Towards Zero- and Few-shot Knowledge-seeking Turn Detection in Task-oriented Dialogue Systems

Di Jin, Shuyang Gao, Seokhwan Kim, Yang Liu, Dilek Hakkani-Tur
Amazon Alexa AI
Sunnvayle, CA 94089
{djinamzn,shuyag,seokhwk,yangliud,hakkanit}@amazon.com

Abstract

Most prior work on task-oriented dialogue systems is restricted to supporting domain APIs. However, users may have requests that are out of the scope of these APIs. This work focuses on identifying such user requests. Existing methods for this task mainly rely on fine-tuning pre-trained models on large annotated data. We propose a novel method, REDE, based on adaptive representation learning and density estimation. REDE can be applied to zero/few-shots cases, and quickly learn a high-performing detector that is comparable to the full-supervision setting with only a few shots by updating less than 3K parameters. We demonstrate REDE's competitive performance on DSTC9 Track 1 dataset and our newly collected test set.

1 Introduction

Current task-oriented dialog systems often rely on pre-defined APIs to complete target tasks [20, 4] and filter out any other requests beyond the APIs as out-of-domain cases. However, some of these out-of-domain user requests can be addressed by incorporating external domain knowledge from the web or any other sources [11]. To address this problem, [12] recently organized a benchmark challenge on task-oriented conversational modeling with unstructured knowledge access in DSTC9 [8]. This challenge includes the knowledge-seeking turn detection task to determine whether to invoke a knowledge-driven responder or just rely on available API functions. Table 1 provides one data sample, where the user utterance of turn 5 cannot be addressed by APIs and thus needs to be identified as a knowledge-seeking turn to invoke the subsequent unstructured knowledge-grounded responder. The state-of-the-art systems [9, 19, 14] implemented this detector by fine-tuning a large pre-trained model on the training dataset (about 72K samples) as a binary classifier, and achieved an F1 score of over 95% on the benchmark test set. However, after close investigation, we find those user queries in the test set are very limited in topic coverage and language variation. To evaluate the detector performance on real-world user queries, we specially curate a new contrast set following [7] by manually collecting questions posted by real users on Tripadvisor forums. We found that the detector trained on DSTC9 Track 1 training samples had a large performance degradation on this contrast set (F1 score dropped by over 15%), suggesting the need for methods with better generalization.

In this work, we propose a method that can quickly learn a knowledge-seeking turn detector with much fewer out-of-domain samples, such as only a few shots or even zero shot. Our method is composed of two stages: **RE**presentation learning and **DE**nsity estimation (REDE). First, we learn a representation model via fine-tuning a pre-trained sentence encoder on all non-knowledge-seeking turns (utterances that can be supported by APIs) via masked language modeling (MLM). Then we learn a density estimator using these representation vectors. During inference, the density estimator produces a density score for a given user utterance. If it is above a threshold, this utterance is counted as an in-domain API turn, otherwise as a knowledge-seeking turn. To incorporate out-of-domain examples, we propose to use principle component analysis to quickly learn a projection matrix with few knowledge-seeking turn samples and then use this matrix to linearly transform the representation

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

Table 1: One example of task-oriented conversations with unstructured knowledge access. The most appropriate FAQ pair for answering turn 5 is highlighted in bold font.

Turn	Speaker	Utterance	Sampled Knowledge Snippets from FAQs
1	User	I'm looking for a place to stay in the south of town. It doesn't need to have free parking.	
2	Agent	There are 4 hotels that are in the area you are looking for. Would you prefer a 3 or 4 star rated hotel?	
3	User	I don't care about the star rating as long as it's expensive.	
4	Agent	The ***Lensfield Hotel*** is the only expensive hotel in the south area. Would you like any more information on this location?	
5	User	I'm interested in knowing, do they have a workout facility on the premises?	
6	Agent	There are both a fitness center and gym available on the premises. Does this sound ok?	Q1: Do you have room service for your guests? A1: Yes, the Lensfield Hotel provides room services. **Q2: Is there a gym available at your location?** **A2: There is both a fitness center and gym available on the premises.**
7	User	That is perfect can you book that for me please.	
8	Agent	The Lensfield Hotel is located in the South. It has a 3 star rating and is expensive. There is free parking and internet. I have booked it for you.	
9	User	Great, thank you!	

vectors. We conduct experiments on the DSTC9 Track 1 data as well as our new contrast test set. We demonstrate that REDE can achieve competitive performance as other supervised methods in the full-shot setting and outperform them by a large margin in the low-resource setting. More importantly, our approach generalizes much better in the new contrast test set that we created.

