Semi-supervised Intent Discovery with Contrastive Learning

Xiang Shen Yinge Sun Yao Zhang Mani Najmabadi
Expedia Group
{xianshen, yinsun, yaozhang, mnajmabadi}@expediagroup.com

Abstract

User intent discovery is a key step in developing a Natural Language Understanding (NLU) module at the core of any modern Conversational AI system. Typically, human experts review a representative sample of user input data to discover new intents, which is subjective, costly, and error-prone. In this work, we aim to assist the NLU developers by presenting a novel method for discovering new intents at scale given a corpus of utterances. Our method utilizes supervised contrastive learning to leverage information from a domain-relevant, already labeled dataset and identifies new intents in the corpus at hand using unsupervised K-means clustering. Our method outperforms the state-of-the-art by a large margin up to 2% and 13% on two benchmark datasets, measured by clustering accuracy. Furthermore, we apply our method on a large dataset from the travel domain to demonstrate its effectiveness on a real-world use case.

1 Introduction

Conversational AI systems such as chatbots and virtual assistants are widely used in a variety of applications to assist users. Natural Language Understanding (NLU), as an integral part of a conversational AI system, is the process of classifying the user's input into a set of pre-defined categories, generally referred to as intents. In most applications, the task of NLU is achieved by developing a supervised text classification model. Understanding and identifying user intents is key to developing intent classification models as it directly impacts the performance of the system.

Generally, the set of intents that can be recognized by the model is defined by human experts based on domain knowledge and business requirements. This process usually requires a significant amount of effort to manually review large-scale user input data. In addition, this task becomes increasingly complex as the number of potential intents in the dataset grows.

To address these challenges, intent discovery methods that aim to detect new user intents, either automatically or semi-automatically, from a large number of unlabeled utterances, have been developed in recent years.

A popular approach for intent discovery is based on unsupervised clustering algorithms. For instance, Shi et al. (2018) conduct hierarchical clustering on sentence representations from autoencoder based on a combination of word embeddings, POS tagging, keywords, and topic modeling. Vedula et al. (2019, 2020) establish a multi-stage procedure which detects out-of-domain utterances with a classification model and then group the text with unknown intents by clustering. Chatterjee and Sengupta (2020) extend DBSCAN (Ester et al., 1996) to discover intents in conversations based on sentence embeddings from universal sentence encoder (Cer et al., 2018). However, the unsupervised approaches often fail to produce highly accurate and granular intents. As such, researchers have developed semi-supervised methods to leverage existing domain-relevant labeled data in order to improve the intent discovery results. Zhang et al. (2021) propose deep aligned clustering (DAC) in a semi-supervised framework to leverage the prior information of known intents. DAC uses BERT to generate sentence representations. However, BERT has poor performance with respect to generating semantically meaningful sentence representations (Reimers and Gurevych, 2019; Li et al., 2020). Alternatively, Sahay et al. (2021) use sentence-BERT (SBERT) to learn sentence representations but fail to achieve significantly better results using the DAC algorithm without adjusting their training methods for SBERT.

Proceedings of the Third Workshop on Natural Language Processing for Conversational AI, pages 120–129

