a particular layer of the hierarchy.

The key challenge in our problem scenario is dealing with the inherent uncertainty associated with inferred labels. This stems from the fact that we start with a sparse and limited set of labels and then attempt to extrapolate this information to the entire spatial domain. A common approach would be to predict binary labels (flooded vs non-flooded) for each location. However, this ignores the degree of confidence (or uncertainty) in these predictions.

To better handle this, we replace the ground truth labels y_i in the Binary Cross Entropy loss function with the inferred uncertain labels $\hat{y}_i$. This allows us to directly optimize for the accuracy of the predicted probabilities, rather than merely the binary labels. The loss function thus becomes:

$$L_{DL} = - \sum_{i=1}^N \hat{y}_i \log p_i + (1 - \hat{y}_i) \log(1 - p_i) \quad (7)$$

where $\hat{y}_i \in \hat{Y}$ is the inferred uncertain label. Currently, the cross entropy measures the difference between the predicted probability distribution of deep learning model and the probability distribution of inferred labels. This modification effectively incorporates uncertainty information into the training process and can improve the model's performance when dealing with ambiguous cases.

3.3.2 Addressing Multi-instance Learning Scenarios. In scenarios where we partition the spatial domain into non-overlapping cells, or

instances, and construct a hierarchical structure for label inference, we encounter a multi-instance learning scenario [7, 23, 39]. Since we have different resolution levels, here, instead of assigning a single label to each pixel, we need to compute an aggregate probability output for each pixel to capture the overall likelihood of the event (e.g., flooding) occurring within the corresponding coarser cell. As shown in the right-bottom of Figure 2, to make an alignment between the structure of deep learning output and the inferred label, we have to combine the predictions of some pixels together, e.g., aggregate the 4×4 pixels in the output mask to train with the only 1 label inferred by logical reasoning.

In this context, the loss function should be updated again to reflect the cell-level nature of the labels:

$$L_{DL} = - \sum_{i=1}^{N_k} \hat{y}_{k,i} \log P_{k,i} + (1 - \hat{y}_{k,i}) \log(1 - P_{k,i}) \quad (8)$$

The probability output $P_{k,i}$ for each cell sample $s_{k,i}$ is computed as:

$$P_{k,i} = \frac{1}{|s_{k,i}|} \sum_{s_{0,j} \in s_{k,i}} p_j \quad (9)$$

where $P_{k,i}$ is the average of the predicted probabilities p_j for all the samples s_j within the coarse cell $s_{k,i}$. $|s_{k,i}|$ represents the number of finest resolution pixels in the cell. This is a natural way to aggregate the predictions of individual locations within a cell to produce a cell-level prediction. It allows the model to handle different levels of granularity in the spatial domain, making it flexible and adaptable to various spatial scales.

4 EVALUATION

4.1 Experiments setup

Dataset Description: We use two real-world flood mapping datasets collected from North Carolina during Hurricane Mathew in 2016. The explanatory features comprise the red, green, and blue bands within the aerial imagery obtained from the National Oceanic and Atmospheric Administration's National Geodetic Survey¹. In addition, digital elevation imagery was sourced from the University of North Carolina Libraries². Each piece of data was subsequently resampled to a 2-meter by 2-meter resolution to standardize the information. For Dataset 1, the image has a shape of 2500×1800 with 4.5 million pixels. For Dataset 2, the image has a shape of 3400×8400 with 28.56 million pixels.

Candidate Methods: In our experiments, we compare our proposed SKI-HL model with a variety of baselines that represent different approaches to handling spatial data and infusing knowledge into deep learning.

- **Pretrain:** In this method, the deep learning model is trained with the initially labeled pixels for each dataset.
- **Self-training:** The model adds patches with high confidence from Pretrain to the training dataset and re-trains the model.
- **DeepProbLog [22]:** This is a programming language that integrates deep learning with probabilistic logic programming. It allows for the incorporation of neural networks within a logic program, and these neural networks can be used to define probabilistic facts.

¹<https://www.ngs.noaa.gov/>

²<https://www.lib.ncsu.edu/gis/elevation>

Table 2: Accuracy versus Uncertainty (AvU).

