Hierarchical Novelty Detection via Fine-Grained Evidence Allocation

Anonymous Authors[1]

Abstract

By leveraging a hierarchical structure of known classes, Hierarchical Novelty Detection (HND) offers fine-grained detection results that pair detected novel samples with their closest (known) parent class in the hierarchy. Prior knowledge on the parent class provides valuable insights to better understand these novel samples. However, traditional novelty detection methods try to separate novel samples from all known classes using uncertainty or distance based metrics so they are incapable of locating the closest known parent class. Since the novel class is also part of the hierarchy, the model can more easily get confused between samples from known classes and those from novel ones. To achieve effective HND, we propose to augment the known (leaf-level) classes with a set of novel classes, each of which is associated with one parent (i.e., non-leaf) class in the original hierarchy. Such a structure allows us to perform novel fine-grained evidence allocation to differentiate known and novel classes guided by a uniquely designed loss function. Our thorough theoretical analysis shows that fine-grained evidence allocation creates an evidence margin to more precisely separate known and novel classes. Extensive experiments conducted on real-world hierarchical datasets demonstrate the proposed model outperforms the strongest baselines and achieves the best HND performance.

1. Introduction

Novelty detection aims to tackle the challenging real scenarios, where test samples may come from previously unseen classes outside of the training distribution. Various novelty detection techniques have been developed with promising detection performance (Chen et al., 2021; Vaze et al., 2021; Chen et al., 2020; Zhang et al., 2020). Uncertainty or distance based metrics are commonly leveraged to quantify how a testing sample is different from known ones. However, most existing methods only provide a *binary* detection result, indicating whether the sample is novel or not. Such a coarse-grained result does not offer additional insight on the nature of the novel sample to further inform decision-making. For example, when detecting a new type of malware, it may be beneficial to identify the closest software family it belongs to, which can help security engineers quickly develop a defense strategy. Similar cases can be found in many other domains: when a newly synthesized protein is discovered, locating the most similar existing protein type can equip biologists with valuable prior knowledge to study the novel one and advance scientific discovery.

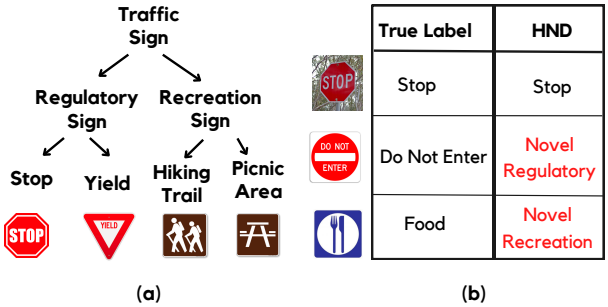

Figure 1. An illustrative example of Hierarchical Novelty Detection (HND): (a) A hierarchy of traffic signs with known classes (Stop, Yield, Hiking Trail and Picnic Area); (b) Testing samples from both known and novel classes.

To perform *fine-grained* novelty detection, it is beneficial to leverage existing hierarchical structures that humans commonly use to organize information. For example, most real-world objects can be described using a hierarchical structure based on their relationship with other relevant objects. Many benchmark datasets also organize the training classes into a hierarchical structure. With a hierarchy of known classes in place, fine-grained novelty detection can be achieved by simultaneously performing novelty detection while accurately identifying a parent class within the hierarchy that the novel sample is most similar to. We refer to this problem as *hierarchical novelty detection (HND)*. As shown in Figure 1, given a set of known types of traffic

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

signs organized by a hierarchy in (a) used in model training, the testing samples may come from known classes (Stop) or represent new types of traffic signs, including Do Not Enter and Food as shown in (b). For those novel samples, a properly trained HND not only needs to detect that they are not part of the existing hierarchy but also assign them to the closest parent class: Do Not Enter $\rightarrow$ Novel Regulatory and Food $\rightarrow$ Novel Recreational.

To achieve good novelty detection performance, a general detection model tries to separate novel data samples from known classes as much as possible. As a result, the model tends to assign a high uncertainty (or distance) score for novel data samples so that they can be clearly differentiated from known samples. However, directly applying existing novelty detection techniques does not meet the unique requirement of HND. *A fundamental challenge lies in that the novel samples are no long totally unbounded* as in the standard novelty detection setting. In contrast, they are also part of the hierarchy and the novelty arises because their corresponding class was not included during the training time. Thus, simply assigning a high uncertainty/distance score to a novel data sample may push it outside the entire hierarchy, hence is not able to identify a close parent class to better understand the nature of the sample.

HND is only sparsely pursued by existing efforts. One viable solution is to conduct hierarchical classification (HC) augmented with Novelty Detection techniques (Lee et al., 2018; Wang et al., 2022) (referred to as HC-ND). For each node followed by the HC process, a novelty score is predicted and compared with a pre-defined threshold to determine whether HC-ND should continue or stop. For samples from a known class, HC should proceed to the bottom layer of the hierarchy and assign them to the corresponding leaf node; for a novel sample, HC-ND should identify a right non-leaf node to stop when sufficient novelty is detected, making it impossible to further assign it into one of the existing child classes. The effectiveness of HC-ND heavily hinges on the HC model, as a mistake made at any point during the hierarchical classification process will result in a wrong detection result. Consequently, the detection error accumulates quickly with the depth of the hierarchy. Furthermore, a different novelty threshold may be assigned depending on the depth of the hierarchy, which further complicates the detection process.

To avoid a fast accumulating detection error in HC-HD, one can convert a multi-level hierarchy into a flat structure (Lee et al., 2018). To allow a novel data sample to be assigned to any non-leaf node as its closest parent class, the flat structure augments all known leaf classes with a set of novel classes, each of which associates with one non-leaf node in the original hierarchy. Figure 2 shows an example of the flat structure, where the augmented novel classes are highlighted in red. One remaining challenge lies in the lack of training samples from the novel classes. To overcome this, a Leave-One-Out (LOO) training process has been developed that iteratively removes classes from the hierarchy and treats them as the novel class to support training. This process can effectively avoid assigning a novel sample to any known classes. Nevertheless, since samples from known classes are used as novel ones during training, the model may have trouble in differentiating the known classes from the novel one during testing (as shown in Figure 5).

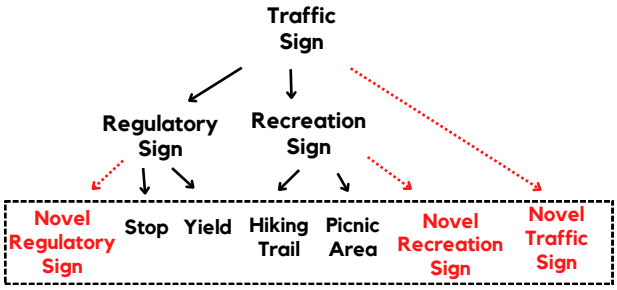

Figure 2. Flatten the hierarchy of Figure 1(a) to convert the problem into a multi-class (leaf classes and novel non-leaf classes) classification problem. Novel non-leaf classes include Novel Traffic Sign, Novel Regulatory Sign, and Novel Recreation Sign.

To achieve effective HND, we propose to conduct novel fine-grained evidence allocation for hierarchical novelty detection. By leveraging the flattened structure, we perform evidence-based multi-class classification to train a model that can allocate different amounts of evidence to known classes and novel ones, respectively, which forms a margin to separate them more precisely. In particular, for testing samples from known classes, the model is trained to assign high evidence to the corresponding leaf class, so it can be clearly differentiated from other known classes as well as the novel classes; for a novel data sample, the model can assign moderate evidence to the corresponding novel class while ensuring a low evidence to all other classes. Model training is guided by a uniquely designed loss function with strong theoretical guarantees to create an evidence margin for improved HND detection. In addition, prior belief on the existence of certain novel classes can be incorporated in a principled way by adjusting the base rate in the innovative evidential formulation. Our empirical results confirm that HND performance can indeed benefit from such prior belief. Our contribution of the paper is threefold:

- We propose to conduct novel evidential hierarchical novelty detection (E-HND) that leverages fine-grained evidence to more precisely differentiate samples of known class from those of novel ones in the same hierarchy.
- We design a unique loss function that can create an evidence margin to ensure good separation of known and novel samples with sound theoretical guarantees.

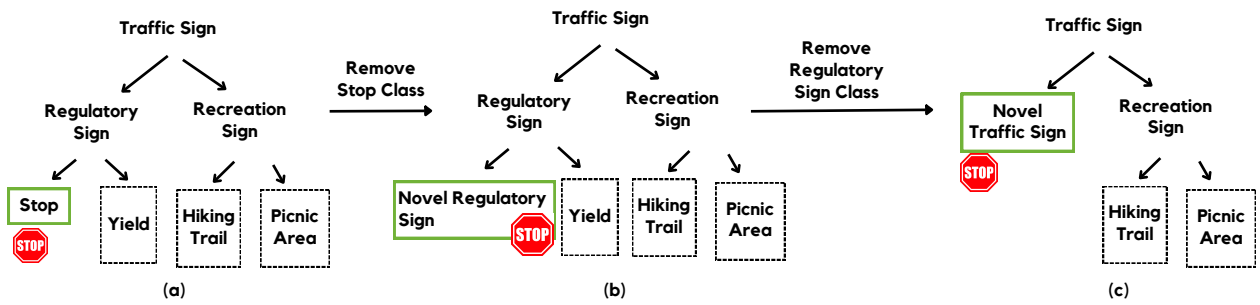

Figure 3. Working mechanism of leave-one-out (LOO) training. A training sample from the stop class is used to train HND. The green color represents Ground Truth(GT), and the dashed line represents the non-ground truth classes. (a) GT is the known leaf class (b) By removing the known leaf class (stop) from the hierarchy, GT becomes its novel parent class (Novel Regulatory sign). (c) By removing the Regulatory sign class, GT becomes the Novel Traffic sign class.

- We leverage the base rate in the evidential formulation to incorporate prior belief on the existence of novel classes.

We perform extensive experiments on multiple real-world hierarchical datasets, which show the effectiveness of the proposed E-HND model. Comparison with the strongest known baselines shows that E-HND achieves the best HND performance to date.

2. Related Works

Novelty Detection. The field of novelty detection aims to identify whether the sample is from a known or novel class. In order to perform novelty detection, some methods use maximum probability or maximum logit value as the score for assigning a sample to a known class (Vaze et al., 2021). Similarly, (Bendale & Boult, 2016) uses a separate class to assign the probability that a sample belongs to a novel class. (Chen et al., 2020; 2021; Yang et al., 2020) learn a prototype based on known classes and assigns the test sample to a known class on the basis of how close they are to prototypes. (Chen et al., 2021) further utilizes an adversarial learning-based training to generate novel samples to further improve the novelty detection performance. Moreover, there are various uncertainty-based methods (Sensoy et al., 2018; Malinin & Gales, 2018; Charpentier et al., 2020) that quantify the uncertainty measures to represent the uncertainty in prediction for novel samples. These methods can not be directly used in HND, as HND has a unique setting to identify the closest parent of the novel sample. Hence, in order to tackle the problem, we need to consider the hierarchical structure within known classes.

Hierarchical Novelty Detection. There are various works (Chang et al., 2021; Chen et al., 2022; Zhao et al., 2021; Du et al., 2020) in the field of hierarchical classification that achieve promising results on identifying samples from fine-grained classes. However, these classifications are not equipped with novelty detection mechanisms. To introduce novelty detection in hierarchy, (Lee et al., 2018) uses KL divergence based confidence score for each local classifier. Further, in order to improve novelty detection, (Wang et al., 2022) uses fuzzy logic as an uncertainty measure. However, using multiple classifiers in the hierarchy causes errors to accumulate while making predictions. Also, it requires us to set multiple thresholds. To avoid the use of multiple thresholds and error propagation, (Lee et al., 2018) flattens the hierarchy to perform multi-class classification for leaf and non-leaf classes together. (Ruiz & Serrat, 2022) uses cosine loss to learn prototypes to leaf and non-leaf classes, and assigns the test sample to the closest learned prototype. These methods leverage the training of known samples as novel non-leaf classes, causing the model to confuse between known and novel classes in the testing phase. In order to address the problem, we conduct HND based on fine-grained evidence allocation that helps in the separation between known and novel classes.

