Universiteit Leiden
The Netherlands

Open-world continual learning via knowledge transfer

Li, Y.

Chapter 5

Improving Open-world Continual Learning under the Constraints of Scarce Labeled Data

In the previous chapter, we introduced *Pro-KT*, a prompt-based framework for open-world continual learning that employs a structured prompt bank to facilitate effective knowledge transfer and robust open-set decision-making. Although *Pro-KT* achieves strong performance in scenarios with abundant labeled data, its dependence on extensive supervision restricts its applicability in real-world settings where labeled samples are scarce—a common challenge in dynamic, open-ended environments. Similarly, prior studies [8, 25, 30] face a critical limitation: their reliance on large-scale training data for each new task.

However, this requirement for abundant labeled samples often conflicts with the inherent nature of open and dynamic data, where new tasks (or classes) typically emerge with only limited supervision—or even just a few labeled examples [105, 106]. For example, on social media platforms, user-uploaded content (e.g., photos or posts) frequently contains novel objects or scenes without corresponding labeled data. Acquiring sufficient ground-truth annotations for every new task or class is often impractical, necessitating open-world continual learning models that can adapt to extreme data scarcity.

This limitation motivates the focus of this chapter, where we extend our investigation to a more challenging yet practical setting: **O**pen-world **F**ew-shot **C**ontinual **L**earning (OFCL). In this setting, the model must not only manage incremental tasks amid open-set unknowns but also operate under severe data scarcity, with both known and unknown classes represented by only a few labeled examples.

As illustrated in Figure 5.1, the scarcity of labeled data increases the two main problems in open-world continual learning, namely the misclassification of open-set samples as known classes and catastrophic forgetting due to inadequate knowledge repre-

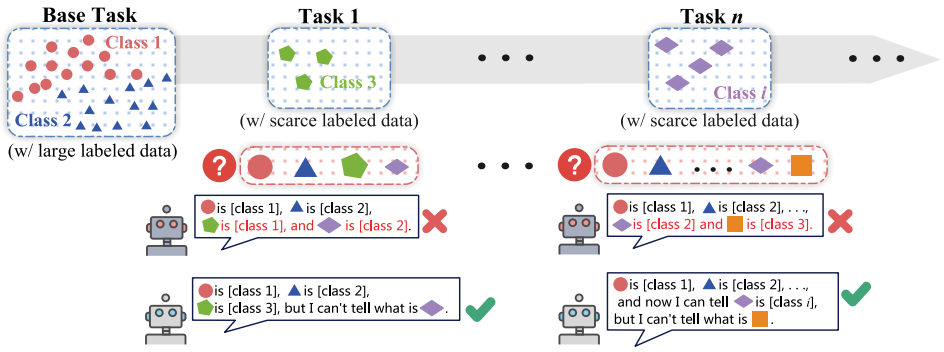

Figure 5.1: Open-world few-shot continual learning framework overview. The model incrementally learns new tasks (task 1, ..., task n, etc.) with few-shot class examples (class 3, ..., class i, etc.) while simultaneously detecting unknown samples during testing and overcoming severe forgetting and overfitting challenges inherent in data-scarce continual learning scenarios.

sentation. The limited training examples further impair the model's ability to learn precise decision boundaries between known and unknown classes, degrading performance in open-world scenarios. Additionally, data scarcity restricts the model's capacity to extract and generalize useful knowledge, hindering effective cross-task knowledge transfer. Consequently, the few-shot learning constraint introduces the following three challenges:

Challenge (1): Representation Collapse. With insufficient training samples, the model tends to learn shallow or overly narrow features that fail to generalize across tasks. This results in weak representations, impairing the distinction between seen and unseen classes and worsening both forgetting and misclassification.

Challenge (2): Open Detection Boundary Ambiguity. The lack of representative examples near decision boundaries makes it difficult for the model to accurately separate known and open-set classes. Consequently, open-set samples are more frequently misclassified into known categories, reducing reliability in open-world settings.

Challenge (3): Knowledge Transfer Degradation. In few-shot learning, the scarcity of task-specific data severely limits the amount of transferable knowledge. This diminishes the model's ability to leverage prior experience for new tasks, making continual adaptation less efficient and more prone to errors.

These challenges underscore the necessity for an open-world continual learning framework capable of operating under limited supervision while preserving strong generalization, robust boundary modeling, and efficient cross-task knowledge transfer. To address these requirements, we propose a **P**rompt-**E**nriched **A**daptive **K**nowledge

transfer (*PEAK*) model, a novel framework specifically designed for open-world continual learning in data-scarce scenarios. Our approach simultaneously enhances three key aspects: (1) knowledge representation quality, (2) decision boundary reliability, and (3) cross-task transferability under extreme label scarcity.

First, we introduce a learnable token injection mechanism that enriches instance-level representations. By implicitly capturing auxiliary semantics through augmented tokens, our *Instance-wise Token Augmentation* (ITA) mitigates representational sparsity while promoting feature alignment across tasks through token-level embedding synchronization.

Secondly, by addressing the few-shot "hubness problem", where certain data points become overly frequent nearest neighbors in high-dimensional spaces, thus biasing predictions [36], we develop a hyperspherical solution inspired by [37]. We propose *Margin-based Open Boundary* (MOB) to establish well-separated decision margins through learnable class centroids with adaptive radii, effectively distinguishing known from unknown classes.

Finally, our dynamic hyperspherical space enables incremental absorption of novel class knowledge, converting "unknown" samples to "knowns". Through continuous refinement of the semantic space with new tasks, we introduce *Adaptive Knowledge Space* (AKS) to enhance both generalization capability and long-term continuity in open-world environments.

Extensive experiments across multiple benchmarks demonstrate that our *PEAK* consistently outperforms state-of-the-art approaches, establishing its effectiveness for few-shot open-world continual learning scenarios.

Our key contributions in this chapter are summarized as follows:

- **Problem Formulation**: We formalize the challenging yet practical setting of open-world continual learning with scarce labeled data, addressing the orthogonal challenges of open-set detection and knowledge transfer under extreme data limitations.
- **Technical Innovations**: We introduce Instance-wise Token Augmentation for enhancing embedding semantics and mitigating forgetting through learnable token injections. We propose a Margin-based Open Boundary to enable robust open-set detection via hyperspherical class representations, and use a novel Adaptive Knowledge Space to dynamically incorporate and update knowledge from both known and unknown classes.
- **Comprehensive Evaluation**: Our framework demonstrates superior performance against state-of-the-art open-world continual learning methods, few-shot incremental learners, and open-set detection baselines. Ablation studies further validate the robustness of each component. The source code is publicly available at `https://github.com/liyj1201/OFCL`.

This chapter is based on our publication [35]:

- **Li, Y.**, Wang, X., Yang, X., Bonsangue, M., Zhang, J., Li, T.: Improving Open-World Continual Learning under the Constraints of Scarce Labeled Data. In: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining. KDD '25 (2025).