		Uncertainty	
		Certain	Uncertain
	Accuracy	Accurate Certain (AC) Inaccurate Certain (IC)	Accurate Uncertain (AU) Inaccurate Uncertain (IU)

- **Abductive Learning (ABL)** [3]: This is a learning framework that combines both reasoning and learning. It works by training a model to make predictions, then using a logic reasoner to validate these predictions against a set of given logic rules. If the prediction contradicts the rules, the learning algorithm will revise its model based on the abductive explanation
- **SKI-HL-Base** This is a simplified version of our model as a candidate method that doesn't implement the selection of uncertain areas in the grounding process. Instead, it ground all atoms for each layer in the hierarchy.

Classification evaluation metrics: We used precision, recall, and F1 score on the flood mapping class to evaluate the pixel-level classification performance.

Uncertainty quantification evaluation metrics: The performance of uncertainty estimations in our model is quantitatively evaluated using the Accuracy versus Uncertainty (*AvU*) measure, as shown in previous work [12, 20]. We set an uncertainty threshold, denoted by T_u , to group uncertainty estimations into 'certain' and 'uncertain' categories. Predictions based on these estimations are then grouped into four categories: Accurate-Certain (AC), Accurate-Uncertain (AU), Inaccurate-Certain (IC), and Inaccurate-Uncertain (IU). Let n_{AC} , n_{AU} , n_{IC} , n_{IU} represent the number of samples in the respective categories. The *AvU* measure evaluates the proportion of AC and IU samples, with the idea being that accurate predictions should ideally be accompanied by certainty, and inaccurate predictions should correspondingly indicate uncertainty. This measure lies in the range [0, 1], with higher values indicating more reliable model performance. Specifically, we compute AvU_A for accurate predictions and AvU_I for inaccurate predictions as follows:

$$AvU_A = \frac{n_{AC}}{n_{AC} + n_{AU}}, AvU_I = \frac{n_{IU}}{n_{IC} + n_{IU}} \quad (10)$$

In our evaluation, we compute the harmonic average of AvU_A and AvU_I to penalize extreme cases:

$$AvU = \frac{2 * AvU_A * AvU_I}{AvU_A + AvU_I} \quad (11)$$

This evaluation approach thus offers a comprehensive measure of the reliability of our model's uncertainty estimations.

Model configuration: When implementing our method and baselines, we considered U-Net, a powerful deep learning model for image segmentation, as the base model. We set the same set of architecture for the U-Net model in all baselines with 5 down-sample operations and 5 upsample operations. There is a batch normalization within each convolutional layer and the dropout rate is 0.2.

For Pretrain, we partitioned an image into 100 by 100 patches, and then let the corresponding patches which contain the labeled pixel as the training patches. We set the label of all pixels in the patches to the label of the initially labeled pixel in it. The pre-training is based on these patches. For self-training, it uses Pretrain predictions

to augment the training dataset. Specifically, those patches with high-confidence predictions were added to the training set. The confidence of one image patch was calculated based on the average predicted class probabilities over all pixels in the patch. This process continues iteratively, gradually expanding the training dataset. We set a set of confident thresholds and then add high confident patches (which are measured by the prediction probabilities of Pretrain) into the training set. For DeepProbLog, ABL, and our proposed SKI-HL, we use the same pretrain U-Net to initialize the deep learning model in these frameworks.

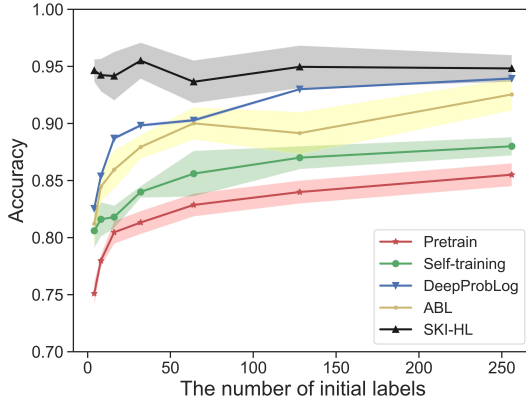
For the hierarchical structure of SKI-HL, we set the grid size constant $\eta = 10$ and $K = 2$ which means there are 3 layers with grid size 100×100 , 10×10 and 1×1 respectively in this hierarchy. And we construct the spatial knowledge base for the flood mapping task based on distance and topology relationships. For the distance relationship, we directly use the neighborhood pair to model. For the elevation, we adopt a Hidden Markov Tree model [17, 33] which can model the topological relationship of each location based on the elevation.

