Towards Real-world Scenario: Imbalanced New Intent Discovery

Shun Zhang[♠♡], **Chaoran Yan**[♠], **Jian Yang**[♠*], **Jiaheng Liu**[♠], **Ying Mo**[♠], **Jiaqi Bai**[♠♡], **Tongliang Li**[♣], **Zhoujun Li**[♠♡]
♠ State Key Laboratory of Complex & Critical Software Environment, Beihang University
♡ School of Cyber Science and Technology, Beihang University
♣ Computer School, Beijing Information Science and Technology University
{shunzhang, ycr2345, jiaya, bjq,lizj}@buaa.edu.cn {tonylingli}@bistu.edu.cn

Abstract

New Intent Discovery (NID) aims at detecting known and previously undefined categories of user intent by utilizing limited labeled and massive unlabeled data. Most prior works often operate under the unrealistic assumption that the distribution of both familiar and new intent classes is uniform, overlooking the skewed and long-tailed distributions frequently encountered in real-world scenarios. To bridge the gap, our work introduces the imbalanced new intent discovery (i-NID) task, which seeks to identify familiar and novel intent categories within long-tailed distributions. A new benchmark (ImbaNID-Bench) comprised of three datasets is created to simulate the real-world long-tail distributions. ImbaNID-Bench ranges from broad cross-domain to specific single-domain intent categories, providing a thorough representation of practical use cases. Besides, a robust baseline model ImbaNID is proposed to achieve cluster-friendly intent representations. It includes three stages: model pre-training, generation of reliable pseudo-labels, and robust representation learning that strengthens the model performance to handle the intricacies of real-world data distributions. Our extensive experiments on previous benchmarks and the newly established benchmark demonstrate the superior performance of ImbaNID in addressing the i-NID task, highlighting its potential as a powerful baseline for uncovering and categorizing user intents in imbalanced and long-tailed distributions[1].

1 Introduction

New intent discovery (NID) has captured increasing attention due to its adaptability to the evolving user needs in open-world scenarios (Mou et al., 2022; Siddique et al., 2021; Yang et al., 2020; Chrabrowa et al., 2023; Raedt et al., 2023). NID methods generally follow a two-stage training process, including a knowledge transfer and a discovery stage. The prior knowledge is injected into the model via pre-training and then the discriminative representation is learned for known and novel intent categories (Zhang et al., 2021a, 2022; Zhou et al., 2023; Zhang et al., 2023b; Shi et al., 2023).

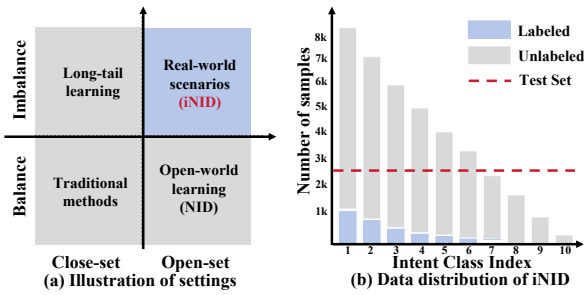


Figure 1: Illustration of proposed i-NID task: (a) i-NID unifies open-world and long-tail learning paradigms; (b) i-NID uses labeled and unlabeled data following a long-tail distribution to identify and categorize user intents.

Despite the considerable advancements in NID, there remain two salient challenges impeding adoption in practical scenarios. In Fig. 1, most NID approaches predominantly address the issue of intent discovery within the framework of balanced datasets. But the distribution of intents often follows a long-tailed pattern (Mou et al., 2022), particularly in dialogue systems, wherein a small number of intents are highly represented and a wide variety of intents (unknown intents) are sparsely exemplified. Secondly, NID methods suffer from severe clustering degradation, where lack of improved methods for unbalanced data distributions and leading to poor performance in unbalanced scenarios. Therefore, we explore the new methods under the **I**mbalanced **N**ew **I**ntent **D**iscovery (**i-NID**) task to bridge the gap between the NID and real-world applications.

To break out the aforementioned limitations, we propose a novel framework ImbaNID, which includes three key components: model pre-training,

*Corresponding author.
[1] https://github.com/Zkdc/i-NID

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 3949–3963

reliable pseudo-labeling (RPL), and robust representation learning (RRL). Specifically, the multi-task pre-training incorporates the generalized prior knowledge into the mode for establishing a robust representational foundation conducive to clustering known and novel intents. The RPL component formulates the pseudo-label generation as a relaxed optimal transport problem, applying adaptive constraints to recalibrate the class distribution for enhanced uniformity. The model bias issues can be mitigated in long-tail settings while furnishing reliable supervisory cues for downstream representation learning. Then, a novel distribution-aware and quality-aware noise regularization technique is introduced in RRL to effectively distinguish between clean and noisy samples. A contrastive loss function is subsequently used to facilitate the formation of distinct and well-separated clusters of representations for known and novel intent categories. The collaborative synergy between RPL and RRL fosters an iterative training process to create a symbiotic relationship. This iterative approach cultivates intent representations conducive to clustering, significantly aiding the i-NID task. For better evaluation of unbalanced distribution, we introduce a comprehensive benchmark ImbaNID-Bench for i-NID evaluation.

Extensive experiments of ImbaNID are evaluated on the previous common benchmarks and our proposed benchmark ImbaNID-Bench. The results demonstrate that ImbaNID consistently achieves state-of-the-art performance across all clusters, notably surpassing standard NID models by an average margin of 2.7% in long-tailed scenarios. The contributions are summarized as follows:

- We introduce the imbalanced new intent discovery (i-NID) task, which first encapsulates the challenges of clustering known and novel intent classes within long-tailed distributions. Different model performances under unbalanced distribution are sufficiently explored.
- We construct three comprehensive i-NID datasets to facilitate further advancements in i-NID research. Our extensive experiments on these datasets validate the superiority of the proposed method ImbaNID.
- For i-NID, we develop a novel ImbaNID approach that iteratively enhances pseudo-label generation and representation learning to ensure cluster-adaptive intent representations.

ImbaNID-Bench	$\lvert\mathcal{Y}^k\rvert$	$\lvert\mathcal{Y}^n\rvert$	$\lvert\mathcal{D}_l\rvert$	$\lvert\mathcal{D}_u\rvert$	$\lvert\mathcal{D}_t\rvert$
CLINC150-LT	113	37	583	6395	2250
BANKING77-LT	58	19	383	4658	3080
StackOverflow20-LT	15	5	510	6669	1000

Table 1: Statistics of the ImbaNID-Bench datasets when $\gamma = 10$. $|\mathcal{Y}^k|$, $|\mathcal{Y}^n|$, $|\mathcal{D}_l|$, $|\mathcal{D}_u|$ and $|\mathcal{D}_t|$ represent the number of known categories, novel categories, labeled data, unlabeled data, and testing data.

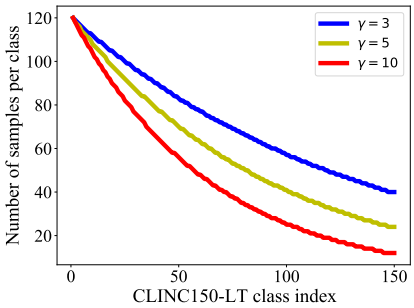


Figure 2: Number of training samples per class in artificially created long-tailed CLINC150-LT datasets with different imbalance factors.

2 Datasets

We introduce a new benchmark (called ImbaNID-Bench) for NID evaluation tailored to long-tail distribution scenarios, which comprises three datasets named CLINC150-LT, BANKING77-LT, and StackOverflow20-LT, derived from CLINC (Larson et al., 2019), BANKING (Casanueva et al., 2020) and StackOverflow (Xu et al., 2015). Comprehensive statistics for each dataset are documented in Appendix B. Here, we describe the details of the ImbaNID-Bench datasets.