2 Related Work

Our work is closely related to those participating systems in DSTC9 Track 1 [11, 12]. All the systems proposed to treat the problem of knowledge-seeking turn detection as a binary classification task and fine-tuned pre-trained models such as RoBERTa, UniLM, PLATO, GPT2, on the whole training set [9, 19, 14], which yielded around 99% and 96% F1 scores on the development and test sets, respectively. Our method differs in two aspects: 1) We do not need to fine-tune the pre-trained model on the training set with labels in a supervised manner, instead, we only need the non-knowledge-seeking turns in the training set and train the pre-trained model on them in an unsupervised way; 2) Our model is at least 5 times smaller and we need less than 5% of training data to achieve similar performance.

Our method is inspired by previous work for out-of-domain (OOD) detection [16, 5, 10] and one-class classification [17]. [11] also tried tackling this problem by applying an unsupervised anomaly detection algorithm, Local Outlier Factor (LOF) [1], which compares the local densities between a given input instance and its nearest neighbors, but did not obtain good results (F1 score is less than 50%). [17] proposed to first learn a representation model via contrastive learning, then learn a density estimator on the obtained representations. They showed decent performance for one-class classification. All these previous work assumed no access to OOD samples, however, we would like to make use of those OOD samples efficiently when they are available. Therefore we extend the general representation learning framework by proposing a novel representation transformation method to learn OOD samples, which leads to significantly boosted detection performance.

3 Methods

3.1 Encoder Adaptation

In this step, we adapt a pre-trained sentence encoder E to the in-domain data, i.e., non-knowledge-seeking turns, $X^{NK} = \{x_1^{NK}, ..., x_N^{NK}\}$, by training the encoder on non-knowledge-seeking turns to optimize the masked language modeling objective [2]. Specifically, 15% of tokens of x_i^{NK} are masked, among which 80% are replaced by a special token "[MASK]", 10% are kept the same, and 10% are randomly replaced with other words in the vocabulary, and then E is trained to predict these masked tokens.

3.2 Representation Transformation

To incorporate the knowledge-seeking turns $X^K = \{x_1^K, ..., x_M^K\}$, a standard solution is to fine-tune E on the combined data of knowledge-seeking and non-knowledge-seeking turns, $X = X^K \cup X^{NK}$, as a supervised binary classifier. However, in few-shot settings where $M << N$, there is an extreme class imbalance problem. In addition, fine-tuning large models may take a long time and much computation power with large data size. Instead, we propose a simple linear transformation to the sentence representation $e = E(x)$ without updating the model parameters, following [18]:

$$\tilde{e} = T(e) = (e - \mu)W \tag{1}$$

where $\mu = \frac{1}{M}\sum_{i=1}^{M} E(x_i^K)$. To calculate W, we first calculate the covariance matrix, $\Sigma = \frac{1}{M}\sum_{i=1}^{M}(E(x_i^K) - \mu)^T(E(x_i^K) - \mu)$, then perform Singular Value Decomposition (SVD) over Σ such that: $\Sigma = U\Lambda U^T$, and finally we obtain $W = U\sqrt{\Lambda^{-1}}$.

The elements in diagonal matrix Λ derived from SVD are sorted in descending order. Therefore, we can retain the first L columns of W to reduce the dimension of transformed vectors $\tilde{e}$, which is theoretically equivalent to Principal Component Analysis (PCA). However, to be noted, both μ and W parameters are obtained using those knowledge-seeking turns instead of non-knowledge-seeking turns and the number of knowledge-seeking turns is much smaller, which can be as small as just a few shots. In another word, we only need a very small size of out-of-domain samples to learn the parameters needed for our representation transformation as defined in Eq. 1 to transform the representations of in-domain data. This is in contrast to the conventional PCA based density estimation method that assumes only having access to in-domain data, i.e. non-knowledge-seeking turns, and needs to learn and perform PCA transformation both on a good amount of those in-domain data.

