Learn to Discover Dialog Intents via Self-supervised Context Pretraining

Anonymous NAACL submission

Abstract

Intent detection is one of the most critical tasks in prevalent task-oriented dialog systems. However, most systems could only identify a fixed set of intents, without covering a ubiquitous space of real-world semantics. Inducing new dialog intents or excluding out-of-scope (OOS) queries are crucial particularly in complex domains like customer support. We present a simple yet effective intent induction schema via pre-training and contrastive learning. In particular, we first transform pretrained LMs into conversational encoders with in-domain dialogs. Then we conduct context-aware contrastive learning to reveal latent intent semantics via coherence from dialog contexts. By composing a fine-grained intent subspace from in-scope domain data, we demonstrate the effectiveness of our approach to induce intents with simple clustering algorithms and detect outliers with probabilistic linear discriminant analysis (pLDA). The experimental results validate the robustness and versatility of our framework, which also achieves superior performances over competitive baselines without label supervision.

1 Introduction

Recent advances of task-oriented agents such as Apple Siri, Amazon Alexa or Google DialogFlow have prompted the success of improving more natural customer service automation (Zhang et al., 2020b). They have been widely adapted in several domains like booking flights, restaurants, and customer support. Usually, the very first step of establishing such systems is to determine appropriate ontologies that constrain the dialog intent and state space for a specific task (Weld et al., 2021).

However, most works with such assumptions usually categorize utterances into few simple intents; yet exclude further pragmatic intent discovery in the real world (Hendrycks and Gimpel, 2018). Dialogs with different levels of stylized interactions may experience disparate variations of intent fluidity, where the scope of intents varies. In complex domains like customer support and healthcare, users may not be fully cognizant of the system capabilities, where ‘out-of-scope’ (OOS) queries would reasonably emerge during dialogs that fall out of bounds of system-supported intent scopes (Larson et al., 2019). Systems which can barely enumerate a complete view of user intents will easily fail to recognize an emerging intent during shifts between domains or chit-chats, as shown in Figure 1.

Second, unlike (written) texts, spoken utterances do not come with clear sentential segmentation by topographic means (Jonathan Ginzburg, 2010). This poses a more challenging understanding task for recent systems. Perkins and Yang (2020) introduced the dialog induction task to cluster user query utterances. It aims to learn a good discriminative classifier that predicts the same intents for utterances and their corresponding contexts. However, the performance may be degraded if representations of encoders trained from misleading objectives are hard to separate with accurate decision boundaries (Vulić et al., 2021). We argue that discovering fine-grained representations may be a more critical aspect of retrieving their latent intents that could be easily differentiated with simple clustering. In addition, OOS queries may still be
Recently, contrastive learning based on pretrained Language Models (LM) such as BERT (Devlin et al., 2019) or RoBERTa (Liu et al., 2019), offers striking state-of-the-art (SOTA) performances across multiple natural language processing tasks (NLP) (Casanueva et al., 2020). They aim to enhance discrimination abilities of models by relying on semantic similarity between utterances, which exactly aligns with the clustering objectives based on the distance metrics. Hence, instead of emphasizing on algorithms that seek decision boundaries between data, it is possible to directly learn a good intent space where projected samples are distributed densely in clusters and easily segregated? Second, how could we quickly adapt pretrained LMs into a task-specific operator that generate such representations and discriminate OOS samples?

To solve the above two concerns and mitigate the data scarcity issue, we propose an entirely unsupervised approach without labeled instances to induce new dialog intents and exclude OOS queries from a large non-annotated corpus. Intuitively, we first deploy two-phase pre-training on LMs to learn a good representation that both inherits the coherence characteristics within dialogs and is easily discriminated semantically. Then we introduce a cluster operator to assign each query representation with their intents and exclude OOS ones with Probabilistic Linear Discriminant Analysis (pLDA) (Ioffe, 2006). We show that our pipeline is versatile to incorporate different LMs, pretraining strategies and cluster algorithms to tailor in each domain. The threshold in pLDA could be flexible chosen to trade off between accuracies of detecting IS/OOS queries.

Our contributions are summarized as follows:

1) We make the first attempt of introducing contrastive learning in the dialog induction task without any label supervision.

2) We introduce a flexible algorithm with a tunable threshold to detect OOS queries by forming clusters with only in-domain queries.

