Generalized Category Discovery with Large Language Models in the Loop

Anonymous ACL submission

Abstract

Generalized Category Discovery (GCD) is a crucial task that aims to recognize both known and novel categories from a set of unlabeled data by utilizing a few labeled data with only known categories. Due to the lack of supervision and category information, current methods usually perform poorly on novel categories and struggle to reveal semantic meanings of the discovered clusters, which limits their applications in the real world. To mitigate above issues, we propose *Loop*, an end-to-end active-learning framework that introduces *Large Language Models (LLMs)* [1] into the training loop, which can boost model performance and generate category names without relying on any human efforts. Specifically, we first propose *Local Inconsistent Sampling (LIS)* to select samples that have a higher probability of falling to wrong clusters, based on neighborhood prediction consistency and entropy of cluster assignment probabilities. Then we propose a *Scalable Query* strategy to allow LLMs to choose true neighbors of the selected samples from multiple candidate samples. Based on the feedback from LLMs, we perform *Refined Neighborhood Contrastive Learning (RNCL)* to pull samples and their neighbors closer to learn clustering-friendly representations. Finally, we select representative samples from clusters corresponding to novel categories to allow LLMs to generate category names for them. Extensive experiments on three benchmark datasets show that *Loop* outperforms SOTA models by a large margin and generates accurate category names for the discovered clusters.

1 Introduction

Although modern machine learning systems have achieved superior performance on many tasks, the vast majority of them follow the closed-world setting that assumes training and test data are from the same set of pre-defined categories (Cao et al., 2021). However, in the real world, many practical problems such as intent detection (An et al., 2023a) and image recognition (Zhong et al., 2021a) are open-world, where the well-trained models may encounter data with unseen novel categories. To cope with this limitation, Generalized Category Discovery (GCD) was proposed and widely studied in both NLP (Zhang et al., 2021; An et al., 2023b) and computer vision fields (Vaze et al., 2022; Wen et al., 2022). GCD requires models to recognize both known and novel categories from a set of unlabelled data based on some labeled data containing only known categories, which can adapt models to the emerging categories without any human efforts.

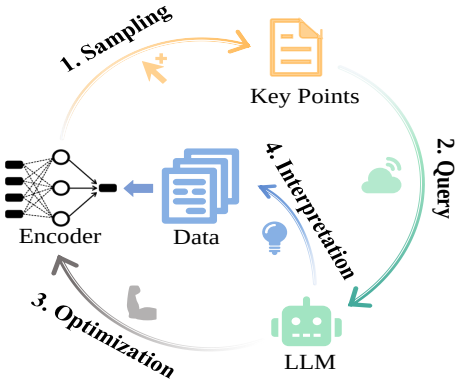


Figure 1: The training loop of our model.

Current methods (An et al., 2023b; Wen et al., 2022; Vaze et al., 2022) usually first perform supervised pretraining on labeled data and self-supervised learning on unlabeled data to train a base model such as BERT (Devlin et al., 2018), then they perform clustering methods such as KMeans to discover both known and novel categories. Even though these methods often improve performance on known categories, they usually perform poorly on novel categories due to the lack of supervision. Furthermore, they also struggle to reveal semantic meanings (e.g., category names or

[1] The LLMs can be either locally deployed models or LLM APIs. In this paper, we use OpenAI's APIs for simplicity.

descriptions) of the discovered clusters due to the lack of prior knowledge for novel categories. Recently, Large Language Models (LLMs) such as ChatGPT have shown extraordinary capabilities for various applications even without any labeled samples (Wei et al., 2023). However, LLMs cannot be directly applied to GCD which requires to cluster thousands of samples. And problems such as data privacy, high inference latency and cost also limit their applications in the real world.

To solve above limitations and enjoy the benefits of both base models and LLMs, we propose *Loop*, an end-to-end active-learning framework that introduces LLMs into the training process. By selecting a few key samples to query LLM APIs and optimize the base model based on the feedback, *Loop* can compensate for the lack of supervision and generate category names for the discovered clusters with very little query cost. Hence, we only need to train and maintain a small base model locally, which can reduce the inference cost and protect data privacy. Specifically, as shown in Fig. 1, we first propose *Local Inconsistent Sampling (LIS)* to select the most informative samples that have a higher probability of falling to wrong clusters. Specifically, we select samples that have high entropy of cluster assignment probabilities and whose neighbors have the most diverse cluster assignments. Intuitively, samples that have high entropy and diverse neighbor predictions seem to violate the clustering assumption (Jiang et al., 2022) and locate near decision boundaries (Fig. 2 dashed circle), so these neighbor-chaotic samples with great uncertainty would have a high probability of falling to wrong clusters (Wang et al., 2023), and correcting them can significantly improve the model performance.

After selecting the key samples, we need to build proper prompts to query the LLMs. However, due to the lack of information for novel categories, we cannot directly query LLMs which category these samples belong to as in traditional active learning. To solve this issue, we propose a *Scalable Query* strategy that allows LLMs to choose true neighbors of the selected samples from multiple candidate neighbor samples (Zhang et al., 2023). Based on the feedback of LLMs, we can solve the local inconsistency problem and decide which clusters these key samples truly belong to. Furthermore, LLMs are more competent at comparing semantic similarities between sentences than choosing from multiple category names. Then based on the refined neighbor relationships, we perform *Refined Neighborhood Contrastive Learning (RNCL)* to pull samples closer to their neighbors to learn clustering-friendly representations. In this way, we can correct these samples by pulling them closer to true clusters they belong to and cluster the rest of samples to form more compact clusters. Finally, we decouple the clusters corresponding to novel categories from the discovered clusters (An et al., 2023b) and select a few samples closest to each center of the clusters to query LLMs to generate category names for novel categories.

Experimental results on three benchmark datasets show that *Loop* outperforms SOTA models by a large margin and generates accurate category names for the discovered clusters. Furthermore, we also validate that the proposed *Local Inconsistent Sampling* can select more informative samples and the proposed *Scalable Query* strategy can help to correct the selected samples effectively with very little query cost.