3. Methodology

3.1. Problem Formulation

Let $\mathcal{H}$ denote a hierarchical relationship between classes. For a class y, let $Pa(y), Ch(y), An(y)$ and $De(y)$ denote the parent, children, ancestors and descendants of y in $\mathcal{H}$, respectively. There are three types of classes: leaf class (with no children), non-leaf class (ancestor of a leaf class), and novel class ($Ch(y) = \phi$ and $y \notin \mathcal{H}$). Out of these classes, only the leaf class and non-leaf class are known during training, forming the hierarchy $\mathcal{H}$. Let $N(y)$ denote the set of novel classes, whose closest known parent class in $\mathcal{H}$ is y. For a test sample belonging to a novel class $N(y)$, the goal of HND is to predict y for that sample.

As mentioned in the introduction, to avoid accumulating errors over the hierarchical classification process and setting layer specific novelty threshold, we leverage a flattened structure to conduct HND. Let $Le(\mathcal{H})$ and $NLe(\mathcal{H})$ represent the set of leaf and non-leaf classes, respectively. To cover the entire hierarchy, we associate each non-leaf class with a novel class, as shown in Figure 2. We refer to these

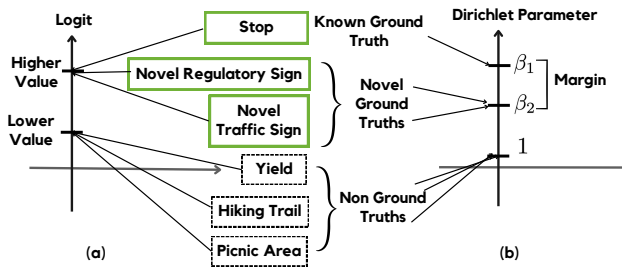

Figure 4. A training image from Stop class is used by cross-entropy based loss function and E-HND (a) Logits mapped to ground truth and non-ground truth classes by LOO method (b) Dirichlet parameters mapped to ground truth and non-ground truth classes by E-HND

novel classes as novel non-leaf classes. The flattened structure allows us to perform multi-class classification in one-shot by including both the known leaf classes and novel non-leaf classes. The model can output the probabilities for the known ($\boldsymbol{p}^{(kn)} = [p_1^{(kn)}, ..., p_{|Le(\mathcal{H})|}^{(kn)}]$) and novel non-leaf classes ($\boldsymbol{p}^{(no)} = [p_1^{(no)}, ..., p_{|NLe(\mathcal{H})|}^{(no)}]$). Once being trained, the model can perform HND by

$$\hat{k} = \underset{k}{\operatorname{argmax}}[p_1^{(kn)}, ..., p_{|Le(\mathcal{H})|}^{(kn)}, p_1^{(no)}, ..., p_{|NLe(\mathcal{H})|}^{(no)}] \quad (1)$$

where $\hat{k}$ represents the index of the class with the highest probability among known leaf and non-leaf novel classes.

3.2. Challenges of Model Training

One key challenge in novelty detection is the lack of samples from novel classes during the training process. LOO is a technique leveraging a flattened structure for novel detection training using the samples solely from known classes (Lee et al., 2018). Let (x, y) be a pair of training sample with a leaf-level label y. To support known sample classification, LOO trains the model to output the highest probability value $p(y|x)$ among all known leaf classes $Le(\mathcal{H})$. To support novelty detection using samples in class y, it recursively removes each of its ancestors $c \in An(y)$ from $\mathcal{H}$, resulting in a new hierarchy $\mathcal{H} \setminus c$. For each c, it maximizes the probability of the new ground truth as $N(Pa(c))$. As an example, consider a sample from Stop class as shown in Figure 3. The sample is first used to maximize the probability of Stop in comparison to non-ground truths shown in the dashed box as shown in (a). When the Stop Sign class is removed from the hierarchy, the same training sample is used to maximize the probability of Novel Regulatory Sign as shown in (b). Finally, when Regulatory Sign class is removed from the hierarchy, the training sample is used to maximize the probability of Novel Traffic Sign as shown in (c). Overall, the following loss function is used to minimize

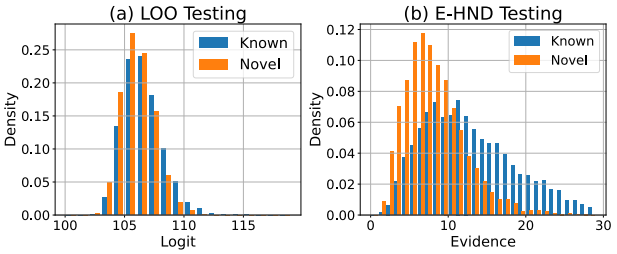

Figure 5. Comparison of the distribution of (a) logits and (b) evidences for known and novel test samples in CUB dataset

the cross-entropy:

$$\mathcal{L}^{CE}(\theta) = \mathop{\mathbb{E}}_{p(x,y)}[-\ln p(y|x;\theta_{Le(\mathcal{H})}) + \sum_{c\in An(y)} -\ln p(N(Pa(c))|x;\theta_{N(Pa(c))\cup Le(\mathcal{H}\setminus c)})] \quad (2)$$

Since the same data sample is used to maximize both the known ground truth (*i.e.,* y) and the novel ground truth (*i.e.,* $N(Pa(c))$) during training, it could lead to a conflict that causes confusion when using the model for testing. For example, given a test Stop image, the model could output a high probability for both the known ground true label and each of the novel ground true labels as shown in Figure 4(a). This kind of training does not allow the model to separate between known and novel samples in testing. The model allocates high logit values for both known and novel samples as we observed in CUB test dataset in Figure 5(a), compromising the novelty detection performance in practical settings.

3.3. Learning the Evidence Margin

To avoid confusion of the model during testing, it is essential to use a more fine-grained loss function that can clearly separate samples from known and novel classes. Maximizing the class probability by minimizing the cross-entropy as in (2) is inadequate. To this end, we propose to conduct evidential HND, which performs fine-grained *evidence-based* training that guides the model to allocate distinct amounts of evidence to known and novel classes, respectively, resulting in a clear separation.

Given a K-way multi-class problem, evidential learning trains a model to assign a belief mass distribution $\boldsymbol{b} = [b_1, b_2, ..., b_K]$ along with an uncertainty mass u to multi-class forming a multinomial opinion w given by:

$$w = (\boldsymbol{b}, u, \boldsymbol{a}), \text{with} \sum_{k=1}^{K} b_k + u = 1 \quad (3)$$

where $\boldsymbol{a} = [a_1, a_2, ..., a_K]$ denotes the base rate distribution representing the prior probabilities associated with each class. The probability that a sample belongs to a class k is

$$P(y = k) = b_k + a_k u \quad (4)$$

Assume that the label distribution is governed by a parameter $\mathbf{p} = [p_1, p_2, ..., p_K]$, *i.e.,* $P(y = k|p_k) = p_k$, which allows us to obtain (4) by marginalizing $\mathbf{p}$. Further, assume that $\mathbf{p}$ is drawn from a Dirichlet PDF $D(\mathbf{p}|\boldsymbol{\alpha})$, where $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_K)^\top$. The parameter α_k represents the effective number of observations for class k. Let r_k represents the observed evidence, then parameter α_k is given by

$$\alpha_k = r_k + a_k W \tag{5}$$

where W provides the weight to the base rates [1]. Such an expression leads to an evidence based representation of class probability $P(y = k)$:

$$P(y = k) = \mathbb{E}[p_k] = \frac{\alpha_k}{\sum_{k=1}^{K} \alpha_k} = \frac{r_k + a_k W}{\sum_{k=1}^{K} r_k + W} \tag{6}$$

Thus, given the ground-truth labels, training an evidential learning model can be achieved by maximizing the ground-true label probability. This is equivalent to assigning high evidence to that label. On the other hand, the amount of evidence also reflects the confidence (or uncertainty) of the prediction. By comparing (6) and (4), when the model predicts a low evidence r_k, the corresponding belief b_k is low and the uncertainty mass u is high due to the summation constraint in (3). For novel samples, it is natural for the model to make a low-confidence prediction because the model has not been exposed to such samples.

Taking advantage of the key properties offered by an evidence-based formulation, we propose a novel loss function that can form an evidence margin to clearly separate known and novel samples, leading to improved novelty detection performance. On one hand, since the model naturally provides low-confidence predictions for novel samples, the loss function simulates that behavior during the training phase by upper bounding the evidence allocated to the novel classes. On the other hand, for the known classes, it allows the model to predict much higher evidence, which ensures confident predictions on known samples. More formally, given the i-th training sample, the proposed loss function comprises two terms that work in a multitask fashion to allocate: (i) high evidence to the ground truth known leaf class and (ii) moderate evidence to the ground truth novel non-leaf classes.

$$\mathcal{L}_i(\theta) = \mathcal{L}_i^{(1)}(\theta) + \mathcal{L}_i^{(2)}(\theta) \tag{7}$$

$$\mathcal{L}_i^{(1)}(\theta) = KL\left[D(\mathbf{p}_i|\boldsymbol{\alpha}_i; \theta_{Le(\mathcal{H})})||D(\mathbf{p}_i|\hat{\boldsymbol{\alpha}}_i; \theta_{Le(\mathcal{H})})\right]$$

$$\mathcal{L}_i^{(2)}(\theta) = \sum_{c \in An(y)} \mathcal{L}_{i,c}^{(2)}(\theta)$$

$$\mathcal{L}_{i,c}^{(2)}(\theta) = KL\left[D(\mathbf{p}_i|\boldsymbol{\alpha}_i; \theta_{Le'(\mathcal{H}\setminus c)})||D(\mathbf{p}_i|\tilde{\boldsymbol{\alpha}}_i; \theta_{Le'(\mathcal{H}\setminus c)})\right]$$

[1] The default values a_k and W are usually set to $\frac{1}{K}$ and K, respectively, leading to $\alpha_k = r_k + 1$ in common settings.

The first term $\mathcal{L}_i^{(1)}(\theta)$ is defined using the KL divergence between a model predicted Dirichlet Distribution $D(\mathbf{p}_i|\boldsymbol{\alpha}_i; \theta_{Le(\mathcal{H})})$ and a sharp baseline Dirichlet Distribution $D(\mathbf{p}_i|\hat{\boldsymbol{\alpha}}_i; \theta_{Le(\mathcal{H})})$ that assigns high evidence (*i.e.,* $\hat{\alpha}_{ij} \gg 1$) to ground-truth label j and zero evidence to other labels. In the second term $\mathcal{L}_i^{(2)}(\theta)$, we iteratively remove the ancestor $c \in An(y)$ from $\mathcal{H}$, resulting in a new hierarchy $\mathcal{H} \setminus c$) each time. For each c, $\mathcal{L}_{i,c}^{(2)}(\theta)$ is defined by the KL divergence between a model predicted Dirichlet Distribution $D(\mathbf{p}|\boldsymbol{\alpha}; \theta_{Le'(\mathcal{H}\setminus c)})$ and a less sharp Dirichlet Distribution $D(\mathbf{p}|\tilde{\boldsymbol{\alpha}}; \theta_{Le'(\mathcal{H}\setminus c)})$, where $Le'(\mathcal{H} \setminus c)$ denotes $N(Pa(c)) \cup Le(\mathcal{H} \setminus c)$. The Dirichlet parameters of the two baseline distributions are given as

$$\hat{\alpha}_{ik} = \begin{cases} \beta_1 \gg 1, & \text{if } k = j^{\mathcal{H}} \\ 1 & \text{otherwise} \end{cases} \tag{8}$$

$$\tilde{\alpha}_{ik} = \begin{cases} 1 < \beta_2 < \beta_1, & \text{if } k = j^{\mathcal{H}\setminus c} \\ 1 & \text{otherwise} \end{cases} \tag{9}$$

where $j^{\mathcal{H}}$ and $j^{\mathcal{H}\setminus c}$ denote the known ground-truth label and and novel known ground-truth labels when c is removed. Figure 4 (b) illustrates the key idea of the proposed loss function, which forms an evidence margin $(\beta_1 - \beta_2)$ to clearly separate the known ground-truth an novel ground-truth labels. Learning such a evidence margin can lead to a much improved HND performance in testing. As shown in Figure 5 (b), the model tends to predict much higher evidence for known samples than the novel samples. In contrast, without learning the evidence, the model predicts very similar logits for both known and novel samples margin as shown in Figure 5 (a).

