Multi-View Zero-Shot Open Intent Induction from Dialogues: Multi Domain Batch and Proxy Gradient Transfer

Hyukhun Koh[1], **Haesung Pyun**[2], **Nakyeong Yang**[1], **Kyomin Jung**[1]
Seoul National University[1], Hanyang University[2]
{hyukhunkoh-ai, yny0506, kjung}@snu.ac.kr
haesung.pyun@gmail.com

Abstract

In Task Oriented Dialogue (TOD) system, detecting and inducing new intents are two main challenges to apply the system in the real world. In this paper, we suggest the semantic multi-view model to resolve these two challenges: (1) SBERT for General Embedding (GE), (2) Multi Domain Batch (MDB) for dialogue domain knowledge, and (3) Proxy Gradient Transfer (PGT) for cluster-specialized semantic. MDB feeds diverse dialogue datasets to the model at once to tackle the multi-domain problem by learning the multiple domain knowledge. We introduce a novel method PGT, which employs the Siamese network to fine-tune the model with a clustering method directly. Our model can learn how to cluster dialogue utterances by using PGT. Experimental results demonstrate that our multi-view model with MDB and PGT significantly improves the Open Intent Induction performance compared to baseline systems.

1 Introduction

In developing Task Oriented Dialogue (TOD) systems, it is important to understand the user's intent, such as opening a new account in banking and reporting accidents in finance. In this circumstance, TOD system is used to capture the intents automatically within some dialogue turns, generating appropriate responses to users. Since Pretrained Language Models (PLMs) have become mainstream among the recent NLP tasks, various kinds of research to apply the PLMs to a dialogue system have appeared. (Henderson et al. (2019), Bao et al. (2020), Majumder et al. (2020), Zhao et al. (2020))

In TOD systems, one of the core tasks is intent detection, which seeks to discern user intentions from their utterances. However, most of the typical intent detection modules depend heavily on the domain to be applied. Due to the domain-lopsided characteristic, it is necessary for typical models to collect the data of the domain to be serviced and go through a cumbersome process to train a model for each domain from scratch. Besides, no standard conventions exist for defining and labeling the intents within the same domain. For example, Banking77, an intent classification dataset, has 77 intents. In contrast, Banking from DSTC 11 has 29 intents and some of the intents, "AskAboutATMFees" and "GetBranchInfo," are not included in Banking77. Hence, it is tricky to apply a model trained with Banking77 directly to Banking dataset. For example, Banking77, an intent classification dataset, has 77 intents. In contrast, Banking from DSTC 11 has 29 intents and some of the intents, "AskAboutATMFees" and "GetBranchInfo," are not included in Banking77. Hence, it is impossible to directly apply a model trained with Banking77 to Banking dataset. In addition, customers may not communicate within only predefined intent sets. Summing up, intent detection modules should induce intents from utterances beyond each domain dataset.

Previous researches utilized BERT to produce hidden representations and passed them to clustering algorithms in order to discern intents of utterances (Lin et al., 2019, Aharoni and Goldberg, 2020). However, utterances in dialogue data are usually shorter and colloquial compared to generic text where the ellipsis and omission are less likely to occur. Furthermore, BERT representations are insufficient to capture the intent information for dialogue since it is only trained to focus on learning the context of the generic text. Additional methods are required to contain the domain-specific information since the meaning of a sentence can also be interpreted in various ways depending on its domain. Therefore, some researchers have trained PLMs with dialogue data (Shen et al., 2021, Zhang et al., 2021a, Zhang et al., 2021b, Zhang et al., 2021c). However, they mostly have conducted experiments within a few-shot setting where some portion of a target dataset is available, rather than

Proceedings of The Eleventh Dialog System Technology Challenge, pages 67–80
September 11, 2023

a zero-shot setting where target domain data is not accessible. In accordance with the study by Zhang et al. (2022), they handled such problems in semi-supervised learning schemes. Zhang et al. (2022) showed that it is beneficial to train PLM with external intent detection datasets. They also proposed a contrastive learning method based on K-nearest neighbor algorithm. Nevertheless, the performance gap between few-shot and zero-shot settings has been still large.

Our whole pipeline is the same as the baseline used in DSTC 11 task2[1]. Details are presented in Section 4 Task Description. Following the baseline pipeline, we only feed some utterances with labeled as InformIntent without any modification. Similar utterances should be clustered in the hidden representation space.

In this paper, we aim to improve Open Intent Induction which requires inducing intent sets in various domains without additional training on target domain datasets (zero-shot setting). We suggest the semantic multi-view model to resolve aforementioned problem using three methods : (1) SBERT for general embedding (GE), (2) Multi Domain Batch (MDB) for dialogue domain knowledge, and (3) Proxy Gradient Transfer (PGT) for cluster-specialized semantics. By MDB and PGT methods, constructed domain-agnostic model is not necessary to update parameters further when applying to a different target domain. MDB makes a model to learn various domain knowledge by composing a batch with six groups of samples, sampling from each of the six different dataset. PGT is a way of fine-tuning with K-means to improve on the clustering capability, especially in inducing intents properly. Although K-means is an integer assignment problem that is impossible to differentiate, PGT allows a differentiable comparison of K-means labels and ground truth labels by adopting Siamese network learning process.

From our experiment, combining the General Embedding (GE) module and the MDB module showed an excellent performance. In addition, when applying the PGT method to these two modules for the clustering, the performance was improved as well. Finally, we found that Spectral clustering with all three modules (GE, MDB, PGT) works better, compared to K-means clustering. The codes are available in the github repository[2]

[1]DSTC11 Track Proposal: Intent Induction from Conversations for Task-Oriented Dialogue

[2]Multi_View_Zero_Shot_Open_Intent_Induction

This paper has three contributions:

1. We suggest the multi-view model composed with GE, MDB, PGT modules.

2. We suggest a novel method PGT to directly fine-tune a model onto a clustering. PGT contributes to better performance and deals with a non-differentiable problem.

3. Our model demonstrates excellent performance in Banking and Finance, outperforming the baseline in Open Intent Induction task.

2 Background

In this section, we will briefly cover PLMs (2.1) and explore how PLMs are applied to TOD systems (2.2). Finally, we will introduce existing clustering methods (2.3).