Our contributions can be summarized as follows:

- **Perspective**: we propose to introduce LLMs into the training loop to enjoy the benefits of both base models and LLMs. To the best of our knowledge, we are the first to utilize LLMs to guide the training process of GCD.
- **Framework**: we propose *Loop*, an end-to-end active-learning framework that can select informative samples with *Local Inconsistent Sampling* and label these samples with *Scalable Query* without any human efforts.
- **Interpretation**: *Loop* can reveal semantic meanings of the discovered clusters by generating category names, which is infeasible in previous methods.
- **Experiments**: Extensive experiments on three benchmark datasets show that *Loop* outperforms SOTA models by a large margin (average 7.67% improvement) and generates accurate category names with very little query cost (average $0.4 for each dataset).

2 Related Work

2.1 Generalized Category Discovery

Under the open-world assumption, GCD (Vaze et al., 2022) requires models trained on a few labeled data with known categories to recognize both

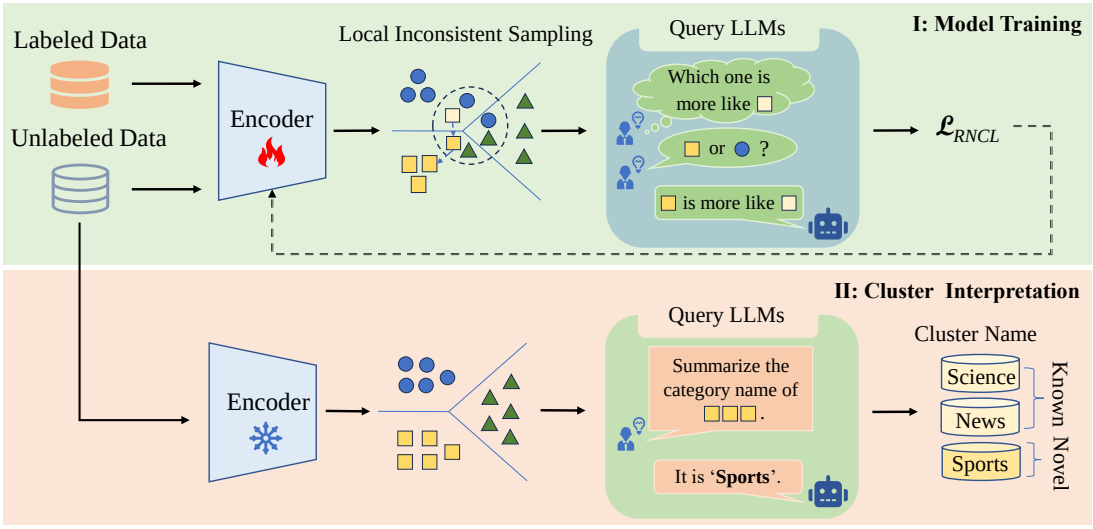


Figure 2: The overall architecture of our model.

known and novel categories from the newly collected unlabeled data. Previous methods mainly performed representation learning on unlabeled data with self-supervised learning (Vaze et al., 2022; Zhong et al., 2021b; Wen et al., 2022) or pseudo-label learning (Han et al., 2019; Ge et al., 2020). For example, Lin et al. (2020); An et al. (2023a) proposed to generate pseudo labels by clustering. Zhang et al. (2022) performed contrastive learning to learn clustering-friendly representations. An et al. (2023b) proposed to decouple known and novel categories with a prototypical network.

2.2 Active Learning

Active Learning (AL) aims to select informative samples for manual labeling to balance model performance and annotation cost. Previous methods are mainly based on Uncertainty-based (e.g., entropy (Zhang et al., 2023), confidence (Wang and Shang, 2014) and margin (Roth and Small, 2006)) or information (e.g., MHPL (Wang et al., 2023) and CAL (Margatina et al., 2021)). Recently, Zhang et al. (2023); Cheng et al. (2023) also utilized LLMs to replace human experts to save annotation cost. However, how to employ active learning with LLMs for the open-world setting has not yet been explored.

3 Method

Problem Setup. Under the open-world assumption, models trained on a labeled dataset $\mathcal{D}^l = \{(x_i, y_i)|y_i \in \mathcal{Y}_k\}$ containing only known categories $\mathcal{Y}_k$ may encounter newly collected unlabeled data $\mathcal{D}^u = \{x_i|y_i \in \{\mathcal{Y}_k, \mathcal{Y}_n\}\}$ with both known categories $\mathcal{Y}_k$ and novel categories $\mathcal{Y}_n$, which can make a model fail. To cope with this challenge, Generalized Category Discovery (GCD) requires a model to recognize both known and novel categories based on $\mathcal{D}^{all} = \mathcal{D}^l \cup \mathcal{D}^u$, without any annotation or category information for novel categories. Finally, model performance will be measured on a testing set $\mathcal{D}^t = \{(x_i, y_i)|y_i \in \{\mathcal{Y}_k, \mathcal{Y}_n\}\}$.

Framework Overview. As shown in Fig. 2, there are two stages in the proposed *Loop* framework. In the first stage, we introduce LLMs to guide the base model to learn better representations. Specifically, we first pre-train the base model for warm up (Sec. 3.1). Then we select informative samples for annotation based on *Local Inconsistent Sampling* (Sec. 3.2). Next, we construct prompt with the *Scalable Query* strategy to query LLMs to acquire correct neighborhood relationships between samples (Sec. 3.3). Finally, we perform *Refined Neighborhood Contrastive Learning* to learn clustering-friendly representations based on the feedback of LLMs (Sec. 3.4). In the second stage, we interpret the discovered clusters by decoupling and generating category names for novel categories (Sec. 3.5).

3.1 Multi-task Pre-training

We use the lightweight language model BERT (Devlin et al., 2018) as the base model to extract features $z_i = F_\theta(x_i)$ for the input sentence x_i. To quickly adapt the base model to current tasks, we pre-train F_θ on both labeled and unlabeled data in a multi-task manner (Zhang et al., 2022) with Cross-Entropy (CE) loss and Masked Language

Modeling (MLM) loss (Devlin et al., 2018):

$$\mathcal{L}_{pre} = \mathcal{L}_{ce}(\mathcal{D}^l) + \mathcal{L}_{mlm}(\mathcal{D}^{all}) \quad (1)$$

Through pretraining, F_θ can acquire both category-specific knowledge and general knowledge from data, which can provide a good representation initialization for subsequent training.

3.2 Local Inconsistent Sampling

To select informative samples that have a higher probability of falling to wrong clusters, we propose *Local Inconsistent Sampling (LIS)* to select samples that make different predictions from their neighbors and have high prediction entropy.