3.4. Theoretical Analysis

In this section, we perform a deep theoretical analysis to understand the key properties of the proposed loss function. First, we analyze how the proposed loss function can guarantee to learn an evidence margin to separate known ground truth and novel ground-truth labels in Theorem 1. We then show that the updates from two loss terms do not conflict with each other in Theorem 2 as the same data sample is leveraged in both terms in a multi-task fashion.

Theorem 1 (Evidence margin learning)**.** *Given a hierarchy $\mathcal{H}$ and a training sample i. The known ground truth class is y with index $j^{\mathcal{H}}$ and the novel ground truth index is $j^{\mathcal{H}\setminus c}, \forall c \in An(y)$. The loss function trains the model to assign evidence such that*

$$1 \leq \alpha_{j^{\mathcal{H}}} \leq \beta_1, \quad 1 \leq \alpha_{j^{\mathcal{H}\setminus c}} \leq \beta_2, \forall c \in An(y) \tag{10}$$

And when the learning converges, the Dirichlet parameters form an evidence margin given by $(\beta_1 - \beta_2)$.

Proof. (Proof sketch) Limited by the space, we provide the proof of the theorem in Appendix. We first define a general

form of KL divergence-based loss function with baseline ground truth β. For $\alpha_j = \beta$ and $\alpha_{k\neq j} = 1$, we show that loss becomes 0. We then show that the loss decreases when α_j starts increasing until it reaches β and further increasing α_j beyond β causes the loss to increase. □

Theorem 2 (Non-conflicting update)**.** *When optimizing the overall loss function in (8) that involves simultaneously minimizing the two loss terms $\mathcal{L}_i^{(1)}(\theta)$ and $\mathcal{L}_i^{(2)}(\theta)$, it does not lead to a conflict in the model predicted Dirichlet parameters $\boldsymbol{\alpha}$.*

Proof. (Proof sketch) We provide the details of the proof in Appendix. We first identify the common parameters between two loss terms. We then show that for the non-common parameters, each loss term updates them independently. Finally, for the common parameters, we show that there is no conflicting update. □

Remarks: Theorem 2 ensures that the model parameters can be updated consistently when optimizing the jointly objective function in (8). Besides, ensuring the evidence margin, both loss terms also try to assign minimum evidence to the non-ground truth labels. As an example in Figure 4(b), the model learns to output a Dirichlet parameter with value 1 for all non-ground truth labels: Yield, Hiking Trail, Picnic Area.

3.5. Incorporating the Prior Belief

The evidential theory allows us to encode a prior belief in the form of base rate distribution $\boldsymbol{a}$ as we calculate the effective number of observations. Base rates denote the prior probabilities of a data sample belonging to the classes when no evidence is observed and W quantifies the weight of the base rates. In the most common setting with no strong prior belief, the Dirichlet parameter is related to evidence as $\alpha_k = r_k + 1$, where evidence r_k represents the observed number of observations in support of class k. Recall that the 1 is the result of $a_k W$ with a weak base rate $\frac{1}{K}$ and setting $W = K$. By adjusting the base rates, we can incorporate more appropriate prior belief that can further improve the HND performance in practice.

In particular, a higher base rate for the known classes denote the belief of completeness of the hierarchy, and a test sample will more likely be assigned to one of the known leaf classes. In contrast, a higher base rate for the novel classes allows us to encode the belief that the current hierarchy is still incomplete. By leveraging this Bayesian formulation, we can assign different base rates given the distinct prior belief on how incomplete each sub-hierarchy is. Applying a higher base rate to class k has the effect of enforcing a stronger prior belief by increasing the *pseudo counts*.

As an example, if we have a stronger belief that only the sub-hierarchy of Recreation Sign is more likely to be incomplete as part of the hierarchy shown in Figure 2, we can modify the base rate distribution by using a higher base rate $a_k > 1/K$ for the corresponding Novel Recreation Sign class. As a result, the *pseudo count* increases for Novel Recreation Sign and decreases for other classes. In this way, a prior belief can be encoded through base rates, resulting in an increased Dirichlet parameter that affects the loss function accordingly.

4. Experiments

We conduct experiments on real-world hierarchical datasets to assess the effectiveness of the proposed method. We investigate the effects of using different sets of hyperparameters to confirm the positive impact of the learning an evidence margin. Finally, we explore the trade-off between known and novel performance by adjusting the different values of base rate distributions. Additional experiments are presented in the Appendix C.

4.1. Datasets

To evaluate the performance of E-HND and other competitive baselines, we use four real-world hierarchical datasets:

- **TinyImagenet** (Le & Yang, 2015): It contains 200 classes each with 500 training, 50 validation, and 50 test images in each class, resulting in a total of 120k images. We randomly select 50 classes as novel and the remaining classes as known classes. For the known class, we create the hierarchy using the hypernym-hyponym relationship from WordNet. The resulting hierarchy contains 150 leaf nodes, 86 non-leaf nodes, and 12 levels.
- **CUB-200-2011** (Welinder et al., 2010): It contains 12k images of fine-grained species of bird with a total of 200 classes. We use a 150-50 split of known-novel classes. We construct the hierarchy using hypernym-hyponym relationships from WordNet. The hierarchy consists of 43 non-leaf nodes, 150 leaf nodes, and 7 levels.
- **Animals With Attributes 2** (Lampert et al., 2014): It contains 37k images of animals with total 50 classes. We use a 40-10 split of known-novel classes. We construct the hierarchy using hypernym-hyponym relationships from WordNet, resulting in 21 non-leaf nodes, 40 leaf nodes, and 7 levels.
- **Mapillary Traffic Sign Dataset** (Ertler et al., 2019) It consists 70k images of traffic signs with total of 203 classes. We use a 164-39 split of known-novel classes from (Ruiz & Serrat, 2022) and construct hierarchy using parent-child relationships from (Ruiz & Serrat, 2022), resulting in 41 non-leaf, 164 leaf nodes, and 4 levels.

4.2. Compared Baselines

We compare E-HND with the following state-of-the-art baselines in HND. It is worth noting that performing Hierarchical Classification augmented with Novelty Detection(HC-ND) requires setting multiple thresholds, as discussed in the introduction. Hence, it is difficult to make a fair comparison,

Table 1. Comparison Results

Method	CUB		Tiny Imagenet		AWA2		Traffic	
	NA@50	AUC	NA@50	AUC	NA@50	AUC	NA@50	AUC
DARTS	40.42	30.07	15.91	12.18	36.75	35.14	34.00	30.36
Relabel	38.23	28.75	18.67	14.73	45.71	40.28	39.67	34.03
Evidential	35.06	25.86	19.35	14.53	44.82	36.44	37.32	32.57
HCL	32.19	25.22	13.45	10.19	36.40	32.80	36.40	32.80
LOO	42.25	32.81	18.93	14.50	47.82	41.95	41.51	35.47
E-HND	**46.18**	**35.31**	**21.44**	**16.03**	**48.22**	**42.37**	**45.09**	**41.02**
TD+LOO	44.42	34.31	19.37	14.87	50.25	42.86	42.41	38.22
TD+E-HND	**46.85**	**35.78**	**21.77**	**16.39**	**52.53**	**45.56**	**47.69**	**43.11**

so we do not include it as one of the competing baselines in table 1. Instead, we use the features from one of the HC-ND methods as input to our method (TD+EHND) and the best-performing baseline (TD+LOO). We discuss results from HC-ND methods in Appendix C.

- **Dual Accuracy Reward Trade-off Search (DARTS)** (Deng et al., 2012): Following the modified version of DARTS (Lee et al., 2018), we obtain the expected rewards for all the classes.
- **Relabel** (Lee et al., 2018): For training the novel classes, the training samples from known leaf classes are randomly relabeled as novel non-leaf classes.
- **Leave-One-Out (LOO)** (Lee et al., 2018): The method removes the class from the hierarchy one at a time, changing its the ground truth as the novel parent.
- **Top-Down Features + Leave-One-Out (TD+LOO)** (Lee et al., 2018): TD+LOO uses features extracted from the TD method as input to the LOO method. In contrast, LOO method directly uses Resnet101 features as input.
- **Evidential** (Sensoy et al., 2018): There are different ways of training an evidential model. We construct a baseline using the evidential log loss.
- **Hierarchical Cosine Loss (HCL)** (Ruiz & Serrat, 2022): As explained earlier, HCL uses a cosine loss to learn prototypes of all the classes in the flattened structure in HND. We follow the code provided by the paper.

4.3. Evaluation Metrics

The test dataset contains samples from known and novel classes. Known Accuracy (K-ACC) denotes the ratio of correctly predicted leaf-level labels by the model out of the total known test samples. Similarly, Novel Accuracy (N-ACC) denotes the ratio of correctly predicted closest parent by the model out of novel test samples. For the practical testing scenario of HND, we should compare our method using both K-ACC and N-ACC. However, these accuracies are in a trade-off relation, *i.e.,* an increase in one accuracy causes the other accuracy to decrease. In order to capture the trade-off relation, we can add a bias term to the logit of novel classes that increases N-ACC and decreases K-ACC. With the use of different biases, we can obtain different sets of K-ACC and N-ACC and plot them to obtain a K-ACC vs N-ACC plot. This allows us to compute the Area Under the Curve **(AUC)** of the plot to fairly compare all the methods. Similarly, we also report N-ACC, where the model has exactly 50% K-ACC, as the evaluation metric denoted by **NA@50**.

4.4. Results and Discussion

We provide the results of the datasets for E-HND along with the baselines in Table 1. From the results, we can see that E-HND outperforms the baselines for all the datasets, achieving the best performance. The superior performance proves the effectiveness of E-HND. Further, we utilize the hierarchical features (TD features) as input to our method and the best-performing baseline, leading to TD+E-HND and TD+LOO, respectively. We see an increased performance for both methods, while our method maintains the superior performance.

We present the K-ACC vs N-ACC plots for the datasets in Figure 6 for E-HND, along with the baselines. The set of biases used to obtain the plot represents the additional logits added to novel classes in the settings of (TD+LOO, relabel, LOO). For evidential models, E-HND, and TD+E-HND, the additional bias represents the 'pseudo counts'. We can alternatively achieve the K-ACC vs N-ACC for our setting by adjusting the base rate distribution. We study the impact of different base rates for novel classes in Section 4.5.

From the K-ACC vs N-ACC plots for hierarchical datasets, we observe that TD+E-HND has superior performance than other methods. Due to the training mechanism to create an evidence margin in our method, we are able to obtain higher novel accuracies than other baselines for different performances of known accuracy.