3.3 Density estimation

In this step, we encode all the non-knowledge-seeking turns in the training set and transform them to obtain $\{\tilde{e}_1^{NK}, ..., \tilde{e}_N^{NK}\}$, normalize them into unit vectors, and then learn a shallow density estimator D over them, such as Gaussian Mixture Model (GMM). Note that in the zero-shot setting when no knowledge-seeking turns are available, the representation transformation step (in Section 3.2) is skipped.

During inference, given a test sample x, we encode it with the encoder E, transform it with T defined in Eq. 1, and then use the learned density estimator D to produce a density score $D(T(E(x)))$. If it is above a pre-set threshold η, x is considered as a non-knowledge-seeking turn, otherwise as a knowledge-seeking turn. This whole pipeline is motivated by the assumption that the well learned representations of in-domain (non-knowledge-seeking turns) and OOD samples (knowledge-seeking turns) should be distributed separately in the latent space, and thus the estimated density of in-domain data by the density estimator should be higher than that of OOD data.

4 Experiments

4.1 Dataset

We use the DSTC9 Track 1 competition data [11, 12], which is an augmented version of MultiWOZ 2.1 [3] with out-of-API-coverage turns grounded on external knowledge sources beyond the original database entries.[1] It contains three sub-tasks: (1) knowledge-seeking turn detection, (2) knowledge selection, and (3) knowledge-grounded response generation. Table 1 shows an example conversation. The user utterance at turn $t = 5$ requests the information about the gym facility, which is out of the coverage of the structured domain APIs and thus needs to be identified as a knowledge-seeking turn (knowledge-seeking turn detection sub-task). The relevant knowledge contents can be found from external sources as in the rightmost column which includes sampled QA snippets from the FAQ lists. With access to these unstructured external knowledge sources, the agent manages to continue the conversation with no friction by selecting the most appropriate knowledge (knowledge selection sub-task) and generating proper responses based on the selected knowledge (knowledge-grounded response generation sub-task). In this work, we focus on sub-task 1: knowledge-seeking turn detection. The data statistics of it are summarized in Table 2.

[1] Data can be downloaded from: https://github.com/alexa/alexa-with-dstc9-track1-dataset

Table 3: Performance on the original test set and contrast set when all knowledge-seeking turns data are used for training. Trainable parameters refer to those parameters that are updated for learning knowledge-seeking turns.

Learning Schema	Sentence Encoder	Model size	Trainable Parameters	Test Set (%)			Contrast Set (%)		
				P	R	F1	P	R	F1
Standard Fine-tuning	RoBERTa-Large	355M	355M	99.19	92.88	95.93	96.61	69.37	80.75
	RoBERTa-Large-NLI	355M	355M	99.46	92.28	95.73	97.54	64.18	77.42
	DistilBERT-Base-NLI-STSB	66M	66M	98.92	92.78	95.75	95.36	66.67	78.44
REDE	DistilBERT-Base-NLI-STSB	66M	3K	97.76	94.65	**96.18**	86.98	94.17	**90.43**

We further curate a new contrast test set by first collecting questions posted by real users in the Tripadvisor forums , then obtaining the questions that cannot be addressed by MultiWOZ API schema [3] (this schema was also used for constructing the DSTC9 Track 1 dataset) as knowledge-seeking turns, and finally manually doing minimal edits on them to make them more like dialogue utterances if needed. We obtained 617 knowledge-seeking turns and mixed them with those non-knowledge-seeking turns in the original test set to form the contrast set. We provide several data samples in the Appendix.

Table 2: Statistics of the knowledge-seeking turn detection benchmark dataset. Pos: knowledge-seeking turns; Neg: non-knowledge-seeking turns.

	Pos	Neg	All
Train Set	19,184	52,164	71,348
Valid Set	2,673	6,990	9,663
Test Set	1,981	2,200	4,181
Contrast Set	617	2,200	2,817

4.2 Baselines and Settings

The baselines are 1) the best performing model in the DSTC9 Track 1 competition [12], which is a fine-tuned RoBERTa-Large model [13] on the training set. 2) Fine-tuned RoBERTa-Large-NLI (obtained by fine-tuning RoBERTa-Large on SNLI and MultiNLI datasets) and DistilBERT-Base-NLI-STSB (obtained by fine-tuning DistilBERT-Base on SNLI, MultiNLI, and STS-B datasets) on the training set.