Specifically, we first perform Kmeans clustering on $\mathcal{D}^{all}$ to calculate cluster centers $\{\mu_i\}_{i=1}^K$ and get pseudo labels $\{\hat{y}_j\}_{j=1}^N$ for all data based on cluster assignments, where $K = |\mathcal{Y}_k| + |\mathcal{Y}_n|$ is the number of categories and $N = |\mathcal{D}^{all}|$ is the number of samples. We assume K is known for a fair comparison and estimate it in Sec. 5.5. Then for each feature z_i, we search its k-nearest neighbors in the feature space and denote $\mathcal{N}_i$ as the index set of the retrieved neighbors:

$$\mathcal{N}_i = \underset{j}{argtop_k}\{sim(z_i, z_j)|j = 1, ..., N\} \quad (2)$$

where $sim()$ is the cosine similarity function $sim(z_i, z_j) = \frac{z_i^T z_j}{\|z_i\| \cdot \|z_j\|}$. According to the clustering assumption (Jiang et al., 2022), samples that are close to each other in the feature space should have the same predictions, so samples with local inconsistent predictions are near decision boundaries and have a higher probability of falling to wrong clusters (dashed circle in Fig. 2). We calculate the local inconsistency degree $\mathcal{C}_i$ by counting the number of neighbors that have different pseudo labels from the query:

$$\mathcal{C}_i = \sum_{j=1}^{k} |\hat{y}_i \neq \hat{y}_{\mathcal{N}_i^j}| \quad (3)$$

where $\mathcal{N}_i^j$ is the index of the j-th neighbor of x_i.

To further select uncertain samples that are far away from cluster centers and near decision boundaries, we also restrict that the selected samples should have high prediction entropy. Specifically, we model the probability that samples belong to different clusters with Student's t-distribution (Xie et al., 2016):

$$q_{ij} = \frac{(1 + \|z_i - \mu_j\|^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{j'}(1 + \|z_i - \mu_{j'}\|^2/\alpha)^{-\frac{\alpha+1}{2}}} \quad (4)$$

where α is the degree of freedom. The entropy can be calculated as:

$$\mathcal{H}_i = -\sum_j q_{ij} log(q_{ij}) \quad (5)$$

Then we can select a set of informative samples that have both high local inconsistency degree and prediction entropy.

$$S = \{z_i \mid \mathcal{C}_i \in top_m(\mathcal{C}) \wedge \mathcal{H}_i \in top_m(\mathcal{H})\} \quad (6)$$

where $\mathcal{C} = \{\mathcal{C}_j\}_{j=1}^N$ and $\mathcal{H} = \{\mathcal{H}_j\}_{j=1}^N$ are the set of local inconsistency degree and the set of prediction entropy for each sample, respectively. m is a hyperparameter that determines the number of samples to be selected.

Discussion. The proposed *LIS* is effective in two aspects. First, the local inconsistency degree can help to select samples whose neighbors have the most diverse cluster assignments. Since these neighbor-chaotic samples may locate near decision boundaries and violate the clustering assumption (Jiang et al., 2022), it will be hard for the model to decide which clusters they truly belong to. Second, the prediction entropy can select samples that are distributed uniformly among several clusters. Since these samples are far away from cluster centers and distributed near decision boundaries, they can be easily assigned into wrong clusters. By combing the two scores together, our model can select samples that are assigned to wrong clusters, and correcting these samples can provide more gains in improving model performance (Sec. 5.2.2).

3.3 Scalable Query Strategy

Given the selected samples, the next step is how to query LLMs to get proper supervision information. However, we cannot directly query LLMs for categories because there is no label information for novel categories and the returned categories are hard to be aligned with the cluster assignments. So inspired by recent work (Zhang et al., 2023), we propose a *Scalable Query* strategy to mitigate the local inconsistent issue by querying LLMs which samples are the true neighbors of the selected samples. In this way, we can find the true cluster assignments of the selected samples by determining the neighborhood relationship between samples. This query strategy is scalable since we can set a different number of neighbor options for LLMs to choose from. Taking the query with $|q|$ options as an example, the prompt can be designed as: "Select

the sentence that better corresponds with the query sentence. Query: [S]. Sentence 1: [S_1]; Sentence 2: [S_2]; ...; Sentence $|q|$: [$S_{|q|}$].", where [S] is the selected query sample and [S_1], [S_2] ... [$S_{|q|}$] are neighbor sentences of [S] from the top $|q|$ clusters that have the most neighbors of the query sample.

Discussion. The proposed query strategy can help to correct the local inconsistent samples by selecting their true neighbors from the chaotic neighborhood. This strategy is scalable since we can add different number of options to query LLMs. Although adding more options will provide a higher probability to select the sample that is from the same category as the query, it will increase the query cost by adding more query tokens (Sec. 5.2.3). Even if we do not find the true neighbor samples, our model can still learn semantic knowledge by pulling similar samples closer.

3.4 Refined Neighborhood Contrastive Learning

Based on the feedback of LLMs, we can refine the neighborhood relationships between samples. For the unselected samples, we randomly select a sample from their neighbors to enhance generalization of our model. Then we can correct the selected samples and learn clustering-friendly representations by pulling samples closer to their neighbor samples with neighborhood contrastive learning (Zhong et al., 2021b):

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} log\frac{exp(\mathcal{A}^T(z_i)\mathcal{A}(z_{\mathcal{N}_i^s})/\tau)}{\sum_{z_j\in\mathcal{B}} exp(\mathcal{A}^T(z_i)\mathcal{A}(z_j)/\tau)} \tag{7}$$

where $\mathcal{A}$ is a data augmentation method, $\mathcal{N}_i^s$ is the index of the selected neighbor of z_i, τ is a hyper-parameter and $\mathcal{B}$ is the current batch. We also add cross-entropy loss $\mathcal{L}_{ce}(\mathcal{D}^l)$ for training to enhance our model performance on known categories.

3.5 Cluster Interpretation

Different from previous work that only performed clustering to discover clusters without any semantic information, we propose to interpret the discovered clusters with the help of LLMs. Specifically, we first utilize the *'Alignment and Decoupling'* strategy (An et al., 2023b) to decouple clusters that correspond to novel categories from the discovered clusters. Then for each decoupled cluster, we select a few samples that are closest to the center of the clusters as representative samples. Next we make LLMs to summarize these samples to generate label names for these novel categories. Experimental results show that this strategy can select representative samples and generate accurate label names for the discovered novel categories (Sec. 5.3).