4.2 Comparison on classification performance

We first test the performance of each model with 4 labeled pixels. In Table 3, the experimental results underscore the efficacy of our proposed SKI-HL approach over the baseline models. We observe that the Pretrain model exhibits the worst performance in both datasets when the labeled data set is small. The reason was that the explanatory features on the surface contained a large number of obstacles that often confused a classifier, so it's unlikely to generalize a model from a small area to large-scale data. The Self-training model demonstrates enhanced performance over the Pretrain. However, "high confidence" predictions that were added to the training set could still contain errors, which somehow confused the model. Moreover, self-training cannot incorporate spatial knowledge and focuses only on imagery features causing poor performance. We notice that both DeepProbLog and ABL, models that combine domain knowledge into learning, display better performance than the previous models. This highlights the importance of integrating spatial knowledge into learning. Yet, they still lag behind the proposed SKI-HL. The main reason is that the limited label cause the label inference in both DeepProbLog and ABL inaccurate. Furthermore, the performance of ABL is generally lower than DeepProbLog, the gap is due to that ABL only use first-order logic as hard constraints to revise label and cannot deal with complex spatial rules which inherently have uncertainty and cannot always be perfectly satisfied. Our proposed SKI-HL method and its base model consistently outshine the baseline models on both datasets. The performance enhancement is likely due to the integration of uncertainty-guided hierarchical label inference and explicit spatial reasoning. These elements enrich the model's capacity to encapsulate complex spatial dependencies and dynamically concentrate on areas of uncertainty. Further, the hierarchical structure eases the demand for dense labeling, a bottleneck for other baseline models. It is noted that though we didn't ground all the pixels in our SKI-HL model, the performance on Dataset 1 is still parallel with its base model that ground all atoms to achieve the best granularity, and even higher on Dataset 2. Taken together, the superior performance

Table 3: Comparison on classification.

Method	Class	Dataset 1					Dataset 2				
		P	R	F1	Avg. F1	Acc	P	R	F1	Avg. F1	Acc
Pretrain	Dry	0.79	0.62	0.70	0.74	0.75	0.88	0.76	0.81	0.74	0.76
	Flood	0.73	0.86	0.79			0.59	0.77	0.67		
Self-training	Dry	0.60	0.83	0.70	0.78	0.81	0.89	0.77	0.83	0.76	0.78
	Flood	0.93	0.81	0.86			0.61	0.80	0.69		
DeepProbLog	Dry	0.73	0.78	0.75	0.81	0.83	0.83	0.87	0.85	0.82	0.82
	Flood	0.88	0.85	0.87			0.82	0.77	0.79		
ABL	Dry	0.66	0.78	0.72	0.79	0.81	0.76	0.86	0.81	0.79	0.79
	Flood	0.90	0.83	0.86			0.82	0.71	0.76		
SKI-HL-Base	Dry	0.95	0.93	0.94	0.95	0.95	0.92	0.94	0.93	0.91	0.91
	Flood	0.96	0.97	0.96			0.91	0.88	0.90		
SKI-HL	Dry	0.96	0.92	0.94	0.95	0.95	0.91	0.97	0.94	0.92	0.93
	Flood	0.95	0.98	0.96			0.95	0.88	0.91		

**Figure 4: Accuracy comparison on different numbers of initial labels.**

of the SKI-HL approach accentuates the importance of efficiently integrating spatial domain knowledge with deep learning and harnessing the benefits of hierarchical label inference, especially in the context of large-scale spatial applications with limited training labels.

4.3 The effect of the number of initial labeled samples

In order to thoroughly evaluate the effectiveness of our proposed SKI-HL method, we conduct experiments using Dataset 1 with varying numbers of initial labeled data, ranging from 4 to 256 in multiples of 2. The classification accuracy over 5 runs is utilized as the evaluation metric, with the mean value represented by markers and standard deviation indicated by shadows in Table 4. All baseline models, including Pretrain, Self-training, DeepProbLog, and ABL, display an upward trend in accuracy with the increase in initial labels. However, the Pretrain and Self-training models exhibit less significant improvement. This phenomenon can be attributed to the fact that these models use labeled pixels to represent the corresponding whole patches during the training process. While ABL's