The sentence encoder E we used is DistilBERT-Base-NLI-STSB [15].[2] The threshold η is chosen based on the highest F1 score on the development set. For the density estimator, we have tried OC-SVM, KDE with various kinds of kernels, and GMM, and we find GMM performs the best and its inference time is the lowest. We set the number of components to 1 for GMM. Dimensionality L is set as 650 for PCA transformation by tuning on the development set. Details of comparison and tuning results are in the appendix. For evaluation metrics, we report precision (P), recall (R), and F1 scores.

5 Results & Discussion

5.1 Main Results

Full supervised setting Table 3 summarizes the comparison of our method REDE with baselines where all knowledge-seeking turn samples in the DSTC9 Track 1 training set are used for training. REDE has two advantages: (1) Once the first step of adaptive pre-training on non-knowledge-seeking turns is done, it only needs to update less than 3K parameters of the density estimator for learning the knowledge-seeking turns, but it can still achieve superior performance on the test set; (2) It can be better generalized to the new contrast set that has distribution shift with respect to the training data.

Low-resource setting We are more interested in exploring how our method performs under the low-resource setting compared with baselines. Therefore, we sub-sampled different numbers of knowledge-seeking turn samples and kept using all non-knowledge-seeking turn samples. We then trained the model and obtained F1 scores on the test set. We performed five times of random sub-sampling and report the average and standard deviation in Figure 1. We put the full results including the average F1 score and its standard deviation obtained under 5 random sub-sampling

[2] https://github.com/UKPLab/sentence-transformers

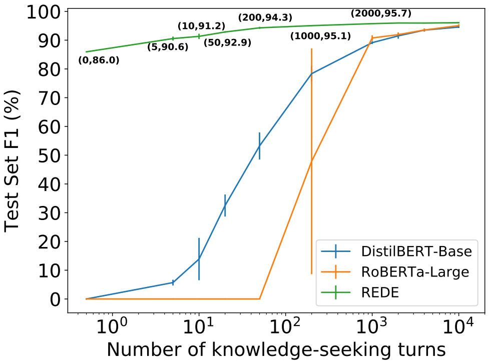


Figure 1: F1 score plots with error bar on the test set with different numbers of knowledge-seeking turns used for training.

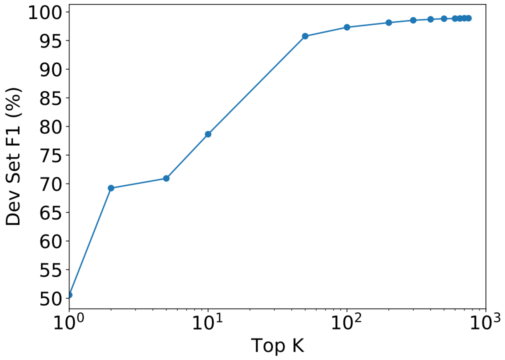


Figure 2: Development set F1 scores by retraining different values of first L columns of W. Full dimension is 768.

in the Appendix. As we can see, REDE is always superior than baselines for all sub-sampling ratios. The performance gap is larger when fewer examples are used. Most notably, for the zero-shot setting without using any knowledge-seeking turns, REDE can still achieve 85.95% of F1 score. For comparison, in the zero-shot setting, we also tested Local Outlier Factor (LOF) [1], which was used in [11], and obtained an F1 score of 73.78% on the test set, which is much lower than our proposed density estimation method. Under the few-shots setting such as 5-shots and 10-shots, REDE can obtain more than 90% of F1, whereas other supervised baselines' scores are under 20%. The extremely low performance of the supervised baselines could be because of the extreme data imbalance situation with data size ratio between two classes (positive and negative classes) being over 100. Notably, RoBERTa-Large performs even worse than DistilBERT-Base when the number of knowledge-seeking turns is less than 1,000 and its variance of performance between different runs is also larger, indicating that larger models do need more data to guarantee good enough fine-tuning performance. Overall, supervised fine-tuning fails in this case while our method REDE can still obtain high performance, indicating its robustness to the data imbalance problem.