Data Construction The first step is to simulate the long-tail distribution frequently encountered in real-world scenarios (Cui et al., 2019). Each class is assigned an index i ($1 \leq k \leq K$), where K denotes the total number of intent categories. $\gamma = \frac{n_{max}}{n_{min}}$ denotes the imbalance ratio, where n_k denotes the data size of class k, $n_{max} = \max_{1\leq k\leq K}(n_k)$, and $n_{min} = \min_{1\leq k\leq K}(n_k)$. We sample from each class based on $n_k = n_{\max}\gamma^{(j-1)/K}$. To explore the impact of data imbalance in NID, we construct ImbaNID-Bench by sampling with diverse imbalance ratios $\gamma \in \{3, 5, 10\}$. Fig. 2 shows the datasets created for CLINC150-LT with different imbalance factors (More details can be found in Appendix B). To simulate an open-world NID setting. We randomly select 75% of intents as known intents, and sample only 10% instances from known intent categories to form a labeled subset, while the remaining in-

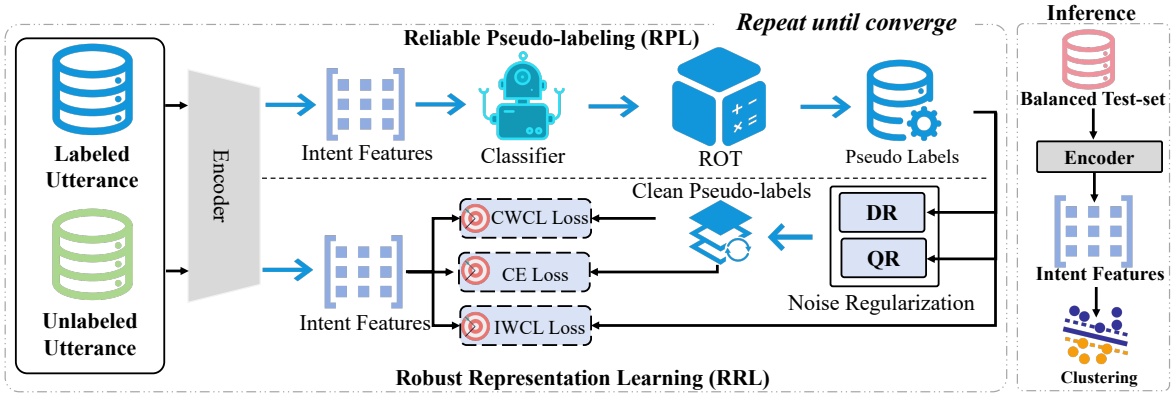


Figure 3: Overview of ImbaNID. The relaxed optimal transport (ROT) technique is used to produce high-quality pseudo-labels. Distribution-aware regularization (DR) and quality-aware regularization (QR) aim at filtering clean pseudo-labels. Finally, our framework incorporates class-wise contrastive learning (CWCL) and instance-wise contrastive learning (IWCL) to embed the data into a representation space where similar samples cluster together.

stances are treated as unlabeled data.

Data Statistics Since different proportions of imbalance ratios γ have different statistics, here we only display the results of $\gamma = 10$ for brevity. Table 1 shows the statistics of CLINC150-LT, BANKING77-LT and StackOverflow20-LT. We will release these datasets for future research.

3 Methodology

3.1 i-NID

Supposing we have a set of labeled intent data $\mathcal{D}_l = \{(x_i, y_i)|y_i \in \mathcal{Y}^k\}$ only comprised of known intent categories $\mathcal{Y}^k$, the deployed model in the wild may encounter inputs from unlabeled data $\mathcal{D}_u = \{x_i|y_i \in \{\mathcal{Y}^k, \mathcal{Y}^n\}\}$.The unlabeled data $\mathcal{D}_u$ contains both known intent categories $\mathcal{Y}^k$ and novel intent categories $\mathcal{Y}^n$, where $\mathcal{Y}^k$ and $\mathcal{Y}^n$ denote the data with the **K**nown and **N**ovel intents data, respectively. Both $\mathcal{D}_l$ and $\mathcal{D}_u$ present a long-tail distribution with imbalance ratio $\gamma > 1$. The goal of i-NID is to classify known classes and cluster novel intent classes in $\mathcal{D}_u$ by leveraging $\mathcal{D}_l$. Finally, model performance will be evaluated on a balanced testing set $\mathcal{D}_t = \{(x_i, y_i)|y_i \in \{\mathcal{Y}^k, \mathcal{Y}^n\}\}$.

3.2 Overall Framework

To achieve the learning objective of i-NID, we propose an iterative method to bootstrap model performance on reliable pseudo-labeling and robust representation learning. As shown in Fig. 3, our model mainly consists of three stages. Firstly, we pre-train a feature extractor on both labeled and unlabeled data to optimize better knowledge transfer (Sec. 3.3). Secondly, we obtain more accurate pseudo-labels by solving a relaxed optimal transport problem (Sec. 3.4). Thirdly, we propose two noise regularization techniques to divide pseudo-labels and employ contrastive loss to generate well-separated clusters of representations for both known and novel intent categories(Sec. 3.5).

3.3 Model Pre-training

Intent Representation Extraction To trigger the power of pre-trained language models in NID, we use BERT (Devlin et al., 2019; Yang et al., 2023) as the intent encoder ($E_\theta : \mathcal{X} \rightarrow \mathbb{R}^H$). Firstly, we feed the i^{th} input sentence x_i to BERT, and take all token embeddings $[t_0, \ldots, t_M] \in \mathbb{R}^{(M+1)\times H}$ from the last hidden layer (t_0 is the embedding of the [CLS] token). The mean pooling is applied to get the averaged sentence representation $\boldsymbol{z}_i \in \mathbb{R}^H$:

$$\boldsymbol{z}_i = \frac{1}{M+1}\sum_{i=0}^{M} t_i \quad (1)$$

where [CLS] is the vector for text classification, M is the sequence length, and H is the hidden size.

Knowledge Sharing To effectively generalize prior knowledge through pre-training to unlabeled data, we fine-tuned BERT on labeled data ($\mathcal{D}_l$) using the cross-entropy (CE) loss and on all available data ($\mathcal{D}_a = \mathcal{D}_l \cup \mathcal{D}_u$) using the masked language modeling (MLM) loss. The training objective of the fine-tuning can be formulated as follows:

$$\mathcal{L}_p = -\mathbb{E}_{x\in\mathcal{D}_l} \log P(y|x) - \mathbb{E}_{x\in\mathcal{D}_a} \log P(\hat{x}|x_{\setminus m(x)}) \quad (2)$$

where $\mathcal{D}_l$ and $\mathcal{D}_u$ are labeled and unlabeled intent corpus, respectively. $P(\hat{x}|x_{\setminus m(x)})$predicts masked

tokens $\hat{x}$ based on the masked sentence $x_{\setminus m(x)}$, where $m(x)$ denotes the masked tokens. The model is trained on the whole corpus $\mathcal{D}_a = \mathcal{D}_l \cup \mathcal{D}_u$.

3.4 Reliable Pseudo-labeling

Optimal Transport Here we briefly recap the well-known formulation of optimal transport (OT). Given two probability simplex vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ indicating two distributions, as well as a cost matrix $\mathbf{C} \in \mathbb{R}^{|\boldsymbol{\alpha}|\times|\boldsymbol{\beta}|}$, where $|\boldsymbol{\alpha}|$ denotes the dimension of $\boldsymbol{\alpha}$, OT aims to seek the optimal coupling matrix $\mathbf{Q}$ by minimizing the following objective:

$$\min_{\mathbf{Q}\in\mathbf{\Pi}(\boldsymbol{\alpha},\boldsymbol{\beta})} \langle \mathbf{Q}, \mathbf{C} \rangle \tag{3}$$

where $\langle \cdot, \cdot \rangle$ denotes frobenius dot-product. The coupling matrix $\mathbf{Q}$ satisfies the polytope $\mathbf{\Pi}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \left\{ \mathbf{Q} \in \mathbb{R}_+^{|\boldsymbol{\alpha}|\times|\boldsymbol{\beta}|} \mid \mathbf{Q}\mathbf{1}_{|\boldsymbol{\beta}|} = \boldsymbol{\alpha}, \mathbf{Q}^\top \mathbf{1}_{|\boldsymbol{\alpha}|} = \boldsymbol{\beta} \right\}$, where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are essentially marginal probability vectors. Intuitively speaking, these two marginal probability vectors can be interpreted as coupling budgets, which control the mapping intensity of each row and column in $\mathbf{Q}$.