3) Experimental results verify the SOTA performance of our approach on two intent induction datasets and demonstrate the effectiveness of detecting OOS queries with self-supervised learning.

2 Problem Formulation

In this section, we first follow Perkins and Yang (2020) to formulate the dialog induction task as the clustering task. Suppose we have a training corpus of unannotated dialogs \(X \in \mathcal{X}\) where each sample has the following format: \(x_i = (u_i, c_i)\), \(u_i\) is the first query utterance and \(c_i\) with \(N-1\) length is the following contexts. We hope to find an optimal operator \(f : \mathcal{X} \rightarrow \mathcal{Y}, \mathcal{Y} \subseteq \mathbb{R}^K\) to assign a new query \(u_{test}\) an one-hot vector \(y\) representing its cluster id \(k\). We introduce a function \(\phi_{\theta} : \mathcal{X} \rightarrow \mathcal{Z}\) that maps an input into a hidden space \(Z \in \mathbb{R}^H\), where we restrict \(\phi_{\theta}\) in a language model function class \(\Phi\). We will finally try to find a good set of parameters \(\theta\) and best \(f\) s.t. 

\[
y_{test} = \arg \min_{\phi \in \Phi} \sum_{y \in \mathcal{Y}} J(\phi_{\theta}(u_{test}), f) \text{ over some clustering objective function } J.\]

We are also interested in detecting an OOS signal \(y_{oos} \in \mathbb{R}^2\) for \(u_{test}\) whether a query is in domain, i.e. \(y_{test} \in \mathcal{Y}\).

3 Methodology

3.1 PI: Self-supervised Pre-training (SSP)

We first deploy three language learning strategies to adapt pretrained LMs into conversational tasks: Mask Language Modeling (MLM), Mask Context Modeling (MCM) and Next Context Prediction (NCP). We first concatenate each utterance \(ui\) and their contexts \(ci\) with ‘[SEP]’ as \(x_i\). Then, we will retrieve the feature representation \(h_i\) for the entire dialog \(x_i\) with each token as \(x'_i\) with a pretrained LM, \(\phi_{\theta}(\cdot)\) that implicitly learns sentential and conversational structures. We use two different projection matrices \(W_v \in \mathbb{R}^{H \times |V|}\) and \(W_c \in \mathbb{R}^{H \times 2}\) to transform \(h_i\) into one-hot vectors \(\hat{y}_v\), \(\hat{y}_c\), where \(H\) is BERT hidden size and \(|V|\) is vocabulary size. We then update the parameters \(\theta\) and \(W_v\) or \(W_c\) to minimize the cross entropy losses as follows:

\[
\mathcal{L}_v(\theta, W_v) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} y_{i} \log \phi_{\theta}(x'_i), W_v \tag{1}
\]

\[
\mathcal{L}_c(\theta, W_c) = -\frac{1}{N} \sum_{i=1}^{N} y_{i} \log \phi_{\theta}(x'_i), W_c \tag{2}
\]

1) MLM: Similarly in BERT, we perform masking on 15% of input tokens \(x_i\) with 80% of the time a special token ‘[MASK]’, 10% of the time random word tokens from vocabulary and others remaining the same; and update LM with loss in Eq 1.

2) MCM: MLM does not take any account of sequential relations between utterances and their contexts. Instead, we only mask tokens in the contexts \(c_i\) to explicitly condition on utterances for token prediction and update LM with loss in Eq 1.

3) NCP: We directly predict whether the entire dialog is reasonable based on \(ui\) and the given \(ci\). And update LM with the loss in Eq 2. For each \(ui\),
we also randomly sample a negative context \( c_{j\setminus i} \) to construct a negative pair \((x_i, c_{j\setminus i})\).