4.5. Ablation Studies

Impact of β_1 and β_2 on performance. For using E-HND, we have two hyperparameters β_1 and β_2. We recommend setting the value as $\beta_1 > \beta_2 \gg 1$. To study the impact of different values of β_1 and β_2, we plot the AUC values on test samples of the CUB dataset for different settings of β_1 and β_2. In Figure 7(a), we keep β_2 to a fixed value and vary

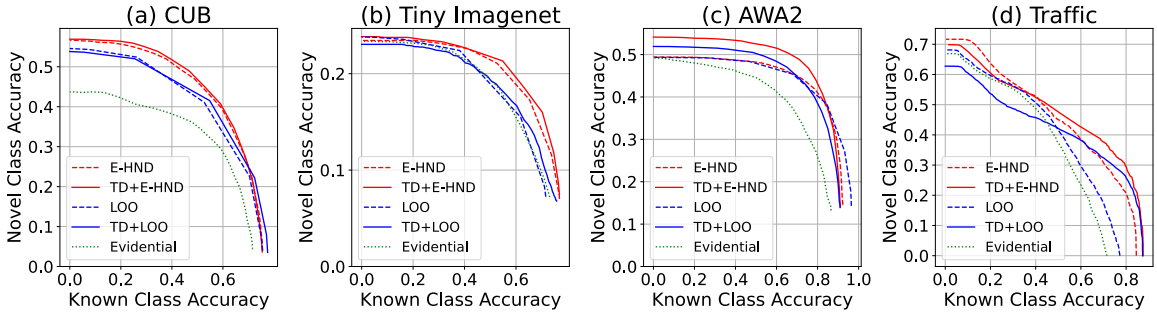

Figure 6. Known accuracy vs Novel accuracy curve for E-HND along with the baselines for 4 hierarchical datasets

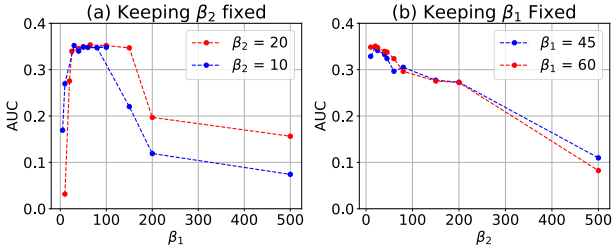

Figure 7. Impact of different values of β_1 and β_2. (a) Vary β_1 for fixed value of $\beta_2 = 10$ and $\beta_2 = 20$ for CUB (b) Vary β_2 for fixed value of $\beta_1 = 45$ and $\beta_1 = 60$

the value of β_1 to obtain performance results on the test set. Looking at the curve, we see that, for $\beta_1 < \beta_2$, the performance is on the lower region. As β_1 starts increasing, then the performance increases, becoming almost constant for a range of values. Then as β_1 reaches the value in a much higher range, the performance starts decreasing. Too high margin has lower performance, as setting $\beta_1 \gg \beta_2$ results in $\mathcal{L}_i^{(1)}(\theta)$ much higher than $\mathcal{L}_i^{(2)}(\theta)$, focusing the effect of overall loss function mostly on training known classes. Similarly, in Figure 7 (b), we keep β_1 to a fixed value and vary the value of β_1 to obtain different performance results on the test set. We see that, in the lower region of β_2, the suitable margin is created, which results in higher performance. However, as β_2 increases and $\beta_2 > \beta_1$, the performance becomes much lower. Both of the plots confirm that a reasonable margin of $\beta_1 > \beta_2$ has a positive impact on the performance. Therefore, these hyperparameters are easy to set as long as we maintain a reasonable margin of $\beta_1 > \beta_2$. However, making β_1 extremely high should be avoided as we see the decreasing performance for $\beta_1 \gg \beta_2$.

Impact of base rate distribution. In this section, we study the impact of using different base rates for novel classes. Let, $\boldsymbol{a}^{(kn)}$ and $\boldsymbol{a}^{(no)}$ denote base rates for known and novel classes respectively to represent completeness and incompleteness of the hierarchy such that

$$\sum_{k=1}^{|Le(\mathcal{H})|} a_k^{(kn)} + \sum_{k=1}^{|NLe(\mathcal{H})|} a_k^{(no)} = 1 \qquad (11)$$

Now, varying $\sum_{k=1}^{|NLe(\mathcal{H})|} a_k^{(no)}$ in the range of $[0, 1]$, we obtain the base rates of each novel class by distributing the novel base rate to all the novel classes. We can obtain corresponding values of the base rate for known classes using (11). Now, for different sets of known and novel base rates, we obtain the corresponding K-ACC and N-ACC. As we increase the value of the novel base rate, we observe in Figure 8 for CUB dataset, that K-ACC starts decreasing, and N-ACC starts increasing. This improvement is caused by an increase in 'pseudo counts' eventually increasing the 'effective' number of observations for novel classes.

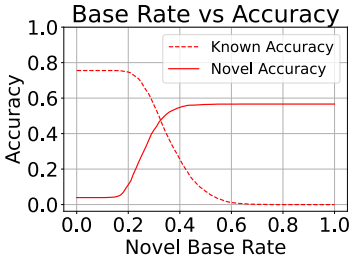

Figure 8. Impact of base rate

However, the increase in the novel base rate can not improve the N-ACC beyond the value of 57% indicating the challenge associated with correctly identifying the closest parent of a novel test sample trained with known samples only. The increase in novel base rate eventually makes the K-ACC 0% referring to the compromise that comes with using the strong prior that the hierarchy is incomplete. The plot clearly demonstrates the trade-off between K-ACC and N-ACC that can be obtained by adjusting the base rates for known and novel classes.

5. Conclusion

In this paper, we formulate a novel evidential framework to address the unique challenges associated with hierarchical novelty detection. The proposed E-HND framework leverages fine-grained evidence quantification, creating an evidence margin to distinguish between known and novel classes in the hierarchy. In order to guide the model to learn the evidence margin, we provide the design of a novel loss function with theoretical guarantees. Further, we provide a natural way to encode prior beliefs of completeness of hierarchy by leveraging base rate distribution. The proposed framework shows effectiveness in our extensive experiments with real-world hierarchical datasets.

Impact Statement

The proposed research provides a way to not only detect novel samples but also identify the closest parent class from the training data hierarchy. The work can be potentially useful in multiple applications to understand the relationship of novel samples with existing classes. Such relationships provide additional insights to make further decisions when tackling novel samples. While the field of novelty detection in AI treats novel samples and known classes as completely different entities, our work explores the idea of novel samples being related to known classes and aims to find that relationship. As a result, this work broadens the application of novelty detection by offering more information on novel samples, that eventually allows practitioners to make suitable decisions from additional insights.

References

Bendale, A. and Boult, T. E. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1563–1572, 2016.

Chang, D., Pang, K., Zheng, Y., Ma, Z., Song, Y.-Z., and Guo, J. Your" flamingo" is my" bird": Fine-grained, or not. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11476–11485, 2021.

Charpentier, B., Zügner, D., and Günnemann, S. Posterior network: Uncertainty estimation without ood samples via density-based pseudo-counts. *Advances in Neural Information Processing Systems*, 33:1356–1367, 2020.

Chen, G., Qiao, L., Shi, Y., Peng, P., Li, J., Huang, T., Pu, S., and Tian, Y. Learning open set network with discriminative reciprocal points. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pp. 507–522. Springer, 2020.

Chen, G., Peng, P., Wang, X., and Tian, Y. Adversarial reciprocal points learning for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8065–8081, 2021.

Chen, J., Wang, P., Liu, J., and Qian, Y. Label relation graphs enhanced hierarchical residual network for hierarchical multi-granularity classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4858–4867, 2022.

Deng, J., Krause, J., Berg, A. C., and Fei-Fei, L. Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3450–3457. IEEE, 2012.

Du, R., Chang, D., Bhunia, A. K., Xie, J., Ma, Z., Song, Y.-Z., and Guo, J. Fine-grained visual classification via progressive multi-granularity training of jigsaw patches. In *European Conference on Computer Vision*, pp. 153–168. Springer, 2020.

Ertler, C., Mislej, J., Ollmann, T., Porzi, L., and Kuang, Y. Traffic sign detection and classification around the world. *ArXiv*, abs/1909.04422, 2019. URL https://api.semanticscholar.org/CorpusID:202542747.

Lampert, C. H., Nickisch, H., and Harmeling, S. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2014. doi: 10.1109/TPAMI.2013.140.

Le, Y. and Yang, X. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Lee, K., Lee, K., Min, K., Zhang, Y., Shin, J., and Lee, H. Hierarchical novelty detection for visual object recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1034–1042, 2018.

Liu, W., Wang, X., Owens, J., and Li, Y. Energy-based out-of-distribution detection. *Advances in neural information processing systems*, 33:21464–21475, 2020.

Malinin, A. and Gales, M. Predictive uncertainty estimation via prior networks. *Advances in neural information processing systems*, 31, 2018.

Neal, L., Olson, M., Fern, X., Wong, W.-K., and Li, F. Open set learning with counterfactual images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

Ruiz, I. and Serrat, J. Hierarchical novelty detection for traffic sign recognition. *Sensors*, 22(12):4389, 2022.

Sensoy, M., Kaplan, L., and Kandemir, M. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems*, 31, 2018.

Vaze, S., Han, K., Vedaldi, A., and Zisserman, A. Open-set recognition: A good closed-set classifier is all you need. In *International Conference on Learning Representations*, 2021.

Wang, Y., Hu, Q., Chen, H., and Qian, Y. Uncertainty instructed multi-granularity decision for large-scale hierarchical classification. *Information Sciences*, 586:644–661, 2022.

Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., and Perona, P. Caltech-ucsd birds 200. 2010.

Yang, H.-M., Zhang, X.-Y., Yin, F., Yang, Q., and Liu, C.-L. Convolutional prototype network for open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2358–2370, 2020.

Zhang, H., Li, A., Guo, J., and Guo, Y. Hybrid models for open set recognition. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pp. 102–117. Springer, 2020.

Zhao, Y., Yan, K., Huang, F., and Li, J. Graph-based high-order relation discovery for fine-grained recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15079–15088, 2021.

Appendix

Organization of the Appendix

- In section A, we provide a summary of notations with the corresponding description used in the paper and proofs.
- In section B, we provide proof of Theorems 1 and 2.
- In section C, we provide the details of experiments and additional results.
- In section D, we discuss the limitations of this work.
- In section E, we provide the link to the source code.