2.1 Pretrained Language Models

There are two streams of PLM which is trained on massive corpus, Auto-Regressive (AR) and Auto-Encoding (AE) model. GPT-3 (Brown et al., 2020) and FLAN (Wei et al., 2022) are well known for AR model using Transformer decoder which are optimized to maximize the likelihood of the next word generation given preceding tokens. In comparison, BERT (Devlin et al., 2019), a representative of AE model, is a pretrained Transformer encoder with Masked Language Modeling (MLM). BERT variant models have been introduced since BERT has shown an outstanding performance in Natural Language Understanding (NLU) tasks. RoBERTa (Liu et al., 2020) enhanced BERT with Dynamic Masking method and larger dataset. Sentence Transformer SBERT (Reimers and Gurevych, 2019) derives more semantically meaningful sentence embeddings by training a model with NLI task in the scheme of Siamese network.

Some studies tried to overcome the disadvantage of AR and AE model by combining MLM with permutation methods: XLNet (Yang et al., 2019), MPNet (Song et al., 2020). XLNet (Yang et al., 2019) proposed an autoregressive pretraining by permutating tokens to capture long dependencies. MPNet unifies MLM and permutation pretraining methods, employing a two-stream attention to capture position information.

2.2 Dialogue Systems with PLMs

Despite its universal applications of PLMs, there is a discrepancy between a general corpus and a dialogue corpus. Therefore, there are some studies which trained the PLMs on a dialogue corpus further, rather than directly applying PLMs to the TOD task. TOD-BERT (Wu et al., 2020) pretrained BERT on TOD corpus. IntentBERT (Zhang et al., 2021b) proposed regularized supervised learning and showed that it is promising in a few-shot intent detection task. Vulic et al. (2021) proposed two-stage procedures. They used a universal conversational encoder with small conversational data samples, subsequently fine-tuned an encoder for a sentence similarity task. These studies focus on pretraining with the dialogue corpus and applying it to a specific dataset.

2.3 Clustering Methods

There are several candidates when cluster task is necessary. K-means (MacQueen, 1967) clusters the inputs based on Euclidean distance. It has problems when the real clusters do not have spherical shape or have a huge imbalance in size. Furthermore, Clustering performance heavily relies on the predefined K. Agglomerative (Chidananda Gowda and Krishna, 1978), one of hierarchical clusterings which build clusters based on the proximity matrix, starts with N clusters. However, it is often deficient in robustness, sensitive to noise, and computationally expensive. DBSCAN (Ester et al., 1996) is a density-based approach, which finds every point's neighbors and identifies the core points. DBSCAN is robust to noise and outliers. Nevertheless, it is slower than K-means clustering. It is also difficult to choose distance threshold epsilon and minimum points. ITER-DBSCAN (Chatterjee and Sengupta, 2020) extends DBSCAN with label propagation technique and dialog act classification. Gaussian Mixture Model (GMM) clusters based on the probability of a mixture of multiple unknown Gaussian distributions. This method does not assume the real cluster to be a specific shape but requires high computational costs. Spectral clustering (Ng et al., 2001) is based on Ratio Cut algorithm using the Laplacian matrix. This algorithm calculates the Laplacian matrix upon the similarity matrix and obtains eigenvectors from the matrix. This method can cluster the data type of non-global shape. Nonetheless, the number of clusters is essential in advance.

3 Related Works

Studies related to OOD detection and intent discovery are very relevant to Open Intent Induction task in terms of the fact that they need to handle unseen intents. The researchers of those studies are interested in how to make a good representation and how to connect it well by using a grouping algorithm. Perkins and Yang (2019) proposed the method which utilized content view and utterance view to yield the cluster assignments. Lin and Xu (2019) introduced margin loss on Bi-Directional Long Short Term Memory (BiLSTM) (Mesnil et al., 2015) for inter-class discrimination and intra-class compactness. They applied local outlier factor (LOF) (Breunig et al., 2000) to detect unknown intents based on the nearest-neighbor's density. Chatterjee and Sengupta (2020) introduced ITER-DBSCAN which extended DBSCAN with a label propagation technique. However, the methods mentioned above require domain specific knowledge and laborious feature engineering. Also, those methods often cannot fully leverage the prior knowledge and cannot accurately capture fine-grained intents. Lin et al. (2020) leveraged PLM and limited labeled data for supervised intent classification. They used the sentence similarity and filtered out low confidence assignment which is for refining clusters. Deep Aligned Clustering (Zhang et al., 2021a) pretrained BERT model with known intent samples to inject prior knowledge. As a next step, they produced pseudo-labels by the cluster assignment for self-supervised learning. CPFT (Zhang et al., 2021c) performed a contrastive learning with MLM to learn discriminative ability on utterance semantics. Shen et al. (2021) suggested the supervised contrastive scheme with dialogue domain labeled dataset on the basis of MPNet. After that, they applied K-means clustering with estimated K based on Bayesian Optimization. Zhou et al. (2022) focused on training fine-grained discriminative semantic features to deal with OOD intents among in-domain (IND) distribution. They utilize K-nearest neighbors of IND intents sample. The limitation of those studies is that they mainly focused on a few-shot performance where the premise is that few target-related labels are available.

Chen et al. (2022) pretrained Transformer with dialogue data and introduced a domain adaptive pretraining method which shows good performance in Open Intent Discovery task. Since their domain adaptive pretraining method shows promising results, we adopt it with some variations. Zhang et al. (2022) embedded rich information for the clustering by exploiting external and internal data. They show comparable results in the setting where target-related labels are unprovided. However, there is a still big gap between a few-shot setting and a zero-shot setting.

In this paper, we suggest two learning methods to induce intents from dialogues efficiently: 1. For MDB, how to appropriately inject prior external knowledge in PLM model to be robust whenever the domain changes and 2. For PGT, how the capability of model can be boosted in terms of clustering performance. Using two methods, the performance of Banking and Finance significantly increased compared to the model exposed in a few-shot setting.