### 3.2 P2: Context-aware contrastive Learning (CACL)

To better align with the downstream metric-based clustering, inspired by the recent success of contrastive learning in few-shot setting (Vulić et al., 2021; Zhang et al., 2021b), we found it is rather appropriate to introduce similarity-based pretraining for retrieving distant-sensitive\(^4\) representations for clustering, instead of pseudo labeling samples directly. As shown in Lee et al. (2021) of contrastive learning setting, if we have the input variable \( X_1 \) and its target random variable \( X_2 \) for the pretext task, given the knowledge of their latent label \( Y \), one can possibly predict \( X_2 \) without knowing much about \( X_1 \). At this case, \( X_2 \) is approximately independent of \( X_1 \) conditional on the label \( Y \) (\( X_1 \perp X_2 | Y \)). In other words, if we try to learn a good representation \( \phi(X_1) \) close to \( \phi^* \) in Eq 3 by semantically minimizing distances between \( X_1 \) and \( X_2 \), we implicitly force LMs to first predict \( Y \); and predict \( X_2 \) from \( Y \). And we could possibly linearly separate such learned representations \( \phi^* \) with small representation errors.

\[
\phi^* = \arg\min_{\phi \in \Phi} \mathbb{E}||\phi(X_1) - \phi(X_2)||^2 \quad (3)
\]

In our case, \( X_1 \) and \( X_2 \) are the given utterance queries and their corresponding contexts. Intuitively, successful dialogs should preserve coherence where \( X_1 \) and \( X_2 \) should be driven by the same underlying intent \( Y \), regardless of being causes or effects of the intent. It elucidates that with a good representation \( \phi^* \), we shall easily cluster \( X_1 \) based on similarity by inducing their underlying intents \( Y \). Therefore, we leverage the dual-encoder architectures to model the relevance between an utterance and their context pair \((u, c)\). We will use the pretrained encoder \( \phi(\cdot) \) from section 3.1 to obtain \((h_u, h_c)\). And we take the final encoding from the \('[CLS]' token via a pooling operation.

**Context Filtering** Before sending data pairs into the encoder, some redundant sentences \( c_i \) in the dialog contexts \( c = \{c_i\} \) may cause frictions to obfuscate user intents. Therefore, instead of encoding all dialog contexts, we first obtain the hidden representations for each sentence in a dialog \((h_u, h_{c^i}, \ldots, h_{c^{i-L}})\) where \( L \) is the dialog length. Then we calculate the distance (cosine similarity) between each dialog context representation and the utterance. Eventually, we take the closest context as the final sentence pair \((h_u, h_{c^i})\), where \( i' = \arg\min_{i \in [1, L-1]} \cos(h_u, h_{c^i}) \).

For a given pair \((h_u, h_{c^i})\), we then introduce three strategies to finetune our encoder \( \phi \).

1) **Context Ranking** We leverage the standard multiple negatives ranking loss (MNEG) (Henderson et al., 2017) to rank the correct context \( c^i \) for the utterance \( u_i \) over other negative contexts \( c'_{j \setminus i} \notin \{c_i\} \) in a single batch. MNEG loss for a single batch \((u_1, c_1), \ldots, (u_{N_U}, c_{N_U})\) is calculated as the following:

\[
L_{\text{MNEG}} = - \sum_{i=1}^{N_U} S \cdot \cos(u_i, c_i) + \sum_{i=1}^{N_U} \log \sum_{j=1, j \neq i}^{N_R} e^{S \cdot \cos(u_j, c_i)} \quad (4)
\]

where \( S \) is a scaling factor. It will try to maximize correct match score and abate negative scores.

2) **Online Contrastive Learning (OCL)** We select only hard positive and negative pairs to update the contrastive loss (Robinson et al., 2021), where

---

\(^4\)Here we mean the mapped representations preserve the topological information within a latent intent space.
\[ d(u_i, c_j) = 1 - \cos(u_i, c_j) \] and \( \gamma \) is the margin:

\[
L_{OCL} = \begin{cases} 
    d(u_i, c_{j'})^2 & \text{if } j = j' \\
    \left(\text{ReLU}(\gamma - d(u_i, c_j))\right)^2 & \text{if } j \neq j'
\end{cases}
\]  
\( (5) \)

3) **Triplet Loss (TL):** Finally we also try to compare positive and negative pairs at once by minimizing the following triplet loss:

\[
L_{TL} = \max(\gamma + \cos(u_i, c_{i'}) - \cos(u_i, c_{j\neq i'}), 0)
\]  
\( (6) \)

It will prompt the minimization of the distance from the anchor example and approximate that of the negative example to the margin \( \gamma \).