5.1 OFCL Problem Definition

In this section, we introduce open-world few-shot continual learning, a realistic but challenging problem that aims to incrementally learn under the open-world assumption, where unseen or open samples may appear during the test phase, and to generalize effectively to new tasks with scarce labeled training data. It is important to note that we assume only the base task provides ample training samples (as shown in Figure 5.1), while each subsequent task ($t > 0$) involves only limited labeled data.

Few-shot learning inherently suffers from the scarcity of labeled data, often leading to under-trained models and sparse knowledge representations. To address this challenge, our approach integrates prompt-based learning with a novel Instance-wise Token Augmentation mechanism, enabling the model to adaptively enrich feature representations even under limited annotation. In addition, the Margin-based Open Boundary module constructs compact and discriminative decision boundaries by leveraging hyperspherical embeddings, thereby enhancing the model's robustness in open-set detection. Complementing this, the Adaptive Knowledge Space introduces a dynamic structure for knowledge accumulation, transfer, and replay, allowing the model to incrementally refine its understanding of both known and unknown classes over time.

Together, these components enable our proposed method to overcome key limitations of previous open-world continual learning approaches, such as the reliance on fixed logits-based thresholds (e.g., *Pro-KT* [30]), and render it well-suited for real-world deployment under limited supervision.

Definition 8 ***OFCL Problem Definition.*** *Given a sequence of tasks* $\{1, ..., t, ...\}$*, each training set is organized in an* N*-way* K*-shot format:* $D^t_{tr} = \{(x^t, C) \mid x^t \in X^t_{tr}(C),\ C \in CLS^t_{tr}\}$*, where* $X^t_{tr}(C)$ *denotes the set of* K *training samples for class* C*, and* CLS^t_{tr} *represents the set of* N *classes in task* t*. Similarly, the test set for task* t *is defined as* $D^t_{te} = \{(x^t, C) \mid x^t \in X^t_{te}(C),\ C \in CLS^t_{te}\}$*, where* CLS^t_{te} *is the set of classes in task* t*.*

Under the open-world assumption, we do not assume any predefined relationships between classes in the test and training sets. Specifically, some test samples may be unknown or open, i.e., belonging to the set $(CLS^t_{te} \setminus CLS^t_{tr})$*. During evaluation, we assess performance on the cumulative test set* $D^1_{te} \cup ... \cup D^t_{te}$ *in a continual manner.*

Accordingly, the objective of open-world few-shot continual learning is to learn a unified classifier f_θ, parameterized by θ, that can (1) detect unknown/open samples, (2) classify known samples accurately, and (3) incrementally transform unknowns into knowns over time.

To support a clear understanding of the above and subsequent formulations and model components, we summarize the key notations used throughout this chapter in Table 5.1.

Table 5.1: Notations and explanations.

Notation	Explanation
t	The session t
CLS_{tr}^t	The set of training classes from t
D_{tr}^t, D_{te}^t	Training and test sets of t
N	# of classes in each training set
K	# of training samples of each class
C	Class C
$\mathbf{x}^+, \mathbf{x}^-$	Positive and negative sets of class C
α, β	Scaling factors for $\mathbf{x}^+$and $\mathbf{x}^-$
$q_\sigma(\cdot)$	A quantile function
$\mathbf{p}^t$	The token set of t
$\mathbf{p}*$	The set of matched tokens
L_P	The length of each token
D_e	The embedding dimension
$\mathbf{h}'^t = \mathbf{h}^t \oplus \mathbf{p}^*$	The augmented embedding
f_{pr}	The pre-trained backbone
f_θ	The trainable classifier
r_C	The learnable radius of class C
$\mathbf{c}_C$	The learnable centroid of class C
m	The margin
$(CLS_{tr}^t, \mathbf{C}^t, \mathbf{R}^t)$	Hyperspheres learned from D_{tr}^t
$(CLS_{open}^t, \overline{\mathbf{C}^t}, \overline{\mathbf{R}^t})$	Hyperspheres learned from unknowns
λ	The trade-off parameter

5.2 Methodology

The challenges outlined above, namely, representation collapse, boundary ambiguity, and knowledge transfer degradation, make clear the need for an approach to Open-world Few-shot Continual Learning that can simultaneously address data scarcity, openness, and catastrophic forgetting. As we have discussed, existing open-world continual learning methods assume enough labeled data, and few-shot learning techniques often disregard continual adaptation. In this section, we close this gap by introducing a unified framework specifically designed for open-world few-shot continual learning. Our solution is schematically presented in Figure 5.2 and consists

of three components discussed in more details below: (1) Instance-wise Token Augmentation to deal with data sparsity through dynamic prompt-based enrichment, (2) a Margin-based Open Boundary mechanism to refine decision frontiers in few-shot conditions, and (3) an Adaptive Knowledge Space to organize and transfer knowledge across tasks.

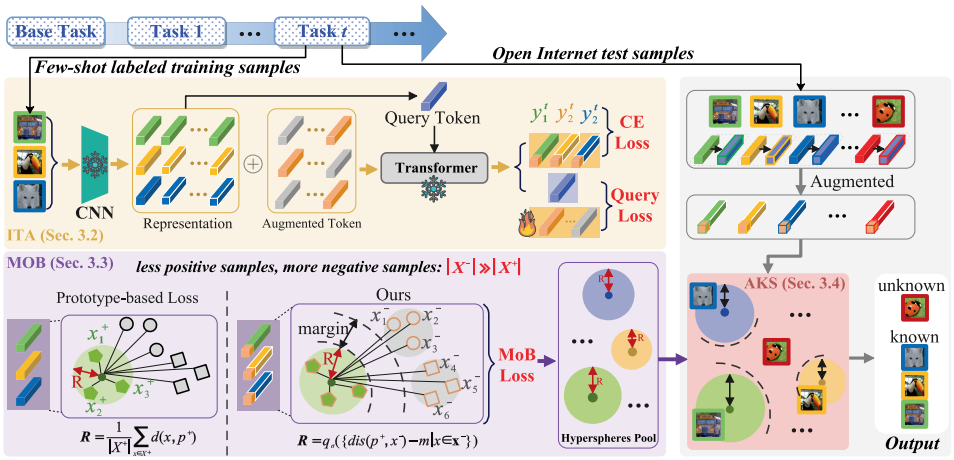

Figure 5.2: The overall *PEAK* model consists of three key components: (1) ITA (Yellow): enhancing the samples by matching them with appropriate additional tokens; (2) MOB (Purple): constructing compact decision boundaries of knowns by margin-based loss function for open detection; (3) AKS (Red): incorporating knowledge learned from both knowns and unknowns, and facilitating the transition from unknowns to knowns.

5.2.1 Instance-wise Token Augmentation

Inspired by the strong few-shot learning ability of prompt tuning methods [32, 83, 107], we propose Instance-wise Token Augmentation, a knowledge augmentation paradigm for *PEAK* designed to address the challenges of sparse supervision in open-world few-shot continual learning. This approach builds directly on the foundation presented in Chapter 4, where we demonstrated how prompt-based methods such as *Pro-KT* can enrich sample representations by injecting additional knowledge. This enriched representation plays a critical role in few-shot settings. Here, the scarcity of labeled data amplifies the risk of knowledge sparsity, undermining both representation quality and transfer efficiency. To alleviate this problem, we extend the prompt-based framework of *Pro-KT* by introducing Instance-wise Token Augmentation, a dynamic mechanism for instance-level knowledge augmentation, enabling more robust representation learning, knowledge accumulation, and cross-task transfer under limited supervision.