performance does show improvement with an increased number of labels, its limited ability to handle complex spatial rules due to the usage of hard logic for label revision makes it unable to reconcile potential conflicts, then fail to achieve higher performance. DeepProbLog's accuracy incrementally improves and eventually stabilizes at a higher level as the number of initial labels increases. Owing to its logic-based framework for computing gradients, the training process maintains a degree of consistency, irrespective of the ground knowledge base. In contrast to the other models, our SKI-HL framework remains relatively unaffected by the number of initial labels, maintaining an accuracy rate of around 0.95. The underlying reason lies in our model's unique label inference process: it starts from a coarse resolution and subsequently refines labels, ensuring accurate label inference. The slight variations in the results can be attributed to training randomness and the initialization process.

4.4 Comparison on uncertainty quantification performance

Table 4: Comparison on uncertainty quantification.

Method	Accuracy	Dataset 1		Dataset 2	
		AvU_A / AvU_I	AvU	AvU_A / AvU_I	AvU
Pretrain	Accurate	0.81	0.45	0.97	0.20
	Inaccurate	0.32		0.11	
Self-training	Accurate	0.85	0.65	0.92	0.37
	Inaccurate	0.53		0.23	
DeepProbLog	Accurate	0.90	0.40	0.79	0.61
	Inaccurate	0.26		0.50	
ABL	Accurate	0.85	0.44	0.91	0.43
	Inaccurate	0.29		0.28	
SKI-HL-Base	Accurate	0.82	0.69	0.51	0.64
	Inaccurate	0.59		0.86	
SKI-HL	Accurate	0.80	0.74	0.80	0.84
	Inaccurate	0.68		0.88	

In Table 4, we notice a clear distinction in uncertainty estimation between our proposed SKI-HL model and the baselines. While Pretrain and Self-training models manifest a larger gap between AvU_A and AvU_I , this discrepancy is mitigated in DeepProbLog and ABL, which effectively incorporate spatial knowledge into

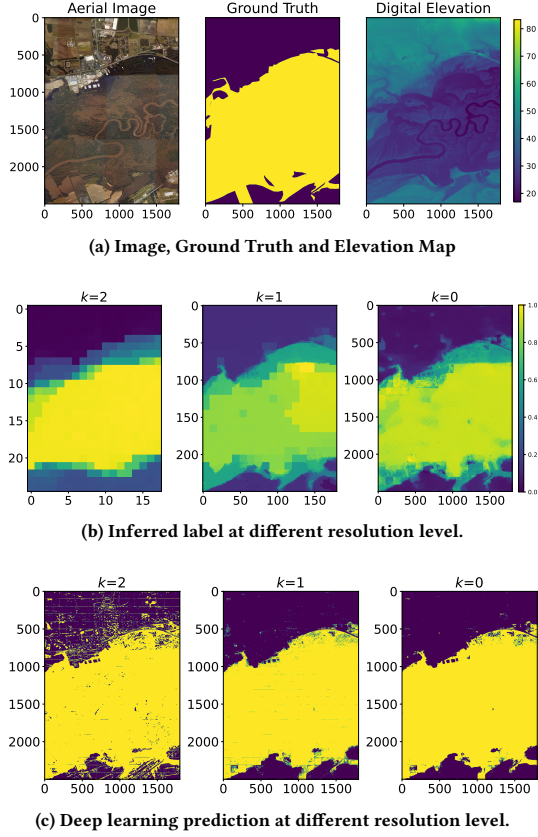


Figure 5: Performance with different resolution in Dataset 1.

learning. However, they still struggle to achieve a balanced AvU_A and AvU_I , particularly in situations of sparse and noisy labels. In stark contrast, our proposed SKI-HL model exhibits a superior performance on both datasets, signifying its robust ability to model complex spatial dependencies and adjust to areas of uncertainty dynamically. The integration of uncertainty-guided hierarchical label inference further mitigates the impact of sparse labeling, a bottleneck for other models. This finding emphasizes the pivotal role of efficiently integrating spatial domain knowledge with deep learning, especially under the constraints of limited training labels, in achieving reliable uncertainty estimation for large-scale spatial applications.