A. Summary of Notations

Notation	Description
$\mathcal{H}$	A training hierarchy of known classes
y	A class from hierarchy $\mathcal{H}$
$Pa(y)$	A set of parents of class y in $\mathcal{H}$
$Ch(y)$	A set of children of a class y in $\mathcal{H}$
$An(y)$	A set of ancestors of a class y in $\mathcal{H}$
$De(y)$	A set of descendants of a class y in $\mathcal{H}$
$N(y)$	A set of novel classes for closest known parent y
$Le(\mathcal{H})$	A set of leaf classes of hierarchy $\mathcal{H}$
$NLe(\mathcal{H})$	A set of non-leaf classes of hierarchy $\mathcal{H}$
$\boldsymbol{p}^{(kn)}$	Probability distribution for known classes
$\boldsymbol{p}^{(no)}$	Probability distribution for novel classes
$\mathcal{H} \setminus c$	Hierarchy obtained by removing class c
$Le'(\mathcal{H} \setminus c)$	Leaf classes for the hierarchy $\mathcal{H} \setminus c$
$\mathcal{L}^{CE}(\theta)$	Cross Entropy based loss function for LOO training
$\boldsymbol{b}$	Belief distribution for K classes
$\boldsymbol{a}$	Base rate distribution for K classes
u	Uncertainty mass
w	Multinomial opinion
$\boldsymbol{\alpha}$	Parameters of Dirichlet PDF
$\boldsymbol{r}$	Evidence distribution for K classes
W	Non-informative prior weight
$\mathcal{L}_i(\theta)$	E-HND loss function for sample i
$\mathcal{L}_i^{(1)}(\theta)$	E-HND loss term when no class is removed
$\mathcal{L}_{i,c}^{(2)}(\theta)$	E-HND loss term when class c is removed
$j^{\mathcal{H}}$	Ground truth index for $\mathcal{L}_i^{(1)}(\theta)$
$j^{\mathcal{H}\setminus c}$	Ground truth index for $\mathcal{L}_{i,c}^{(2)}(\theta)$
β_1	Baseline Dirichlet parameter for $j^{\mathcal{H}}$
β_2	Baseline Dirichlet parameter for $j^{\mathcal{H}\setminus c}$
$\boldsymbol{a}^{(kn)}$	Base rate distribution for known classes
$\boldsymbol{a}^{(no)}$	Base rate distribution for novel classes
$S(\boldsymbol{\alpha}(\mathcal{H}))$	A set of Dirichlet parameters of hierarchy $\mathcal{H}$
$S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H}))$	A set of Dirichlet parameters by $\mathcal{L}_i^{(1)}(\theta)$
$S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H}))$	A set of Dirichlet parameters by $\mathcal{L}_i^{(2)}(\theta)$

B. Proof of Theorems

B.1. Proof of Theorem 1

To prove the theorem, we first define a general KL divergence-based loss with target ground truth parameter β as given by:

$$\mathcal{L} = KL[D(\boldsymbol{p}|\boldsymbol{\alpha})||D(\boldsymbol{p}|\hat{\boldsymbol{\alpha}})] = \ln\Gamma(\alpha_0) - \ln\Gamma(\hat{\alpha}_0) + \sum_{k=1}^{K} \ln\Gamma(\hat{\alpha}_k) - \ln\Gamma(\alpha_k) + \sum_{k=1}^{K}(\alpha_k - \hat{\alpha}_k)[\psi(\alpha_k) - \psi(\alpha_0)] \quad (12)$$

$\hat{\alpha_j} = \beta, \hat{\alpha_0} = \beta + K - 1$. For $k \neq j, \hat{\alpha_k} = 1$, we have the value as:

$$\begin{aligned}\mathcal{L} &= \ln\Gamma(\alpha_0) - \ln\Gamma(\beta + K - 1) + \ln\Gamma(\beta) - \ln\Gamma(\alpha_j) + (\alpha_j - \beta)[\psi(\alpha_j) - \psi(\alpha_0)] \\ &+ \sum_{k=1,k\neq j}^{K} \ln\Gamma(1) - \ln\Gamma(\alpha_k) + \sum_{k=1,k\neq j}^{K} (\alpha_k - 1)[\psi(\alpha_k) - \psi(\alpha_0)] \end{aligned} \quad (13)$$

The non-ground truth Dirichlet parameter $\alpha_k, k \neq j$ is fixed at 1 to make the analysis easier. The loss function becomes

$$\begin{aligned}\mathcal{L}[\alpha_{k,k\neq j} = 1] &= \ln\Gamma(\alpha_0) - \ln\Gamma(\beta + K - 1) + \ln\Gamma(\beta) - \ln\Gamma(\alpha_j) + (\alpha_j - \beta)[\psi(\alpha_j) - \psi(\alpha_0)] \\ &= \ln\Gamma(\alpha_j + K - 1) - \ln\Gamma(\beta + K - 1) + \ln\Gamma(\beta) - \ln\Gamma(\alpha_j) + (\alpha_j - \beta)[\psi(\alpha_j) - \psi(\alpha_j + K - 1)] \end{aligned} \quad (14)$$

For derivative, we use the following relations:

$$\begin{aligned}\frac{\mathrm{d}\ln\Gamma(z)}{\mathrm{d}z} &= \psi(z) \\ \frac{\mathrm{d}\psi(z)}{\mathrm{d}z} &= \psi^1(z) \\ \psi^1(z+1) &= \psi^1(z) - \frac{1}{z^2} \\ \psi(z) &= \ln(z) - \frac{1}{2z}\end{aligned}$$

Taking the derivative w.r.t loss function, we have:

$$\frac{\mathrm{d}\mathcal{L}[\alpha_{k,k\neq j} = 1]}{\mathrm{d}\alpha_j} = \psi(\alpha_j + K - 1) - 0 + 0 - \psi(\alpha_j) + (\alpha_j - \beta)[\psi^1(\alpha_j) - \psi^1(\alpha_j + K - 1)] + 1[\psi(\alpha_j) - \psi(\alpha_j + K - 1)] \quad (15)$$

$$= (\alpha_j - \beta)[\psi^1(\alpha_j) - \psi^1(\alpha_j + K - 1)] \quad (16)$$

$$= (\alpha_j - \beta)[\psi^1(\alpha_j) - \{\psi^1(\alpha_j) - \frac{1}{(\alpha_j + K - 2)^2} - \frac{1}{(\alpha_j + K - 3)^2} - ... - \frac{1}{\alpha_j^2}\}] \quad (17)$$

$$= (\alpha_j - \beta)[\frac{1}{(\alpha_j + K - 2)^2} + \frac{1}{(\alpha_j + K - 3)^2} + ... + \frac{1}{\alpha_j^2}] \quad (18)$$

Now, we prove that $\mathcal{L}$ decreases when there is increase in ground truth parameter α_j till β; it becomes 0 at $\alpha_j = \beta$ and increases after α_j becomes greater than β using Lemma 3, 4, and 5.

Lemma 3. *KL divergence loss becomes zero when ground truth Dirichlet parameter α_j is equal to β.*

Proof.

$$\mathcal{L}[\alpha_j = \beta, \alpha_{k,k\neq j} = 1] = \ln\Gamma(\beta + K - 1) - \ln\Gamma(\beta + K - 1) + \ln\Gamma(\beta) - \ln\Gamma(\beta) + (\beta - \beta)[\psi(\beta) - \psi(\beta + K - 1)] + \sum_{k=1,k\neq j}^{K} \ln\Gamma(1) - \ln\Gamma(1) + \sum_{k=1,k\neq j}^{K} (1-1)[\psi(1) - \psi(\beta + K - 1)] = 0 \tag{19}$$

□

Lemma 4. *KL divergence loss decreases when there is an increase in ground truth Dirichlet parameter α_j until it reaches the fixed value β.*

Proof. Using equation 18, when ground truth $\alpha_j < \beta$, $\frac{\mathrm{d}\mathcal{L}[\alpha_{k,k\neq j}=1]}{\mathrm{d}\alpha_j} < 0$. Hence, for $\alpha_j < \beta$, loss decreases for increases in α_j. □

Lemma 5. *KL divergence loss increases when there is an increase in ground truth Dirichlet parameter, α_j when α_j becomes greater than the fixed value β.*

Proof. Using equation 18, when ground truth $\alpha_j > \beta$, $\frac{\mathrm{d}\mathcal{L}[\alpha_{k,k\neq j}=1]}{\mathrm{d}\alpha_j} > 0$. Hence, for $\alpha_j > \beta$, loss increases for increases in α_j. □

B.2. Proof of Theorem 2

$S(.)$ denotes the operation that converts a vector to a set. The total Dirichlet parameters from the model are given by $S(\boldsymbol{\alpha}(\mathcal{H}))$. For the purpose of analysis, we use a data sample i and separate the parameters trained by $\mathcal{L}_i^{(1)}(\theta)$ and $\mathcal{L}_i^{(2)}(\theta)$ for the sample into two sets: $S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H}))$ and $S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H}))$ respectively. In a more fine-grained manner, let $S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H} \setminus c))$ denote parameters trained by $\mathcal{L}_{i,c}^{(2)}(\theta)$. The total Dirichlet parameters in the model can be obtained as:

$$S(\boldsymbol{\alpha}_i(\mathcal{H})) = S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H})) \bigcup_{y\in Le(\mathcal{H})} \bigcup_{c\in An(y)} S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H} \setminus c)) \tag{20}$$

The common parameters between $S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H}))$ and $S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H} \setminus c))$ denoted by $S(\boldsymbol{\alpha}_i(\mathcal{H}, \mathcal{H} \setminus c))$ is obtained by

$$S(\boldsymbol{\alpha}_i(\mathcal{H}, \mathcal{H} \setminus c)) = S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H})) \cap S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H} \setminus c)) \tag{21}$$

$$= S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H})) \setminus \bigcup_{d\in Le(\mathcal{H}), d\in De(c)} \alpha_{id} \tag{22}$$

The common parameters do not include ground truth parameters from $\mathcal{L}_i^{(1)}(\theta)$ and $\mathcal{L}_{i,c}^{(2)}(\theta)$ given by $\alpha_{ij^{\mathcal{H}}}$ and $\alpha_{ij^{\mathcal{H}\setminus c}}$, but only the non-ground truth parameters for sample i that doesn't belong to descendants of class c. Now, the difference of parameters between $S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H}))$ and $S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H} \setminus c))$ is given by:

$$S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H})) \setminus S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H} \setminus c)) = \{\alpha_{ij^{\mathcal{H}}}\} \bigcup_{d\in Le(\mathcal{H}), d\in De(c)} \alpha_{id} \tag{23}$$

$$S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H} \setminus c)) \setminus S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H})) = \{\alpha_{ij^{\mathcal{H}\setminus c}}\} \tag{24}$$

Now, the common and difference of parameters between $S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H}))$ and $\bigcup_{c\in An(c)} S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H} \setminus c))$ is obtained by

$$S(\boldsymbol{\alpha}_i(\mathcal{H}, \bigcup_{c\in An(c)} \mathcal{H}\setminus c)) = S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H}))\setminus\{\alpha_{ij^{\mathcal{H}}}\} \tag{25}$$

$$S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H}))\setminus \bigcup_{c\in An(c)} S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H}\setminus c)) = \{\alpha_{ij^{\mathcal{H}}}\} \tag{26}$$

$$[\bigcup_{c\in An(c)} S(\boldsymbol{\alpha}_i^{(2)}(\mathcal{H}\setminus c))]\setminus S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H})) = \bigcup_{c\in An(c)}\{\alpha_{ij^{\mathcal{H}\setminus c}}\} \tag{27}$$

From (25), we see that the common parameters between $\mathcal{L}_i^{(1)}(\theta)$ and $\mathcal{L}_i^{(2)}(\theta)$ include parameters only from non-ground truth known leaf classes. Moreover, the parameter only trained by $\mathcal{L}_i^{(1)}(\theta)$ is parameter of ground truth known leaf class. Finally, the parameters only trained by $\mathcal{L}_i^{(2)}(\theta)$ are ground truth novel non-leaf classes. Now, we provide the definition of conflicting update.

Definition 6 (Conflicting update)**.** A conflicting update is defined for $\mathcal{L}_i^{(1)}(\theta)$, $\mathcal{L}_i^{(2)}(\theta)$ and a parameter α when either of the following conditions is true:

- Condition I: $\mathcal{L}_i^{(1)}(\theta)$ increases α and $\mathcal{L}_i^{(2)}(\theta)$ decreases α.
- Condition II: $\mathcal{L}_i^{(1)}(\theta)$ decreases α and $\mathcal{L}_i^{(2)}(\theta)$ increases α.

Next, for ground truth parameters, we prove that there is no conflicting update between loss terms using lemma 7.