4 Task Description

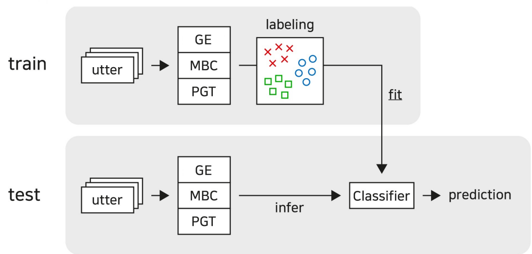

Figure 1: This is the outline of DSTC 11 task 2 Open Intent Induction. By concatenating representations of GE, MDB, and PGT modules, a clustering algorithm clusters those multi-view representations of train-utterances to construct an intent schema which consists of utterances and predicted labels. Next, we train the linear classifier with the intent schema. At test phase, multi-view representations of test-utterances go into the trained classifier to induce labels.

The second track of DSTC11 challenge is Intent Induction task based on TOD, such as Banking and Finance. This challenge aims to automatically induce intent sets for a task-oriented chatbot without training on target datasets which is a customer service interaction between the chatbot and clients.

There are three given datasets(Insurance, Banking, Finance) from 1K customer support spoken conversations, Insurance for development and others for evaluation. Each dataset has two divided data, one for training an evaluation classifier and the other for testing the performance of Open Intent Induction. The challenge assumes that utterances have only speaker roles without labels such as dialog act and intents, the number of which ranges from 5 to 50.

The task consists of three subtasks: Label Construction, Classifier Construction, and Inducing Intent. Label Construction subtask clusters the train-utterances in order to make a set of intents while predicting what the number of intents is. Classifier Construction subtask is for training a classifier with train-utterances and predicted labels which are from Label Construction subtask. Inducing Intent subtask predicts the labels for test-utterances based on a trained classifier.

4.1 Problem Formulation

$$\begin{aligned} \{\, d_1 \cdots d_n \,\} \rightarrow \{\, u_1 \cdots u_m \,\}\,, \\ d_i \subset D \; and \; u_i \subset U \end{aligned} \tag{1}$$

According to equation (1), we denote a dialogue set as $\mathbf{D} = \{\mathbf{d_1} \cdots \mathbf{d_n}\}$, where $\mathbf{d_i}$ represents the i-th utterance in a given dialogue, n is the total number of the dialogue sets. Extracted utterances denote as $\mathbf{U} = \{\mathbf{u_1} \cdots \mathbf{u_m}\}$, where $\mathbf{u_i}$ represents the i-th candidate utterance which seems to have intent, labeled as InformIntent, and m is the number of utterances selected from $\mathbf{D}$. Each single turn utterance becomes an input to the model and $\mathbf{U}$ may have some noises containing utterances without intents.

For Label Construction subtask, K-means attaches labels to train-utterances on hidden representation spaces that our model(GE-MDB-PGT) generates. For Classifier Construction subtask, train-utterances and their labels are used to train Linear Regression classifier. For Inducing Intent subtask, test-utterances sequentially go into our model and trained classifier to induce intents. The whole pipeline of DSTC 11 task 2 is in Figure 1.

5 Methods

In this section, we cover our methodology in detail. We adopt the baseline model structure of the DSTC 11 challenge as it is, but we investigate how the performance can be boosted by further learning. Our goal is for the model to automatically induce and generate meaningful intents from a collection of unlabeled utterances. SBERT, the baseline model proposed by DSTC 11, is used as the backbone.

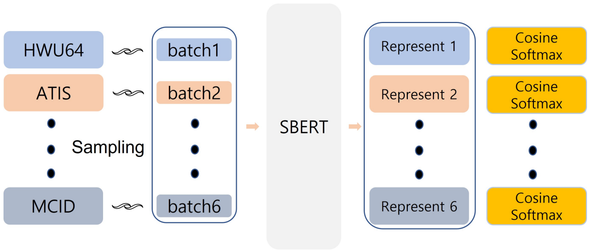

Figure 2: It shows the procedure of MDB to learn how to make appropriate domain-agnostic hidden representations. Multi-domain samples from each dataset construct the total batch (HWU64, ATIS, ..., MCID). Next, SBERT converts the total batch to hidden representations. Finally, the loss is calculated by applying cosine softmax for samples of each dataset.

5.1 Multi-View Model

Utterances from **U** enter into our three modules for taking the different perspectives into account, which is a multi-view approach. By passing U through them, we get each of three hidden representations of utterance.

$$\begin{aligned} Z_1 &= GE(U) \\ Z_2 &= MDB(U) \\ Z_3 &= PGT(U) \\ H &= Concat\,\{Z_1, Z_2, Z_3\} \end{aligned} \tag{2}$$

Hidden representation Z_1 is drawn using the General Embedding(GE) module, which interprets utterances in terms of a general linguistic perspective. Z_2 is created through a module learned by MDB to extract the dialogue-viewed representation. Hidden representation Z_3 is induced by the PGT module for tuning an encoder and a clustering module in order to boost the performance. H is constructed by concatenating all those hidden representations.

In the last step, K-means or Spectral clustering groups the concatenated representations to make intent schema by Bayesian Optimization (BO). BO method samples the number of clusters from a uniform distribution between 5 to 50 and estimates the optimal K based on the silhouette score. In the test process, Hungarian algorithm maps predicted labels to reference labels in order to evaluate the performance.

5.2 Multi Domain Batch Cosine Softmax (MDB)

As mentioned earlier, SBERT (the baseline of DSTC 11) is not enough to make a proper representation of the dialogue domain due to the discrepancy between the generic corpus and the dialogue corpus. Therefore, training PLM with a dialogue corpus is indispensable. In that sense, Multi Domain Batch (MDB) proved to be efficient in Chen et al. (2022). Unlike Chen et al. (2022), we directly apply MDB to PLM with some variations. The Figure 2 explains the process step by step. MDB uses six external datasets which are publicly available. In the process of MDB training, six different dialogue datasets configure the batch. For example, if the batch size is 36, the model would convert six samples for one dataloader to hidden representations where samples for one dataset are independent of the other datasets. The last layer of MDB is cosine softmax which is already proven to be more advantageous than naive softmax in Chen et al. (2022). We add a term τ for a numerical and denominator part.

$$j^{th}\ Cosine\ Softmax = \frac{exp(\bar{h_j}^T \bar{w}^k/\tau)}{\sum_i^{L^k} exp(\bar{h_i}^T \bar{w}^k/\tau)}, \tag{3}$$

where k is the k^{th} dataset, $\bar{h}_i$ for h_i divided by norm(h_i), $\bar{w}^k$ for k^{th} dataset's linear normalized weight, τ for the temperature, and L^k for the k^{th} dataset's number of intent categories.