3.6 Resource Saving

By selecting the most informative samples and reducing the query options, our framework can reduce query cost. To further reduce the computing and query cost, we propose two strategies for our model training.

Interval Update. Since the neighborhood relationships between samples will not change dramatically, we query LLMs and update the neighborhood relationships every a few epochs (5 in our experiments). In this way, we can save the computing resource of neighborhood retrieval and the cost of querying LLMs.

Query Result Storage. Since we may query LLMs for the same sample repeatedly in different epochs, we maintain a dictionary to store the query results to avoid duplicated queries. In this way, we can reuse the query results and reduce the cost of queries.

4 Experiments

4.1 Experimental Setup

4.1.1 Datasets

We perform experiments on three benchmark datasets. **BANKING** (Casanueva et al., 2020) is an intent detection dataset in the bank domain. **StackOverflow** Xu et al. (2015) is a question classification dataset. **CLINC** (Larson et al., 2019) is an intent detection dataset from multiple domains. For each dataset, we randomly select 25% categories as novel categories and 10% data as labeled data. More details are listed in Appendix A.1.

4.1.2 Comparison with SOTA Methods

We compare our model with various baselines and SOTA methods.

Unsupervised Models. (1) DeepCluster (Caron et al., 2018). (2) DCN (Yang et al., 2017). (3) DEC (Xie et al., 2016). (4) KM-BERT (Devlin et al., 2018). (5) AG-GloVe (Gowda and Krishna, 1978). (6) SAE (Liu et al., 2018).

Semi-supervised Models. (1) Simple (Wen et al., 2022). (2) Semi-DC (Caron et al., 2018). (3) Self-

Method	BANKING			StackOverflow			CLINC		
	H-score	Known	Novel	H-score	Known	Novel	H-score	Known	Novel
DeepCluster	13.97	13.94	13.99	19.10	18.22	14.80	26.48	27.34	25.67
DCN	16.33	18.94	14.35	29.22	28.94	29.51	29.20	30.00	28.45
DEC	17.82	20.36	15.84	25.99	26.20	25.78	19.78	20.18	19.40
KM-BERT	21.08	21.48	20.70	16.93	16.67	17.20	34.05	34.98	33.16
AG-GloVe	30.47	29.69	31.29	29.95	28.49	31.56	44.16	45.17	43.20
SAE	37.77	38.29	37.27	62.65	57.36	69.02	45.74	47.35	44.24
Simple	40.52	49.96	34.08	57.53	57.87	57.20	62.76	70.60	56.49
Semi-DC	47.40	53.37	42.63	64.90	63.57	61.20	73.41	75.60	71.34
Self-Label	48.19	61.64	39.56	59.99	78.53	48.53	61.29	80.06	49.65
CDAC+	50.28	55.42	46.01	75.78	77.51	74.13	69.42	70.08	68.77
DTC	52.13	59.98	46.10	63.22	80.93	51.87	68.71	82.34	58.95
Semi-KM	54.83	73.62	43.68	61.43	81.02	49.47	70.98	89.03	59.01
DAC	54.98	69.60	45.44	63.64	76.13	54.67	78.77	89.10	70.59
GCD	55.78	75.16	44.34	64.63	82.00	53.33	63.08	89.64	48.66
PTJN	60.69	77.20	50.00	77.48	72.80	82.80	83.34	91.79	76.32
DPN	60.73	80.93	48.60	83.13	85.29	81.07	84.56	92.97	77.54
MTP	61.59	80.08	50.04	77.23	84.75	70.93	80.32	91.69	71.46
***Loop* (Ours)**	**74.60**	**83.99**	**67.10**	**91.57**	**87.56**	**90.53**	**90.74**	**94.45**	**87.31**
Improvement	+13.01	+3.06	+17.06	+8.44	+2.27	+7.73	+6.18	+1.48	+9.77

Table 1: Model comparison results (%) on testing sets. Average results over 3 runs are reported. Some results are cited from An et al. (2023b).

Labeling: (Yu et al., 2022). (4) CDAC+ (Lin et al., 2020). (5) DTC (Han et al., 2019). (6) Semi-KM (Devlin et al., 2018) (7) DAC (Zhang et al., 2021). (8) GCD (Vaze et al., 2022). (9) PTJN (An et al., 2023a). (10) DPN (An et al., 2023b). (11) MTP (Zhang et al., 2022).

4.1.3 Evaluation Metrics

We measure model performance with clustering accuracy with Hungarian algorithm (Kuhn, 1955). (1) **H-score**: harmonic mean of the accuracy of known and novel categories (Saito and Saenko, 2021). (2) **Known**: accuracy of known categories. (3) **Novel**: accuracy of novel categories.

4.1.4 Implementation Details

We use the pre-trained Bert-base-uncased model (Wolf et al., 2019) as the base model and the GPT-3.5 Turbo API as the LLM. For hyper-parameters, k is set to {50, 50, 500} for BANKING, CLINC and StackOverflow, respectively. α is set to 1, m is set to 500, $|q|$ is set to 2 and τ is set to 0.07. The pre-training epoch is set to 100 and the training epoch is set to 50 on an NVIDIA 3090 GPU. The learning rate for pretraining and training is set to $5e^{-5}$ and $1e^{-5}$, respectively. For masked language modeling, the mask probability is set to 0.15 following previous works. Random Token Replace (Zhang et al., 2022) is used for data augmentation.

5 Experimental Analysis

5.1 Main Results

We show the comparison results in Table 1. From the results we can see that our model gets the best performance on all datasets and evaluation metrics (average **7.67%** improvement), which can show the effectiveness of our model. Specifically, our model gains average **9.21%** improvement in H-score, which means that our model can better balance model performance on known and novel categories and alleviate the effects of model bias towards known categories. Average **2.27%** improvement in accuracy of known categories shows that our model can acquire semantic knowledge from both labeled and unlabeled data to enhance our model performance. Last but not least, our model gains average **11.52%** improvement in accuracy of novel categories. We attribute the remarkable improvement to following reasons. First, *Local In-*

Model	H-score	Known	Novel
Loop (Ours)	74.60	83.99	67.10
w/o $\mathcal{L}_{ce}$	72.77	82.43	65.13
w/o LLMs	70.02	78.15	63.42
w/ Entropy	74.06	84.07	66.18
w/ Margin	72.88	82.73	65.13
w/ Random	72.33	82.23	64.56
w/ Confidence	72.08	82.44	64.03
$\|q\| = 3$	75.91	84.25	69.08
$\|q\| = 4$	77.30	83.84	71.71
OverClustering	74.07	80.54	68.56

Table 2: Ablation study with different model variants.