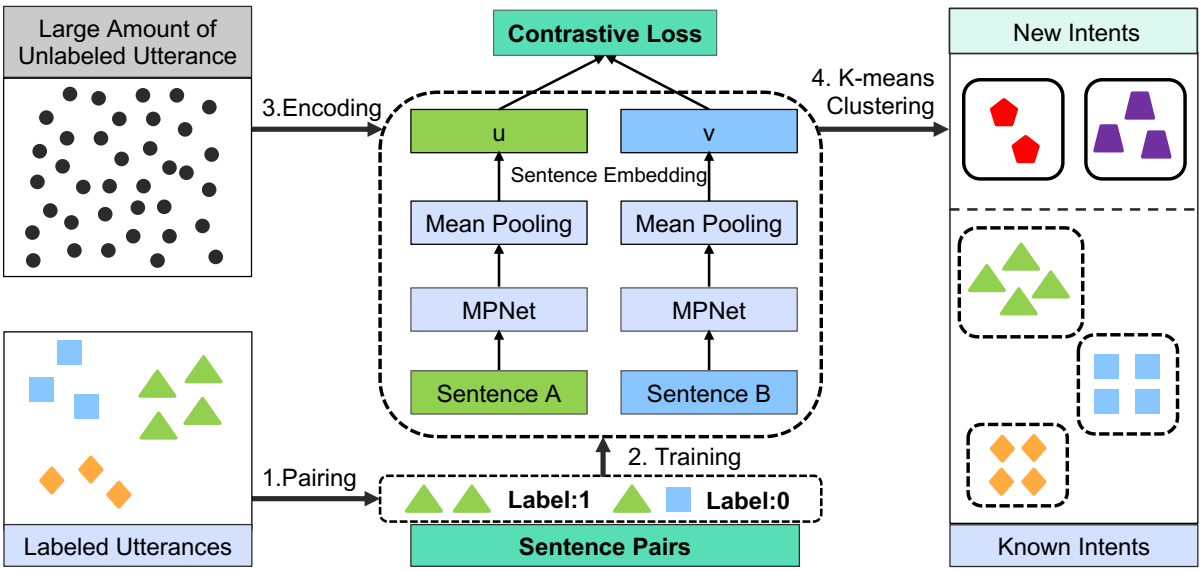

Figure 1: The architecture of the proposed method. First, we use labeled data to pair utterances based on their labels. Then, we train the sentence transformer, MPNet, in the Siamese Network structure with contrastive loss to learn sentence representations using sentence pairs in a supervised way. Next, we encode unlabeled data by the trained model. Finally, we estimate the optimal values of the number of clusters and perform K-means to discover new intents from identified clusters.

To address the limitations in the methods mentioned above and to further improve the results, we propose a novel semi-supervised method as illustrated in Figure 1. We use a Siamese Network (Reimers and Gurevych, 2019) to learn semantically meaningful sentence representations, in which a model is trained to differentiate pairs of sentences with the same intent and those with different intents. A pre-trained model that achieves the state-of-the-art performance among sentence transformers, MPNet (Song et al., 2020), is used in the Siamese Network structure combined with contrastive learning to learn similar and dissimilar representations for sentences with the same and different intents respectively. K-means algorithm is then applied to cluster the unlabeled utterances based on their sentence representations for discovering new intents at scale. In addition, we propose a novel way to determine the optimal number of clusters k based on the concept of clustering stability, where k selection solely depends on the dataset at hand as opposed to relying on any prior knowledge. Inspired by self-supervised learning with aligned pseudo labels introduced in DAC (Zhang et al., 2021), we further propose and experiment with four different pseudo label training (PLT) strategies in the setting of Siamese Network, and examine the impact of PLT in our experiments.

The contributions of our work are summarized as follows. First, we propose a novel semi-supervised framework that learns sentence representations effectively via supervised contrastive learning and discovers new intents at scale via unsupervised clustering. Second, we propose a simple and robust method based on clustering stability to determine the optimal number of clusters for K-means. Third, we conduct extensive experiments to show that our method achieves state-of-the-art performance on two public benchmark datasets, widely used for intent discovery. In addition, we successfully apply our methods to a real-world dataset from an application in the travel domain.

2 Related Work

In this section we review the previous work focused on the main components of the intent discovery process in more detail.

2.1 Sentence Representation

Sentence representation has a significant impact on the quality of the intent discovery results. A simple method is to employ mean pooling of word embeddings such as word2vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014), but it does not capture the information in the order of words. InferSent (Conneau et al., 2017) obtains sentence embeddings from pre-training bidirectional LSTM and max-pooling the output. Cer et al. (2018) in-

troduce universal sentence encoder (USE) which trains a transformer network on multiple tasks and achieves great performance on tasks through transfer learning.

A common approach to generate sentence embeddings using pre-trained transformer networks such as BERT (Devlin et al., 2019) is to extract the embeddings of CLS token, a special token inserted at the beginning of the input, from the last layer. However, this approach leads to anisotropic semantic space and performs poorly on textual similarity tasks (Reimers and Gurevych, 2019). Reimers and Gurevych (2019) demonstrate that averaging over the embeddings of all tokens improves the performance and further present a fine-tuned BERT in Siamese network architecture (SBERT) pre-trained on sentence pairs. Li et al. (2020) tackle the anisotropy issue and transform the embeddings to Gaussian distribution through a normalizing flow. Su et al. (2021) provide an even simpler solution to boost the performance based on the whitening of embeddings from BERT.