The benefit of this methodology is that we do not have to preprocess collected datasets to unify the intent category sets. It saves lots of time to merge the datasets because different domains have different intent label categories, and even the same domain has different ways of defining intent labels. Next, the model can acquire multiple dialogue domain knowledge at once, which is not dependent on a specific dataset. It indicates that we no longer train the model further when applying the model to other TOD industrial fields. Besides, as well known fact that catastrophic forgetting could occur when sequentially training the model, MDB prevents this problem by parallelly training which is much more efficient in terms of learning time.

5.3 Proxy Gradient Transfer(PGT)

In Open Intent Induction task, the model induces a set of intents by clustering methods. Therefore, the model needs to know how to cluster well from the perspective of clustering methods. In that sense, Proxy Gradient Transfer(PGT) is a way of learning how to interpret utterances in terms of clustering methods, grounded on the MDB module. Since K-means self-supervised learning is proven to be

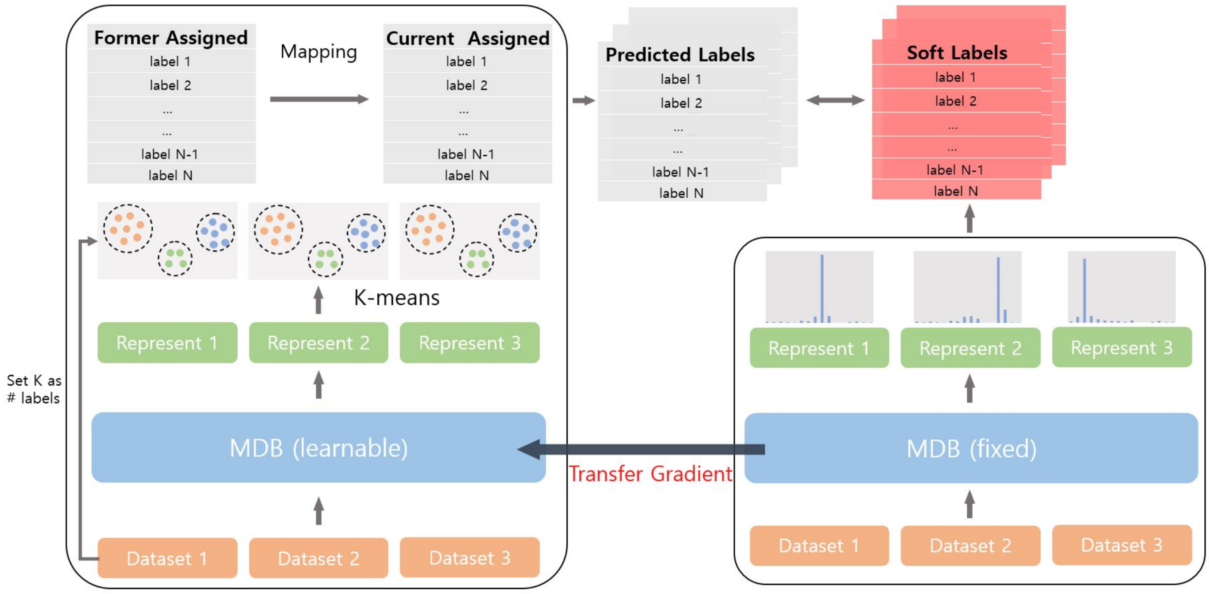

Figure 3: MDB signifies the SBERT trained by MDB method. First, two same MDB are initialized. Second, MDB (learnable) predicts the labels for train utterances using K-means. Third, MDB (fixed) calculates gradients based on cross entropy loss between predicted labels and soft labels. Fourth, MDB (fixed) transfers the gradients to MDB (learnable), considering a weight difference. Finally, MDB (learnable) is optimized with transferred gradients.

efficient in Zhang et al. (2021a), we choose K-means as a clustering module. Details are in Figure 3. However, there are three obstacles to overcome. It is not straightforward to select the K for K-means and it is impossible to differentiate. Furthermore, K-means is non-convex, whose result highly depends on fortunate initialized centroids. At last, the labels of clusters can vary each step even though their contents do not change.

First, we use the number of intent labels for each dataset to determine K. The premise is that K does not change during PGT training. Second, Siamese network learning process makes the model differentiate. In the process of PGT training, one of two identical encoders in Siamese network (MDB-learnable) predicts labels from K-means clustering. The other (MDB-fixed) generates representations to use them as soft labels and vicariously calculates gradients based on the cross entropy loss between predicted labels and soft labels. Afterward, the gradients are assigned from MDB(fixed) to MDB(learnable), reflecting equation (4).

$$Grad_{learned} = Grad_{fixed} + W_{learned} - W_{fixed} \quad (4)$$

where $Grad_{learned}$ is the gradient of MDB (learnable), $Grad_{fixed}$ for the gradient of MDB (fixed), $W_{learned}$ for the weight of MDB (learnable), and W_{fixed} for the weight of MDB (fixed). In the next step, MDB(learnable) is optimized with allotted gradients. Third, when the performance of MDB is reasonably good, the effect of initialized centroids diminishes. Though few initial randomnesses can still remain, we consider it as an augmentation. Lastly, the cluster label is adjusted through Hungarian mapping according to the similarity among the clusters' centroids.

6 Experiment

6.1 Dataset

We separate our datasets into two groups, a train dataset and a test dataset. For MDB and PGT training, we introduce six datasets.