4.5 Case Study

In our case study, we visually analyze the effectiveness of our model across varying resolution levels. As depicted in Figure 5a, we present the aerial earth imagery, ground truth label, and digital elevation map from Dataset 1. It is noted that we didn't use the ground truth to train our label, instead, it was only used for testing. Figures 5b and 5c illustrate the evolution of inferred labels and deep learning predictions at different resolution levels. The resolution of the inferred labels refines progressively from a coarse resolution of 25 by 18 to the finest resolution of 2500 by 1800. This process allows for

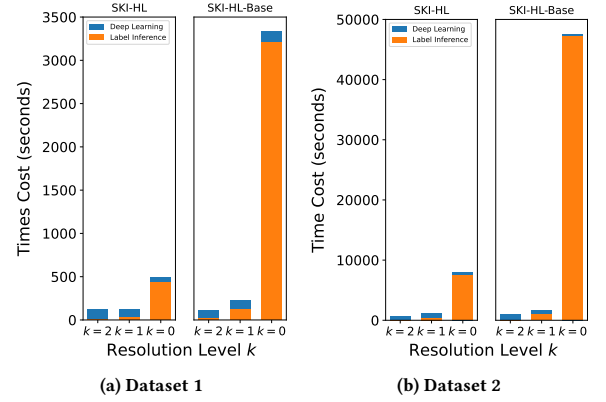


Figure 6: Comparison on time cost.

the accurate detection and refinement of uncertain areas, which often represent flood boundaries. Simultaneously, the granularity increase of the training labels results in an improved output from the deep learning model. A clear reduction in misclassified pixels can be observed, appearing as noise within each class of the area. This improvement can be attributed to the fact that multi-instance learning, used with coarse resolution labels, cannot provide supervision to every pixel. Hence, as our approach refines the label resolution, the deep learning model is able to generate more accurate predictions.

4.6 Analysis of time costs with hierarchical label inference

In order to demonstrate the computational efficiency of our proposed approach, we conducted a set of experiments evaluating the time costs of SKI-HL and its base model across different resolution levels. These experiments were executed on a AMD EPYC 7742 64-Core Processor CPU and an NVIDIA A100 GPU equipped with 80 GB of memory.

Figure 6 presents the time costs associated with the uncertainty-aware deep learning model training (blue bar) and the hierarchical label inference module training (orange bar) at each resolution level. Notably, the training time costs of the deep learning model remain relatively stable across different iterations, whereas the label inference process exhibits a strong dependency on the number of ground atoms. In particular, the label inference module requires the most significant computational resources when the number of ground atoms is large.

To illustrate, in dataset 2, the label inference at the finest resolution consumed approximately 12.5 hours without implementing a selective grounding process. However, when adopting the uncertainty-guided grounding strategy, the model achieved a considerable time-saving factor of 6.3 and 5.0 times compared to grounding all atoms for datasets 1 and 2, respectively. This stark difference in computational time underlines the necessity and effectiveness of our proposed uncertainty-guided hierarchical label inference in the context of large-scale spatial data.

5 CONCLUSION AND FUTURE WORKS

In this paper, we proposed a novel Spatial Knowledge-Infused Hierarchical Learning (SKI-HL) framework that successfully addresses the limitations of existing neuro-symbolic models, particularly with regard to spatial data, through a system of iteratively inferring labels within a multi-resolution hierarchy. Our model outperformed several baseline methods on real-world flood mapping datasets.

However, there is still a vast scope for further exploration and improvement in various aspects of the work. First, the model can incorporate temporal dynamics features and capture changes in earth imagery over time, which is critical for many applications such as deforestation tracking. Second, we can expand to other geospatial applications to further validate the generalizability and adaptability of the proposed SKI-HL framework.

ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation (NSF) under Grant No. IIS-2147908, IIS-2207072, CNS-1951974, OAC-2152085, and the National Oceanic and Atmospheric Administration grant NA19NES4320002 (CISESS) at the University of Maryland.