2.2 Contrastive Learning

Contrastive learning has demonstrated its ability to learn effective representations primarily in the computer vision field. In recent years, several studies have explored constructing sentence embeddings using contrastive learning. The core idea of contrastive learning is to create positive and negative sentence pairs such that representations of positive pairs are pulled together while negative pairs are pushed apart. Different strategies have been proposed to achieve this goal. Fang et al. (2020) utilize back-translation to perform sentence augmentation. Giorgi et al. (2020) sample text segmentations within the same document to construct sentence pairs. Gao et al. (2021) create positive instances by simply adding dropout masks to the original sentences.

2.3 Clustering

Clustering as an essential part of the intent discovery process identifies and groups similar sentence representations in an unsupervised setting. Popular algorithms include density-based methods (e.g., DBSCAN (Ester et al., 1996)) and centroid-based methods (e.g., K-means (MacQueen et al., 1967; Arthur and Vassilvitskii, 2006)). DBSCAN is not scalable for large datasets and not efficient for high-dimensional data. In contrast, K-means is a fast and scalable algorithm. Therefore, it is more suitable for identifying clusters from large datasets (usually thousands of or even hundreds of thousands of utterances) with high dimensional sentence representations (e.g., 768).

The K-means algorithm requires the number of clusters (k) to be provided by the user. There are several methods proposed in the literature that aim to identify the optimal k. The elbow method (Thorndike, 1953), the silhouette method (Rousseeuw, 1987), and the information criterion methods, such as Akaike information criterion (Akaike, 1974) and Bayesian information criterion (Schwarz, 1978), use simple measures to select k based on the tightness or separation of clusters. Ben-David et al. (2007); Levine and Domany (2001) introduce the notion of clustering stability to determine k that produces the most stable clustering results. DAC (Zhang et al., 2021) selects k by eliminating low confidence clusters from K' (a large pre-determined value), however, this method produces unstable clustering results that are highly dependent on the pre-determined value.

3 Our Approach

In this section, we describe the proposed method in detail. As shown in Figure 1, we start by constructing the sentence pairs from our labeled dataset. Then, we train a binary classifier with contrastive loss in a Siamese network structure which uses a pre-trained MPNet model to identify if two utterances have the same intent or not. Next, we encode a large number of unlabeled utterances using the trained MPNet. Finally, we find the optimal k for K-means based on the concept of clustering stability and group the utterances using K-means to obtain new intents.

3.1 Sentence Representation

To construct effective sentence embeddings, we train a MPNet with contrastive loss as a binary encoder using the labeled data. MPNet is a transformer-based language model which improves the pre-trained BERT model by introducing masked and permuted language models. The new pre-training technique maintains the advantages of the masked language modeling (MLM) from BERT as well as the permuted language modeling (PLM) from XLNet (Yang et al., 2019). In this study, we employ a pretrained version of MPNet model 'paraphase-mpnet-base-v2' from Reimers and Gurevych (2019).

While feeding a sentence with m different tokens to BERT or MPNet, token embeddings are extracted from the last hidden layer as $[CLS, T_1, T_2, \cdot, T_m]$ where CLS is a special token and short for classification. Then the sentence representation is obtained by applying mean-pooling of the token embeddings with fixed length:

$$u = \text{mean-pooling}([CLS, T_1, T_2, \cdot, T_m])$$

A binary classifier is added on the top of a sentence pair with representation u and v obtained from the MPNet model with the same weights. The binary classifier identifies if two sentences have the same intent or not. We transform the labeled utterances into sentence pairs in a way that sentences with the same intent are labeled 1 and labeled as 0 otherwise. In this case, a total number of N labeled utterances will lead to $N(N-1)/2$ sentence pairs. The supervised contrastive learning objective is to minimize the contrastive loss:

$$L = y*(D(u,v))^2 + (1-y)*(max(0, m - D(u,v))^2$$

Where $D(u, v)$ is the distance of u and v. This supervised contrastive learning (SCL) method aims to pull together utterances with the same intent and push apart utterances with different intents.