Train Banking77 (Casanueva et al., 2020) is banking dialogue divided into Out-Of-Domain (OOD) intent and In-Domain (IND) intent. We use fine-grained 77 intents as train data. ATIS (Hemphill et al., 1990) is made of airplane travel transcripts. We preprocess them to extract the 17 intent categories. Clinc150 (Larson et al., 2019) is for evaluating the intent detection system. HWU64 (Liu et al., 2019) is for developing a home robot with various scenarios to serve as personal assistance, comprised of multi-domain(21 domains). MCID (Levy and Wang, 2020) is involved in Covid-19, a chatbot dataset consisting of user and agent. Restaurants-8k (Coope et al., 2020) is a set of 8,198 utterances gathered from the actual restaurant booking system. The details of the training dataset with specific data statistics are in table 1.

Test We test our model on two different domain datasets without further training. Finance and Banking are the released test datasets of DSTC 11.

6.2 Baseline

In DSTC 11 task2 Open Intent Induction, "glove-840b-300d" and "sentence-transformer/all-mpnet-base-v2" are base encoders. Afterward, K-means clusters the hidden representations from encoders to induce an intent schema. Utterances with "InformIntent" are used as input, which implies that some noises might be included as well.

- KGlove + K-means clustering
- SBERT + K-means clustering

6.3 Metrics

There are five evaluation metrics: accuracy (ACC), normalized mutual information (NMI), Adjusted rand index (ARI), Precision, Recall, and F1 score. ACC is the main evaluation. NMI is defined as

$$\mathbf{NMI}(\mathbf{Y}, \mathbf{C}) = \frac{2 * I(Y;C)}{[H(Y) + H(C)]}, \quad (5)$$

where Y is class labels, C for cluster labels, H for entropy, and I for mutual information. Following the baseline code, we used hyperopt packages to identify the optimal alignment.

6.4 Implementation

The "sentence-transforemr/all-mpnet-base-v2" model is our backbone. In MDB training process, we split each dataset into three sets (Train 70 : valid 20 : test 10). Since each dataset has its own various sizes, we set the number of iterations per epoch according to the largest dataset's length (i.e., 22,500 of CLINC150 dataset) divided by the batch size 64.

We train MDB and PGT modules 40 epochs in total. The optimizer is AdamW with cosine annealing and warm-up scheduler, whose learning rate is 5e-6, weight decay for 1e-2, and warm-up steps for 10% of total iterations. To calculate the loss in the MDB training, we set temperature τ as 0.05 for each dataset, whereas PGT uses cross entropy loss.

In our experiments with the grouping algorithms, K-Means and Spectral clustering use n init 10. As mentioned in Section 5 Methods, we sample from the uniform distribution between 5 and 50 until finding optimal K. The type of affinity is 'nearest neighbors' for Spectral clustering.

7 Results

We show the results on the Banking and Finance DSTC 11 datasets in Table 2. We ablate each module to find the best combination. We notate T(True) and F(False) to signify each module's existence (GE-MDB-PGT). For example, TTF-kmeans means that we concatenate representations of GE and MDB, subsequently clustering those with K-means algorithm.

First, an encoder with Spectral clustering outperforms one with K-Means clustering in most cases by at least around 5% in ACC and at most 16% in ACC. Spectral clustering additionally constructs an affinity matrix using the similarity based on the nearest neighbors and clusters utterances based on K-means in a Laplacian embedding space. Therefore, the result implies that Spectral clustering is advantageous for Finance and Banking as it clusters data by considering the connectivity between data points rather than compactness around the cluster center. More details are in Appendix A and Appendix B

Next, the MDB module significantly plays a significant role in Open Intent Induction task beyond the domain. For instance, on Banking dataset, FTF-spectral outperforms Baseline by around 13% in ACC, 5% in NMI, and 7% in F1 score. On Finance Dataset, it also shows higher performance, around 12% in ACC, 7% in NMI, and 11% in F1 score. These results symbolize that MDB method is significantly profitable for learning how to discriminate similar utterances without a domain dependency. Additionally, by observing TTF model's performance improvement upon Baseline (TTF-kmeans), we can derive the same result.

Furthermore, comparing FTF and FTT, the PGT module boosts performance by bridging the gap between clustering and our model. On Banking dataset, our FTT-spectral improves FTF-spectral upon around 3.5% in ACC, 1.7% in NMI, and 3.3% in F1. And, on Finance dataset, 1.5% in ACC, 0.3% in NMI, and 1% in F1.

Comparing FTF and TTF, the general representations induced by GE module contribute to the performance. On Banking dataset, our TTF-spectral improves FTF-spectral upon around 6% in ACC, 3% in NMI, 6% in F1. And, on Finance dataset,

Table 1: Intent Detection Datasets Summary

Dataset	#Domain	#Intents	#Utters	Vocab	Length(min / max / avg)
ATIS	1	25	5,781	938	1 / 46 / 11.2
BANKING77	1	77	13,072	2,636	2 / 78 / 11.7
CLINC150	10	150	22,500	6,420	1 / 28 / 8.3
HWU64	21	64	11033	4661	1 / 25 / 6.6
MCID	1	16	1745	1415	1 / 20 / 6.7
RESTAURANT8K	1	13	4181	3949	1 / 84 / 15.1

	Banking			Finance			Summary		
Models	ACC	NMI	F1	ACC	NMI	F1	ACC	NMI	F1
Baseline (TFF-kmeans)	70.76	83.97	79.9	56.46	76.22	67.83	63.61	80.095	73.865
FTF - kmeans	74.4472	87.338	82.5436	70.177	82.717	78.1243	72.3121	85.0275	80.33395
FTF - spectral	83.7838	89.4275	86.402	68.5841	83.8913	78.7872	76.18395	86.6594	82.5946
FTT - kmeans	75.1843	85.8193	81.5178	65.2212	80.6355	74.0729	70.20275	83.2274	77.79535
FTT - spectral	87.2236	91.1721	89.7293	70.177	84.2486	79.6696	78.7003	87.71035	84.69945
TTF - kmeans (Submit)	79.36	87.71	83.66	57.52	78.44	70.38	73.62385	86.476	82.4171
TTF - spectral	89.6806	92.6607	92.3042	70.6195	**84.9659**	**80.1991**	80.15005	**88.8133**	**86.25165**
TTT - kmeans (Submit)	74.2	86.98	82.1	60.8	80.48	73.73	66.91385	82.1102	76.64015
TTT - spectral	**90.172**	**92.9917**	**92.6801**	**71.0619**	84.2045	79.7353	**80.6170**	88.6008	86.2077

Table 2: Results represent the performance of our method applied to Banking and Finance domain datasets. We notate T(True) and F(False) to signify each module's existence (GE-MDB-PGT). FTF refers to a model using MDB, FTT refers to a model using MDB and PGT, TTF refers to a model using GE and MDB, and finally TTT refers to a model using GE, MDB and PGT

2% in ACC, 1% in NMI, 1.5% in F1. GE module is prominently helpful for banking dataset. We conjecture that Banking utterances are closer to ordinary language than Finance utterances. Finally, TTF-spectral and TTT-spectral show that the combination of GE and PGT boosts ACC on Banking and Finance, around 0.5% for the two datasets.