Following the architectural intuition of *Pro-KT*, the instance-wise token augmentation module operates through three key steps: token initialization, instance-wise token matching, and embedding augmentation.

Token Initialization. For a task t with training set D_{tr}^t, we initialize a set of learnable tokens $\mathbf{p}^t$ as:

$$\mathbf{p}^t = (p_1^t, ..., p_i^t, ..., p_l^t), \text{with } p_i^t \in \mathbb{R}^{L_p \times D_e}, , \tag{5.1}$$

where l denotes the number of tokens, L_P their length, and D_e the embedding dimension. During training, we maintain a frequency set $\nu^t = (\nu_1, ..., \nu_k, ...)$ to track how often each token p_k^t is selected for augmentation.

Instance-wise Token Matching. To identify the most relevant tokens for a given sample, we employ a top-K selection mechanism based on cosine similarity between the sample's embedding and token-associated keys:

$$(\mathbf{k}^*, \mathbf{p}^*) = \underset{K}{\text{argmin}} \sum_{i=1}^{l} sim(\mathbf{h}^t, k_i^t) \cdot \nu_i, \text{ with } \mathbf{h}^t = \mathbf{Q}(x^t), \tag{5.2}$$

where k_i^t is the key for token p_i^t, $sim(\cdot, \cdot)$ measures cosine similarity, and $\mathbf{Q}$ is a query function projecting the input x^t into the keys space.

Embedding Augmentation. The selected tokens $\mathbf{p}^*$ are concatenated with the original embedding $\mathbf{h}^t$ to produce an augmented representation:

$$\mathbf{h}'^t = \mathbf{h}^t \oplus \mathbf{p}^*, \tag{5.3}$$

where $\oplus$ denotes dimension-wise concatenation.

The training objective combines classification loss with a token alignment term to ensure selected keys remain semantically relevant:

$$\mathcal{L}_{aug} = \mathcal{L}(f_\theta(f_{pr}(\mathbf{h}'^t), \mathbf{y}^t) + \lambda \cdot \sum_{i=1}^{l} sim(\mathbf{h}^t, k_i^t), \tag{5.4}$$

where fpr is the pre-trained backbone, $\mathbf{y}^t$ are task labels, and λ is a parameter to balance classification accuracy and token-feature alignment. Here, the first addend denotes the classification loss, and the second is a surrogate loss for pulling selected keys closer to corresponding query features.

Unlike prior prompt-based continual learning methods, the instance-wise token augmentation module stores all task-specific tokens in a unified token bank $(\mathbf{K}, \mathbf{P})$, treating them as reusable knowledge. During inference, the model retrieves and applies the most relevant tokens from this bank for each test sample to mitigate forgetting by preserving prior task knowledge, reduce overfitting through dynamic, instance-specific augmentation, and enhance generalization by leveraging cross-task token sharing as demonstrated empirically in Section 5.3.

5.2.2 Margin-based Open Boundary

Our approach introduces augmented training samples to facilitate the formation of precise, compact decision boundaries for open sample detection, a capability in *PEAK* where traditional methods often fail. Unlike existing approaches that simply combine continual learning with open-set recognition, we propose a Margin-based Open Boundary strategy implemented in hyperspherical space to construct compact boundaries from known samples or training classes. Our solution addresses the two challenges in open-world few-shot continual learning: the hubness problem [37], where certain points disproportionately dominate nearest-neighbor lists, leading to misclassifications, and the need for robust adaptive representations in dynamic, low-sample environments.

For each class $C \in CLS_{tr}^{t}$, we organize the training samples into two distinct sets following contrastive learning principles where training samples belonging to C serve as positive samples, while all other training samples are treated as negative instances (illustrated in Figure 4.3): the positive set $\mathbf{x}^{+} = \{x^{t} | (x^{t}, C) \in D_{tr}^{t}\}$ contains all samples belonging to class C at task t, and the negative set $\mathbf{x}^{-} = \{x^{t} | (x^{t}, C') \in D_{tr}^{t}, C' \neq C\}$ comprises all other samples from the same task. This division enables our margin-based open boundary framework to learn discriminative hyperspherical decision boundaries through the following loss function:

$$\begin{aligned} \mathcal{L}_{margin} = \frac{1}{N} \cdot \sum_{C \in CLS_{tr}} & (\lambda \cdot r_C^2 \\ & + \frac{1}{\alpha} \cdot log[1 + \sum_{x \in \mathbf{x}^{+}} e^{\alpha \cdot (dis(\mathbf{c}_C, f_{pr}(x)) - r_C)}] \\ & + \frac{1}{\beta} \cdot log[1 + \sum_{x \in \mathbf{x}^{-}} e^{-\beta \cdot (dis(\mathbf{c}_C, f_{pr}(x)) - (r_C + m))}]), \end{aligned} \tag{5.5}$$

where N is the number of classes in the training set, m is the margin enforcing separation between classes, r_C is the hypersphere radius for class C around centroid $\mathbf{c}_C$, $dis(\cdot, \cdot)$ is the distance metric between samples, λ is a trade-off parameter for radius regularization, and α, β are the scaling factors for positive and negative set contributions, respectively. Note that the pre-trained backbone f_{pr} processes raw samples directly here, unlike in Equation 5.4 where it operates on embeddings. This design choice preserves the original feature relationships for boundary learning.

Our margin-based open boundary framework operates through three key components: (1) *known classes centroids* for establishing hypersphere centers for each known class, (2) *margin and radius learning* for optimizing class boundaries through contrastive principles, and (3) *open boundary detection* for implementing robust unknown identification.

Known Classes Centroids. Hyperspherical embeddings offer clear advantages for open-set scenarios by naturally distributing data more uniformly and avoiding

central clustering. Following approaches like noHub [37], our method maintains both class structure preservation and uniform feature distribution, a balance that allows continuous adaptation in dynamic, low-sample environments.

We construct this embedding space by representing each class as a hypersphere with a learned centroid. For any class C from task t, its centroid $\mathbf{c}C$ is computed as:

$$\mathbf{c}_C = \frac{1}{|\mathbf{x}^+|} \sum x^t \in \mathbf{x}^+ f_{pr}(x^t). \tag{5.6}$$

The model incrementally stores these centroids throughout the continual learning process $1, ..., t, ...$ Each centroid serves as the center for its corresponding class hypersphere, with an optimal radius to be determined in the next step to establish well-separated decision boundaries between known and unknown samples.

Margin and Radius Learning. In our few-shot learning scenario, a given class C typically has significantly fewer positive samples $\mathbf{x}^+$ than negative samples $\mathbf{x}^-$. To maintain a clear separation between classes while accounting for this imbalance, we enforce two key geometric constraints. First, we require that all positive samples must lie within the class hypersphere:

$$dis(\mathbf{c}_C, f_{pr}(x)) < r_C, \tag{5.7}$$

where r_C is learned through optimization using Equation 5.5. The second constraint imposes that all negative samples reside outside an expanded boundary:

$$dis(\mathbf{c}_C, f_{pr}(x)) > r_C + m, \tag{5.8}$$

with margin m ensuring sufficient separation.