In the next step, we feed unlabeled utterances into the trained MPNet to obtain sentence embeddings for K-means clustering.

3.2 K-means Clustering

We use K-means to group sentences with similar sentence embeddings into clusters. Each resulting cluster consists of sentences with the same intent (either known or unknown). However, a key hyperparameter in K-means, the number of clusters k, is often unknown in practice due to lack of information about the corpus, e.g., the total number of intents in the corpus in our case. Therefore, we propose a novel method to determine the optimal value of k.

Our selection of k is based on the concept of clustering stability inspired by Ben-David et al. (2007); Levine and Domany (2001). The stability refers to the robustness of a clustering model to small perturbations in the data. Intuitively, the optimal number of clusters should generate a clustering model with high stability. In other words, if we repeat a clustering algorithm on different samples of the data, then the clustering results on those samples with the optimal k should be similar. We measure the similarity between a pair of different clustering results (taking one clustering result in the pair as the ground truth) by ACC, an unsupervised equivalent of classification accuracy. Given that different clustering results may have different labels for the same cluster, the Hungarian algorithm (Kuhn, 1955) is employed in the process of calculating ACC to map labels from one clustering to those from the other. We then define the stability score as the mean ACC across all pairs (the number of pairs is equal to the number of samples r), which is a function of k. The optimal value of k is estimated by the k with the highest stability score from a pre-defined range of $[k_{min}, k_{max}]$.

To further speed up the process of selecting the optimal value of k described above, we use Bayesian Optimization (BO), which is an efficient method to optimize functions of any forms and thus is suitable for hyperparameter tuning when the objective function is expensive to compute (Snoek et al., 2012). In BO, a surrogate model of the objective function is an easy-to-optimize probability model (commonly Gaussian Process) of hyperparameters. It is easy to find its extreme point (the hyperparameters that reach the minimum/maximum of the surrogate model) and then update the model

Algorithm 1 Select the number of clusters k

1: Draw r random samples (without replacement) from the complete training data, each containing $\beta\%$ messages.
2: **for** k in k_{min} to k_{max} with increment of s **do**
3: Apply K-means with k clusters on the complete training data.
4: **for** i in 1 to r **do**
5: Apply K-means with k clusters on the ith sample.
6: Calculate ACC between the clustering labels from the complete training data and those from the ith sample.
7: Calculate stability score for k.
8: Fit s pairs of (k, stability score at k) to a Gaussian Process model M_{GP}.
9: **for** t in 1 to T **do**
10: Predict k_{new} by model M_{GP}.
11: Calculate stability score at k_{new} (step 3-7 with k = k_{new}).
12: Update model M_{GP} with (k_{new}, stability score at k_{new}).
13: Select k with the highest stability score.

with the value of objective function at the extreme point. Such BO process conducts optimization-update at each step and thus enables faster convergence. We adopt Discrete-BO proposed in Luong et al. (2019) as the hyperparameter, number of clusters k, is a discrete variable. More specifically, instead of calculating the stability score (i.e., the objective function) for each k in $[k_{min}, k_{max}]$, we only calculate at k from $[k_{min}, k_{max}]$ in increments of s and build a surrogate model based upon those calculated values. Then we repeat the optimization-update step for T times. The procedure (named Clustering Stability with Bayesian Optimization) is summarized in Algorithm 1.

4 Experimentation

4.1 Datasets

We conducted experiments using two benchmark datasets, CLINC (Larson et al., 2019) and BANKING (Casanueva et al., 2020), which are also used by Zhang et al. (2021) and Sahay et al. (2021).

For our use case in travel domain, we collected utterances sent by travelers to Expedia virtual assistant. We used a labeled dataset of 3082 utterances as the training set and 512 utterances as the validation set. A random sample of 100,000 English utterances from one month in 2021 was used for intent discovery. Those utterances are minimally preprocessed by removing invalid characters and lowercasing letters.

The detailed statistics of the datasets is shown in Table 1.