5.2 Analysis

5.2.1 Effect of MLM Adaptation

As shown in Table 4, after removing the MLM adaptation step, our method has significant performance degradation, especially for the contrast set, indicating the importance of adapting the general pre-trained model to the target dataset via unsupervised learning. We have also tried adopting contrastive learning for such unsupervised adaptation (i.e., SimCSE), which has shown state-of-the-art performance for unsupervised representation learning [6]. Results in Table 4 show that it is worse than MLM.

Table 4: Ablation study for MLM adaptation by removing it or replacing it with SimCSE (a contrastive learning method). All training samples are used here.

Settings	Test Set F1	Contrast Set F1
REDE	**96.18**	**90.43**
no MLM	93.49	75.65
MLM → SimCSE	92.00	74.39

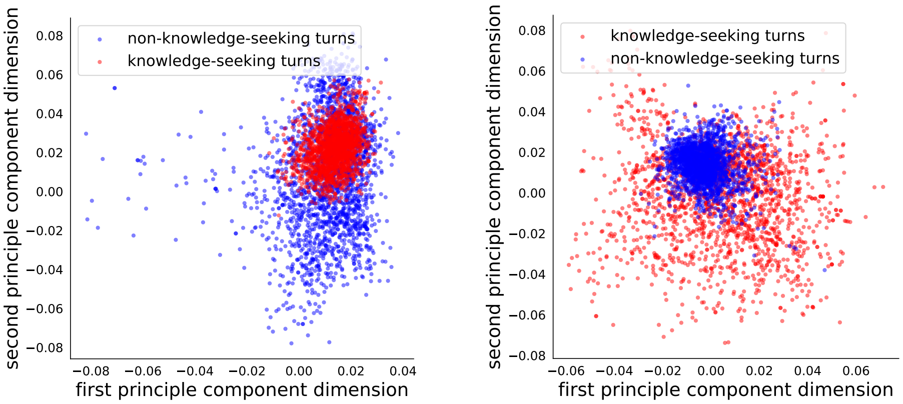
5.2.2 Comparison of Density Estimators

For the density estimator, we have tried OC-SVM, KDE with various kinds of kernels, and GMM, which are summarized in Table 5. All these estimators are implemented using Scikit-Learn library.[3] From Table 5, we see that GMM performs the best while being the fastest for inference, therefore we chose it as the density estimator in our work. We have also tried other kernels for the KDE estimator, such as 'tophat', 'epanechnikov', 'linear', and 'cosine', but they all perform poorly. As for the number of components for GMM, we empirically found 1 works the best and the full results of comparison can be found in the Appendix.

[3] https://scikit-learn.org/stable/

Table 5: Comparison of different density estimators. Inference time is measure on the whole test set using the same machine.

Estimator	Test Set (%)				Contrast Set (%)		
	P	R	F1	Inference Time (s)	P	R	F1
OC-SVM	92.30	88.34	90.28	74.36	68.81	88.33	77.36
KDE-Gaussian	92.81	91.17	91.98	377.43	72.49	88.82	79.83
KDE-Exponential	92.29	91.87	92.08	373.76	73.18	88.01	79.91
GMM	**97.76**	**94.65**	**96.18**	**0.07**	**86.98**	**94.17**	**90.43**



(a) PCA with non-knowledge-seeking turns, F1 = 68.59%.

(b) PCA with knowledge-seeking turns, F1 = 78.49%.

Figure 3: Scatter plot using top two principal components of PCA on test samples. F1 score is measured on the test set with top two dimensions only.

5.2.3 Effects of L

We can retrain only the first L columns of W for the PCA transformation, which can help us reduce the dimension of transformed representation vector $\tilde{e}$. Figure 2 shows the development set performance under different values of L when all knowledge-seeking turns are used for training. We see that the first 50 dimensions can achieve over 95% F1 score and 300 dimensions are already enough to realize the peak performance, whereas the full dimension is 768.