Strategy	BANK.	Stack.	CLINC
Random	33.00	20.50	17.50
Margin	77.00	68.50	68.50
Entropy	80.00	84.50	61.00
Confidence	81.00	78.00	66.50
***LIS* (Ours)**	**88.48**	**90.97**	**72.25**
Improvement	+7.48	+6.47	+3.75

Table 3: Proportion of the selected 200 samples that fall into wrong clusters.

$\|q\|$	2	3	4
Cost ($)	0.39	0.47	0.55

Table 4: Query cost with different number of options.

consistent Sampling can help to select samples that have a higher probability of falling to the wrong clusters. And correct them can provide more information gain for the model training. Second, *Scalable Query* can provide supervision by choosing the true neighbors, which can help to mitigate the local inconsistency problem. Last, *Neighborhood Contrastive Learning* with the refined neighbors can help to pull samples from the same category closer and learn clustering-friendly representations.

5.2 Ablation Study

We validate the effectiveness of different components of our model on the BANKING dataset in Table 2.

5.2.1 Main Components

From the results we can see that removing cross-entropy loss $\mathcal{L}_{ce}$ can lead to slight performance degradation since it is responsible for providing accurate supervision for known categories. And removing feedback from LLMs will lead to severe performance decline on both known and novel categories, which can reflect the importance of introducing LLMs to the training loop to provide supervision information.

5.2.2 Analysis of *LIS*

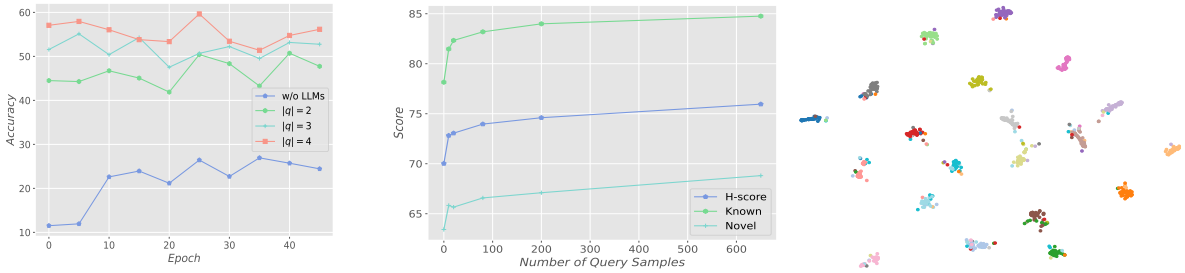
To validate the proposed *Local Inconsistent Sampling (LIS)* strategy, we compare the model performance with different sampling strategies. As shown in Table 2, *LIS* outperforms other sampling strategies, which demonstrates the effectiveness of our *LIS* strategy. To further validate the proposed *LIS* strategy, we also compare the accuracy of different strategies for selecting samples that fall into wrong clusters. From Table 3 we can see that *LIS* outperforms other strategies by a large margin, which means that *LIS* can select more informative samples to boost our model performance.

5.2.3 Analysis of *Scalable Query*

To validate the *Scalable Query* strategy, we compare the model performance with different number of options $|q|$. As shown in Table 2, Increasing $|q|$ can improve our model performance because we can select more accurate neighbors as $|q|$ grows. As shown in Fig. 3(a), our scalable query strategy can correct many samples compared to the method without LLM queries, which shows the effectiveness of our query strategy. And with $|q|$ increasing, our model can correct more samples and get better model performance, which shows the scalability of our query method. However, the query cost will increase with the growth of $|q|$ due to the growth of query tokens (Table 4), so the *Scalable Query* strategy provides users with options to balance query cost and model performance.

5.3 Cluster Interpretation

In addition to the improved model performance, our model can also interpret the discovered clusters by generating category names for them. As shown in Table 5, our model can select representative samples for novel categories and generate accurate names for them, which can provide more convenience for real-world applications of our model. More results are listed in Appendix A.7.



(a) Accuracy of the corrected samples. (b) Effect of the number of query samples. (c) t-SNE Visualization.

Figure 3: Analysis of the quality of representation learning and neighborhood retrieval.

Selected Sentences	Ground Truth	Prediction
Can I change my PIN if I want to? Can I change my PIN? Do I have to change my PIN at a bank?	Change PIN	Change PIN
What will the weather be this weekend? Tell me what the weather is like. What's the weather like?	Weather	Weather forecast

Table 5: Examples of the selected sentences, ground-truth category names and predicted category names.

5.4 Influence of the Number of Samples

We investigate the influence of the number of selected samples for query in Fig. 3(b). From the results we can see that increasing the number of samples can improve our model performance. However, the growth rate gradually slows down because it becomes increasingly difficult to select informative samples as the number of samples increases.

5.5 Real-world Applications

In the real world, the number of categories K is usually unknown. To solve this issue, we utilize the filtering strategy (Zhang et al., 2021) to estimate K. As shown in Table 6, our model obtains the most accurate estimation with only a little error, which shows the effectiveness of our model. To further investigate the influence of K, we perform OverClustering by over-estimating K used for inference by a factor of 1.2. Results in Table 2 show that our model gets close performance even without knowing the ground-truth K, which validates the robustness of our model.

5.6 Visualization

We visualize the learned embeddings of our model on the Stack. dataset with t-SNE in Fig. 3(c). From the figure we can see that our model can learn separable clusters and decision boundaries for different categories, which indicates that our model can learn discriminative features for clustering.

Method	BANK.	Stack.	CLINC
Ground Truth	77	20	150
DAC	66	15	130
DPN	67	18	137
Ours	**78**	**19**	**145**

Table 6: Estimation of the number of categories.