In sum, simply changing from K-means to Spectral clustering improves the performance significantly in the model we submitted. Our TTT-spectral shows remarkable performance in two different domains (e.g., Banking, Finance) without additional training process. Mainly, the MDB method boosts performance by a large margin, and the PGT module raises the performance to a higher level. We empirically recognize that Spectral clustering shows better synergy with our models than K-means clustering. Furthermore, MDB and PGT methods work well regardless of the domain without additional fine-tuning processes.

8 Conclusion

In TOD systems, it is noteworthy to capture users' intents regardless of domain. We figure out that simply adopting PLM does not perform well in dialogue corpus. Therefore, PLM needs more training. We suggest the multi-view model with GE, MDB, and PGT modules. The MDB method with cosine softmax utilizes an existing dataset to bridge between PLM and dialogue domain. The novel PGT method, a methodology for fine-tuning based on K-means, is presented to enhance the clustering capability. As a result, the performance increases in Finance and Banking datasets. According to our clustering experiments, we found that Spectral clustering is best fitted for our model, resulting in more significant and higher performance than simple K-means methods.

Limitations

The first limitation of our approach is that the hidden representation dimension increases as more views are used. Therefore, the clustering phase may take longer than other models. Second, interactive information among these multi-views is not considered in our model. Thus, we will introduce meta-attention to harmonize them, not just a concatenation, to reflect interactive information among modules. Lastly, in order to apply MDB, an existing task-related dataset must exist. Otherwise, it is impossible to train the MDB module.

Acknowledgement

K. Jung is with ASRI, Seoul National University, Korea. This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) [No. 2022-0-00184, Development and Study of AI Technologies to Inexpensively Conform to Evolving Policy on Ethics]. This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) [NO.2021-0- 02068, Artificial Intelligence Innovation Hub (Artificial Intelligence Institute, Seoul National University) & NO.2021-0-01343, Artificial Intelligence Graduate School Program (Seoul National University)]

References

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7747–7763, Online. Association for Computational Linguistics.

Siqi Bao, Huang He, Fan Wang, Hua Wu, and Haifeng Wang. 2020. PLATO: Pre-trained dialogue generation model with discrete latent variable. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 85–96, Online. Association for Computational Linguistics.

Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. 2000. Lof: Identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, SIGMOD '00, page 93–104, New York, NY, USA. Association for Computing Machinery.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

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*, pages 38–45, Online. Association for Computational Linguistics.

Ajay Chatterjee and Shubhashis Sengupta. 2020. Intent mining from past conversations for conversational agent. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4140–4152, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Minhua Chen, Badrinath Jayakumar, Michael Johnston, S. Eman Mahmoodi, and Daniel Pressel. 2022. Intent discovery for enterprise virtual assistants: Applications of utterance embedding and clustering to intent mining. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track*, pages 197–208, Hybrid: Seattle, Washington + Online. Association for Computational Linguistics.

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

Samuel Coope, Tyler Farghly, Daniela Gerz, Ivan Vulić, and Matthew Henderson. 2020. Span-ConveRT: Few-shot span extraction for dialog with pretrained conversational representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 107–121, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*.

Matthew Henderson, Paweł Budzianowski, Iñigo Casanueva, Sam Coope, Daniela Gerz, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulić, and Tsung-Hsien Wen. 2019. A repository of conversational datasets. In *Proceedings of the First Workshop on NLP for Conversational AI*, pages 1–10, Florence, Italy. Association for Computational Linguistics.

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 *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1311–1316, Hong Kong, China. Association for Computational Linguistics.

Sharon Levy and William Yang Wang. 2020. Cross-lingual transfer learning for COVID-19 outbreak alignment. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

Ting-En Lin and Hua Xu. 2019. Deep unknown intent detection with margin loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5491–5496, Florence, Italy. Association for Computational Linguistics.

Ting-En Lin, Hua Xu, and Hanlei Zhang. 2019. Discovering new intents via constrained deep adaptive clustering with cluster refinement. *CoRR*, abs/1911.08891.

Ting-En Lin, Hua Xu, and Hanlei Zhang. 2020. Discovering new intents via constrained deep adaptive clustering with cluster refinement. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8360–8367.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2019. Benchmarking natural language understanding services for building conversational agents. *CoRR*, abs/1903.05566.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Ro{bert}a: A robustly optimized {bert} pretraining approach.

J. MacQueen. 1967. Some methods for classification and analysis of multivariate observations.

Bodhisattwa Prasad Majumder, Shuyang Li, Jianmo Ni, and Julian McAuley. 2020. Interview: Large-scale modeling of media dialog with discourse patterns and knowledge grounding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8129–8141, Online. Association for Computational Linguistics.

Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, and Geoffrey Zweig. 2015. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):530–539.

Andrew Ng, Michael Jordan, and Yair Weiss. 2001. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, volume 14. MIT Press.

Hugh Perkins and Yi Yang. 2019. Dialog intent induction with deep multi-view clustering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4016–4025, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

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*, pages 120–129, Online. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*, volume 33, pages 16857–16867. Curran Associates, Inc.

Ivan Vulic, Pei hao Su, Sam Coope, Daniel Gerz, Paweł Budzianowski, Iñigo Casanueva, Nikola Mrkvsi'c, and Tsung-Hsien Wen. 2021. Convfit: Conversational fine-tuning of pretrained language models. In *EMNLP*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Chien-Sheng Wu, Steven C.H. Hoi, Richard Socher, and Caiming Xiong. 2020. TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 917–929, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Hanlei Zhang, Hua Xu, Ting-En Lin, and Rui Lyu. 2021a. Discovering new intents with deep aligned clustering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(16):14365–14373.