5.2.4 Understanding PCA Transformation

In Section 3.2, the sentence representation is transformed with PCA learned from knowledge-seeking turns. Table 6 shows the F1 score on the test set using top L principal components with PCA learned using different data. Overall, we can see that PCA with knowledge-seeking turns achieves better performance, and using more principal components is always beneficial. PCA is well-known to help construct new subspaces by maximizing the global variance. Intuitively, by learning PCA over knowledge-seeking turns, we expect the manifolds on knowledge-seeking turns to spread out and non-knowledge seeking turns condense. Figure 3 shows the scatter plot of the top two principal components of transformed features. In Figure 3a, we learn PCA from non-knowledge-seeking turns, which results in the manifold of knowledge-seeking turns (red dots) to be within that of non-knowledge-seeking turns (blue dots). It hurts the performance since the density estimation is performed over non-knowledge-seeking turns, as confirmed by the zero shot result in Table 6 in comparison to that in Fig 1. In contrast, in Figure 3b, we learn PCA with knowledge-seeking turns, which makes knowledge-seeking

Table 6: F1 score on test set for top 5, 50, 500, and all principal components under three different settings: zero-shot (PCA over non-knowledge-seeking turns), ten-shot, and full-shot.

Dimensions	Zero-shot	Ten-shot	Full-shot
Top 5	65.67	76.64	78.38
Top 50	71.04	82.23	92.40
Top 500	77.16	91.73	96.32
All (768)	77.05	92.37	96.09

turns (red dots) spread out and non-knowledge-seeking turns (blue dots) condense. By estimating the density of this condensed blue area, we obtain higher F1 score because all the red dots falling outside of the region of blue dots will be classified as out-of-distribution correctly.

6 Conclusion

In this work, we propose a novel method REDE based on domain-adapted representation learning and density estimation for knowledge-seeking turn detection in tasked-orientated dialogue systems. Compared with previous SOTA models, REDE can achieve comparable performance in the full supervised setting and significantly superior performance for the low-resource setting. Besides, REDE has much better generalization capability onto a new contrast set we curated.

References

[1] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, June 2019. Association for Computational Linguistics.

[3] Mihail Eric, Rahul Goel, Shachi Paul, Adarsh Kumar, Abhishek Sethi, Peter Ku, Anuj Kumar Goyal, Sanchit Agarwal, Shuyang Gao, and Dilek Hakkani-Tur. Multiwoz 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. *arXiv preprint arXiv:1907.01669*, 2019.

[4] Mihail Eric, Lakshmi Krishnan, Francois Charette, and Christopher D. Manning. Key-value retrieval networks for task-oriented dialogue. In *Proceedings of the SIGDIAL 2017 Conference*, pages 37–49, 2017.

[5] Varun Gangal, Abhinav Arora, Arash Einolghozati, and Sonal Gupta. Likelihood ratios and generative classifiers for unsupervised out-of-domain detection in task oriented dialog. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7764–7771, 2020.

[6] Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*, 2021.

[7] Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323, Online, November 2020. Association for Computational Linguistics.

[8] R. Chulaka Gunasekara, Seokhwan Kim, Luis Fernando D'Haro, Abhinav Rastogi, Yun-Nung Chen, Mihail Eric, Behnam Hedayatnia, Karthik Gopalakrishnan, Yang Liu, Chao-Wei Huang, Dilek Hakkani-Tür, Jinchao Li, Qi Zhu, Lingxiao Luo, Lars Liden, Kaili Huang, Shahin Shayandeh, Runze Liang, Baolin Peng, Zheng Zhang, Swadheen Shukla, Minlie Huang, Jianfeng Gao, Shikib Mehri, Yulan Feng, Carla Gordon, Seyed Hossein Alavi, David R. Traum, Maxine Eskénazi, Ahmad Beirami, Eunjoon Cho, Paul A. Crook, Ankita De, Alborz Geramifard, Satwik Kottur, Seungwhan Moon, Shivani Poddar, and Rajen Subba. Overview of the ninth dialog system technology challenge: DSTC9. *CoRR*, abs/2011.06486, 2020.

[9] Huang He, Hua Lu, Siqi Bao, Fan Wang, Hua Wu, Zhengyu Niu, and Haifeng Wang. Learning to select external knowledge with multi-scale negative sampling. *arXiv preprint arXiv:2102.02096*, 2021.

[10] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019.

[11] Seokhwan Kim, Mihail Eric, Karthik Gopalakrishnan, Behnam Hedayatnia, Yang Liu, and Dilek Hakkani-Tur. Beyond domain APIs: Task-oriented conversational modeling with unstructured knowledge access. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 278–289, 1st virtual meeting, July 2020. Association for Computational Linguistics.