The optimal radius r_C is determined through:

$$r_C = q_\sigma(\{dis(\mathbf{c}_C, f_{pr}(x)) - m \mid x \in \mathbf{x}^-\}), \tag{5.9}$$

where $q_\sigma(\cdot)$ is a quantile function controlling boundary tightness via parameter σ governing deviations. For each task t, we maintain a complete hypersphere representation:

$$(CLS^t_{tr}, \mathbf{C}^t, \mathbf{R}^t) = \{(C^t_1, \mathbf{c}_{C^t_1}, r_{C^t_1}), ..., (C^t_N, \mathbf{c}_{C^t_N}, r_{C^t_N})\}, \tag{5.10}$$

where N is the number of classes in CLS^t_{tr}. Throughout continual learning $\{1, ..., t, ...\}$, the model aggregates all learned hyperspheres together:

$$(CLS_{tr}, \mathbf{C}, \mathbf{R}) = \{(CLS^1_{tr}, \mathbf{C}^1, \mathbf{R}^1), ..., (CLS^t_{tr}, \mathbf{C}^t, \mathbf{R}^t), ...\}. \tag{5.11}$$

Open Boundary Detection. During testing, our open-set detection procedure evaluates each test sample x against all learned hyperspheres through hypersphere projection and membership verification. The first projects x into the hyperspherical

embedding space containing all learned class representations, while the latter checks if x resides within any class hypersphere by evaluating if there is a class C such that x falls within its hypersphere and thus classifies x as known in class C, and otherwise identifies x as unknown.

More precisely, the open detection process at task t consists of three steps:

Step 1: Identify the nearest centroid $\mathbf{c}_C^*$ to x using $\mathbf{c}_C^* = \underset{\mathbf{c}_C}{\text{argmin}} dis(\mathbf{c}_C, f_{pr}(x))$

Step 2: Compute the distance metric $dis(\mathbf{c}_C^*, f_{pr}(x))$ between the centroid $\mathbf{c}_C^*$ and the test sample x;

Step 3: Decide if x can be directly classified into a known class C if $d^* \leq r^*$, based on the output logits score, and otherwise, classify x as `[unknown]`.

The margin-based open boundary framework treats hypersphere centers as dynamically learnable parameters that evolve through continual learning, updating their positions based on deep feature representations. This approach extends conventional metric learning by combining margin-based optimization with hyperspherical geometry to address hubness while maintaining discriminative power. The centers serve as adaptive prototypes that progressively refine within our Adaptive Knowledge Space, enabling stable knowledge accumulation while accommodating new classes through geometric constraints. This dynamic formulation naturally handles distribution shifts and maintains robust decision boundaries throughout the learning process.

5.2.3 Adaptive Knowledge Space

Considering $(CLS_{tr}, \mathbf{C}, \mathbf{R})$ as the knowledge learned from knowns, we present an adaptive knowledge space, designed not only to store and transfer knowledge about unknowns acquired from previously learned tasks, but also to facilitate updating unknown knowledge into known knowledge for *PEAK*.

Unknown Knowledge Representation. After identifying unknowns, we then employ clustering on the identified unknown samples as follows: Given a detected unknown sample x let $\Gamma_\epsilon(x)$ represent the ϵ-neighborhood of x, $MinPts$ denotes the minimum number of objects in $\Gamma_\epsilon(x)$, and $\rho(x) = |\Gamma_\epsilon(x)|$ be its density value. We define the boolean function $T(x) = 1$ if $\rho(x_j) \geq MinPts$ and $T(x) = 0$ otherwise. If $T(x) = 1$, the unknown sample x is clustered with density-reachable points, and the entire space is partitioned into M groups $\{G_1, G_2, ..., G_M\}$. Otherwise, x is treated as noise. Each group is then assigned a group centroid, which is determined using Equation 5.6. The corresponding radius is obtained by Equation 5.9.

Since there are no *ground-truth labels* for unknowns, we generate a set of pseudo-labels and assign them to each group. Similar to hyperspheres learned from known

samples, we have :

$$(CLS^t_{open}, \overline{\mathbf{C}^t}, \overline{\mathbf{R}^t}) = \{(\overline{C^t_1}, \mathbf{c}_{\overline{C^t_1}}, r_{\overline{C^t_1}}), ..., (\overline{C^t_M}, \mathbf{c}_{\overline{C^t_M}}, r_{\overline{C^t_M}})\}. \tag{5.12}$$

Updating Unknowns to Knowns. At task t the integrated adaptive knowledge space consists of all the learned hyperspheres during training $(CLS_{tr}, \mathbf{C}, \mathbf{R})$ together with all the new $(CLS^t_{open}, \overline{\mathbf{C}^t}, \overline{\mathbf{R}^t})$ associated with the unknowns at all task until t. Because of this incremental growth, if previously encountered unknowns reappear, they are classified with corresponding pseudo-labels. However, if a new hypersphere learned during a subsequent task's training overlaps with an existing hypersphere with a pseudo-label, then we integrate the corresponding pseudo-category into the newly trained categories, thus facilitating a transition from unknowns to knowns.

5.2.4 Overall Objective and Process

At any task t, *PEAK* first obtains L additional tokens via $\mathcal{L}_{aug}$ and stores all the newly learned additional tokens together with those previously learned. Subsequently, each sample is augmented by additional knowledge, and an adaptive knowledge space containing knowledge learned from knowns and unknowns is constructed. Hence, the overall objective is:

$$\min_{(\mathbf{C}^t, \mathbf{R}^t), \theta, (\mathbf{k}^t, \mathbf{p}^t)} \gamma \cdot \mathcal{L}_{margin} + (1 - \gamma) \cdot \mathcal{L}_{aug}, \tag{5.13}$$

where the γ is a balance between $\mathcal{L}_{margin}$ and $\mathcal{L}_{agu}$.

To better illustrate the proposed *PEAK* framework, we detail the training process in algorithm 3. First, the token bank and classifier are initialized. Then, for each time step t, multiple training epochs are conducted, where each sample $(\mathbf{x}^t, \mathbf{y}^t)$ undergoes feature mapping, similarity computation with stored keys, augmentation via token selection, and enhancement through the backbone network. Class-wise centroids, positive and negative sets, margins, and radii are computed to derive the overall loss, which is minimized via backpropagation. Finally, the AKS is updated with hyperspheres representing both known and unknown knowledge, enabling continual adaptation to evolving data distributions.

Accordingly, after learning task t, the inference procedure unfolds through four sequential stages:

Step 1: The model integrates newly acquired task knowledge—specifically, the information derived from labeled samples—into its existing knowledge space. In this process, previously unknown instances are transitioned into the known category space once their true labels have been observed.