Haode Zhang, Yuwei Zhang, Li-Ming Zhan, Jiaxin Chen, Guangyuan Shi, Xiao-Ming Wu, and Albert Y.S. Lam. 2021b. Effectiveness of pre-training for few-shot intent classification. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1114–1120, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jianguo Zhang, Trung Bui, Seunghyun Yoon, Xiang Chen, Zhiwei Liu, Congying Xia, Quan Hung Tran, Walter Chang, and Philip Yu. 2021c. Few-shot intent detection via contrastive pre-training and fine-tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1906–1912, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yuwei Zhang, Haode Zhang, Li-Ming Zhan, Xiao-Ming Wu, and Albert Lam. 2022. New intent discovery with pre-training and contrastive learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 256–269, Dublin, Ireland. Association for Computational Linguistics.

Xueliang Zhao, Wei Wu, Can Xu, Chongyang Tao, Dongyan Zhao, and Rui Yan. 2020. Knowledge-grounded dialogue generation with pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3377–3390, Online. Association for Computational Linguistics.

Yunhua Zhou, Peiju Liu, and Xipeng Qiu. 2022. KNN-contrastive learning for out-of-domain intent classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5129–5141, Dublin, Ireland. Association for Computational Linguistics.

A Kmeans vs Spectral in Banking Dataset

A.1 Analysis about Banking Dataset

Table 3 shows some simple statistics for Banking Dataset. Some utterances do not belong to the gold intent sets, regarded as noises. As shown in the Tabel 4, the utterances are colloquial and usually include filler words. In some cases, samples include utterances which are different from its label. Furthermore, some samples require context to fully capture their intention, such as "I need to make a deposit like now." Additionally, some utterances could be annotated with multiple labels. For example, "Can I close my savings account and transfer the remaining balance from my savings account into my checking account?", can be labeled as not only CloseBankAccount but also, InternalFundTransfer.

We used the utterances, including noise, to make an intent schema where 2,325 instances (63%) were noise. They are daily conversations regardless of banking. Therefore, it is important to distinguish noise from other gold labels effectively.

There is a huge data imbalance with respect to the number of instances included in each gold cluster; for example, the label "CheckAccountBalance" has 243 samples, while "AskAboutCreditScore" has only three samples. The number of samples for each intent label is shown in Table 5.

A.2 Analysis about Intent Schema and Test Prediction

For noise detection, in the case of Kmeans clustering, this algorithm finds 28 intent clusters out of 30 gold clusters, whereas Spectral clustering found 34 intent clusters. Kmeans clustering detect 251 noise utterances out of 2325 noise instances and showed an F1 score of 0.189, while Spectral clustering found 838 noise instances and showed a significantly higher F1 score of 0.478. According to detected labels, the intent schema created through Kmeans clustering contained a lot of noise, resulting in an average F1 score of 0.288 without noise. In contrast, the intent schema created through Spectral clustering achieved a significantly higher average F1 score of 0.379 without noise.

Spectral clustering presented a better performance for labeled data with noise. However, Kmeans clustering tended to concentrate on certain words and did not consider the meaning of the entire sentence. This makes it difficult to distinguish noise from the other gold labels and decide the boundaries between clusters, which share similar semantic with some variations such as "UpdateEmail" and "UpdateStreetAddress." In contrast, Spectral clustering tended to consider the meaning of the entire sentence, filtering out noises consisting of daily conversations. In fact, noises are labeled sentences within additional four clusters.

Spectral clustering also showed better performance for labeled data without noise. Throughout table 6, Spectral clustering clustered sentences which have similar meanings. It also appropriately handled unrelated utterances as NaN, while Kmeans clustering failed to do so. Analyzing the results among the labels, i.e., "UpdateEmail" and "UpdateStreetAddres," Spectral clustering considered the meaning of the entire sentence more, further accurately distinguishing noise from labeled utterances. We recognize that it detects seemingly insignificant word changes such as "email" and "street," preserving overall similar semantics. However, Kmeans clustering focused on specific words rather than the entire meaning of the sen-

Table 3: Simple statistics about Banking Dataset

Dataset file name	# of Utterances	# of Intent	# of Noise
Dialogue.jsonl (InformIntent)	3696	30	2325
Test-utterance.jsonl	407	18	0

Table 4: Examples of Samples in Banking Dataset

Utterances	Intent
I need to make a deposit like now.	FindBranch
No, can you help me find one please?	FindBranch
Well I've got the mobile app. And I'll probably just go ahead and use that because I think I could get directions straight from it. Yeah, I'm new to this new place, anyways. So what about fees? What do they do with those cause I know you when I lived in Alabama we had locations you everywhere and I didn't have to worry about those new fees because I was always using you Intellibank's ATMs but so what what's like the fees like if I if I don't use an Intellibank?	AskAboutATMFees
Can I close my savings account and transfer the remaining balance from my savings account into my checking account?	CloseBankAccoun
All right. So where are you from?	N/A
All righty. So are are you in Raleigh?	N/A
gonna take very long?	N/A
I just need the balance for my checking account, please.	N/A
All right. That sounds great. Is is there a fee associated with this?	N/A
And I need some money transferred to my daughter to pay her bills.	N/A

Table 5: Number of samples included in each gold intent cluster in Banking

AskAboutATMFees	8	GetBranchHours	64
AskAboutCardArrival	10	GetBranchInfo	20
AskAboutCashDeposits	15	GetWithdrawalLimit	20
AskAboutCreditScore	3	InternalFundsTransfer	130
AskAboutTransferFees	11	OpenBankingAccount	83
AskAboutTransferTime	24	OpenCreditCard	5
CheckAccountBalance	243	OrderChecks	9
CheckAccountInterestRate	10	ReportLostStolenCard	70
CheckTransactionHistory	23	ReportNotice	17
CloseBankAccount	63	RequestNewCard	15
DisputeCharge	89	SetUpOnlineBanking	18
ExternalWireTransfer	109	UpdateEmail	34
FindATM	86	UpdatePhoneNumber	20
FindBranch	110	UpdateStreetAddress	49
GetAccountInfo	13	SUM	1371

tence, resulting in misclassifying instances to noise or wrong label even if it contains the exact same keyword with the label. Moreover, their aligned utterances share semantic information; update something. Thus, they tend to locate nearby, obscuring boundaries in terms of the distance-based method. That is, Kmeans clustering cannot distinguish the subtle changes in meaning.