Relaxed Optimal Transport for Pseudo-labeling The variables $\mathbf{Q} \in \mathbb{R}_+^{N\times K}$ and $\mathbf{P} \in \mathbb{R}_+^{N\times K}$ represent pseudo-labels matrix and classifier predictions, respectively, where N is the number of samples, and K [2] is the number of classes. The OT-based PL considers mapping samples to class and the cost matrix $\mathbf{C}$ can be formulated as $-\log \mathbf{P}$. So, we can rewrite the objective for OT-based PL based on the problem (3) as follows:

$$\begin{aligned} &\min_{\mathbf{Q},\boldsymbol{b}} \langle \mathbf{Q}, -\log \mathbf{P} \rangle + \lambda H(\mathbf{Q}) \\ &s.t.\ \mathbf{Q}\mathbf{1} = \boldsymbol{\alpha}, \mathbf{Q}^T\mathbf{1} = \boldsymbol{\beta}, \mathbf{Q} \geq 0 \end{aligned} \tag{4}$$

where the function H is the entropy regularization, λ is a scalar factor, $\boldsymbol{\alpha} = \frac{1}{N}\mathbf{1}$ is the sample distribution and $\boldsymbol{\beta}$ is class distribution. So the pseudo-labels matrix $\mathcal{U}_a$ can be obtained by normalization: $N\mathbf{Q}$. However, in the i-NID setup, the class distribution is often long-tailed and unknown, and the model optimized based on the problem (4) tends to learn degenerate solutions. This mismatched class distribution will lead to unreliable pseudo-labels. To mitigate this issue, we impose a soft constraint (ROT) on the problem (4). Instead of the traditional equality constraint (Asano et al., 2020; Caron et al., 2020a), we employ a Kullback-Leibler (KL) divergence constraint to encourage a uniform class distribution. This constraint is crucial for preventing degenerate solutions in long-tailed scenarios while allowing for the generation of imbalanced pseudo-labels due to its more relaxed nature compared to an equality constraint. The formulation of ROT is articulated as follows:

$$\begin{aligned} &\min_{\mathbf{Q},\boldsymbol{\beta}} \langle \mathbf{Q}, -\log \mathbf{P} \rangle + \lambda_1 H(\mathbf{Q}) + \lambda_2 D_{\mathrm{KL}}(\frac{1}{K}\mathbf{1}, \boldsymbol{\beta}) \\ &s.t.\ \mathbf{Q}\mathbf{1} = \boldsymbol{\alpha}, \mathbf{Q}^T\mathbf{1} = \boldsymbol{\beta}, \mathbf{Q} \geq 0, \boldsymbol{\beta}^T\mathbf{1} = 1 \end{aligned} \tag{5}$$

where λ_2 is a hyper-parameter and D_{KL} is the Kullback-Leibler divergence. The optimization problem (5) can be tractably solved using the Sinkhorn-Knopp algorithm (Cuturi, 2013) and we detail the optimization process in Appendix A.

3.5 Robust Representation Learning

Directly using generated pseudo-labels for representational learning is risky due to significant noise in early-stage pseudo-labeling. Consequently, we categorize pseudo-labels as clean or noisy based on their distribution and quality, applying contrastive loss to achieve cluster-friendly representations.

Noise Regularization We initially introduce a *distribution-aware regularization* (DR) to align the sample selection ratio with the class prior distribution, effectively mitigating selection bias in i-NID setup. This regularization combines small-loss instances with class distributions, ensuring inclusive representation of all classes, particularly Tail categories, during training. Specifically, the final set of selected samples S' is represented as follows:

$$S' = \bigcup_{j=1}^{K} s'_j \tag{6}$$

where K is total classes, s'_j is the set of samples selected from the j-th category slice s_j, defined as:

$$s'_j = \{h \mid (h \in s_j) \wedge (\mathrm{sort}(l(h)) \leq k_j)\} \tag{7}$$

where $l(h)$ is the instance-level loss of h, ρ is threshold hyper-parameter, r_j is the class distribution, $k_j = \min(|s_j|, \lceil N\rho r_j \rceil)$.

In addition, to select high-confidence pseudo-labels that closely align with the predicted labels, we propose a *quality-aware regularization* (QR). Specifically, we calculate confidence scores for each pseudo-label and then select the clean samples, denoted as h, whose confidence scores exceed a certain threshold τ_g:

$$\mathcal{A}' = \{h \mid (h \in \mathcal{U}_a) \wedge (\max(\boldsymbol{p}) > \tau_g)\} \tag{8}$$

[2] We estimate the number of classes K based on previous works (Zhang et al., 2021a) to ensure a fair comparison. We provide a detailed discussion on estimating K in Appendix E.

where $\boldsymbol{p}$ is the probability vector for h and $\tau_g \in [0, 1]$ is a confidence threshold hyper-parameter. Then the overall pseudo-labels $\mathcal{U}_a$ can filter out the clean pseudo-labels $\mathcal{U}_{clean}$ as follows:

$$\mathcal{U}_{clean} = \{h \mid (h \in \mathcal{S}') \vee (h \in \mathcal{A}')\} \tag{9}$$

Contrastive Clustering Following the extraction of clean pseudo-labels, we extend the traditional contrastive loss (Khosla et al., 2020) to utilize label information, forming positive pairs from same-class samples within $\mathcal{U}_{clean}$. Additionally, to enhance the model's emphasis on clean samples, we introduce a method for encoding soft positive correlation among pseudo-positive pairs, enabling adaptive contribution. Specifically, for an intent sample x_i, we first acquire its $L2$-normalized embedding z_i. By multiplying the confidence scores q of two samples, we obtain an *adaptive weight* $w_{ij} = q_i \cdot q_j$. The class-wise contrastive loss (CWCL) is then defined as follows:

$$\mathcal{L}_c(i) = \sum_{p \in P(i)} w_{ip} \cdot \log \frac{\exp(z_i \cdot z_p/\tau)}{\sum_j \mathbb{1}_{i \neq j} \exp(z_i \cdot z_j/\tau)} \tag{10}$$
$$P(i) = \{p \mid (p \in \mathcal{U}_{clean}) \wedge (c_i = c_p)\}$$

where $P(i)$ represents the indices of instances sharing the same label as x_i, and τ is a hyper-parameter. Fundamentally, CWCL loss brings intents of the same class closer together while distancing clusters of different classes, effectively creating a clustering effect. To enhance the generalization of intent representation, we incorporate instance-wise contrastive learning (Chen et al., 2020). The augmented views of instances in $\mathcal{U}_a$ are used as positive examples. The instance-wise contrastive loss (IWCL) is defined as follows:

$$\mathcal{L}_i(i) = -\log \frac{\exp(z_i \cdot \bar{z}_i/\tau)}{\sum_j \mathbb{1}_{i \neq j} \exp(z_i \cdot z_j/\tau)} \tag{11}$$

where z_i, $\bar{z}_i$ regard an anchor and its augmented sample, respectively, and $\bar{z}_i$ denotes the random token replacement augmented view of z_i.

Joint Training To mitigate the risk of catastrophic forgetting of knowledge, we incorporate cross-entropy loss on $\mathcal{U}_{clean}$ into the training process. Overall, the optimization of ImbaNID is to minimize the combined training objective:

$$\mathcal{L}_{all} = \omega \cdot \Big(\sum_{i \in N} \frac{1}{1 + |P(i)|} (\mathcal{L}_c(i) + \mathcal{L}_i(i))\Big) + (1 - \omega) \cdot \mathcal{L}_{ce} \tag{12}$$

where ω is a hyper-parameter and $|\cdot|$ is the cardinality computation. When x_i is a noisy example, $\mathcal{L}_c(i) = 0$ and $|P(i)| = 0$. During inference, we only utilize the cluster-level head and compute the argmax to get the cluster results.

4 Experiments

4.1 Experimental Setup

Baseline Methods We compare our method with various baselines and state-of-the-art methods, including DeepAligned (Zhang et al., 2021a), GCD (Vaze et al., 2022), CLNN (Zhang et al., 2022), DPN (An et al., 2023), LatentEM (Zhou et al., 2023), and USNID (Zhang et al., 2023b). Please see Appendix C for more comprehensive comparison and implementation details.

Evaluation Metrics We adopt three metrics for evaluating clustering results: Normalized Mutual Information (NMI), Adjusted Rand Index (ARI), and clustering Accuracy (ACC) based on the Hungarian algorithm. Furthermore, to more easily assess the impact of long tail distribution on performance, we divide $\mathcal{Y}^k$ and $\mathcal{Y}^n$ into three distinct groups {Head, Medium, Tail} with the proportions $|\text{Head}| : |\text{Medium}| : |\text{Tail}| = 3 : 4 : 3$.

Implementation Details To ensure a fair comparison for ImbaNID and all baselines, we adopt the pre-trained 12-layer bert-uncased BERT model[3] (Devlin et al., 2019) as the backbone encoder in all experiments and only fine-tune the last transformer layer parameters to expedite the training process (Zhang et al., 2021a). We adopt the AdamW optimizer with the weight decay of 0.01 and gradient clipping of 1.0 for parameter updating. For CLNN (Zhang et al., 2022), the external dataset is not used as in other baselines, the parameter of top-k nearest neighbors is set to {100, 50, 500} for CLINC, BANKING, and StackOverflow, respectively, as utilized in Zhang et al. (2022). For all experiments, we set the batch size as 512 and the temperature scale as τ = 0.07 in Eq. (10) and Eq. (11). We set the parameter ρ = 0.7 in Eq. (7) and the confidence threshold τ_g = 0.9 in Eq. (8). We adopt the data augmentation of random token replacement as Zhang et al. (2022). All experiments are conducted on 4 Tesla V100 GPUs and averaged over 3 runs.