6 Conclusion

In this paper, we propose *Loop*, an active-learning framework that introduces LLMs to the training loop for GCD, which can boost our model performance without any human efforts. We further propose *Local Inconsistent Sampling* to select informative samples and utilize *Scalable Query* to correct these samples with the feedback of LLMs. By pulling samples closer to their refined neighbors, our model can learn clustering-friendly representations. Finally, we generate label names for the discovered clusters to facilitate real-world applications. Experiments show that *Loop* outperforms SOTA models by a large margin and generates accurate category names for the discovered clusters.

Limitations

Even though the proposed *Loop* framework achieves superior performance on the GCD task, it still faces the following limitations. First, when increasing the number of samples to query LLMs, the performance of *Loop* improves slowly, which is because it becomes harder to select informative samples. So how to revise the sample selection strategy to select more informative samples is a key question. Second, the *Scalable Query* can only provide neighborhood information, which is relatively weak supervision compared to category supervision in traditional active learning. So how to design query strategy to acquire more accurate supervision is another key question. Last, *Loop* relies on the feedback of LLM APIs, which is uncontrollable, and uploading data to query LLMs may be risky for some sensitive industries.

References

Wenbin An, Feng Tian, Ping Chen, Qinghua Zheng, and Wei Ding. 2023a. New user intent discovery with robust pseudo label training and source domain joint-training. *IEEE Intelligent Systems*.

Wenbin An, Feng Tian, Qinghua Zheng, Wei Ding, QianYing Wang, and Ping Chen. 2023b. Generalized category discovery with decoupled prototypical network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12527–12535.

Kaidi Cao, Maria Brbic, and Jure Leskovec. 2021. Open-world semi-supervised learning. *arXiv preprint arXiv:2102.03526*.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149.

Inigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. *arXiv preprint arXiv:2003.04807*.

Qinyuan Cheng, Xiaogui Yang, Tianxiang Sun, Linyang Li, and Xipeng Qiu. 2023. Improving contrastive learning of sentence embeddings from ai feedback. *arXiv preprint arXiv:2305.01918*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yixiao Ge, Dapeng Chen, and Hongsheng Li. 2020. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. *arXiv preprint arXiv:2001.01526*.

K Chidananda Gowda and G Krishna. 1978. Agglomerative clustering using the concept of mutual nearest neighbourhood. *Pattern recognition*, 10(2):105–112.

Kai Han, Andrea Vedaldi, and Andrew Zisserman. 2019. Learning to discover novel visual categories via deep transfer clustering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8401–8409.

Zhen Jiang, Yongzhao Zhan, Qirong Mao, and Yang Du. 2022. Semi-supervised clustering under a "compact-cluster" assumption. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):5244–5256.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2:83–97.

Stefan Larson, Anish Mahendran, Joseph J Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K Kummerfeld, Kevin Leach, Michael A Laurenzano, Lingjia Tang, et al. 2019. An evaluation dataset for intent classification and out-of-scope prediction. *arXiv preprint arXiv:1909.02027*.

Ting-En Lin, Hua Xu, and Hanlei Zhang. 2020. Discovering new intents via constrained deep adaptive clustering with cluster refinement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8360–8367.

Guifang Liu, Huaiqian Bao, and Baokun Han. 2018. A stacked autoencoder-based deep neural network for achieving gearbox fault diagnosis. *Mathematical Problems in Engineering*, 2018:1–10.

Katerina Margatina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras. 2021. Active learning by acquiring contrastive examples. *arXiv preprint arXiv:2109.03764*.

Dan Roth and Kevin Small. 2006. Margin-based active learning for structured output spaces. In *Machine Learning: ECML 2006: 17th European Conference on Machine Learning Berlin, Germany, September 18-22, 2006 Proceedings 17*, pages 413–424. Springer.

Kuniaki Saito and Kate Saenko. 2021. Ovanet: One-vs-all network for universal domain adaptation. In *Proceedings of the ieee/cvf international conference on computer vision*, pages 9000–9009.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. 2022. Generalized category discovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7492–7501.

Dan Wang and Yi Shang. 2014. A new active labeling method for deep learning. In *2014 International joint conference on neural networks (IJCNN)*, pages 112–119. IEEE.

Fan Wang, Zhongyi Han, Zhiyan Zhang, Rundong He, and Yilong Yin. 2023. Mhpl: Minimum happy points learning for active source free domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20008–20018.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, et al. 2023. Zero-shot information extraction via chatting with chatgpt. *arXiv preprint arXiv:2302.10205*.

Xin Wen, Bingchen Zhao, and Xiaojuan Qi. 2022. A simple parametric classification baseline for generalized category discovery. *arXiv preprint arXiv:2211.11727*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487. PMLR.

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*, pages 62–69.

Bo Yang, Xiao Fu, Nicholas D Sidiropoulos, and Mingyi Hong. 2017. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *international conference on machine learning*, pages 3861–3870. PMLR.

Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. 2022. Self-labeling framework for novel category discovery over domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Hanlei Zhang, Hua Xu, Ting-En Lin, and Rui Lyu. 2021. Discovering new intents with deep aligned clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Yuwei Zhang, Zihan Wang, and Jingbo Shang. 2023. Clusterllm: Large language models as a guide for text clustering. *arXiv preprint arXiv:2305.14871*.

Yuwei Zhang, Haode Zhang, Li-Ming Zhan, Xiao-Ming Wu, and Albert Lam. 2022. New intent discovery with pre-training and contrastive learning. *arXiv preprint arXiv:2205.12914*.

Dataset	$\lvert\mathcal{Y}_k\rvert$	$\lvert\mathcal{Y}_n\rvert$	$\lvert\mathcal{D}^l\rvert$	$\lvert\mathcal{D}^u\rvert$	$\lvert\mathcal{D}^t\rvert$
BANK.	58	19	673	8,330	3,080
Stack.	15	5	1,350	16,650	1,000
CLINC	113	37	1,344	16,656	2,250

Table 7: Statistics of datasets. $\lvert\mathcal{Y}_k\rvert$, $\lvert\mathcal{Y}_n\rvert$, $\lvert\mathcal{D}^l\rvert$, $\lvert\mathcal{D}^u\rvert$ and $\lvert\mathcal{D}^t\rvert$ represent the number of known and novel categories, labeled, unlabeled and testing data, respectively.

Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. 2021a. Neighborhood contrastive learning for novel class discovery. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10867–10875.

Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. 2021b. Neighborhood contrastive learning for novel class discovery. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10867–10875.

A Appendix

A.1 Datasets

To validate the effectiveness of our *Loop* framework, we perform experiments on three benchmark datasets **BANKING** (Casanueva et al., 2020), **StackOverflow** (Xu et al., 2015) and **CLINC** (Larson et al., 2019). For each dataset, we randomly select 25% categories as novel categories and then select 10% data from each known category as labeled data. After training, we test model performance on the testing set in an inductive manner. We also perform experiments with different known category ratios in Sec. A.3. Statistics of the datasets are listed in Table 7.

A.2 Prompt Design

Query Prompt. Following Zhang et al. (2023), we design the query prompt as follows:

"Select the sentence that better corresponds with the query sentence in terms of intents or categories. Please respond with 'Sentence 1' or 'Sentence 2' ... or 'Sentence $|q|$' without explanation.

Query: $[S]$. Sentence 1: $[S_1]$; Sentence 2: $[S_2]$; ...; Sentence $|q|$: $[S_{|q|}]$."

Interpretation Prompt. To generate category names for the discovered clusters that correspond to novel categories, we select three samples that are closest to the center of the clusters as representative samples. And we design the interpretation prompt as follows:

"Given the following sentences, return a word or a phrase to summarize the common intent or category of these sentences without explanation.

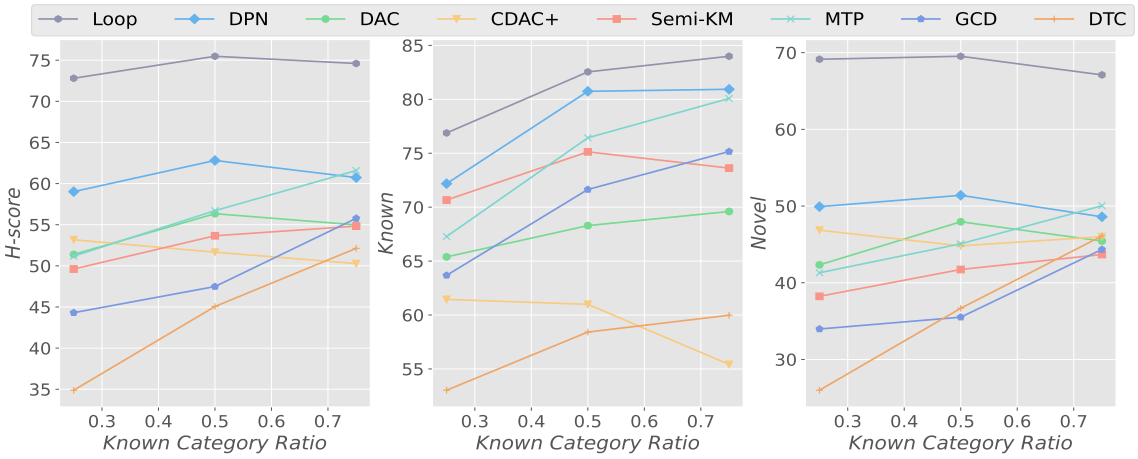


Figure 4: Model performance with different known category ratios.

Sentence 1: [S_1]; Sentence 2: [S_2]; Sentence 3: [S_3]."

A.3 Influence of the Known Category Ratio

In the real world, the ratio of known categories may vary in different applications and the number of novel categories may exceed the number of known ones. To validate the robustness of our model towards the changing known category ratios, we perform experiments with known category ratios in the set $\{25\%, 50\%, 75\%\}$ on the BANKING dataset. As shown in Fig. 4, our model gets the best performance on all known category ratios and evaluation metrics, which can show the effectiveness and robustness of our model towards different known category ratios. Furthermore, our model outperforms other methods by a large margin on the accuracy of novel categories and H-score, which can further validate that our *Loop* framework can learn better representations based on the feedback of LLMs.

A.4 Influence of the Number of Neighbors

To investigate the influence of the number of neighbors k, we perform experiments with k in the set $\{25, 50, 100, 150, 200\}$ on the BANKING dataset. As shown in Fig. 5, our model gets the similar performance when k is less than 100. However, when k exceeds 100 by a lot, our model performance drops quickly. This is because when k exceeds the average number of samples for each category by a lot (e.g., approximately 110 for the BANKING dataset), there is a higher probability for neighborhood contrastive learning to randomly select samples from other categories as the positive key, which can introduce much noise for model training and degrade the model performance.

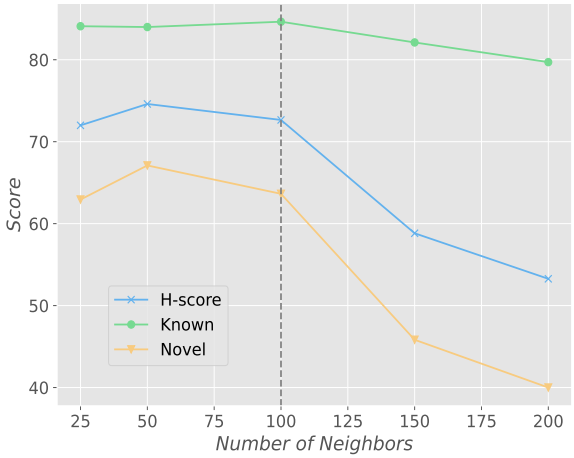


Figure 5: Model performance with different number of neighbors.

A.5 More Results of Feature Visualization

We also visualize the learned embeddings by previous SOTA methods (MTP and DPN) on the StackOverflow dataset in Fig. 6. Compared to the visualization results of our model in Fig. 3(c), we can see that some clusters are mixed together for the compared methods, which can indicate that our model can learn more discriminative features and form more separatable decision boundaries for different categories. Furthermore, if we remove the feedback of LLMs from our model (*Loop* w/o LLM query), clusters corresponding to novel categories will be mixed together due to the lack of supervision, which can further validate the effectiveness of our active-learning framework.

(a) MTP. (b) DPN. (c) *Loop* w/o LLM query.