### 3.3 Clustering

After two-phase context learning, our main goal is to induce the intents from a collection of new user queries. We follow Perkins and Yang (2020) to formulate it as an unsupervised clustering task that involves two popular clustering algorithms \( k\text{-means} \) and \( \text{spectral clustering} \) as an operator \( f : x \to k(x) = m^k(x) \), where \( k(x) \) is the cluster id associated with its mean \( m^k(x) \). \( k\text{-means} \) only requires the query-view vectors from our pre-trained encoder \( \Phi \) while \( \text{multi-view spectral clustering (MVSC)} \) will solicit both query-view and content-view representations for matching.

### 3.4 Out-of-scope (OOS) Detection

After forming the query clusters, we are further interested whether such in-domain gaussian-like distributions could help us exclude the future out-of-scope queries. Here we propose to use the idea of **Probabilistic Linear Discriminant Analysis (pLDA)** (Ioffe, 2006) with predicted labels generated from our \( k\text{-means} \) operator \( f(\cdot) \). The goal of pLDA is to further project query samples onto a latent space such that samples from the same class remain in the same distribution. It can generate class center using continuous non-linear functions even from single example of unseen class. By first obtaining the predicted labels \( y_{i} \) from clustering, we could model both the distribution of utterances \( x \sim N(x|y_i, \phi_{u_i}) \) and class labels \( y \sim N(y|m, \Phi_b) \), where \( m \) is the parameter, \( \Phi_u \) and \( \Phi_b \) are the within/between-class covariance matrices. Latent variables \( u, v \) will represent the example of class and class variable in the projected space. For each class \( y_k \), we compute the mean latent representations \( \bar{u}_k \) from utterances belonging to class \( y_k \). Finally, for a new test query \( x_{test} \), we could calculate the likelihood ratio \( R_k \) with each class’s \( \bar{u}_k \):

\[
R(u_{test}, \bar{u}_k) = \frac{P(u_{test}, \bar{u}_k)}{P(u_{test})P(\bar{u}_k)}
\]  
\( (7) \)

If all of \( R(u_{test}, \bar{u}_k) \) are below a threshold \( T \), we could conclude the query is out-of-scope that does not belong to any of the in-domain clusters.

**Supervised Fine-tuning** To further validate the pLDA-based OOS detection performance from trained representations, we also perform similarity-based contrastive learning on datasets with \( K \) labels available. We generate a sets of \( n \) positive pairs \((x_i, x_j)\) for each intent \( y_k \) and obtain totally \( K \times n \) samples. Then, we use LM encoder \( \phi(\cdot) \) to generate representations \( \phi(x_i), \phi(x_j) \). Then we adopt MNEG loss mentioned above to conduct the ranking task. Finally, we use the pairs \((\phi(x_i), y_{i})\) to train pLDA model and perform OOS detection.

### 4 Experimental Setup

#### 4.1 Datasets

We collect four datasets with two of them (TwACS, AskUbuntu) having smaller amount of annotated user intents (Perkins and Yang, 2020) and two of them (CLINC, BANKING) (Larson et al., 2019) with much larger intent annotated corpus. Table 1 shows the data statistics. For pretraining and contrastive learning purposes, we only use unlabeled corpus for training. For supervised contrastive learning, we use labeled training corpus from CLINC and BANKING. For evaluation datasets, there are three experimental settings:

1) **Clustering:** we use all annotated dialogs in TwACS and AskUbuntu to verify our model’s performance on the dialog intent induction task.

2) **Unsupervised OOS:** we first generate pseudo training set by first sampling \( n = 6000 \) dialogs \( x_i \in X \) as \( X^* \); then obtain their representations \( \phi(x_i) \) and their assigned clusters \( y_i = f(\phi(x_i)) \) via \( k\text{-means} \) as data pairs \( (\phi(x_i), y_i) \). We use these training datasets to assess necessary parameters from pLDA. For evaluation, we randomly select 1000 samples from OOS test sets from CLINC and BANKING as OOS samples and select 1000 samples from IS training sets \( X^*_c \in X^* \cap X_c = \emptyset \).

3) **Supervised OOS:** we directly use the training data \((\phi(x_i), y_i)\) where \( y_i \) is the true label for pLDA.
training. For evaluation, we combine the in-domain test set and OOS test set as the final testing set.