[3] https://huggingface.co/bert-base-uncased

Methods	CLINC150-LT											
	$\gamma = 1$			$\gamma = 3$			$\gamma = 5$			$\gamma = 10$		
	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC
GCD	91.13	67.44	77.50	87.61	59.71	73.07	84.18	53.04	67.96	80.21	47.64	61.91
DeepAligned	93.89	79.75	86.49	92.29	73.79	81.78	90.93	70.19	79.02	88.43	62.47	71.47
CLNN	95.45	84.30	89.46	93.52	78.02	85.42	92.54	73.05	79.38	89.52	63.92	72.00
DPN	95.11	86.72	89.06	94.84	79.98	85.64	94.51	79.32	84.49	92.43	70.62	77.51
LatentEM	95.01	83.00	88.99	93.74	78.16	84.62	93.39	77.23	83.78	92.01	72.77	80.22
USNID	96.55	88.43	92.18	94.67	80.30	85.33	94.06	77.60	82.49	91.62	68.61	74.40
ImbaNID	**97.26**	**91.78**	**95.64**	**95.60**	**85.36**	**90.44**	**94.65**	**81.90**	**88.04**	**93.40**	**76.21**	**82.40**
Methods	**BANKING77-LT**											
	$\gamma = 1$			$\gamma = 3$			$\gamma = 5$			$\gamma = 10$		
	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC
GCD	77.86	46.87	58.95	71.92	42.35	56.98	69.16	37.93	53.41	66.89	33.38	46.92
DeepAligned	79.39	53.09	64.63	78.93	51.65	63.64	77.99	48.56	60.06	75.01	44.11	54.03
CLNN	86.19	66.98	77.22	85.64	65.34	75.75	82.95	58.87	70.65	79.99	52.04	62.63
DPN	82.58	61.21	72.96	84.43	61.36	72.27	80.88	49.75	61.69	77.17	43.41	57.95
LatentEM	84.02	62.92	74.03	83.37	61.23	73.08	81.38	56.78	69.51	80.55	55.65	65.05
USNID	87.53	69.88	79.92	86.62	67.01	75.03	83.59	60.56	70.06	80.49	54.26	63.15
ImbaNID	**87.66**	**70.13**	**81.14**	**86.79**	**67.35**	**76.72**	**83.60**	**61.18**	**72.89**	**81.08**	**55.80**	**66.59**
Methods	**StackOverflow20-LT**											
	$\gamma = 1$			$\gamma = 3$			$\gamma = 5$			$\gamma = 10$		
	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC
GCD	62.07	45.11	66.81	61.86	40.59	65.30	57.84	36.15	59.10	48.04	27.55	48.60
DeepAligned	76.47	62.52	80.26	75.27	62.73	77.10	75.47	64.19	78.50	73.47	61.82	73.80
CLNN	77.12	69.36	82.90	78.78	68.98	84.30	77.67	65.81	76.70	75.29	60.46	76.60
DPN	61.13	52.59	48.09	79.64	69.22	85.00	78.91	51.81	81.00	76.56	63.15	78.30
LatentEM	77.32	65.70	80.50	75.54	63.04	77.40	77.42	65.72	79.20	77.07	65.20	78.17
USNID	81.47	76.08	86.43	81.99	74.64	86.90	81.34	72.28	83.00	78.09	66.24	78.90
ImbaNID	**83.52**	**77.06**	**88.30**	**82.12**	**75.09**	**87.40**	**81.42**	**73.09**	**86.50**	**79.78**	**71.15**	**82.60**

Table 2: The main results on three datasets under various imbalance ratios γ ($\gamma = 1$ is the balanced NID setting). We set the known class ratio $|\mathcal{Y}^k|/|\mathcal{Y}^k \cap \mathcal{Y}^n|$ to 0.75, and the labeled ratio of known intent classes to 0.1 to conduct experiments. Results are averaged over three random run (p-value < 0.01 under t-test). We bold the **best result**.

4.2 Main Results

ImbaNID achieves SOTA results in both balanced and imbalanced settings. In Table 2, we present a comprehensive comparison of ImbaNID with prior start-of-the-art baselines in both balanced and multiple imbalanced settings. We observe that ImbaNID significantly outperforms prior rivals by a notable margin of 3.9% under various settings of imbalance ratio. Specifically, on the broad cross-domain CLINC150-LT dataset, ImbaNID beats the previous state-of-the-art with an increase of 3.5% in `ACC`, 0.7% in `NMI`, and 3.9% in `ARI` on average. On the StackOverflow20-LT with fewer categories, ImbaNID demonstrates its effectiveness with significant improvements of 2.6% in `ACC`, 0.6% in `NMI`, and 2.4% in `ARI` on average, consistently delivering substantial performance gains across each imbalanced subset. When applied to the specific single-domain BANKING77-LT datasets, ImbaNID reliably achieves significant performance improvements, underscoring its effectiveness in narrow-domain scenarios with indistinguishable intents. These results show the conventional NID models with naive pseudo-labeling and representation learning methods encounter a great challenge in handling the i-NID task. Our method efficiently produces accurate pseudo-labels under imbalanced conditions by employing soft constraints and utilizes these pseudo-labels to construct cluster-friendly representations.

Effectiveness on Long-tailed Distribution We also provide a detailed analysis of the results for the `Head`, `Medium`, and `Tail` classes, offering a more comprehensive understanding of our method's performance across three i-NID datasets. Fig. 4 presents the comparative accuracy among various groups under the condition $\gamma = 3$. It is noteworthy that in `Tail` classes, the gaps between ImbaNID and the best baseline are 4.2%, 3.5% and 3.7% across three datasets. In contrast, most baselines exhibit degenerated performance, particularly on CLINC150-LT and BANKING77-LT. Moreover, ImbaNID retains a competitive performance on `Head` classes. These results highlight the effective-

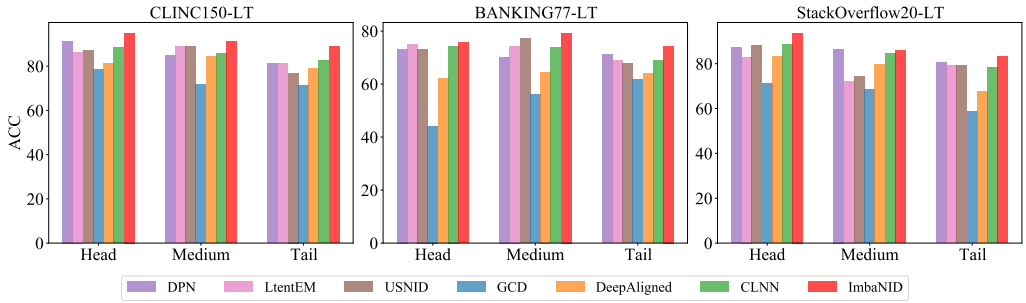


Figure 4: Head, Medium, and Tail comparison on the ImbaNID-Bench datasets.

Methods	CLINC150-LT			BANKING77-LT			StackOverflow20-LT		
	Head	Medium	Tail	Head	Medium	Tail	Head	Medium	Tail
ImbaNID	**82.52**	**90.67**	**71.26**	**68.26**	**66.05**	**65.87**	**90.67**	**87.25**	**81.67**
① w/ COT	72.74	87.44	58.67	62.72	63.11	48.70	86.63	85.75	79.67
② w/ EOT	81.41	83.00	65.33	66.59	65.40	57.61	90.00	86.11	81.60
③ w/ MOT	69.33	57.67	30.52	62.07	57.34	26.20	88.97	66.00	64.33
④ w/o DR	80.74	88.57	71.21	67.17	65.08	49.67	88.33	86.75	81.33
⑤ w/o QR	82.50	88.94	70.52	63.91	65.42	59.02	87.67	86.00	81.57
⑥ w/o DR and QR	81.19	87.19	71.05	67.50	64.88	50.00	88.33	86.51	80.33
⑦ w/o Adaptive Weight	82.37	90.22	71.11	68.18	65.81	64.57	90.30	87.00	79.67
⑧ w/o CWCL	81.93	90.11	70.81	67.83	66.03	58.70	90.33	85.22	78.00
⑨ w/o IWCL	81.78	86.44	71.23	65.54	64.22	65.20	90.51	76.75	80.33

Table 3: Experimental results of the ablation study on the ImbaNID-Bench datasets at imbalance ratios $\gamma = 10$.