Lemma 7. *For the ground truth parameters, $\{\alpha_{ij^{\mathcal{H}}}\}$ and $\bigcup_{c\in An(c)}\{\alpha_{ij^{\mathcal{H}\setminus c}}\}$, conditions I and II from Definition 6 do not hold.*

Proof.
- $\frac{\mathrm{d}\mathcal{L}_i^{(2)}(\theta)}{\mathrm{d}\alpha_{ij^{\mathcal{H}}}} = 0$ as $\mathcal{L}_i^{(2)}(\theta)$ is not the function of $\alpha_{ij^{\mathcal{H}}}$.
- From the definition 6, both conditions I and II do not hold, as $\mathcal{L}_i^{(2)}(\theta)$ neither increases nor decreases $\alpha_{ij^{\mathcal{H}}}$.
- $\forall c, c\in An(y), \frac{\mathrm{d}\mathcal{L}_i^{(1)}(\theta)}{\mathrm{d}\alpha_{ij^{\mathcal{H}\setminus c}}} = 0$ as $\mathcal{L}_i^{(1)}(\theta)$ is not the function of $\alpha_{ij^{\mathcal{H}\setminus c}}$.
- From the definition 6, both conditions I and II do not hold, as $\mathcal{L}_i^{(1)}(\theta)$ neither increases nor decreases $\alpha_{ij^{\mathcal{H}\setminus c}}$.
- Hence, there are no conflicting updates for the parameters: $\{\alpha_{ij^{\mathcal{H}}}\}$ and $\bigcup_{c\in An(c)}\{\alpha_{ij^{\mathcal{H}\setminus c}}\}$

□

Now, for analysis of updates for non-ground truth parameters, we prove that KL divergence-based loss trains the model to output 1 using Lemma 8.

Lemma 8. *KL divergence loss increases when there is an increase in non-ground truth Dirichlet parameter α_k, for $k\neq j$.*

For ease of analysis, fix $\alpha_j = \beta$. For K Dirichlet parameters, suppose K is the ground truth index, $\alpha_1, \alpha_2, ..., \alpha_{K-1}$ are non-ground truth Dirichlet parameters.

$$\mathcal{L}[\alpha_j = \alpha_K = \beta] = \ln\Gamma(\alpha_0) - \ln\Gamma(\beta+K-1) + \sum_{k=1,k\neq j}^{K-1}\ln\Gamma(1) - \ln\Gamma(\alpha_k) + \sum_{k=1,k\neq j}^{K-1}(\alpha_k-1)[\psi(\alpha_k)-\psi(\alpha_0)] \tag{28}$$

Taking the derivative of equation 28 w.r.t α_1, we have

$$\frac{\mathrm{d}\mathcal{L}[\alpha_j=\alpha_K=\beta]}{\mathrm{d}\alpha_1} = \psi(\alpha_0)-\psi(\alpha_1)+(\alpha_1-1)[\psi^1(\alpha_1)-\psi^1(\alpha_0)]+\psi(\alpha_1)-\psi(\alpha_0)+\sum_{k=2,k\neq j}^{K-1}(\alpha_k-1)[-\psi^1(\alpha_0)] \tag{29}$$

$$= (\alpha_1-1)[\psi^1(\alpha_1)-\{\psi^1(\alpha_1)-\frac{1}{(\alpha_0-1)^2}-\frac{1}{(\alpha_0-2)^2}-...-\frac{1}{(\alpha_1)^2}\}]+\sum_{k=2,k\neq j}^{K-1}(\alpha_k-1)[-\psi^1(\alpha_0)] \tag{30}$$

$$= (\alpha_1-1)[\frac{1}{(\alpha_0-1)^2}+\frac{1}{(\alpha_0-2)^2}+...+\frac{1}{(\alpha_1)^2}]+\sum_{k=2,k\neq j}^{K-1}(\alpha_k-1)[-\psi^1(\alpha_0)] \tag{31}$$

Here, the first term is positive, and for the second, term we use the limit definition to take the derivative of $\psi(\alpha_0)$, we have the relation:

$$\begin{aligned}\psi^1(\alpha_0) &= \lim_{\triangle\alpha_1\to 0}\frac{\psi(\alpha_0+\triangle\alpha_1)-\psi(\alpha_0)}{\triangle\alpha_1}\\ &= \lim_{\triangle\alpha_1\to 0}\frac{\ln(\alpha_0+\triangle\alpha_1)-\frac{1}{2(\alpha_0+\triangle\alpha_1)}-\ln(\alpha_0)+\frac{1}{2\alpha_0}}{\triangle\alpha_1}\\ &= \lim_{\triangle\alpha_1\to 0}\frac{\ln(1+\frac{\triangle\alpha_1}{\alpha_0})+\frac{1}{2}\frac{\triangle\alpha_1}{\alpha_0(\alpha_0+\triangle\alpha_1)}}{\triangle\alpha_1}\end{aligned}$$

This is a $\frac{0}{0}$ form. Hence, using L'Hopital rule, taking derivative on both numerator and denominator, we have

$$\begin{aligned}\psi^1(\alpha_0) &= \lim_{\triangle\alpha_1\to 0}\frac{\frac{1}{\alpha_0+\triangle\alpha_1}+\frac{1}{2(\alpha_0+\triangle\alpha_1)^2}-\frac{1}{\alpha_0}-\frac{1}{2\alpha_0^2}}{1}\\ \psi^1(\alpha_0) &= \lim_{\triangle\alpha_1\to 0}-\frac{\triangle\alpha_1}{\alpha_0(\alpha_0+\triangle\alpha_1)}-\frac{2\alpha_0\triangle\alpha_1+\triangle\alpha_1^2}{2\alpha_0^2(\alpha_0+\triangle\alpha_1)^2}<0\end{aligned}$$

Hence, the second term is also positive, making $\frac{\mathrm{d}\mathcal{L}[\alpha_j=\alpha_K=\beta]}{\mathrm{d}\alpha_1}>0$. Therefore, KL divergence loss increases when there is an increase in non-ground truth Dirichlet parameter α_1. The minimum allowed value of the Dirichlet parameter is 1. Hence, the non-ground truth parameter approaches 1. This can be proved similarly for other non-ground truth Dirichlet parameters $\alpha_k, k\neq j$.

Finally, for common parameters between loss terms: non-ground truth parameters, we prove that there is no conflicting update between loss terms using lemma 9.

Lemma 9. *For the common parameters, $S(\boldsymbol{\alpha}_i(\mathcal{H},\bigcup_{c\in An(c)}\mathcal{H}\setminus c))$, conditions I and II from definition 6 do not hold.*

Proof.
- We observe that the common parameter $\alpha_i\in S(\boldsymbol{\alpha}_i^{(1)}(\mathcal{H}))\setminus\{\alpha_{ij^{\mathcal{H}}}\}$ is a non-ground truth parameter for both $\mathcal{L}_i^{(1)}(\theta)$ and $\mathcal{L}_{i,c}^{(2)}(\theta), c\in An(y)$.
- For a non-ground truth parameter, $\alpha_{ik}, k\neq j^{\mathcal{H}}, k\neq j^{\mathcal{H}\setminus c}, \forall c\in An(y)$, the updates from loss terms are given by $\frac{\mathrm{d}\mathcal{L}_i^{(1)}(\theta)}{\mathrm{d}\alpha_{ik}}>0$ and $\frac{\mathrm{d}\mathcal{L}_i^{(2)}(\theta)}{\mathrm{d}\alpha_{ik}}>0$.
- From the definition 6, condition I is false as $\mathcal{L}_i^{(1)}(\theta)$ decreases α_{ik}.
- From the definition 6, condition II is false as $\mathcal{L}_i^{(2)}(\theta)$ decreases α_{ik}. Hence, both conditions I and II do not hold.
- Hence, there are no conflicting updates for the common parameters: $S(\boldsymbol{\alpha}_i(\mathcal{H},\bigcup_{c\in An(c)}\mathcal{H}\setminus c))$.

□

Table 2. Significance Test

	CUB		Tiny Imagenet		AWA2		Traffic	
	NA@50	**AUC**	**NA@50**	**AUC**	**NA@50**	**AUC**	**NA@50**	**AUC**
TD+LOO	44.82±0.52	34.50±0.14	19.16±0.33	14.31±0.36	50.13±0.12	42.23±0.24	42.41±0.50	38.22±0.39
TD+E-HND	**46.92** ± 0.23	**35.83** ± 0.07	**21.80** ± 0.10	**16.39** ± 0.14	**53.76** ± 1.2	**45.39** ± 0.40	**47.69**±0.14	**43.11**±0.11
p-value	8.20×10^{-10}	7.71×10^{-16}	3.94×10^{-15}	2.98×10^{-13}	1.94×10^{-8}	3.14×10^{-14}	2.13×10^{-17}	1.08×10^{-18}

C. Details of Experiments and Additional Results

C.1. Significance Test

The result presented in table 1 is from a single run of the method. Therefore, we perform a t-test to find out if the difference between evaluation metrics between our method and the baselines is significant enough. We run our method(TD+HND) and the best performing baseline(TD+LOO) for 10 times using different values of random seed. The obtained mean, standard deviation and p-values obtained are presented in table 2. We can see from the table that p-values obtained for the combination of all the datasets and evaluation metrics are low. Hence, the gap between the evaluation metrics of TD+E-HND and TD+LOO is significant.

C.2. Training Details

To speed up the training, we use a standard Resnet-101 architecture as a backbone to extract the features from the training samples for CUB, AWA2 and Tinyimagenet datasets. For traffic(MTSD) dataset, we use Resnet101 features provided by (Ruiz & Serrat, 2022). We train the model using the full batch of Resnet-101 features with Adam optimizer and an initial learning rate of 10^{-2}. We use the validation set to select the suitable hyperparameters β_1 and β_2. The validation set does not include samples from the novel classes. We use the set of (β_1, β_2) values of $(65, 20)$, $(30, 5)$, $(20, 5)$ and $(40, 5)$ for CUB, AWA2, Tiny Imagenet and Traffic respectively. For obtaining evidence from the classification network, we use Softplus activation on logit. The detailed algorithm is provided in Algorithm 1.

All the experiments are conducted using NVIDIA GeForce RTX 3060 with 32GB memory. The training algorithm is implemented in pytorch version: 1.13.0 and cuda version: 11.6. The hierarchy information associated with datasets is first computed and stored in a .npy file using numpy(a library of python) to avoid runtime computations of hierarchy information during the model training.