### 4.2 Model Variants and Baselines

We experiment with competitive baselines from some popular transformer-based LMs and clustering approaches. The pretraining methods include:

1. **Autoencoders** (Perkins and Yang, 2020) which are finetuned on TwACS and AskUbuntu datasets.
2. **Quick thoughts (QT)** (Logeswaran and Lee, 2018) which is a strong representation learning baseline that is adopted in BERT.
3. **DialoGPT** (Zhang et al., 2020a) which is a generative model trained with dialog corpus.
4. **BERT** (Devlin et al., 2019) which is a contextualized representation model based on transformers.
5. **TOD-BERT** (Wu et al., 2020) which is finetuned on conversational data based on BERT.
6. **Sentence-BERT (SBERT)** (Reimers and Gurevych, 2019) which deploys dual-encoder structure for similarity training.

We also experiment with a range of model variants enabled by our framework, and compare their performances against an array of cutting-edge conversational sentence encoders. The entire pipeline is specified as LM+P1+P2-LOSS, where we adopt a LM structure and pretrain it with Phase-1 (P1) pretraining and Phase-2 (P2) learning with specific loss (LOSS). For LMs, we use 1) BERT, 2) RoBERTa (ROB) as an improved variant of BERT, LM-pretrained with more data and 3) DistilRoBERTa (DROB), a distilled version of RoBERTa, trained with around 4 times fewer data than the teacher RoBERTa model. All of models contain 768-dimensional transformer layers with 12 (BERT, ROB) or 6 (DROB) attention layers. For P1, we have pretraining methods: MLM, MCM, NCP. And we use MNEG, TL, OCL for P2-LOSS.

For OOS detection, we use binary classification with pretrained LMs as our baselines. BERT-all indicates that we train all in-scope data and 250 out-of-scope queries. To avert class imbalancing, we further subsample only 1000 in-scope data for BERT, ROB and DROB model. We also pretrained BERT with in-scope data beforehand as BERT*. Eventually we train different LMs as specified in § 3.4 and detect OOS queries with pLDA.

### 4.3 Experimental Setting

We use Sentence-Transformers package (Reimers and Gurevych, 2019) to implement our framework. We first pretrain the LMs with P1-SSP. We train for 4 epochs in batches of 16. We also set the max length as 200 and mask probability of 0.15. For P2-CACL, we train each model with 2 epochs in batches of 8. And for supervised contrastive learning, we randomly sample \( n = 500 \) positive query pairs for each class and train models with 10 epochs in batches of 10. Margin \( \gamma \) is set to be 5. We set the threshold \( T \) in pLDA as -1.7 for CLINC and 0 for BANKING empirically. We use the AdamW optimizer and \( 2e - 5 \) learning rate and weight decay rate is 0.01. We run each experiment 5 times and report the average. We follow the metrics used in Perkins and Yang (2020) for clustering precision/recall, f1 score and accuracy.

### 5 Results and Discussion

#### 5.1 Main Clustering Results

The main results are summarized in Table 2. First we observe that by directly deploying general...
Table 3: Clustering Performance (%) of our proposed approach with different pretraining LMs and losses. The best results are in bold. We use MLM pretraining for all BERT-based structures.

<table>
<thead>
<tr>
<th>Pretraining LM</th>
<th>Dataset</th>
<th>TwACS</th>
<th>AskUbuntu</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>P</td>
<td>R</td>
</tr>
<tr>
<td>DROB (Sanh et al., 2020)</td>
<td>MNEG</td>
<td>55.8</td>
<td>42.2</td>
</tr>
<tr>
<td></td>
<td>TL</td>
<td>51.4</td>
<td>42.5</td>
</tr>
<tr>
<td>ROB (Liu et al., 2019)</td>
<td>MNEG</td>
<td>51.3</td>
<td>46.7</td>
</tr>
<tr>
<td></td>
<td>OCL</td>
<td>47.9</td>
<td>34.6</td>
</tr>
<tr>
<td>BERT (Devlin et al., 2019)</td>
<td>MNEG</td>
<td>55.8</td>
<td>48.8</td>
</tr>
<tr>
<td></td>
<td>OCL</td>
<td>54.9</td>
<td>41.6</td>
</tr>
</tbody>
</table>

BERT or DialoGPT, universal representations do not reveal clear boundaries for intent separation, causing relatively low performances. Training with conversational data like TOD-BERT offers better disentanglement of intent features for each sample. We then observe even competitive performance when we apply SBERT dual-encoder structure. The results suggest that the pretraining objectives and corpus may substantially influence the downstream clustering task. This prompts us to finetune pre-trained LMs like BERT and TOD-BERT with P1-SPP. The results unanimously suggest the improved performance even over the baselines by using previous SOTA Avkmeans algorithm in AskUbuntu.