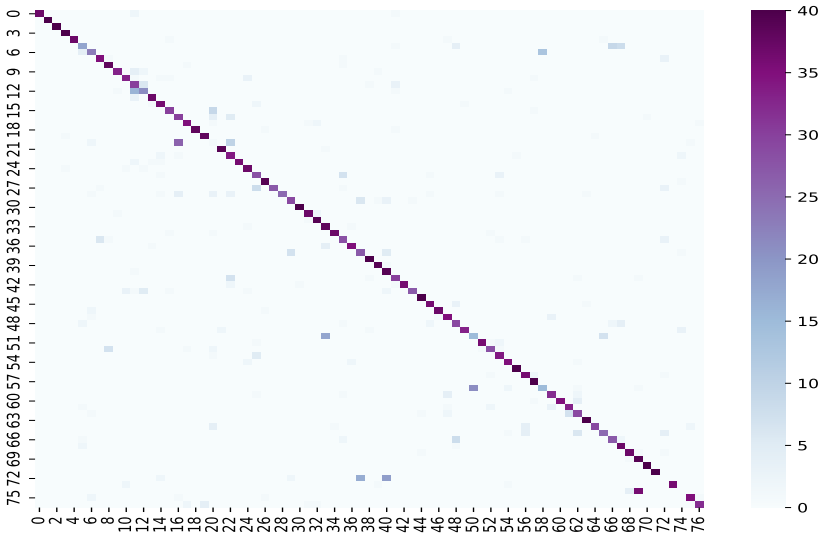
Figure 6: t-SNE Visualization for the compared methods.

A.6 Visualization for Confusion Matrix

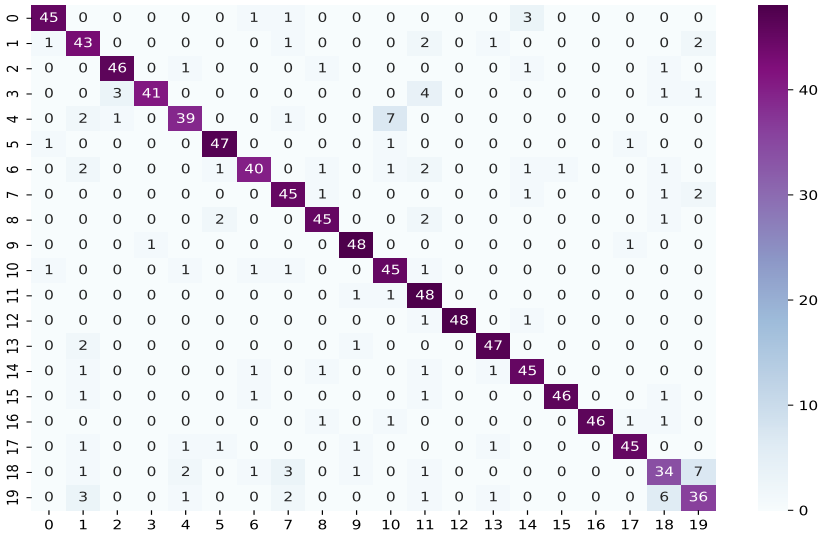
To investigate the performance of our model on different categories, we illustrate the confusion matrix of our model on the three datasets in Fig. 7. From the figure we can see that our model can make good distinctions for most of categories. However, our model still needs to be improved for some fine-grained categories that can be easily confused and misclassified (e.g., some categories of the BANKING dataset).

A.7 More Results of Cluster Interpretation

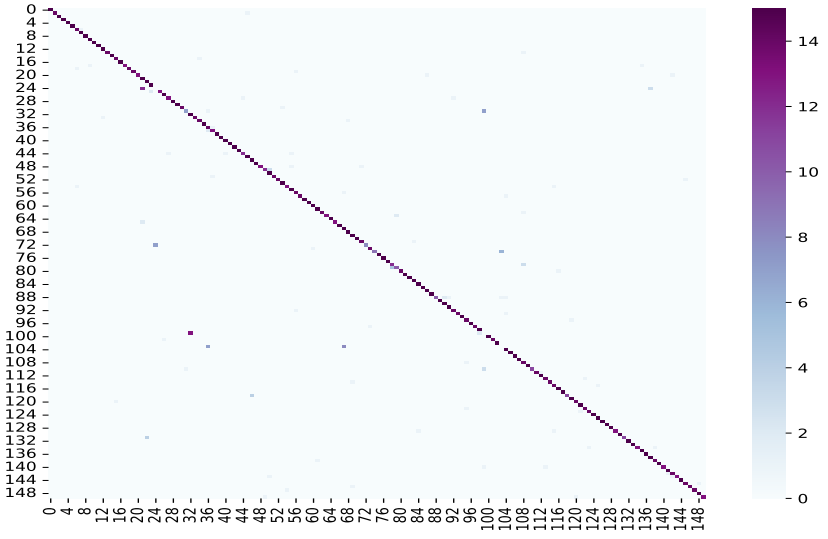
We provide more examples of the selected sentences and generated category names on the three datasets in Table 8. The results can further validate that our model can select representative samples and generate accurate names for the discovered novel categories, which can validate the effectiveness of our interpretation strategy.



(a) BANKING.



(b) StackOverflow.



(c) CLINC.

Figure 7: Confusion matrix on different datasets.

Selected Sentences	Ground Truth	Prediction
Can I top up with check? Where do I find how to top off with a check? Can I top up my account with a check?	Top up by cash or check	Top up with a check
Why is my cash withdrawal still showing as pending? My cash withdrawal is showing as pending, why? My cash withdrawal shows as pending still.	Pending cash withdrawal	Pending cash withdrawal
When can I expect a transfer from Europe to go through? When will my transfer arrive from Europe? When will I receive a transfer from Europe?	Transfer timing	Transfer timing from Europe
Look up the calories in an apple. What's the amount of calories in a cheesy omelette? Look up the calories in Cheetos.	Calories	Calorie information for food
Tell me how much my state taxes amount to. What is the amount of my state taxes? What is the amount of my federal taxes?	Taxes	Tax amount
Has my recent order shipped? Is my order already here? What is the status of my delivery order?	Order status	Order status
How to convert excel sheet column names? Setup an excel template. How do you prevent printing dialog of excel?	Excel	Excel functionalities and operations
How to pass URL variables into a WordPress page? Get three posts before a certain date in WordPress. Where to place a query to show posts in wordpress?	WordPress	WordPress Customization
Get order increment id in magento. How to get the attribute group in magento? How to get store information in magento?	Magento	Magento functionality

Table 8: Examples of the selected sentences, ground-truth category names and predicted category names.