Algorithm 3: : Training Process of the *PEAK* framework

Input: the training set $D_{tr}^t = \{(\mathbf{X}^t, \mathbf{Y}^t)\}$, the set of training classes CLS_{tr}^t, the token bank $(\mathbf{K}, \mathbf{P}) = \{(\mathbf{k}^t, \mathbf{p}^t)\}_{t=1}^T$, a trainable classifier f_θ, a backbone f_{pr} and the hyperparameter balance γ.
Initialize: $(\mathbf{K}, \mathbf{P})$, f_θ.

for t = 1, 2, ..., T **do**
  for each epoch **do**
    for $(\mathbf{x}^t, \mathbf{y}^t)$ in each training set D_t^{tr} with class CLS_{tr}^t **do**
      Map $\mathbf{x}^n$ to $\mathbf{h}^t$;
      Get the similarity of h^t and each key of $(\mathbf{k}^t, \mathbf{p}^t)$;
      Select augmentation tokens with $(\mathbf{k}^*, \mathbf{p}^*)$ Equation 5.2;
      Enhance $\mathbf{h}^t$ to $\mathbf{h}'^t$ with $\mathbf{p}^*$ via Equation 5.3;
      Calculate the augmented feature by $f_{pr}(\mathbf{h}'^t)$;
      Calculate $\mathbf{c}_C$ for each class C in CLS_{tr}^t via Equation 5.6;
      Obtain positive and negative set of class $C : \mathbf{x}^+, \mathbf{x}^-$;
      Obtain margin and radius via Equation 5.9
      Calculate the overall loss via Equation 5.13;
    end for
    Update f_θ, $(\mathbf{K}, \mathbf{P})$, the radius r_C and margin m by backpropagation;
  end for
  Update the overall AKS with **1)** Hyperspheres learned from D_{tr}^t: $(CLS_{tr}^t, \mathbf{C}^t, \mathbf{R}^t)$ and **2)** the Hyperspheres learned from unknowns: $(CLS_{open}^t, \overline{\mathbf{C}^t}, \overline{\mathbf{R}^t})$.
end for

Step 2: During the inference phase, the model performs open-set recognition by actively identifying and isolating open (i.e., previously unseen) samples, thereby mitigating the risk of erroneous assignment to existing known classes.

Step 3: For samples confidently recognized as belonging to known classes, the model conducts accurate classification while simultaneously preserving knowledge acquired from previous tasks, thus alleviating catastrophic forgetting.

Step 4: The knowledge space is continuously and adaptively updated as the model incrementally incorporates novel information extracted from open samples, thereby refining its decision boundaries and expanding its representational capacity.

In summary, the proposed *PEAK* framework not only enhances the discriminative representations of known classes but also enforces tighter and more robust decision boundaries to improve open-set detection performance. This integrated approach reflects a principled extension of existing continual learning paradigms, effectively addressing the dual challenges of knowledge retention and open-world generalization intrinsic to open-world continual learning scenarios.

5.3 Evaluation

Having established the foundations of our *PEAK* framework, we now turn to empirical validation to assess its effectiveness in realistic open-world continual learning scenarios. Through experiments on standard benchmarks, we evaluate the framework's ability to detect unknown samples while maintaining stability on known classification under few-shot conditions, as well as the practical efficiency of our adaptive knowledge mechanisms. These evaluations not only quantify *PEAK*'s advantages over existing approaches but also validate our design choices through ablation studies and sensitivity analyses.

5.3.1 Experimental Setup

Datasets. Our evaluation follows the few-shot continual learning benchmark configuration from TOPIC [108], employing two standard datasets: (1) the *CUB200 dataset* [109] (11,788 images, 200 species) partitioned into 100 base classes for task 0 followed by 10 incremental tasks (10 classes each, 5 samples per class), and (2) the *MiniImageNet dataset* [110] (60,000 images, 100 classes) divided into 60 base classes plus 8 incremental tasks (5 classes each, 5 samples per class).

Implementation Protocol. Consistent with few-shot continual learning standards, we randomly shuffle session orders (10 for CUB200, 8 for MiniImageNet) and treat subsequent session training classes as unknowns during the current session testing. All reported results average three random shuffles. Experiments are conducted on NVIDIA GeForce RTX 3090 GPUs using the PyTorch library. Input images are resized to 224×224, normalized to [0,1] range, with models trained via Adam optimizer (batch size=25, learning rate=0.03). Baseline hyperparameters follow their original optimal configurations. Our implementation includes fully reproducible PyTorch code for both *PEAK* and comparative baselines in open-world continual learning scenarios.

Baselines. To empirically validate *PEAK*'s effectiveness, we conduct comparisons against 23 state-of-the-art baselines across two standard datasets, establishing new performance benchmarks. Our evaluation includes both *Unknowns Detection* (8 representative for OOD and OSR methods) and *Knowns Classification* (15 leading few-shot continual learning approaches), providing complete coverage of the problem dimensions.

i) Unknowns Detection Baselines:

- *MSP* [80] uses the maximum softmax probability of classifier predictions as the OOD score.
- *KL* [111] proposes a scalable framework for OOD detection, resulting in large-scale datasets and improved evaluation protocols to enhance robustness and generalization.

- *SSD* [112] is a unified self-supervised method that integrates diverse pretext tasks to improve representations.
- *MaxLogits* [101] enhances OOD detection performance by utilizing the maximum logit score.
- *Energy* [63] assigns lower energies to ID data and higher energies to OOD data to create an energy gap for OOD detection.
- *ViM* [113] is an OOD detection method with matching virtual logits, improving the separation between in-distribution and out-of-distribution samples through a robust scoring mechanism.
- *KNN-based OOD* [114] presents a deep nearest neighbor-based method for OOD detection, utilizing the local structure of feature spaces to identify anomalies.
- *NNGuide* [115] is a nearest neighbor guidance strategy for OOD detection, enhancing the discriminative power of feature representations by leveraging proximity-based metrics.

ii) Knowns Classification Baselines:

- *iCaRL* [42] is an incremental classifier and representation learning method designed for FSCIL.
- *TOPIC* [108] is a model tailored for few-shot class-incremental learning (FSCIL), emphasizing effective adaptation to new classes with limited samples.
- *CEC* [116] accommodates new classes while preserving previous knowledge for incremental learning.
- *Meta-FSCIL* [117] is a meta-learning-based approach, leveraging meta-learning principles to facilitate fast adaptation to new sessions.
- *ALICE* [118] employs an attention-based approach to efficiently learn new classes with limited training samples.
- *FeSSSS* [105] addresses the challenge of FSCIL by utilizing feature space segmentation for improved adaptation.
- *Pro-KT* [30] Our *Pro-KT* framework (introduced in the previous chapter) that integrates task-generic and task-specific prompt libraries to encode and transfer knowledge, while leveraging task-aware open-set boundaries to effectively identify unknowns.
- *LIMIT* [119] focuses on efficient knowledge transfer and adaptation to new classes while mitigating the impact on the existing knowledge.
- *NC-FSCIL* [120] is a neural clustering-based model, utilizing clustering techniques to enhance adaptation.