The power of contrastive learning By introducing the P2-CACL, the proposed approaches achieve the best performance across two datasets, particularly at NCP+CACL+k-means in TwACS and MCM+CACL+MVSC in AskUbuntu. They successfully establish an intent subspace where utterances are close to their contexts, which congruently approximates the process of disclosing their underlying latent intents. Such cross-instance similarities are especially useful when we compare these representations based on distance metrics, rather than simply learning a MLP-based mapping from each sentence to its class. In addition, we also see that NCP by differentiating correct utterance-context pair has slight improvement in TwACS over MLM, while we see MCM surpasses MLM more in AskUbuntu setting with MVSC approach. The context-aware pretraining (MCM, NCP) seem to be more beneficial in MVSC where contexts are considered, than single-view k-means approach.

Impacts of input LMs and losses Table 3 displays the results by adopting different pretrained LMs and losses from § 3.1 during P2-CACL. It verifies the versatility and wide applicability of our proposed method. But the model variants may still naturally impact the absolute clustering performance. Surprisingly, we see a better performance of BERT-based structure over RoBERTa, which decreases the pretraining objectives may play crucial roles of dominating how clusters are formed. The comparison between DROB and BERT reveals that loss regime influences more than the parameter capacity in TwACS; reversely in AskUbuntu. Overall, both MNEG and TL yield similar strong results, while OCL is poor by using only hard examples.

Amount of training examples We also analyze what amount of in-domain corpus is adequate to fairly discriminate utterances’ intents, i.e. to be decently separable upon projection onto the latent intent space. We reduce the amount of data through subsampling. The scores are provided in Figure 3. As expected, more training data yield better performance on average while the absolute scores reach the bottleneck after increasing up to 1/4 of total training data. It implies that even few samples may fairly discriminate the utterance and surpass the models skipping Phase-2.

5.2 Ablation Study
A more careful ablation comparison is conducted in Table 4 to investigate the effects of different stages of pretraining and filtering method. Experimental
results indicate that all of components are necessary to achieve the peak performance in both datasets. Around 3-4% performance drop is observed when we do not perform context filtering where redundant noises may disturb. Forfeiting P1-SSP will result in further drops where we believe it will benefit the model discrimination ability of semantically similar utterances. In addition, we found large performance drops by only adopting P1-SSP in all three pretraining methods, which indicates the necessity of contrastive learning in differentiating samples with aid of their contexts. We also found that NCP is more beneficial in TwACS and MLM in AskUbuntu than P2-only models.

### 5.3 Out-of-scope Detection

**Unsupervised detection** Figure 4 shows the ROC curves of our unsupervised outlier detectors in both datasets. We could observe with varying threshold $T$, the specificity (True negative rate) will be inversely proportional to recall where our classifier predicts more outliers correctly and incorrectly. But surprisingly, without any label supervision, pLDA could still achieve around average 0.7 AUC score. We observe models pretrained with MLM perform better in TwACS and MCM in AskUbuntu. It alludes that models rely more on contexts in AskUbuntu with more turns than TwACS. In addition, we care more about the recall than the precision since precision errors will prompt fallback responses to make users reiterate. Hence, we could sacrifice little on in-scope accuracy and achieve higher recall by tuning $T$.

**Supervised detection** Table 5 indicates the results of supervised OOS detection. We could observe that binary classification with LMs may give a high in-domain accuracy and low recall since the model tends to overfit, significantly depending on the amount of OOS data during training (extremely low recall by leveraging all in-scope data). Instead, even not relying on OOS train data, our method could still achieve competitive true negative rates while maintaining superior recall especially in BANKING. In this case, Phase-1 pretraining seems to be less conducive to assist in out-of-scope detection. Models with RoBERTa backbone perform much better than the other model variants. Figure 4 also shows the PR curve for OOS detection. By specifying the same range of available thresholds, we observe that models with RoBERTa backbone have both higher precision and recall. Even models with DistilRoBERTa backbone may have competitive results with BERT-based model.