REFERENCES

- [1] Stephen H Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. 2017. Hinge-loss markov random fields and probabilistic soft logic. (2017).
- [2] Bibo Cai, Xiao Ding, Bowen Chen, Li Du, and Ting Liu. 2022. Mitigating Reporting Bias in Semi-supervised Temporal Commonsense Inference with Probabilistic Soft Logic. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 10454–10462.
- [3] Wang-Zhou Dai, Qiuling Xu, Yang Yu, and Zhi-Hua Zhou. 2019. Bridging machine learning and logical reasoning by abductive learning. *Advances in Neural Information Processing Systems* 32 (2019).
- [4] Michelangelo Diligenti, Marco Gori, and Claudio Sacca. 2017. Semantic-based regularization for learning and inference. *Artificial Intelligence* 244 (2017), 143–165.
- [5] Ivan Donadello, Luciano Serafini, and Artur D’Avila Garcez. 2017. Logic tensor networks for semantic image interpretation. *arXiv preprint arXiv:1705.08968* (2017).
- [6] Giuseppe Fabrizio, Alfonso Farina, and Antonio De Maio. 2006. Knowledge-based adaptive processing for ship detection in OTH Radar. In *2006 International Radar Symposium*. IEEE, 1–5.
- [7] James Foulds and Eibe Frank. 2010. A review of multi-instance learning assumptions. *The knowledge engineering review* 25, 1 (2010), 1–25.
- [8] Lianru Gao, Yiqun He, Xu Sun, Xiuping Jia, and Bing Zhang. 2019. Incorporating negative sample training for ship detection based on deep learning. *Sensors* 19, 3 (2019), 684.
- [9] Artur d’Avila Garcez, Sebastian Bader, Howard Bowman, Luis C Lamb, Leo de Penning, BV Illumino, Hoifung Poon, and COPPE Gerson Zaverucha. 2022. Neural-symbolic learning and reasoning: a survey and interpretation. *Neuro-Symbolic Artificial Intelligence: The State of the Art* 342, 1 (2022).
- [10] Ira Harmon, Sergio Marconi, Ben Weinstein, Sarah Graves, Daisy Zhe Wang, Alina Zare, Stephanie Bohlman, Aditya Singh, and Ethan White. 2022. Injecting Domain Knowledge Into Deep Neural Networks for Tree Crown Delineation. *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022), 1–19.
- [11] Wenchong He and Zhe Jiang. 2023. A Survey on Uncertainty Quantification Methods for Deep Neural Networks: An Uncertainty Source Perspective. *arXiv preprint arXiv:2302.13425* (2023).
- [12] Wenchong He, Zhe Jiang, Marcus Kriby, Yiqun Xie, Xiaowei Jia, Da Yan, and Yang Zhou. 2022. Quantifying and Reducing Registration Uncertainty of Spatial Vector Labels on Earth Imagery. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 554–564.
- [13] Wenchong He, Arpan Man Sainju, Zhe Jiang, Da Yan, and Yang Zhou. 2022. Earth Imagery Segmentation on Terrain Surface with Limited Training Labels: A Semi-supervised Approach based on Physics-Guided Graph Co-Training. *ACM Transactions on Intelligent Systems and Technology (TIST)* 13, 2 (2022), 1–22.
- [14] Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. 2016. Harnessing deep neural networks with logic rules. *arXiv preprint arXiv:1603.06318* (2016).
- [15] Yu-Xuan Huang, Wang-Zhou Dai, Jian Yang, Le-Wen Cai, Shaofen Cheng, Ruizhang Huang, Yu-Feng Li, and Zhi-Hua Zhou. 2020. Semi-supervised abductive learning and its application to theft judicial sentencing. In *2020 IEEE international conference on data mining (ICDM)*. IEEE, 1070–1075.
- [16] Neal Jean, Sherrie Wang, Anshul Samar, George Azzari, David Lobell, and Stefano Ermon. 2019. Tile2vec: Unsupervised representation learning for spatially distributed data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 3967–3974.
- [17] Zhe Jiang, Miao Xie, and Arpan Man Sainju. 2019. Geographical hidden Markov tree. *IEEE Transactions on Knowledge and Data Engineering* 33, 2 (2019), 506–520.
- [18] Anuj Karpatne, Ramakrishnan Kannan, and Vipin Kumar. 2022. *Knowledge Guided Machine Learning: Accelerating Discovery Using Scientific Knowledge and Data*. CRC Press.
- [19] Angelika Kimmig, Stephen Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. 2012. A short introduction to probabilistic soft logic. In *Proceedings of the NIPS workshop on probabilistic programming: foundations and applications*. 1–4.