4.2 Evaluation Setup

To evaluate the effectiveness of SCL on two benchmark datasets, we split the datasets into training, validation, and test sets, and use different known intent class ratios of 25%, 50%, and 75% to compare the performance of different methods. For each known intent, 10% of training data is randomly sampled and used as labeled data. We use the labeled training set to train MPNet, and select the best model with the validation set. While Zhang et al. (2021) uses a validation set which is much larger than the labeled training set, our validation set is a smaller portion of the original validation set. More specifically, we use 30% of the original validation data within known intents to maintain the number of sentence pairs with a training validation ratio around 9:1. We use the trained MPNet to calculate sentence embeddings for the test set and evaluate K-means clustering performance on the test set. The number of clusters is fixed as the ground-truth number of intents for fair comparison. We report the average results over five runs of experiments with different random seeds.

In addition, to evaluate the performance of our proposed method on the selection of k, we set k_{min} to $0.5 \cdot k_{true}$ and k_{max} to $1.5 \cdot k_{true}$ to cover a wide range of possible values of k. We set the increment $s = 10$ in Algorithm 1 to reduce the number of times the stability score needs to be calculated to 10%. We set $T = 10$ as we observe 1 usually converges in a few steps. We try various numbers of random samples r (i.e., 10, 20, 30, 40, 50) and observe similar trends and peaks for the stability scores. Therefore, we set $r = 10$ for all experiments to reduce the computation cost. In each of the random samples, we set the sampling ratio, $\beta = 80$ to ensure each sample is representative of the entire dataset.

We also evaluate the effectiveness of pseudo label training (PLT) inspired by Zhang et al. (2021). In PLT, labels are updated at the end of each epoch to learn better sentence representations over the course of training. We compare different strategies for PLT with our proposed baseline SCL. More specifically, we experiment with 4 strategies.

- Inclusive pairing: pair up all sentences from the entire training data by pseudo labels from K-means to continue training.
- Exclusive pairing: pair up sentences from the labeled training set by true labels and pair up sentences from the unlabeled training set by pseudo labels to continue training.
- Alignment-A: align pseudo labels from K-means to the true labels of the labeled training set by maximizing ACC between pseudo labels and true labels, and then replace aligned pseudo labels of the labeled training set with true labels to pair up sentences and continue training.
- Alignment-C: align pseudo labels from K-means to the true labels of the labeled training set by minimizing cluster centroid distance between pseudo labels and true labels, and then replace aligned pseudo labels of the labeled training set with true labels to pair up sentences and continue training.

Dataset	#Classes	#Training	#Validation	#Test	Vocabulary	Length (mean/max)
CLINC	150	18,000	2,250	2,250	7,283	8.31/28
BANKING	77	9,003	1,000	3,080	5,028	11.91/79
TRAVEL	18	3,082	512	–	2,624	12.98/184

Table 1: Statistics of datasets.

		CLINC			BANKING		
	Method	ACC	ARI	NMI	ACC	ARI	NMI
25%	BERT	58.28	44.91	80.23	40.92	27.64	61.53
	DAC	**75.20**	**65.36**	**89.12**	47.58	35.49	68.88
	SMPNET	68.37	58.54	87.68	56.97	45.11	76.33
	SCL	71.23	62.02	88.30	**58.73**	**47.47**	**76.79**
50%	BERT	70.75	59.61	86.33	57.48	44.20	73.02
	DAC	**80.70**	**72.26**	**91.50**	59.44	47.07	76.14
	SMPNET	73.49	64.81	89.75	62.80	50.94	78.28
	SCL	78.36	70.71	91.38	**67.28**	**55.50**	**80.25**
75%	BERT	80.62	72.17	91.06	67.11	54.20	78.59
	DAC	86.40	79.56	93.92	63.68	52.11	78.77
	SMPNET	83.24	74.28	93.39	71.82	58.82	82.22
	SCL	**86.91**	**81.64**	**94.75**	**76.55**	**65.43**	**85.04**

Table 2: The clustering results on two datasets at known class ratio of 25%, 50% and 75%.