- *MCNet* [121] implements memory-augmented neural networks for effective knowledge retention.
- *SoftNet* [106] utilizes the soft parameter sharing to facilitate efficient learning of new classes.
- *CoSR* [122] is an FSCIL model that employs a collaborative self-regularization mechanism for effective adaptation while preserving previous knowledge.
- *M-FSCIL* [123] proposes a memory-based label-text tuning method that leverages textual label embeddings and a memory bank to enhance generalization in few-shot class-incremental learning.
- *FSIL-GAN* [124] introduces a semantics-driven generative replay strategy, where a text-to-image model is used to synthesize class-representative samples for mitigating forgetting in few-shot incremental learning.
- *CPE-CLIP* [125] presents a multimodal parameter-efficient approach that integrates visual and textual information while freezing most model weights, enabling effective few-shot class-incremental learning with minimal overhead.

Metrics. We employ three complementary metrics: (1) For unknown detection we use the average area under the ROC (AUC_N) and false positive rate (FPR_N) across all N tasks [74]; (2) for known classification, the averaged accuracy ACC_N over N new tasks; and (3) for catastrophic forgetting, the performance dropping rate $PD = ACC_0 - ACC_N$, where ACC_0 denotes base task accuracy [30].

5.3.2 Main Results

Unknown Detection Performance. As shown in Table 5.2, conventional baselines struggle significantly in the open-world few-shot continual learning setting, revealing their limitations for real-world deployment. Our proposed *PEAK* framework demonstrates consistent superiority across both datasets, achieving state-of-the-art performance with 69.72% AUC_N and 69.96% FPR_N on CUB200 (10-way 5-shot), along with 76.20% AUC_N and 71.30% FPR_N on MiniImageNet (5-way 5-shot). These results represent substantial improvements over all baseline categories: confidence-based (MSP, MaxLogits), energy-based (Energy, ViM), distance-based (KNN-OOD), and guidance-based (NNGuide) approaches.

PEAK's advantages stem from the integrations of three novel components: (1) Dynamic modeling of evolving knowledge spaces maintains discriminative boundaries for both closed-set classification and open-set detection; (2) Adaptive knowledge space updates enable incremental incorporation of novel open samples while preserving stability; (3) Joint optimization strategy provides principled uncertainty handling that reduces both false positives and catastrophic forgetting. The framework's particularly strong FPR_N performance (critical for safety-sensitive applications) demonstrates the ability to avoid misclassifying unknown samples, while its

Table 5.2: Results (%) regarding unknown detection. We report the results over 10 tasks for CUB200 (10-way 5-shot) and 8 tasks for MiniImageNet (5-way 5-shot).

Dataset	Methods	$AUC_N \uparrow$	$FPR_N \downarrow$
CUB200 (10-way 5-shot)	MSP	61.34	92.95
	KL	60.81	93.10
	SSD	37.70	99.52
	MaxLogits	<u>61.31</u>	<u>92.43</u>
	Energy	60.81	93.10
	ViM	54.16	97.91
	KNN-based OOD	37.20	99.16
	NNGuide	50.26	96.97
	PEAK	**69.72**	**69.96**
MiniImageNet (5-way 5-shot)	MSP	49.78	94.91
	KL	46.61	95.10
	SSD	50.64	95.18
	MaxLogits	46.78	94.65
	Energy	46.61	95.10
	ViM	<u>59.56</u>	<u>89.87</u>
	KNN-based OOD	52.88	93.78
	NNGuide	52.47	90.95
	PEAK	**76.20**	**71.30**

high AUC_N reflects a robust balance between known-class recognition and unknown rejection during continual expansion.

Overall, these results validate *PEAK* as a good solution to the fundamental challenge of maintaining robust open-set detection in continually evolving learning systems, addressing an important bottleneck in real-world deployment scenarios.

Results on Known Classification. Next, we evaluate our model's ability to acquire novel classes while preserving prior knowledge under the challenging few-shot class-incremental learning setting. Tables Table 5.3 and Table 5.4 present the averaged classification accuracies ACC_N by different methods across incremental tasks on CUB200 and MiniImageNet datasets, respectively.

PEAK demonstrates resilience against limited data, class imbalance, and catastrophic forgetting (the three main challenges of few-shot class-incremental learning), showing state-of-the-art performance on both benchmarks. The *PEAK* framework's final accuracy and low performance degradation (PD) validate the effectiveness of our instance-wise token augmentation and adaptive knowledge space mechanism, which jointly contribute to alleviating catastrophic forgetting while preventing overfitting under the few-shot context.

On the *MiniImageNet* benchmark, *PEAK* achieves a strong final accuracy of 87.12% with a performance degradation (PD) of 10.88, outperforming both *Pro-KT* (PD = 28.86) and M-FSCIL (which, however, exhibits a better PD of 8.08). Notably,

Table 5.3: Averaged classification accuracy (%) on CUB200 (10-way 5-shot).

Methods	ACC_N in each task (%) ↑											PD↓
	0	1	2	3	4	5	6	7	8	9	10	
iCaRL	68.68	52.65	48.61	44.16	36.62	29.52	27.83	26.26	24.01	23.89	21.16	47.52
TOPIC	68.68	62.49	54.81	49.99	45.25	41.40	38.35	35.36	32.22	28.31	26.28	42.40
CEC	75.85	71.94	68.50	63.50	62.43	58.27	57.73	55.81	54.83	53.52	52.28	23.57
Meta-FC	75.90	72.41	68.78	64.78	62.96	59.99	58.30	56.85	54.78	53.82	52.64	23.26
ALICE	77.40	72.70	70.60	67.20	65.90	63.40	62.90	61.90	60.50	60.60	60.10	17.30
FeSSSS	79.60	73.46	70.32	66.38	63.97	59.63	58.19	57.56	55.01	54.31	52.98	26.62
Pro-KT	**82.90**	74.15	72.82	62.46	59.69	54.88	51.56	48.57	47.32	46.86	47.68	35.22
LIMIT	76.32	74.18	72.68	69.19	68.79	65.64	63.57	62.69	61.47	60.44	58.45	17.87
NC-FSCIL	80.45	75.98	72.30	70.28	68.17	65.16	64.43	63.25	60.66	60.01	59.44	21.01
MCNet	77.57	73.96	70.47	65.81	66.16	63.81	62.09	61.82	60.41	60.09	59.08	18.49
SoftNet	78.07	74.58	71.37	67.54	65.37	62.60	61.07	59.37	57.53	57.21	56.75	21.32
CoSR	74.87	73.15	68.23	63.50	62.72	59.10	57.46	55.73	53.28	52.31	51.75	23.12
M-FSCIL	81.04	79.73	76.62	73.30	71.22	68.90	66.87	65.02	63.90	62.49	60.40	20.64
FSIL-GAN	81.27	78.03	75.61	73.72	70.49	68.19	66.58	65.63	63.21	60.92	59.43	21.84
CPE-CLIP	81.58	78.52	76.68	71.86	71.52	70.23	67.66	66.52	65.09	64.47	64.60	16.98
PEAK	76.67	**81.09**	**76.80**	**76.20**	**74.80**	**72.44**	**71.52**	**69.17**	**70.33**	**68.17**	**68.92**	**7.74**

although CPE-CLIP achieves the lowest PD (7.46), it does so at the cost of lower accuracy compared to M-FSCIL. This advantage is likely attributed to the overlap between CLIP's pretraining data and the MiniImageNet dataset. In contrast, *PEAK* achieves a superior base session accuracy ($\text{ACC}_N = 90.23\%$), further validating our model's stronger initialization and representation capabilities—even without relying on large-scale external pretraining as required by CLIP.