[12] Seokhwan Kim, Mihail Eric, Behnam Hedayatnia, Karthik Gopalakrishnan, Yang Liu, Chao-Wei Huang, and Dilek Hakkani-Tur. Beyond domain apis: Task-oriented conversational modeling with unstructured knowledge access track in dstc9. *arXiv preprint arXiv:2101.09276*, 2021.

[13] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.

[14] Haitao Mi, Qiyu Ren, Yinpei Dai, Yifan He, Jian Sun, Yongbin Li, Jing Zheng, and Peng Xu. Towards generalized models for beyond domain api task-oriented dialogue. *AAAI-21 DSTC9 Workshop*, 2021.

[15] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

[16] Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[17] Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, and Tomas Pfister. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*, 2021.

[18] Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*, 2021.

[19] Liang Tang, Qinghua Shang, Kaokao Lv, Zixi Fu, Shijiang Zhang, Chuanming Huang, and Zhuo Zhang. Radge relevance learning and generation evaluating method for task-oriented conversational system-anonymous version. *AAAI-21 DSTC9 Workshop*, 2021.

[20] Jason D. Williams, Kavosh Asadi, and Geoffrey Zweig. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, 2017.

A Contrast Set

Table 1 shows several data samples for the newly curated contrast set. These user queries collected from real users are much more diverse than those in the benchmark test set of DSTC9 Track 1 dataset since all knowledge-seeking turns in DSTC9 are collected by paraphrasing the given frequently asked questions which are limited in question types. Among these examples, the user query of "How much do you charge for parking?" is actually quite challenging for knowledge-seeking turn detection since this query is very close to one of the available API functions that is responsible for checking whether there is free parking. However, in order to answer this query, we still need to invoke the knowledge module to retrieve external unstructured knowledge.

Table 1: Examples of newly collected user questions in the contrast set. These user queries collected from real users are much more diverse than those in the benchmark test set of DSTC9 Track 1 dataset.

Domains	**Examples**
Attraction	Is it necessary to buy tickets in advance?
Attraction	How long it could take to see it all ? 4 hours it would be enough?
Hotel	Is there a minimum check in age?
Hotel	How much do you charge for parking?
Restaurant	Would there be room for a stroller with a sleeping baby during dinner?
Restaurant	Can I order crab cakes take out for eight servings ?

B Full Results of Low-resource Setting

We put the full results including the average F1 score and its standard deviation obtained under 5 random sub-sampling in Table 2. We find that our method REDE performs much better than those fine-tuned baselines under the low-resource setting as well as performs more consistently under different sub-sampling strategies.

Table 2: Averaged F1 score and standard deviation under the low-resource setting by randomly sub-sampling different number of knowledge-seeking turns for five times. DistillBERT is DistillBERT-Base-NLI-STSB while RoBERTa is RoBERTa-Large.

Samples	DistillBERT	RoBERTa	REDE
5	5.70 ± 0.99	0.00 ± 0.00	90.56 ± 0.69
10	13.88 ± 7.37	0.00 ± 0.00	91.34 ± 0.92
20	32.51 ± 3.84	0.00 ± 0.00	92.85 ± 0.26
50	53.22 ± 4.70	0.00 ± 0.00	94.29 ± 0.35
200	78.35 ± 2.57	47.87 ± 39.28	95.05 ± 0.20
1,000	89.14 ± 0.49	90.76 ± 0.94	95.74 ± 0.13
2,000	91.49 ± 0.97	91.92 ± 0.79	95.97 ± 0.12
4,000	93.49 ± 0.53	93.55 ± 0.49	95.95 ± 0.18
10,000	94.56 ± 0.32	95.06 ± 0.46	96.07 ± 0.17

C Number of Components for GMM

Table 3 shows the performance under different number of components for the GMM density estimator. From it, we see that the number of components has minor influence on the performance so we decide to use 1 as the number of components in this work.

Table 3: Comparison of performance (in percentage) by using different number of components for the GMM estimator.

Components #	Dev Set F1	Test Set F1	Contrast Set F1
1	98.71	96.18	90.43
2	98.88	95.82	90.61
3	98.97	96.12	90.35
4	99.03	96.04	89.72