We applied the test-utterance dataset to the classifier trained on the Intent schema to conduct the final evaluation. As mentioned in the Results 7, the classifier trained on the schema created using spectral clustering showed remarkably high performance. When Spectral clustering algorithm creates the schema, it groups similar utterances discerning noises. On the other hand, Kmeans focused on specific words or misclassified daily conversations irrelevant to the topic, which lowered the quality of the schema and caused the classifier to make many errors. As a result, spectral clustering showed significantly higher performance in ACC.

A.3 Alignment

In the organizer's code, predicted labels are assigned one-to-one with reference labels. The one-to-one assignment is the Hungarian mapping method relying on the number of references, so duplicate allocation to the same reference is not allowed. Therefore, when predicted clusters (23 labels) exceed the number of gold clusters (18 labels), almost similar clusters are divided into two different intents, one of which is abandoned. If we aligned the reference label to the predicted cluster allowing overlapping, both Kmeans and Spectral results show that ACC increased by about 4.5% (0.786 and 0.953, respectively). Spectral F1 score also increased by 2%, while Kmeans F1 score raised by 0.1%. It indicates that spectral clustering was well-trained in sensitivity to noise and effectively separated existing clusters.

B Kmeans vs Spectral in Finance Dataset

This section presents an empirical analysis of the results obtained from "TTT-kmeans" and "TTT-spectral" on the finance dataset. The Kmeans clustering tends to prioritize specific keywords over the overall context. For instance, in the first example provided in Table 7, taking a look at the bold phrases, it can infer "ScheduleAppointment" from "see a notary," but may provide incorrect answers if it fails to comprehend such chunks. Moreover, in cases such as "OrderCheck," where it encounters the word "check," it tends to blindly map it to "CancelCheck," and for sentences containing the word "pin," it predicts "ChangePin." Additionally, when it incorrectly predicts "UpdateEmail," it directly maps the word "update" to "UpdateEmail," and there is a tendency to link the word "liquidity" to "purchase."

On the other hand, spectral clustering tends to focus less on specific keywords than Kmeans clustering, but may still make errors in distinguishing relatively ambiguous labels in Table 8. In cases where it makes incorrect predictions, there are often instances of multi-labeling depending on how to interpret utterances. For example, in table 9, the sentence "what's the nearest branch I can visit in person?" carries an ambiguous meaning, such as where I can go and which branch is currently open. However, given the erroneous mapping of "change statement" to "UpdateEmail" and the misinterpretation of "add to business account" as "PhoneNumber" due to its focus on the word "number," and the misclassification of the entity "Gerald Smith" as a location name, there is still much room for improvement.

Table 6: Examples about classifiers' predicted labels in Banking

Utterance	Gold intent	Predicted label (Kmeans)	Predicted label (Spectral)
And and is it. Is it free? Is it free?	NaN	AskAboutATMFees	AskAboutATMFees
Oh, thanks. What about you? Are you having weather?	NaN	AskAboutCardArrival	NaN
Do you guys have an app?	NaN	SetUpOnlineBanking	NaN
OK. that's what what what about my my savings account.	NaN	InternalFundsTransfer	CheckAccountBalance
I want to put money into his account.	InternalFundsTransfer	ExternalWireTransfer	InternalFundsTransfer
no, if possible, could I update my email with you? That's an old one.	UpdateEmail	UpdateStreetAddress	UpdateEmail
OK. I appreciate that. And I wanted to also update my number.	UpdatePhoneNumber	UpdateStreetAddress	UpdatePhoneNumber
... I have money taken of my account that I didn't that I did not do.	DisputeCharge	CheckTransactionHistory	DisputeCharge

Table 7: Kmeans clustering's incorrect predictions and representative examples in Finance.

Predict	Reference	utter
AddUserToAccount	ScheduleAppointment	Good morning. I need to **see a notary** for some documents by at least 2pm today please.
CancelCheck	OrderCheck	I need to order **checks** ASAP!
CancelCheck	OrderCheck	I want to order 500 **checks** for my business.
ChangePin	CloseAccount	please close account #098716253 with **pin** #65491
ChangePin	GetCreditReport	... I need my business credit report sent to me via mail. My **pin** is 94442
UpdateEmail	UpdatePhoneNumber	I'd like to **update** my phone number is 761-451-9850
UpdateEmail	UpdateStreetAddress	I'd like to **update** my street address please, my name is Helen Smithfield
PurchaseStocks	AskLiquidityRatio	i want to check **liquidity** ratio
PurchaseStocks	AskLiquidityRatio	i am jorah mont, i want to check my **liquidity** ratio
OpenAccount	CloseAccount	Hi please close my **account**, I'm switching to a different **bank**

Table 8: Spectral clustering's top incorrect prediction-label pairs in Finance.

Predict	Reference
ApplyCreditCard	RequestNewCard
checkAccountBalance	CheckCreditCardBalance
MakeCreditCardPayment	SetAutoPayment
FindBranch	ScheduleAppointment

Table 9: Spectral clustering's incorrect predictions and representative examples in Finance.

Predict	Reference	utter
GetBranchHours	FindBranch	What's the nearest branch I can visit in person?
FindBranch	ScheduleAppointment	I need to see a person about my taxes. In person thank you.
GetBranchHours	FindBranch	Which of your branches can I visit in person?
UpdateEmail	ChangeStatementDelivery	hey I want to **change** my monthly statement delivery to **email** pls
UpdatePhoneNumber	AddUserToAccount	... I need to add Luke Walker to my business account. The **account number** is 45000222
UpdateStreetAddress	AddUserToAccount	I'm Eric Daniels. I need to add **Gerald Smith** to my business account.