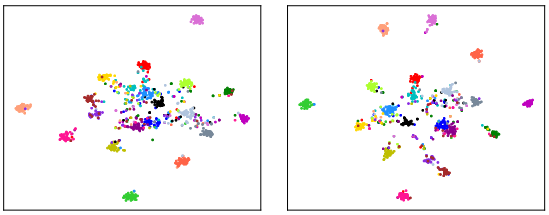
ness of ImbaNID in i-NID setup, making it particularly advantageous for Head and Tail classes.

4.3 Effect of Pseudo-label Assignment

To evaluate ROT in reliable pseudo-labels generation of the i-NID setup, we compare three OT-based optimizations for pseudo-labels generation, including COT (Caron et al., 2020a), EOT (Asano et al., 2020), and MOT (Li et al.). (1) COT denotes the removal of the KL term from our optimization problem (5). (2) EOT signifies the replacement of the KL term in our optimization problem (5) with a typical entropy regularization $\mathrm{KL}(\boldsymbol{\beta}\|\hat{\boldsymbol{\beta}})$. (3) MOT operates without any assumption on the class distribution $\boldsymbol{\beta}$, allowing $\boldsymbol{\beta}$ to be updated by the model prediction using a moving-average mechanism. Specifically, $\boldsymbol{\beta} = \mu\hat{\boldsymbol{\beta}} + (1-\mu)\boldsymbol{v}$, where μ is the moving-average parameter, $\hat{\boldsymbol{\beta}}$ is the last updated $\boldsymbol{\beta}$ and $v_j = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\left(j = \arg\max \mathbf{P}_i\right)$. From Table 3, we can observe that ImbaNID outperforms the model ①, which indicates the necessity of imposing constraints on the class distribution. Compared to the model ②, ImbaNID achieves the most gains for Head and Tail classes, indicating it better constrains the class distribution towards uniformity. Finally, when compared to the above strategies, the performance of the model ② in the Tail classes is notably inferior. The results stem from inadequate constraints on the category distribution, leading to a decline in cluster quality. The comparisons underscore that ImbaNID demonstrates strong proficiency in generating accurate pseudo-labels within the i-NID setup.

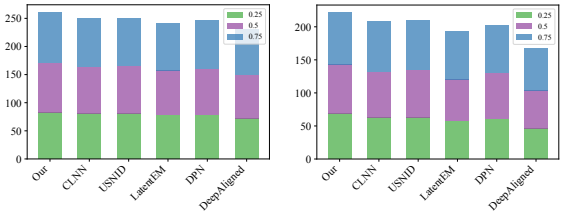
4.4 Effect of Noise Regularization

To investigate the effectiveness of noise regularization (NR) in filtering noisy pseudo-labels, we conduct ablation experiments to analyze its contributions. In Table 3, eliminating DR diminishes intent discovery performance, particularly in Tail classes. This occurs because a higher proportion of Head classes in pseudo-labels inevitably results in model bias. Furthermore, removing QR results in decreased performance, primarily because fewer examples are initially selected due to the classifier's low confidence, leading to degenerate solutions. Notably, considering all pseudo-labels as clean leads to significant performance drops across all datasets, indicating that numerous noisy pseudo-labels may cause model overfitting and reduced generalization. The results indicate that NR is indispensable to ImbaNID in handling i-NID setup.



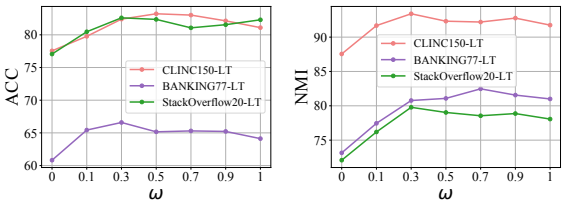
(a) USNID (b) ImbaNID

Figure 5: t-SNE visualization of embeddings on the StackOverflow20-LT dataset. The known class ratio $|\mathcal{Y}^k|/|\mathcal{Y}^k \cap \mathcal{Y}^n|$ is 0.75, and the labeled ratio is 0.1.



(a) Impact on CLINC-LT (b) Impact on BANKING-LT

Figure 6: Impact of varying the known class ratio on two datasets. The x-axis represents different models and the y-axis denotes their corresponding accuracy values.



(a) Effects on ACC (b) Effects on NMI

Figure 7: Effects of ω on ImbaNID-Bench.

4.5 Effect of Contrastive Clustering

To assess the impact of contrastive clustering in representation learning, we carry out ablation experiments to analyze its individual effects in Table 3. When the adaptive weight strategy is removed from Eq. (10), the model disregards probability distribution information and becomes more susceptible to noisy pseudo-labels. Then, removing `CWCL` or `IWCL` from Eq. (12) results in performance degradation, suggesting that class-wise and instance-wise contrastive learning respectively aid in developing compact cluster representations and enhancing representation generalization. In Fig. 5, we use t-SNE to illustrate embeddings learned on the StackOverflow20-LT dataset, where ImbaNID visibly forms more distinct clusters than comparative methods, underscoring the effectiveness of our model.

4.6 Effect of Known Class Ratio

To investigate the impact of varying numbers of known intents, we vary the ratio of known intents ranging in $\{25\%, 50\%, 75\%\}$ during training. Fig. 6 illustrates the comparative accuracy among various ratio of known intents under the condition $\gamma = 3$. We observe that even when only a few known intents are available, our method still performs better than other strong baselines. This demonstrates its strength in learning from labeled data and discovering inherent patterns from unlabeled data. Meanwhile, we note a rise in performance as the volume of labeled data incorporated increases, aligning with anticipated outcomes. In short, our proposed method exhibits strong robustness and generalization capability.

4.7 Effect of Exploration and Utilization

The weight of the multitask learning ω in Eq. 12 adjusts the contribution of two objectives. Intuitively, the first term aims to explore cluster-friendly intent representations across all samples, while the second term focuses on mitigating the risk of catastrophic forgetting, ensuring the effective utilization of knowledge derived from clean samples. We vary the value of ω and conduct experiments on ImbaNID-Bench ($\gamma = 10$) to explore the effect of ω, which also reflects the inference of exploration and utilization. In Fig. 7, only utilizing clean samples ($\omega = 0.0$) or only exploring($\omega = 1.0$) the intent representation will not achieve the best results. Interestingly, the effect of ω shows a similar trend (increase first and then decrease) on all metrics and datasets, which indicates that we can adjust the value of ω to give full play to the role of both so that the model can make better use of known knowledge to discover intents accurately.

4.8 Comparison of Time Complexity

The majority of existing methods (Zhang et al., 2022; An et al., 2023; Zhou et al., 2023) are mostly based on k-means for pseudo-labeling, while we propose a novel ROT approach for

pseudo-labeling. We discuss the comparison and selection of time complexity between pseudo-labeling methods based on k-means and ROT. Specifically, the k-means method is a clustering-based approach that iteratively computes distances between data points and assigns them to k cluster centers. Its time complexity, typically around $O(nkt)$, depends on the dataset size (n), the number of cluster centers (k), and the convergence speed (t). While the k-means method has lower time complexity, it is sensitive to the selection of initial cluster centers and convergence, leading to potentially unstable outcomes. On the other hand, ROT involves iteratively optimizing the distance or similarity between two data distributions to find the best mapping. Although the time complexity of ROT methods, such as those based on the Sinkhorn algorithm, is typically polynomial (e.g., $O(n^2m)$ where n is the number of source domain data points and m is the number of target domain data points), they generally provide high-quality pseudo-labels.

5 Related Work

5.1 New Intent Discovery

New Intent Discovery (NID) is similar to generalized category discovery (GCD) (Vaze et al., 2022), which originates from computer vision and aims to discover novel intents by utilizing the prior knowledge of known intents. Lin et al. (2020) conducts pair-wise similarity prediction to discover novel intents, and Zhang et al. (2021a) uses aligned pseudo-labels to help the model learn discriminative intent representations. Recent works further advance NID by incorporating contrastive learning (Shen et al., 2021; Kumar et al., 2022; Zhang et al., 2022, 2023b), knowledge transfer (An et al., 2023), probabilistic frameworks (Zhou et al., 2023), pseudo-label learning (Zhang et al., 2024a) or prototype attracting and dispersing (Zhang et al., 2024b) to capture cluster-friendly intent representation. However, those methods operate under the unrealistic assumption that the distribution of both known and new intent classes is uniform, overlooking the long-tailed distributions frequently encountered in real-world scenarios. In this work, we explore the imbalanced NID scenario.