The difference between inclusive pairing and exclusive pairing is that the former does not utilize the information of true labels from the labeled training set while the latter does but it never pairs up sentences from the labeled training set with sentences from the unlabeled training set due to the mismatch between true labels and pseudo labels. In addition, alignment-A and alignment-C are two ways to tackle the mismatch problem and then leverage the true label information and to train with all pairs. We freeze the first 11 layers of MPNet to speed up the pseudo label training process and preserve universal features learned from SCL.

When applying our method to travel domain data, we first use the labeled data to train SCL. Then, we run our k selection algorithm on 100,000 unlabeled data to determine k. Finally, we apply K-means with the selected k to unlabeled data to get the intent clusters.

4.3 Evaluation Metrics

We employ three standard metrics for evaluating the performance of clustering: Clustering Accuracy (ACC), Normalized Mutual Information (NMI), and Adjusted Rand Index (ARI). To calculate ACC, Hungarian algorithm is used to map the clustering labels to the true labels.

For our travel domain data, we visualize the clustering results by t-SNE and manually review the utterances within each cluster to discover new intents.

5 Results and Discussion

5.1 Effectiveness of SCL

In this section, we compare our method SCL with three baseline methods, BERT, DAC and SMPNet. BERT refers to using the BERT model trained with labeled data to generate sentence representations on unlabeled data for K-means clustering. DAC is considered as the current state-of-the-art in the area of intent discovery. DAC uses pseudo labels generated by clustering to train BERT iteratively in a self-supervised learning way while BERT does not use clustering predictions for further training. SMPNet is simply trained by Siamese MPNet with sentence pairs using cosine similarity loss instead of contrastive loss used in SCL. SMPNet is included here to show the effectiveness of contrastive learning. We conduct experiments on two benchmark datasets described in Section 4.1. As shown in Table 2, our method consistently achieves better clustering performance compared to DAC by a large margin on the BANKING dataset across different known label ratios with respect to all 3 metrics. Our method also achieves comparable re-

	Method	CLINC (k = 150)					BANKING (k = 77)				
		$\hat{k}$	Error	ACC	ARI	NMI	$\hat{k}$	Error	ACC	ARI	NMI
25%	DAC	123	18.00	**67.22**	**59.28**	**87.50**	**63**	**18.18**	47.13	35.83	69.09
	SCL	**131**	**12.67**	66.05	57.86	87.09	47	38.96	**48.41**	**39.81**	**73.07**
50%	DAC	126	16.00	72.16	65.70	90.01	**64**	**16.88**	56.92	46.00	75.76
	SCL	**141**	**6.00**	**77.40**	**69.49**	**91.14**	60	22.08	**59.22**	**49.49**	**78.24**
75%	DAC	129	14.00	77.03	71.88	92.35	65	15.58	61.94	51.35	78.73
	SCL	**151**	**0.67**	**86.90**	**81.57**	**94.72**	**69**	**10.39**	**73.03**	**62.45**	**84.25**

Table 3: The results of k selection at known class ratio of 25%, 50% and 75%.

			CLINC			BANKING		
		PLT strategy	ACC	ARI	NMI	ACC	ARI	NMI
25%	SCL only	-	71.23	62.02	88.30	58.73	47.47	76.79
	SCL+PLT	Inclusive pairing	72.86	63.88	88.99	59.47	47.66	76.69
		Exclusive pairing	73.01	64.05	89.15	60.29	48.89	**77.26**
		Alignment-A	**73.77**	**64.78**	**89.31**	59.10	47.69	76.85
		Alignment-C	72.68	63.82	89.04	**61.09**	**48.91**	77.25
50%	SCL only	-	78.36	70.71	91.38	**67.28**	**55.50**	**80.25**
	SCL+PLT	Inclusive pairing	79.58	72.36	91.91	66.12	55.14	80.24
		Exclusive pairing	**80.59**	**73.25**	**92.21**	66.21	54.86	80.18
		Alignment-A	79.89	72.62	91.96	66.81	55.45	80.40
		Alignment-C	78.28	70.56	91.39	66.39	54.80	80.10
75%	SCL only	-	86.91	81.64	94.75	**76.55**	**65.43**	**85.04**
	SCL+PLT	Inclusive pairing	88.28	82.32	94.95	74.81	64.51	84.82
		Exclusive pairing	**88.68**	**83.44**	**95.25**	75.18	64.44	84.77
		Alignment-A	88.49	83.09	95.11	75.66	64.91	85.02
		Alignment-C	84.58	78.99	94.04	73.09	62.18	83.52