- [20] Ranganath Krishnan and Omesh Tickoo. 2020. Improving model calibration with accuracy versus uncertainty optimization. *Advances in Neural Information Processing Systems* 33 (2020), 18237–18248.
- [21] Qing Li, Siyuan Huang, Yining Hong, Yixin Chen, Ying Nian Wu, and Song-Chun Zhu. 2020. Closed loop neural-symbolic learning via integrating neural perception, grammar parsing, and symbolic reasoning. In *International Conference on Machine Learning*. PMLR, 5884–5894.
- [22] Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. 2018. Deepproblog: Neural probabilistic logic programming. *advances in neural information processing systems* 31 (2018).
- [23] Oded Maron and Tomás Lozano-Pérez. 1997. A framework for multiple-instance learning. *Advances in neural information processing systems* 10 (1997).
- [24] Giuseppe Marra and Ondřej Kuželka. 2021. Neural markov logic networks. In *Uncertainty in Artificial Intelligence*. PMLR, 908–917.
- [25] Mengjia Qiao, Xiaohui He, Xijie Cheng, Panle Li, Qianbo Zhao, Chenlu Zhao, and Zhihui Tian. 2023. KSTAGE: A knowledge-guided spatial-temporal attention graph learning network for crop yield prediction. *Information Sciences* 619 (2023), 19–37.
- [26] Meng Qu and Jian Tang. 2019. Probabilistic logic neural networks for reasoning. *Advances in neural information processing systems* 32 (2019).
- [27] Matthew Richardson and Pedro Domingos. 2006. Markov logic networks. *Machine learning* 62 (2006), 107–136.
- [28] Ryan Riegel, Alexander Gray, Francois Luus, Naweed Khan, Ndivhuwo Makondo, Ismail Yunus Akhalwaya, Haifeng Qian, Ronald Fagin, Francisco Barahona, Udit Sharma, et al. 2020. Logical neural networks. *arXiv preprint arXiv:2006.13155* (2020).
- [29] Marc Rußwurm and Marco Korner. 2017. Temporal vegetation modelling using long short-term memory networks for crop identification from medium-resolution multi-spectral satellite images. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 11–19.
- [30] Arpan Man Sainju, Wenchong He, and Zhe Jiang. 2020. A hidden markov contour tree model for spatial structured prediction. *IEEE Transactions on Knowledge and Data Engineering* 34, 4 (2020), 1530–1543.
- [31] Jidong Tian, Yitian Li, Wenqing Chen, Liqiang Xiao, Hao He, and Yaohui Jin. 2022. Weakly Supervised Neural Symbolic Learning for Cognitive Tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 5888–5896.
- [32] Leon Weber, Pasquale Minervini, Jannes Münchmeyer, Ulf Leser, and Tim Rocktäschel. 2019. Nlprolog: Reasoning with weak unification for question answering in natural language. *arXiv preprint arXiv:1906.06187* (2019).
- [33] Miao Xie, Zhe Jiang, and Arpan Man Sainju. 2018. Geographical hidden markov tree for flood extent mapping. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2545–2554.
- [34] Yaqi Xie, Ziwei Xu, Mohan S Kankanhalli, Kuldeep S Meel, and Harold Soh. 2019. Embedding symbolic knowledge into deep networks. *Advances in neural information processing systems* 32 (2019).
- [35] Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, and Guy Broeck. 2018. A semantic loss function for deep learning with symbolic knowledge. In *International conference on machine learning*. PMLR, 5502–5511.
- [36] Dongran Yu, Bo Yang, Qianhao Wei, Anchen Li, and Shirui Pan. 2022. A probabilistic graphical model based on neural-symbolic reasoning for visual relationship detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10609–10618.
- [37] Yuyu Zhang, Xinshi Chen, Yuan Yang, Arun Ramamurthy, Bo Li, Yuan Qi, and Le Song. 2020. Efficient probabilistic logic reasoning with graph neural networks. *arXiv preprint arXiv:2001.11850* (2020).
- [38] Yichao Zhou, Yu Yan, Ruijun Han, J Harry Caulfield, Kai-Wei Chang, Yizhou Sun, Peipei Ping, and Wei Wang. 2021. Clinical temporal relation extraction with probabilistic soft logic regularization and global inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 14647–14655.
- [39] Zhi-Hua Zhou. 2004. Multi-instance learning: A survey. *Department of Computer Science & Technology, Nanjing University, Tech. Rep 1* (2004).

- [40] Zhi-Hua Zhou. 2019. Abductive learning: towards bridging machine learning and logical reasoning. *Science China Information Sciences* 62 (2019), 1–3.