5.2 Optimal Transport

Optimal Transport (OT) aims to find the most efficient transportation plan while adhering to marginal distribution constraints. It has been used in a broad spectrum of various tasks, including generative model (Gulrajani et al., 2017), semi-supervised learning (Taherkhani et al., 2020; Tai et al., 2021), clustering (Caron et al., 2020a; Zhang et al., 2023a) and new intent discovery (Zhang et al., 2024a). However, these methods typically impose an equality constraint when solving the OT problem. In contrast, we explore generating pseudo-labels by solving a relaxed OT problem. This approach encourages a uniform class distribution and addresses class degeneration in long-tailed

5.3 Contrastive Learning

Contrastive Learning (CL) has been widely adopted to generate discriminative sentence representations for various scenarios (Chen et al., 2020; Khosla et al., 2020; Li et al., 2021), such as out-of-domain detection (Zhang et al., 2023c,d), machine translation (Yang et al., 2020, 2021a,b, 2022b, 2024), and named entity recognition (Yang et al., 2022a; Mo et al., 2024). In essence, the primary intuition behind CL is to pull together positive pairs in the feature space while pushing away negative pairs. Motivated by its superior performance, contrastive learning has also been leveraged for intent recognition where it is used for NID. In this work, we design both class-wise and instance-wise contrastive learning objectives to learn cluster-friendly intent representations.

6 Conclusion

In this work, we first propose the i-NID task to identify known and infer novel intents within these long-tailed distributions. Then, we develop an effective ImbaNID baseline method for the i-NID task, where pseudo-label generation and representation learning mutually iterate to achieve cluster-friendly representations. Comprehensive experimental results on our ImbaNID-Bench benchmark datasets demonstrate the effectiveness of our ImbaNID method for i-NID. We hope our work will draw more attention from the community toward a broader view of tackling the i-NID problem.

Limitations

To better enlighten the follow-up research, we conclude the limitations of our method as follows: (1) Enhancing interpretability. Our ImbaNID automatically assigns labels to unlabeled utterances in real-world long-tail data distributions, yet it does not generate interpretable intent names for each cluster. (2) Integration with LLMs. Large-scale language models (LLMs) have shown an impressive ability in a variety of NLP tasks, we plan to explore the integration of ImbaNID with LLMs to boost performance in practical scenarios. (3) Reducing time complexity. The time complexity of relaxed optimal transport (ROT) is $O\left(n^2\right)$, we plan to further develop a fast matrix scaling algorithm to reduce the complexity.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. U1636211, U2333205, 61672081, 62302025, 62276017), a fund project: State Grid Co., Ltd. Technology R&D Project (ProjectName: Research on Key Technologies of Data Scenario-based Security Governance and Emergency Blocking in Power Monitoring System, Proiect No.: 5108-202303439A-3-2-ZN), the 2022 CCF-NSFOCUS Kun-Peng Scientific Research Fund and the Opening Project of Shanghai Trusted Industrial Control Platform and the State Key Laboratory of Complex & Critical Software Environment (Grant No. SKLSDE-2021ZX-18).

References

Wenbin An, Feng Tian, Qinghua Zheng, Wei Ding, QianYing Wang, and Ping Chen. 2023. Generalized category discovery with decoupled prototypical network. In *Proc. of AAAI*.

Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. 2020. Self-labelling via simultaneous clustering and representation learning. In *Proc. of ICLR*.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018. Deep clustering for unsupervised learning of visual features. In *Proc. of ECCV*.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020a. Unsupervised learning of visual features by contrasting cluster assignments. In *Proc. of NeurIPS*.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020b. Unsupervised learning of visual features by contrasting cluster assignments. In *Proc. of NeurIPS*.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proc. of ICML*.

Aleksandra Chrabrowa, Tsimur Hadeliya, Dariusz Kajtoch, Robert Mroczkowski, and Piotr Rybak. 2023. Going beyond research datasets: Novel intent discovery in the industry setting. In *Proc. of ACL Findings*.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge J. Belongie. 2019. Class-balanced loss based on effective number of samples. In *Proc. of CVPR*.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Proc. of NeurIPS*.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of AACL*.

Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. 2017. Improved training of wasserstein gans. In *Proc. of NeurIPS*.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Proc. of NeurIPS*.

Rajat Kumar, Mayur Patidar, Vaibhav Varshney, Lovekesh Vig, and Gautam Shroff. 2022. Intent detection and discovery from user logs via deep semi-supervised contrastive clustering. In *Proc. of NAACL*.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An evaluation dataset for intent classification and out-of-scope prediction. In *Proc. of EMNLP*.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven C. H. Hoi. 2021. Prototypical contrastive learning of unsupervised representations. In *Proc. of ICLR*.

Ziyun Li, Ben Dai, Furkan Simsek, Christoph Meinel, and Haojin Yang. Imbagcd: Imbalanced generalized category discovery.

Ting-En Lin, Hua Xu, and Hanlei Zhang. 2020. Discovering new intents via constrained deep adaptive clustering with cluster refinement. In *Proc. of AAAI*.

Ying Mo, Jian Yang, Jiahao Liu, Qifan Wang, Ruoyu Chen, Jingang Wang, and Zhoujun Li. 2024. MCL-NER: cross-lingual named entity recognition via multi-view contrastive learning. In *Proc. of AAAI*.

Yutao Mou, Keqing He, Yanan Wu, Pei Wang, Jingang Wang, Wei Wu, Yi Huang, Junlan Feng, and Weiran Xu. 2022. Generalized intent discovery: Learning from open world dialogue system. In *Proc. of COLING*.

Maarten De Raedt, Fréderic Godin, Thomas Demeester, and Chris Develder. 2023. IDAS: intent discovery with abstractive summarization. *CoRR*.

Xiang Shen, Yinge Sun, Yao Zhang, and Mani Najmabadi. 2021. Semi-supervised intent discovery with contrastive learning. In *Proceedings of the 3rd Workshop on Natural Language Processing for Conversational AI*.

Wenkai Shi, Wenbin An, Feng Tian, Qinghua Zheng, QianYing Wang, and Ping Chen. 2023. A diffusion weighted graph framework for new intent discovery. *arXiv preprint arXiv:2310.15836*.

A. B. Siddique, Fuad T. Jamour, Luxun Xu, and Vagelis Hristidis. 2021. Generalized zero-shot intent detection via commonsense knowledge. In *Proc. of SIGIR*.

Fariborz Taherkhani, Ali Dabouei, Sobhan Soleymani, Jeremy M. Dawson, and Nasser M. Nasrabadi. 2020. Transporting labels via hierarchical optimal transport for semi-supervised learning. In *Proc. of ECCV*.

Kai Sheng Tai, Peter Bailis, and Gregory Valiant. 2021. Sinkhorn label allocation: Semi-supervised classification via annealed self-training. In *Proc. of ICML*.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. 2022. Generalized category discovery. In *Proc. of CVPR*.

Jiaming Xu, Peng Wang, Guanhua Tian, Bo Xu, Jun Zhao, Fangyuan Wang, and Hongwei Hao. 2015. Short text clustering via convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*.

Jian Yang, Hongcheng Guo, Yuwei Yin, Jiaqi Bai, Bing Wang, Jiaheng Liu, Xinnian Liang, Linzheng Cahi, Liqun Yang, and Zhoujun Li. 2024. m3p: Towards multimodal multilingual translation with multimodal prompt. *arXiv preprint arXiv:2403.17556*.

Jian Yang, Shaohan Huang, Shuming Ma, Yuwei Yin, Li Dong, Dongdong Zhang, Hongcheng Guo, Zhoujun Li, and Furu Wei. 2022a. CROP: zero-shot cross-lingual named entity recognition with multilingual labeled sequence translation. In *Proc. of EMNLP Findings*.

Jian Yang, Shuming Ma, Li Dong, Shaohan Huang, Haoyang Huang, Yuwei Yin, Dongdong Zhang, Liqun Yang, Furu Wei, and Zhoujun Li. 2023. Ganlm: Encoder-decoder pre-training with an auxiliary discriminator. In *Proc. of ACL*.

Jian Yang, Shuming Ma, Haoyang Huang, Dongdong Zhang, Li Dong, Shaohan Huang, Alexandre Muzio, Saksham Singhal, Hany Hassan, Xia Song, and Furu Wei. 2021a. Multilingual machine translation systems from microsoft for WMT21 shared task. In *Proceedings of the Sixth Conference on Machine Translation, WMT@EMNLP 2021, Online Event, November 10-11, 2021*.