A similar trend is observed on the *CUB200* dataset, where *PEAK* achieves a final accuracy of 68.92% (PD = 7.74), surpassing all major baselines including Meta-FSCIL, FeSSSS, NC-FSCIL, and SoftNet. This consistent superiority across diverse datasets demonstrates the strong generalization capability of *PEAK* under varying domain complexities and task granularities.

Long-term Stability Analysis. Figure 5.3 shows that *PEAK* performance advantage over baseline methods increases consistently as more tasks are learned. This growing performance gap demonstrates two key strengths: superior long-term stability in maintaining knowledge across extended task sequences and exceptional adaptability to evolving knowledge boundaries, the capabilities needed for real-world continual few-shot learning scenarios. These findings further confirm the robustness and scalability of our framework in addressing the challenges of long-term incremental learning.

5.3.3 Ablation Studies

In this sub-section, we evaluate the three main components of our framework: (1) the token selection mechanism in ITA, (2) the complete instance-wise token augmentation module, and (3) the full adaptive knowledge space AKS. Performance is

Table 5.4: Averaged classification accuracy (%) on MiniImageNet (5-way 5-shot).

Methods	ACC_N in each task (%) ↑									PD ↓
	0	1	2	3	4	5	6	7	8	
iCaRL	61.31	46.32	42.94	37.63	30.49	24.00	20.89	18.80	17.21	44.10
TOPIC	61.31	50.09	45.17	41.16	37.48	35.52	32.19	29.46	24.42	36.89
CEC	72.00	66.83	62.97	59.43	56.70	53.73	51.19	49.24	47.63	24.37
Meta-FSCIL	72.04	67.94	63.77	60.29	57.58	55.16	52.9	50.79	49.19	24.41
ALICE	80.60	70.60	67.40	64.50	62.50	60.00	57.80	56.80	55.70	24.90
FeSSSS	81.50	77.04	72.92	69.56	67.27	64.34	62.07	60.55	58.87	22.63
Pro-KT	**98.59**	79.82	77.85	71.82	70.34	69.18	70.13	69.27	69.73	28.86
LIMIT	72.32	68.47	64.30	60.78	57.95	55.07	52.70	50.72	49.19	23.13
NC-FSCIL	84.02	76.80	72.00	67.83	66.35	64.04	61.46	59.54	58.31	25.71
MCNet	72.33	67.70	63.50	60.34	57.59	54.70	52.13	50.41	49.08	23.25
SoftNet	76.63	70.13	65.92	62.52	59.49	56.56	53.71	51.72	50.48	26.15
CoSR	71.92	66.91	62.71	59.59	56.63	53.78	51.01	49.23	47.93	23.99
M-FSCIL	93.45	**91.82**	87.09	**88.07**	**86.75**	87.15	85.68	84.80	85.37	8.08
FSIL-GAN	69.87	62.91	59.81	58.86	57.12	54.07	50.64	48.14	46.14	23.73
CPE-CLIP	90.23	89.56	87.42	86.80	86.51	85.08	83.43	83.38	82.77	**7.46**
PEAK	96.00	91.65	**89.17**	84.90	86.46	**87.18**	**87.57**	**85.71**	**87.12**	10.88

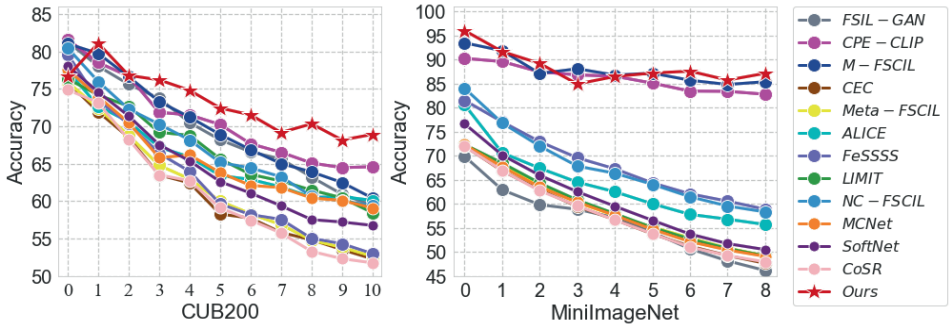

Figure 5.3: Visualizations of model performance on known classification.

measured by the average final accuracy ACC_N across all N incremental tasks, with results presented in Table 5.5.

Table 5.5 shows three key insights about *PEAK*'s architecture. First, the token selection mechanism in ITA proves essential for both classification and detection tasks, with its removal causing notable performance drops, particularly in maintaining knowledge boundaries and open-set detection. Second, the complete ITA module demonstrates the most substantial impact on classification accuracy, confirming its crucial role in preventing overfitting and catastrophic forgetting. Third, the AKS module emerges as fundamental for open-set capability, as its ablation completely disables unknown detection while also degrading known-class classification (Table 5.5), exposing the limitations of softmax-only approaches.

The *PEAK* framework's design, operating at input/output levels rather than an intermediate layer, ensures interesting backbone-agnostic properties. This is vali-

Table 5.5: Ablation studies.

Unknowns	CUB200		MiniImageNet	
Detection	AUC_N	FPR_N	AUC_N	FPR_N
w/o selection mechanism	66.68	69.96	75.73	80.54
w/o ITA	67.06	75.83	70.61	86.71
w/o AKS	-	-	-	-
PEAK	**69.72**	**69.85**	**76.20**	**71.30**
Knowns	CUB200		MiniImageNet	
Classification	ACC_N	PD	ACC_N	PD
w/o selection mechanism	54.52	12.48	61.34	29.66
w/o ITA	50.33	31.14	51.42	34.58
w/o AKS	55.24	28.42	63.02	34.78
PEAK	**63.80**	**8.20**	**65.24**	**25.56**

dated through ViT experiments: On CUB200, *PEAK* achieves 63.80% classification accuracy versus just 21.19% for fine-tuned ViT alone. This 3× performance gap demonstrates that raw representational power is insufficient without our structured knowledge transfer mechanisms for continual few-shot adaptation.

5.3.4 Additional Results

Time and Space Complexity. The Adaptive Knowledge Space shows linear scaling complexity $\mathcal{O}(n)$ as new tasks emerge, demonstrating efficient practical deployment with a compact memory footprint of only 0.048 MB for CUB200 and 0.024 MB for MiniImageNet after complete training. The framework achieves real-time performance with 0.001s token matching latency and under 0.01s classification decisions, while training requires just 0.4284s per epoch on CUB200 and 0.8279s on MiniImageNet, all within a constrained peak memory budget of 28.87 MB (i.e., 28869632B), confirming its suitability for continual learning scenarios.