### 5.4 t-SNE visualization

To understand how well the latent space is formulated where encoded utterances are mapped onto, we perform t-SNE visualization in Figure 5 on three types of representations generated with disparate pretrained encoders. First, we could see without any finetuning, utterances with distinct intents are entangled into a single mixed cluster. After Phase-1 of transforming LMs into conversational encoders, we start to see some clusters with intents apart. Eventually, Phase-2 further specializes the sentence encoder to learn meaningful task-related semantic clusters even without any label supervision.
Figure 4: (a) ROC curves for proposed unsupervised OOS approach. The score behind the model name in each legend indicates the Area-Under-Curve (AUC) score. (b) PR curves for proposed supervised OOS approach.

Figure 5: t-SNE plots of encoded utterances from AskUbuntu. The representations are created by BERT-based encoder \( \phi \) (a) without any finetuning (b) after Phase 1 MCM pretraining and (c) after Phase 1 and phase 2 contrastive learning. The colors are true labels.

6 Related Work

User intent clustering Clustering user utterances to discover new intents is a critical task that disambiguates utterances. Early works focused on feature engineering for query traits to perform high-quality clustering (Kathuria et al., 2010; Cheung and Li, 2012; Padmanabhan, 2016). Haponchyk et al. (2018) further leveraged supervised signals to detect duplicate questions. Samir Kanaan-Izquierdo and Perera-Lluna (2018) also proposed multi-view spectral clustering to consider consistent cluster assignments across different views. Deep clustering was proposed to better represent data features (Xie et al., 2016; Zhang et al., 2021a; Chang et al., 2017; Nguyen et al., 2021). Perkins and Yang (2020) further finetuned pretrained representations for multi-view k-means clustering.

Out-of-scope detection Until recently, the idea of OOS data is considered to address possibility of queries that fall out of intended output classes (Hendrycks and Gimpel, 2018). Usually the problem is formed as a binary classification task where Larson et al. (2019) provided a large-scale evaluation dataset with some benchmark classifier results. Data augmentation (Zhan et al., 2021; Chen and Yu, 2021) and outlier detection (Lin and Xu, 2019; Yilmaz and Toraman, 2020; Cavalin et al., 2020) are two main streams to tackle such problem. Here we introduce new concepts of detecting OOS queries from probabilistic formed intent clusters.

Contrastive learning Models trained with contrastive learning have rendered striking performances in many NLP tasks (Casanueva et al., 2020; Gnel et al., 2021; Zhang et al., 2021b). In particular, Gunel et al. (2021) proposed supervised contrastive learning for pretrained models on several benchmark tasks. Wu et al. (2019) trained models to learn dialog orders. Zhang et al. (2021b) and Vulić et al. (2021) proposed to transform pretrained LMs by masking tokens and response re-ranking; leveraging similarity-based learning for few-shot intent detection. Our work differs from them that we specifically tackle the intent clustering task with entirely unsupervised fashion.

7 Conclusion

In this work, we present a simple yet effective two-phase self-supervised pipeline to induce dialog intents with clustering and detect OOS queries accordingly. We propose to transform LMs into conversational encoders and retrain them with context-aware similarity-based learning. We demonstrate that by extracting such intent-aware representations, it is possible to separate dialogs based on their underlying intents. Without any supervision, we could flexibly obtain decent recall on precluding queries that do not belong in task-specific domains.
References


Li-Ming Zhan, Haowen Liang, Bo Liu, Lu Fan, Xiaoming Wu, and Albert Y. S. Lam. 2021. Out-of-scope intent detection with self-supervision and discriminative training.

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

Jianguo Zhang, Trung Bui, Seunghyun Yoon, Xiang Chen, Zhiwei Liu, Congying Xia, Quan Hung Tran, Walter Chang, and Philip Yu. 2021b. Few-shot intent detection via contrastive pre-training and fine-tuning.


Zheng Zhang, Ryuichi Takanobu, Qi Zhu, Minlie Huang, and Xiaoyan Zhu. 2020b. Recent advances and challenges in task-oriented dialog system.