Algorithm 1 E−HND Training

Require Hyperparameters: β_1, β_2
Require Hierarchy definition: $\mathcal{H}$ of the known classes.
Input Initialized Model: θ.
Input A set of training samples with ground truth labels $\{(x_i, y_i)\}_{i=1}^{N}$.
while not StopCriterion **do**
  for a pair of training sample with ground truth label (x_i, y_i) **do**
    Calculate observed evidences for all the classes $\boldsymbol{r}_i = [r_{i1}, r_{i2}, ...r_{iK}]$ using the model using the equation: $r_k = \text{softplus}(\theta(x_i))$
    Calculate model predicted Dirichlet parameters $\boldsymbol{\alpha}_i$ for each class k using equation 5.
    Calculate the loss $\mathcal{L}_i(\theta)$ using equation 8.
  end for
  Calculate the total loss for all the samples $\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_i(\theta)$.
  Update model parameter θ by backpropagating loss $\mathcal{L}(\theta)$
end while

Table 3. Comparison with the evidential design of loss in HND

Method	CUB	
	NA@50	AUC
Log loss (Sensoy et al., 2018)	35.06	25.86
Digamma loss (Sensoy et al., 2018)	37.01	26.77
MSE loss (Sensoy et al., 2018)	13.15	15.32
E-HND	**46.18**	**35.31**

C.3. Additional Experiment Results

Impact of Evidential Loss

We designed the loss function to learn fine-grained evidence for known and novel classes. In this section, we study the impact of using the loss function based on the evidential formulation provided by (Sensoy et al., 2018). We carry out experiments on the CUB dataset. We design these loss functions in the setting of Hierarchical Novelty Detection:

$$\mathcal{L}_i^{\log}(\theta) = \sum_{k=1}^{|Le(\mathcal{H})|} y_{ik}[\ln(St_i^{\mathcal{H}}) - \ln(\alpha_{ik})] + \sum_{c \in An(y)} \sum_{k=1}^{|Le'(\mathcal{H}\backslash c)|} y_{ik}[\ln(St^{\mathcal{H}\backslash c}) - \ln(\alpha_{ik})] \tag{32}$$

$$\mathcal{L}_i^{\text{digamma}}(\theta) = \sum_{k=1}^{|Le(\mathcal{H})|} y_{ik}[\psi(St_i^{\mathcal{H}}) - \psi(\alpha_{ik})] + \sum_{c \in An(y)} \sum_{k=1}^{|Le'(\mathcal{H}\backslash c)|} y_{ik}[\psi(St_i^{\mathcal{H}\backslash c}) - \psi(\alpha_{ik})] \tag{33}$$

$$\begin{aligned}\mathcal{L}_i^{\text{mse}}(\theta) = &\sum_{k=1}^{|Le(\mathcal{H})|} [(y_{ik} - \alpha_{ik}/St_i^{\mathcal{H}})^2 + \frac{\alpha_{ik}(St_i^{\mathcal{H}} - \alpha_{ik})}{(St_i^{\mathcal{H}})^2(St_i^{\mathcal{H}} + 1)}] \\ &+ \sum_{c \in An(y)} \sum_{k=1}^{|Le'(\mathcal{H}\backslash c)|} [(y_{ik} - \alpha_{ik}/St_i^{\mathcal{H}\backslash c})^2 + \frac{\alpha_{ik}(St_i^{\mathcal{H}\backslash c} - \alpha_{ik})}{(St_i^{\mathcal{H}\backslash c})^2(St_i^{\mathcal{H}\backslash c} + 1)}]\end{aligned} \tag{34}$$

The strengths are calculated using the following relations:

$$St_i^{\mathcal{H}} = \sum_{k=1}^{|Le(\mathcal{H})|} \alpha_{ik}, \quad St_i^{\mathcal{H}\backslash c} = \sum_{k=1}^{|Le'(\mathcal{H}\backslash c)|} \alpha_{ik} \tag{35}$$

The results are presented in table 3. We observe that E-HND has the best performance as provided by NA@50 and the AUC. As other forms of evidential loss from equations [32, 33, 34] do not allow to upper bound the evidence for ground truth. We can not use these formulations to create evidence-margin between known and novel classes. Hence, the design of the loss function of E-HND is justified.

Impact of base rate distribution

We show the impact of the base rate distribution using the CUB dataset in the main paper. Here, we show the result on the two remaining datasets: AWA2, Tiny Imagenet and Traffic in Figure 9. The trend for these datasets is similar to CUB. As we see in (a), (b) and (c), as the novel base rate increases, the N-ACC starts increasing due to the effect of an increase in pseudo counts for novel classes. We observe that the highest N-ACCs obtained by adjusting the base rate are 58%, 22% and 72% for AWA2, Tiny Imagenet, and Traffic respectively. The upper bound of N-ACC is lower for Tiny Imagenet as the hierarchy is deeper than AWA2 and Traffic, making the task of identifying the closest parent for novel samples much more difficult. In addition, the base rate curve clearly shows the trade-off between K-ACC and N-ACC.

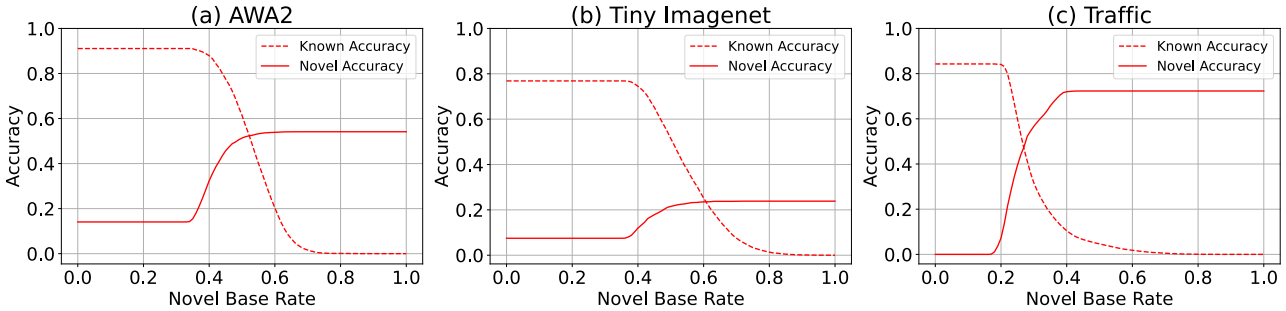

Figure 9. Impact of using base rate distribution for (a) AWA2, (b) TinyImagenet, (c) Traffic dataset

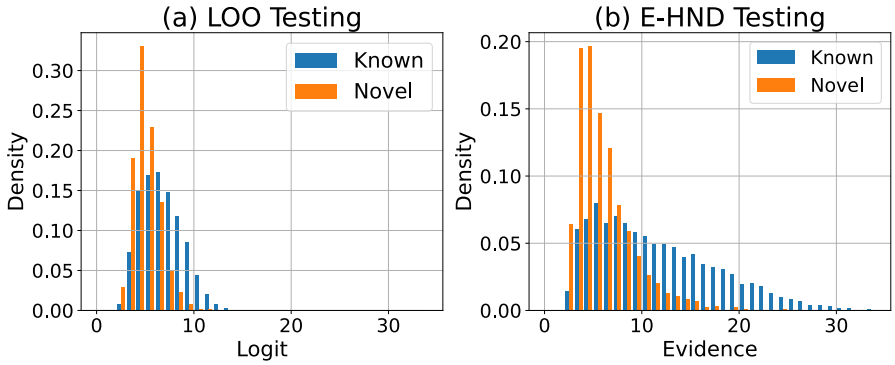

Figure 10. Comparison of distribution of (a) logits and (b) evidences for known and novel test samples in Tiny Imagenet dataset

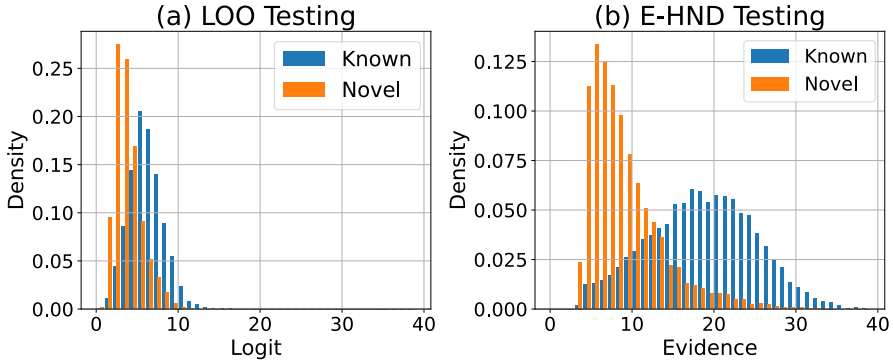

Figure 11. Comparison of distribution of (a) logits and (b) evidences for known and novel test samples in AWA2

Impact of the evidence margin

We show the effect of learning evidence margin in Figure 5 for CUB test samples in the main paper. Learning to create evidence margin for known and novel classes has the effect of lower evidence allocation to novel test samples than known test samples, while LOO allocates high logits to both known and novel test samples. We plot the logits and evidence distribution of (a) E-HND and (b) LOO methods for Tiny Imagenet dataset in Figure 10, AWA2 dataset in Figure 11 and Traffic dataset in Figure 12. A similar effect is observed for known and novel test samples in the rest of the datasets. Due to this effect, our method has improved novelty detection performance in comparison to LOO.

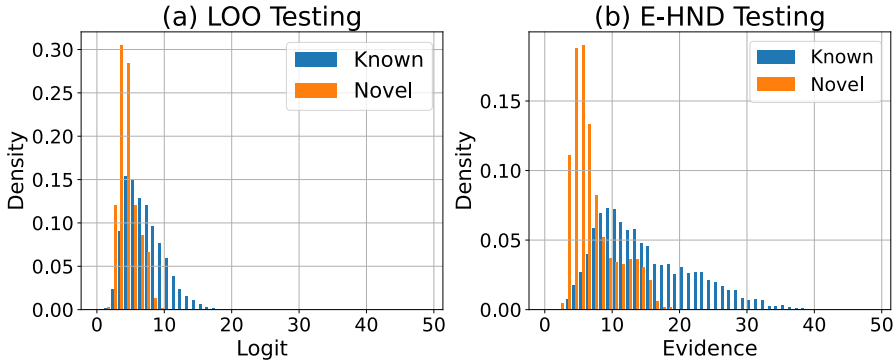

Figure 12. Comparison of distribution of (a) logits and (b) evidences for known and novel test samples in Traffic dataset

Comparison with Novelty Detection Methods

In table 1, we use the features from one of the hierarchical classification augmented with novelty detection(HC-ND) methods as input to LOO and E-HND methods. We see that, when features from HC-ND method are used, the performance increases for all the datasets. In this section, we compare the result for (i$^+$) flatten methods with (ii*) HC-ND methods. For HC-ND methods, each non-leaf node is treated as a classifier. $\mathcal{O}(nle))$ are the leaf nodes that do not belong to the descendants of the non-leaf node nle as a classifier. We describe HC-ND methods as:

- **Top-Down(TD)*** (Lee et al., 2018): The classifiers are trained with cross-entropy loss. A regularization is used to induce uniform probability values for samples that do not belong to the descendants of the classifier. The loss function and confidence score comparison are defined by (Lee et al., 2018) for every non-leaf class in the hierarchy as:

$$\forall nle \in NLe(\mathcal{H}), \mathcal{L}^{\text{TD}} = \mathop{\mathbb{E}}_{p(x,y|nle)}[-\ln p(y|x, nle; \theta_{nle})] + \mathop{\mathbb{E}}_{p(x,y|\mathcal{O}(nle))} KL[U(.|nle)||p(.|x, nle; \theta_{nle})] \tag{36}$$

$$KL[U(.|nle)||p(.|x, nle; \theta_{nle})] \geq \lambda_{nle} \tag{37}$$

- **Maximum Softmax Probability(MSP)*** (Vaze et al., 2021) Maximum Softmax Probability is one of the most widely used baselines in the field of novelty detection. For MSP baseline, we use the loss from equation 36. For the final prediction, we use the confidence score comparison as:

$$max(Pr(.|x, nle; \theta_{nle})) \geq \lambda_{nle} \tag{38}$$

- **HC-ND with $|Ch(nle)|+1$ class*** (Neal et al., 2018) The baseline denotes novel class by $K+1$ for novelty detection in multi-class classification of K classes. Following the method, we modify the HC-ND method such that each non-leaf class contains an extra novel class to classify. Now, instead of using regularization that induces uniform probability distribution, data samples from $\mathcal{O}(nle)$ are used as the samples of novel $|Ch(nle)|+1$ class. The resulting baseline becomes threshold free and contains $|NLe(\mathcal{H})|$ novel nodes. We define the loss function and confidence score comparison as:

$$\forall nle \in NLe(\mathcal{H}), \mathcal{L}^{|Ch(nle)|+1} = \mathop{\mathbb{E}}_{p(x,y|nle)}[-\ln p(y|x, nle; \theta_{nle})] + \mathop{\mathbb{E}}_{p(x,y|\mathcal{O}(nle))}[-\ln p(N(nle)|x, nle; \theta_{nle})] \tag{39}$$

$$max(Pr(.|x, nle; \theta_{nle})) \geq p(N(nle)|x, nle; \theta_{nle}) \tag{40}$$

- **Evidential Uncertainty*** (Sensoy et al., 2018) We use uncertainty mass from evidential theory as an uncertainty measure. We use log loss (Sensoy et al., 2018) to train the classifier to classify correct class and output low uncertainty to known samples. To train the classifier to output high uncertainty for novel samples, we define the regularization using KL divergence between uniform Dirichlet distribution and model Dirichlet distribution:

Table 4. Comparison with the novelty detection methods for CUB dataset

Category	Method	Known Accuracy	Novel Accuracy	Harmonic Mean
Hierarchical	TD*	45.94	24.69	32.12
	Max Softmax Probability*	46.34	25.65	33.02
	HC-ND with $\|Ch(nle)\|+1$ class*	46.83	26.02	33.45
	Energy Score*	47.69	30.56	37.25
	Max Logit Score*	47.80	30.76	37.43
	Evidential uncertainty*	47.37	27.50	34.80
Flatten	Relabel+	50.00	38.23	43.33
	LOO+	50.00	42.25	45.80
	E-HND+	**50.00**	**46.18**	**48.01**

$$\forall nle \in NLe(\mathcal{H}), \mathcal{L}^{\text{HC-evidential}} = \mathop{\mathbb{E}}_{p(x,y|nle)} \sum_{k=1}^{|Ch(nle)|} y_{ik}[\ln(St_i^{nle}) - \ln(\alpha_{ik})] + \mathop{\mathbb{E}}_{p(x,y|\mathcal{O}(nle))} KL[D(.|x,nle;\theta_{nle})||D(.|<1,1,...,>,nle)] \tag{41}$$

$$\frac{|Ch(nle)|}{St_i^{nle}} \leq \lambda_{nle} \tag{42}$$

- **Energy Score*** (Liu et al., 2020) We use the loss function as equation 41 to train the model. If $f_k(x;\theta_{nle})$ represents the logit for k^{th} children of non-leaf node nle. The uncertainty comparison is given by:

$$-\ln \sum_{k=1}^{|Ch(nle)|} e^{f_k(x;\theta_{nle})} \leq \lambda_{nle} \tag{43}$$

- **Maximum Logit Score*** (Vaze et al., 2021) We use the loss function as equation 41 to train the model. The uncertainty comparison is given by:

$$max(f_k(x;\theta_{nle})) \geq \lambda_{nle} \tag{44}$$

For the prediction, a sample is classified at each classifier starting from top from towards leaf nodes. Each classifier quantifies a confidence/uncertainty score to denote how confident/uncertain the classifier is towards the prediction. If the confidence/uncertainty is greater/smaller than a threshold, then its predicted class is used as next classifier till we get the final prediction. Thresholds required for confidence/uncertainty comparison are calculated using validation set that maximizes the harmonic mean between known and novel accuracy. Since the validation set does not contain real novel samples, novel samples are defined as samples from leaf nodes that do not belong to the descendants of the classifier. For flatten methods(+): Relabel, LOO, and our method(E-HND), we use the result where known accuracy is fixed at 50%. We also report the harmonic mean between known accuracy and novel accuracy for each method.

We use CUB dataset to report the results for the baselines along with our method(E-HND) in table 4. We see that in comparison to TD method, using the maximum softmax probability score is effective for both known and novel classes. Similarly, with the use of other novelty scores like energy scores, maximum logit scores, and evidential uncertainty, the performance improves for both known and novel classes. However, these are still outperformed by all the compared flatten baselines like Relabel, LOO, and E-HND. A reason behind this might be the need to set the total of $|NLe(\mathcal{H})|$ thresholds, each for a classifier, in HC-ND methods. Even when we eliminate the need to set thresholds as in HC-ND with $|Ch(nle)|+1$ class*, the result does not improve much. All HC-ND methods share a common top-down inference mechanism that can cause error accumulations. However, this is not prevalent in flatten methods. The comparison results show that the common baselines for novelty detection do not yield the best performances in the setting of hierarchical novelty detection.

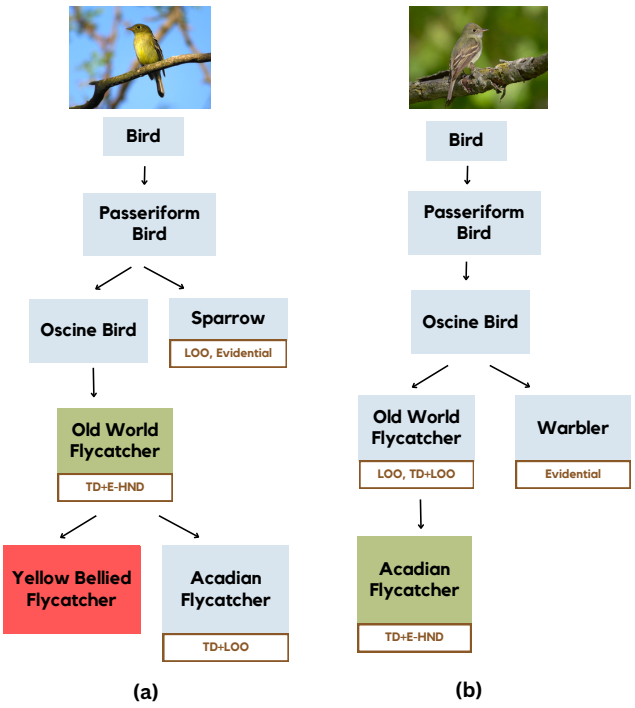

Figure 13. Qualitative study for representative test samples: (a) Prediction for novel sample from Yellow Bellied Flycatcher (b) Prediction for known sample from Acadian Flycatcher.

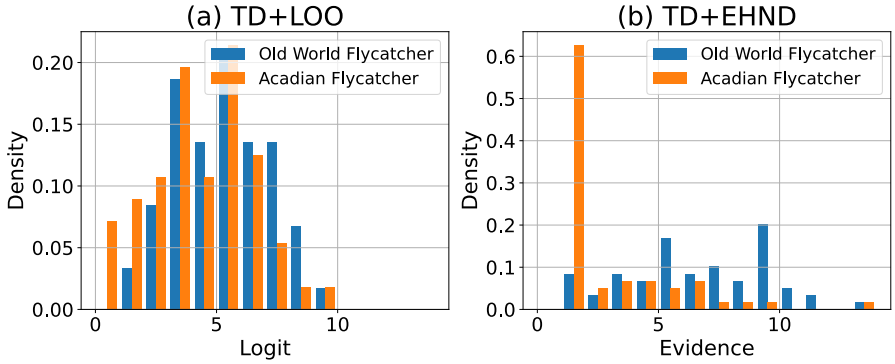

Figure 14. Comparison of the distribution of (a) logit and (b) evidence for Acadian Flycatcher and Old World Flycatcher by (a) TD+LOO and (b) TD+E-HND

C.4. Qualitative Study

We show the prediction of TD+E-HND with some of the baselines, visualized with the corresponding hierarchy for the representative samples from CUB dataset in Figure 13. For the novel sample in (a), the true label and its closest parents are coded in red and green respectively in the hierarchy. Since the true novel class is not present in the hierarchy, the ground truth label is its parent. For the known sample in (b), the true label (also the ground truth) is coded in green. For the representative sample of Yellow Bellied Flycatcher in Figure 13(a), we observe that our method is able to identify the closest parent Old World Flycatcher. However, other methods get confused about the sample with other classes. In particular, TD+LOO method assigns it to Acadian Flycatcher, a child class of the ground truth. We note that Acadian Flycatcher is a known leaf class in the hierarchy. The training method of TD+LOO leverages the samples from Acadian Flycatcher as a

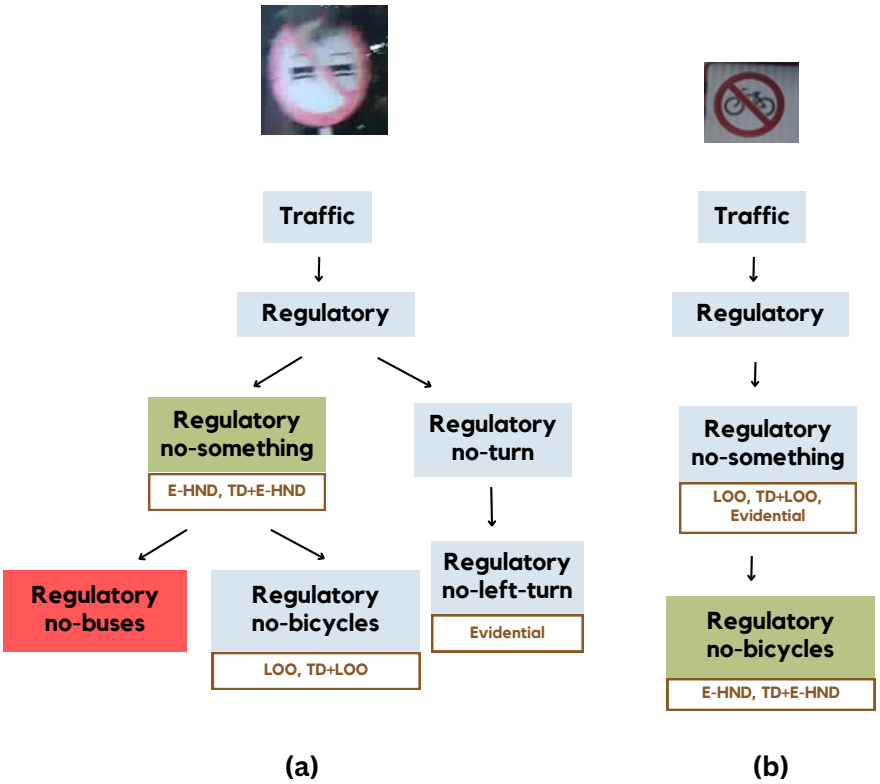

Figure 15. Qualitative study for representative test samples: (a) Prediction for novel sample from Regulatory no-buses (b) Prediction for known sample from Regulatory no-bicycles.

sample of novel Old World Flycatcher. As mentioned in the challenges of LOO training, this can create confusion as the model assigns high logit to both novel and known classes. As a result, the TD+LOO is not able to distinguish whether the sample belongs to Acadian Flycatcher or Novel Old World Flycatcher. We plot the distribution of logits allocated by TD+LOO for the novel samples of Yellow Bellied Flycatcher to classes: Old World Flycatcher and Acadian Flycatcher in Figure 14(a). We observe that both classes are assigned logits in the high region, making them indistinguishable. However, in our case of TD+E-HND in Figure 14(b), we see that Acadian Flycatcher has evidence in the lower region than Old World Flycatcher. Therefore, for the particular sample in Figure 13(a), our method is able to distinguish the closest parent class. Similarly, in Figure 13(b), we have the prediction for the known sample from Acadian Flycatcher class. LOO and TD+LOO predict the representative sample as Old World Flycatcher. This is due to confusion between logits of Old World Flycatcher and Acadian Flycatcher.

We also show the prediction of TD+E-HND with some of the baselines for the representative samples from Traffic dataset in Figure 15. A similar trend can be found in representative samples, as seen with the CUB dataset. Baseline like LOO, and TD+LOO mistakes the novel samples with known samples, and can predict a novel sample from Regulatory–no-buses sign as Regulatory–no-bicycles. Similarly, a known sample from Regulatory–no-bicycles is mistaken to be its parent class Regulatory–no-something.

D. Limitations

While the proposed work is generalizable towards all the domains given the construction of a hierarchy associated with the training classes, the relationship between training classes may occur in a different format, *e.g.,* a graph. It would be

interesting to explore novelty detection in the other forms of relationship that can occur between novel samples and training data.

E. Source Code

The source code can be accessed here: https://anonymous.4open.science/r/Evidential_hierarchical_novelty_detection-A3C3/README.md