Parameter Sensitivity Analysis. The framework's performance depends on several hyperparameters, including the separation margin m and the deviation constraint σ (governing boundary tightness in Equation 5.9), along with prompt configuration parameters L_P, K, and l for token length, token size K, and the number of tokens per task, respectively.

Experiments show that excessively large K or L_P values cause knowledge underfitting (Figure 5.4a), while the token quantity l maintains stable performance within $5 \leq l \leq 30$ (Figure 5.4b). The deviation constraint σ shows an interesting interaction with m: at $m = 0.5$, performance degrades with increasing σ, whereas larger margins benefit from higher σ values (Figure 5.5), highlighting the margin's pivotal role in open-detection robustness.

The loss weighting factor γ used for balancing classification, ITA similarity, and MOB distance terms in Equation 5.13, demonstrates consistent performance across

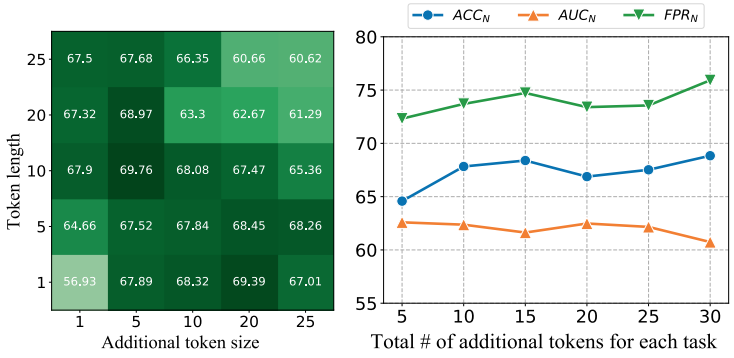

Figure 5.4: (a): ACC_N w.r.t token length L_P and additional token size K, given $l = 25$. (b): ACC_N w.r.t. l (i.e., the total number of additional tokens of each task) with $L_P = 5$ and $K = 5$ (take CUB200 for illustration).

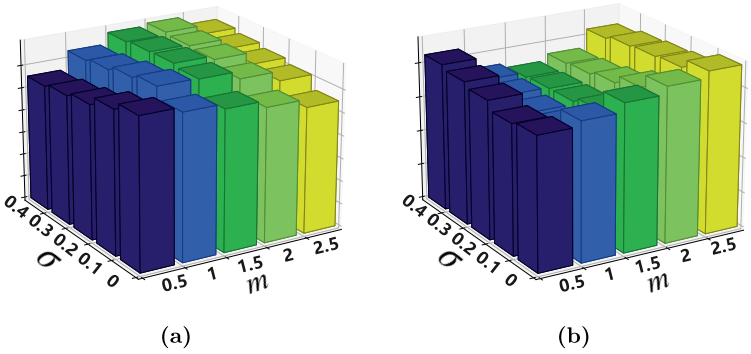

Figure 5.5: (a) and (b): AUC_N and FPR_N w.r.t constraint governing deviations σ and margin m (CUB200 dataset).

varying configurations (Table 5.6), demonstrating the *PEAK* framework's stability to loss-weight adjustments.

In summary, the *PEAK* framework demonstrates consistent performance across diverse realistic hyperparameter configurations, confirming its inherent robustness to parameter variations. This stability suggests practical advantages for real-world deployment, where exhaustive hyperparameter tuning may be impractical. While optimal performance requires tuning 4–5 key parameters, a reasonable engineering trade-off, our open-sourced implementation achieves state-of-the-art results even with commonly used default settings. These characteristics position the proposed *PEAK* as both a robust and readily adoptable solution for open-world continual learning scenarios.

Table 5.6: Performance under different γ configurations on the CUB200 dataset.

γ	ACC_N	AUC_N	FPR_N
1:00:01	57.18	62.12	73.81
1:0.1:0.9	57.78	62.37	72.85
1:0.3:0.7	57.77	62.47	73.25
1:0.5:0.5	58.05	62.46	73.35
1:0.7:0.3	57.42	62.50	72.30
1:0.9:0.1	57.53	62.08	73.75
1:01:00	57.94	62.29	73.03
0.5:0.01:1	59.25	62.14	73.61
0.1:0.01:1	61.24	61.94	73.54

5.3.5 Visualization and Analysis

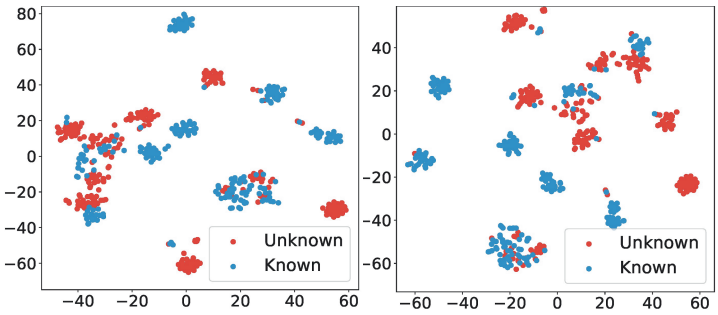

Figure 5.6: Visual comparison of test sample representations: (left) baseline Transformer backbone versus (right) our proposed *PEAK* framework.

To show the effectiveness of *PEAK* in learning the open decision boundary, we compare the visualizations of the final embeddings of all test samples in an arbitrary session using the vanilla backbone and our proposed *PEAK* framework.

As illustrated from the left part of Figure 5.6, we observe that there is no clear boundary between known points and unknown points, making it difficult to distinguish between them effectively. In the right part of Figure 5.6, we can observe distinct boundaries between the known points and unknown points, along with a more balanced distribution of clusters. These visualizations provide compelling evidence that our proposed *PEAK* significantly improves the learning of discriminative and compact decision boundaries for open detection.

5.4 Summary

In this chapter, we introduced *PEAK*, a novel and versatile framework designed to tackle the challenges of open-world continual learning under limited supervision.

At its core, *PEAK* dynamically models the evolving knowledge space, enabling effective extraction, representation, and progressive integration of information from both known and previously unseen classes. By jointly optimizing instance-wise token augmentation and adaptive knowledge accumulation, *PEAK* not only enriches the semantic representation of known categories but also enforces tighter, more discriminative decision boundaries—significantly boosting open-set recognition performance.

A major advantage of *PEAK* is its inherently *backbone-agnostic* design, allowing seamless integration with diverse feature extractors and embedding architectures. This plug-and-play flexibility ensures broad applicability across domains and datasets while maintaining strong generalization, even under varying data distributions and task complexities.

Looking ahead, several promising research directions emerge for future research. Integrating large language models (LLMs) could enhance cross-modal reasoning and provide richer semantic priors, further improving robustness in open-world scenarios. Also, improving the transparency of learned representations by interpretable knowledge structures may yield deeper insights into model decision-making, aiding both debugging and real-world trustworthiness. Finally, developing efficient strategies for rapid generalization and dynamic adaptation in the testing phase could further strengthen *PEAK*'s ability to handle evolving class spaces in truly open-ended dynamic environments.