Jian Yang, Shuming Ma, Dongdong Zhang, Shuangzhi Wu, Zhoujun Li, and Ming Zhou. 2020. Alternating language modeling for cross-lingual pre-training. In *Proc. of AAAI*.

Jian Yang, Yuwei Yin, Shuming Ma, Haoyang Huang, Dongdong Zhang, Zhoujun Li, and Furu Wei. 2021b. Multilingual agreement for multilingual neural machine translation. In *Proc. of ACL*.

Jian Yang, Yuwei Yin, Shuming Ma, Dongdong Zhang, Shuangzhi Wu, Hongcheng Guo, Zhoujun Li, and Furu Wei. 2022b. UM4: unified multilingual multiple teacher-student model for zero-resource neural machine translation. In *Proc. of IJCAI*.

Chuyu Zhang, Ruijie Xu, and Xuming He. 2023a. Novel class discovery for long-tailed recognition. *CoRR*.

Hanlei Zhang, Hua Xu, Ting-En Lin, and Rui Lyu. 2021a. Discovering new intents with deep aligned clustering. In *Proc. of AAAI*.

Hanlei Zhang, Hua Xu, Ting-En Lin, and Rui Lyu. 2021b. Discovering new intents with deep aligned clustering. *Proc. of AAAI*.

Hanlei Zhang, Hua Xu, Xin Wang, Fei Long, and Kai Gao. 2023b. USNID: A framework for unsupervised and semi-supervised new intent discovery. *CoRR*.

Shun Zhang, Jiaqi Bai, Tongliang Li, Zhao Yan, and Zhoujun Li. 2023c. Modeling intra-class and inter-class constraints for out-of-domain detection. In *Proc. of DASFAA*.

Shun Zhang, Tongliang Li, Jiaqi Bai, and Zhoujun Li. 2023d. Label-guided contrastive learning for out-of-domain detection. In *Proc. of ICASSP*.

Shun Zhang, Chaoran Yan, Jian Yang, Changyu Ren, Jiaqi Bai, Tongliang Li, and Zhoujun Li. 2024a. Ronid: New intent discovery with generated-reliable labels and cluster-friendly representations. *CoRR*.

Shun Zhang, Jian Yang, Jiaqi Bai, Chaoran Yan, Tongliang Li, Zhao Yan, and Zhoujun Li. 2024b. New intent discovery with attracting and dispersing prototype. *arXiv preprint arXiv:2403.16913*.

Yuwei Zhang, Haode Zhang, Li-Ming Zhan, Xiao-Ming Wu, and Albert Lam. 2022. New intent discovery with pre-training and contrastive learning. In *Proc. of ACL*.

Yunhua Zhou, Guofeng Quan, and Xipeng Qiu. 2023. A probabilistic framework for discovering new intents. In *Proc. of ACL*.

A ROT

In this section, we provide a comprehensive optimization process for the ROT problem (5), the ROT objective is:

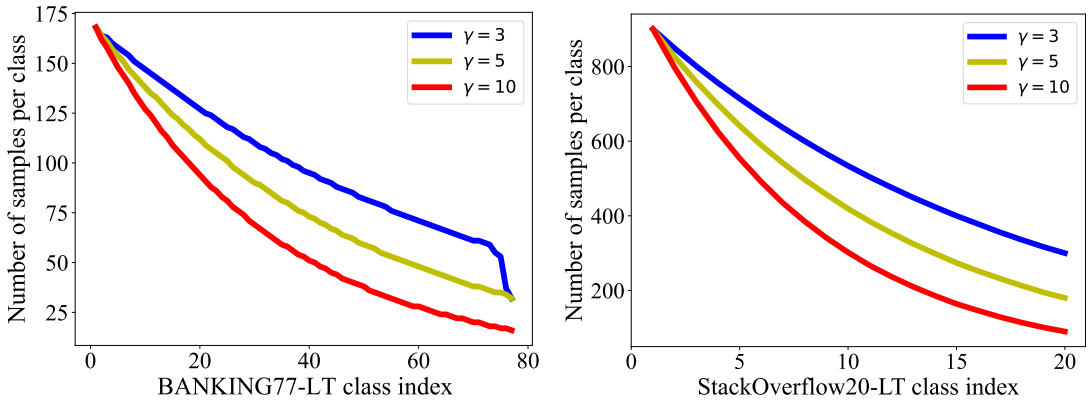
$$\begin{aligned} &\min_{\mathbf{Q},\boldsymbol{\beta}} \langle \mathbf{Q}, -\log \mathbf{P} \rangle + \lambda_1 H(\mathbf{Q}) + \lambda_2 D_{\mathrm{KL}}(\frac{1}{K}\mathbf{1}, \boldsymbol{\beta}) \\ &s.t.\ \mathbf{Q}\mathbf{1} = \boldsymbol{\alpha}, \mathbf{Q}^T\mathbf{1} = \boldsymbol{\beta}, \mathbf{Q} \geq 0, \boldsymbol{\beta}^T\mathbf{1} = 1 \end{aligned} \tag{13}$$

where λ_1 and λ_2 are hyper-parameters, and $D_{\mathrm{KL}}(\boldsymbol{A}, \boldsymbol{B})$ denotes the Kullback-Leibler Divergence. We utilize the Lagrangian multiplier algorithm for optimization:

$$\begin{aligned} L(\mathbf{Q}, \boldsymbol{\beta}, \boldsymbol{f}, \boldsymbol{g}, h) = &\langle \mathbf{Q}, -\log \mathbf{P} \rangle + \lambda_1 H(\mathbf{Q}) \\ &+ \lambda_2 D_{\mathrm{KL}}(\frac{1}{K}\mathbf{1}, \boldsymbol{\beta}) - \boldsymbol{f}^T(\mathbf{Q}\mathbf{1} - \boldsymbol{\alpha}) \\ &- \boldsymbol{g}^T(\mathbf{Q}^T\mathbf{1} - \boldsymbol{\beta}) - h(\boldsymbol{\beta}^T\mathbf{1} - 1) \end{aligned} \tag{14}$$

Dataset	Classes	#Training	#Validation	#Testing	Vocabulary	Length (Max / Avg)
CLINC	150	18000	2250	2250	7283	28 / 8.32
BANKING	77	9003	1000	3080	5028	79 / 11.91
StackOverflow	20	12000	2000	1000	17182	41 / 9.18

Table 4: Statistics of original datasets. # denotes the total number of utterances.



(a) BANKING77-LT (b) StackOverflow20-LT

Figure 8: Number of training samples per class in artificially created long-tailed BANKING77-LT and StackOverflow20-LT datasets with different imbalance factors.

ImbaNID-Bench ($\gamma = 3$)	$\lvert\mathcal{Y}^k\rvert$	$\lvert\mathcal{Y}^n\rvert$	$\lvert\mathcal{D}_l\rvert$	$\lvert\mathcal{D}_u\rvert$	$\lvert\mathcal{D}_t\rvert$
CLINC150-LT	113	37	868	9995	2250
BANKING77-LT	58	19	607	7163	3080
StackOverflow20-LT	15	5	830	10140	1000
ImbaNID-Bench ($\gamma = 5$)	$\lvert\mathcal{Y}^k\rvert$	$\lvert\mathcal{Y}^n\rvert$	$\lvert\mathcal{D}_l\rvert$	$\lvert\mathcal{D}_u\rvert$	$\lvert\mathcal{D}_t\rvert$
CLINC150-LT	113	37	719	8164	2250
BANKING77-LT	58	19	487	5924	3080
StackOverflow20-LT	15	5	686	8350	1000

Table 5: Statistics of the ImbaNID-Bench datasets when $\gamma = 3$ and $\gamma = 5$. $|\mathcal{Y}^k|$, $|\mathcal{Y}^n|$, $|\mathcal{D}_l|$, $|\mathcal{D}_u|$ and $|\mathcal{D}_t|$ represent the number of known categories, novel categories, labeled data, unlabeled data, and testing data.

where $\boldsymbol{f}$, $\boldsymbol{g}$, and h are Lagrangian multipliers. Differentiating Eq. (14) yields the following result:

$$\frac{\partial L}{\partial Q_{ij}} = \lambda_1 log(Q_{ij}) - log(P_{ij}) - f_i - g_j \tag{15}$$

$$\frac{\partial L}{\partial f_i} = -(\sum_j^K Q_{ij}) + \alpha_i \tag{16}$$

$$\frac{\partial L}{\partial g_j} = -(\sum_i^N Q_{ij}) + \beta_j \tag{17}$$

$$\frac{\partial L}{\partial \beta_j} = -\frac{\lambda_2}{K\beta_j} + g_j - h \tag{18}$$

$$\frac{\partial L}{\partial h} = -(\sum_j^K \beta_j) + 1 \tag{19}$$

Initially, we fix $\boldsymbol{\beta}$ and h, and then update $\mathbf{Q}$, $\boldsymbol{f}$, and $\boldsymbol{g}$. By setting $\frac{\partial L}{\partial Q_{ij}}$, $\frac{\partial L}{\partial f_i}$, and $\frac{\partial L}{\partial g_j}$ to zero, we obtain the following results:

$$\begin{aligned} Q_{ij} &= \exp(\frac{f_i + \log(P_{ij}) + g_j}{\lambda_1}) \\ &= \exp(\frac{f_i}{\lambda_1}) \cdot \exp(\frac{\log(P_{ij})}{\lambda_1}) \cdot \exp(\frac{g_j}{\lambda_1}) \end{aligned} \tag{20}$$

$$\sum_j^K Q_{ij} = \alpha_i, \sum_i^N Q_{ij} = \beta_j \tag{21}$$

Based on Eq. (20), we derive the following:

$$\mathbf{Q} = \text{diag}(\exp(\frac{\boldsymbol{f}}{\lambda_1})) \exp(\frac{\log \mathbf{P}}{\lambda_1}) \text{diag}(\exp(\frac{\boldsymbol{g}}{\lambda_1})) \tag{22}$$

Considering the constraints (21) and the conditions $\boldsymbol{\beta}^T\mathbf{1} = \boldsymbol{\alpha}^T\mathbf{1} = 1$, we solve Eq. (22) to determine the values of $\mathbf{Q}$, $\boldsymbol{f}$, and $\boldsymbol{g}$ using the Sinkhorn algorithm (Cuturi, 2013). Subsequently, with $\boldsymbol{f}$, $\boldsymbol{g}$, and $\mathbf{Q}$ fixed, we update $\boldsymbol{\beta}$ and h. Setting Eq. (18) to zero yields the following solution:

$$\beta_j = \frac{\lambda_2}{K(g_j - h)} \tag{23}$$

Take Eq. (23) into the Eq. (19) and let Eq. (19) equal to 0, we can obtain:

$$(\sum_j^K \beta_j(h)) - 1 = 0 \tag{24}$$

Algorithm 1 The optimization of ROT

Input: The cost matrix: $-\log \mathbf{P}$.
Output:
The transport matrix: $\mathbf{Q}$,
The class distribution: $\boldsymbol{\beta}$.
Procedure:

1: Initialize $\boldsymbol{\beta}$ as uniform distribution;
2: **for** $i = 1$ to T **do**
3: Fix $\boldsymbol{\beta}$ and h, calculate $\mathbf{Q}$, $\boldsymbol{f}$ and $\boldsymbol{g}$ with Sinkhorn algorithm.
4: Fix $\mathbf{Q}$, $\boldsymbol{f}$ and $\boldsymbol{g}$, update $\boldsymbol{\beta}$ and h with Eq. (23) and (24).
5: **end for**
6: Return $\mathbf{Q}$ and $\boldsymbol{\beta}$.

We obtain h from Eq.(24) using the bisection method and subsequently determine the corresponding $\boldsymbol{\beta}$. In the final step, we iteratively update $\boldsymbol{f}$, $\boldsymbol{g}$, $\mathbf{Q}$, and $\boldsymbol{\beta}$, h. The iterative optimization process for ROT is outlined in Algorithm1.

B Statistics of Datasets

We present detailed statistics of the CLINC (Larson et al., 2019), BANKING (Casanueva et al., 2020) and StackOverflow (Xu et al., 2015) datasets in Table 4. In addition, we display the number of samples per class for BANKING77-LT and StackOverflow20-LT under various imbalance factors, as shown in Fig. 8. We also provide dataset statistics for the ImbaNID-Bench datasets with imbalance factors of 3 and 5, as shown in Table 5.

C Comparison Methods

In this work, we compare the proposed ImbaNID method against several representative baselines including:

GCD (Vaze et al., 2022) introduces a combination of supervised and self-supervised contrastive learning to learn distinctive representations, which are then clustered using k-means.

DeepAligned (Zhang et al., 2021a) is an improved DeepClustering (Caron et al., 2018) that uses an alignment strategy to alleviate the label inconsistency problem.

MTP-CLNN (Zhang et al., 2022) is a method that applies multi-task pre-training and nearest neighbors contrastive learning for NID.

DPN (An et al., 2023) proposes a decoupled prototypical network that, by framing a bipartite matching problem for category prototypes, separates known and novel categories to meet their distinct training objectives and transfers category-specific knowledge for capturing high-level semantics.

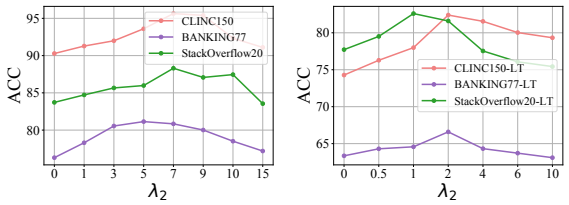
LatentEM (Zhou et al., 2023) introduces a principled probabilistic framework optimized with the EM algorithm. In the E-step, it assigns pseudo-labels, and in the M-step, it learns cluster-friendly representations and updates parameters through contrastive learning.

USNID (Zhang et al., 2023b) is a two-stage framework for both unsupervised and semi-supervised NID with an efficient centroid-guided clustering mechanism.

D Implementation Details

To ensure a fair comparison for ImbaNID and all baselines, we consistently adopt the pre-trained 12-layer bert-uncased BERT model[4] (Devlin et al., 2019) as the backbone encoder in all experiments and only fine-tune the last transformer layer parameters to expedite the training process as suggested in (Zhang et al., 2021a). We adopt the AdamW optimizer with 0.01 weight decay and 1.0 gradient clipping for parameter update. During pre-training, we set the learning rate to 5e-5 and adopt the early stopping strategy with a patience of 20 epochs. For CLNN (Zhang et al., 2022), the external dataset is not used as in other baselines, the parameter of top-k nearest neighbors is set to {100, 50, 500} for CLINC, BANKING, and StackOverflow, respectively, as utilized in Zhang et al. (2022). For all experiments, we set the batch size as 512 and the temperature scale as $\tau = 0.1$ in Eq. (10) and Eq. (11). We set the parameter $\rho = 0.65$ in Eq. (7), the confidence threshold $\tau_g = 0.9$ in Eq. (8). We adopt the data augmentation of random token replacement as Zhang et al. (2022). All experiments are conducted on 4 Tesla V100 GPUs and averaged over 3 runs. we split the datasets into train, valid, and test sets, and randomly select 25% of categories as unknown and only

[4] https://huggingface.co/bert-base-uncased



(a) Balanced (b) Imbalanced

Figure 9: Effects of λ_2 on ImbaNID-Bench.

10% of training data as labeled. The number of intent categories is set as ground truth.

E Estimate the Number of Intents (K)

In practical dialogue systems, new intents emerge constantly and we cannot know the exact number of the intent clusters. In this paper, following the work of (Zhang et al., 2021b), we take the full usage of the well-initialized intent features to automatically estimate the intent cluster number K. Specifically, we first assign a big K' as the initial intent cluster number. Then we directly use the pre-trained model to extract the feature representations for the training data and perform the K-means algorithm to group these feature representations into different clusters. From these clusters, we can distinguish the dense and boundary-clear clusters as the real intent clusters, while the remaining low-size clusters are filtered out. The filtering function can be formulated as follows:

$$K = \sum_{i=1}^{K'} \delta\left(|T_i| \geq t\right) \tag{25}$$

where $|T_i|$ is the size the i_{th} grouped cluster, t is the threshold of filtering. $\delta(\cdot)$ is the indicator function, whose output is 1 if the condition is satisfied.

F Hyper-Parameter Analyses

To investigate the sensitiveness of the hyper-parameters in Eq. 5, we first referred to the experience from previous studies (Asano et al., 2020; Caron et al., 2020b) and identified $\lambda_1 = 0.05$ on the all datasets. Then we examine the impact of λ_2 on model performance by varying the value of λ_2 to observe the performance changes. The results are reported in Fig. 9. Specifically, Fig. 9(a) shows the impact of λ_2 variation on the performance of balanced datasets, while Fig. 9(b) demonstrates the effect of λ_2 on the performance of imbalanced datasets. Empirically, we choose $\lambda_2 = 7$ on the balanced datasets, and $\lambda_2 = 2$ on the imbalanced ImbaNID-Bench datasets.