Table 4: The results of 4 different strategies for pseudo label training on two datasets at known class ratio of 25%, 50% and 75%.

sults on the CLINC dataset in which each intent has an equal number of utterances in training, validation, and test set. DAC performers slightly better on two settings in this experiment when the true k is provided. However, the advantages disappear when k is unknown, as demonstrated in the next subsection 5.2. In addition, it is to be noted that SCL significantly outperforms the baseline by 10% in ACC on the BANKING dataset, which is imbalanced and is more aligned with real-world cases. In comparison, the CLINC dataset has a perfect balance in terms of the number of data points for each intent. Our method, SCL, which utilizes information from each sentence pair provides more robust results when applied to imbalanced datasets.

We suppose the reasons for better results from our method include the following. First, constructing sentence pairs to train the Siamese network leverages the labeled data more effectively than only using individual sentences with labels to train a BERT classification model. Also, as the base model in our training, the pre-trained MPNet(i.e., 'paraphase-mpnet-base-v2') provides better sentence embedding performance than 'bert-base-uncased' with a similar model structure. Additionally, the contrastive loss function fits the training task better than the cosine similarity loss.

5.2 Selection of k

We compare the performance on the selection of k between our proposed method and DAC. More specifically, we calculate the error rate of the predicted k ($\hat{k}$) as well as ACC, ARI, and NMI for both methods at different known class ratios (25%, 50%, 75%) on the CLINC dataset and the BANKING dataset. Table 3 summarizes the average results over five runs of experiments with different random seeds. It shows our proposed method achieves significantly better results on estimating k than DAC by reducing the average error rate from 16.44%

to 10.36%, except for the BANKING dataset with 25% known classes. The reason that our method has a higher error rate on the BANKING dataset with 25% or 50% known classes is mainly due to the imbalance of the dataset in which small clusters with related intents tend to be grouped together as one large cluster. It is worth noting that the ACC, ARI, and NMI derived from our predicted k are much higher than those from DAC. In the case of 75% known class ratio, the results are comparable to those from the ground-truth k. That is compelling evidence that our proposed method works better in real-world cases when the number of intents is unknown.

5.3 Pseudo Label Training

Table 4 shows the results of SCL and PLT with 4 strategies at different known class ratios. The performance on the CLINC dataset across 3 known class ratios is further improved by PLT using the first 3 strategies. The fourth strategy, alignment-C, does not perform as well as the other methods. There are two reasons that could explain this performance drawback. First, matching by cluster centroids does not achieve a high alignment ACC. Second, the error induced by mismatching propagates along with training epochs. In addition, the performance on the BANKING dataset at 25% known class ratio is further improved by all PLT strategies.

5.4 Performance on Travel Domain Data

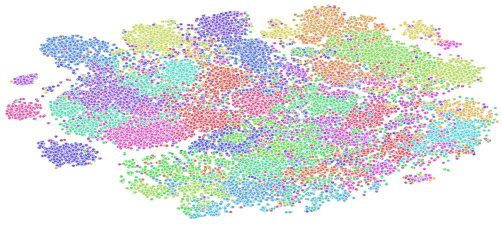

Figure 2: Visualization of the clustering result in 2D embeddings via t-SNE.

We apply our method to select k and obtain $k = 34$. In Figure 2, we further visualize the sentence representations of a large number of unlabeled utterances learned by SCL via t-SNE. Dots with different colors represent utterances in different clusters. Evidently, there is a clear margin between clusters captured by the 2D sentence representations learned by our model. We further analyze utterances within clusters and define new intents according to the business need. For confidentiality reasons, we do not disclose the details of new intents and example utterances.

6 Conclusion and Future Work

In this work, we propose a new semi-supervised framework to discover new intents by a sentence representation network via supervised contrastive learning followed by unsupervised K-means clustering. The method effectively leverages prior knowledge of existing intents to learn sentence representations and discovers new intents by grouping utterances with similar sentence representation. In the future, we will extend the work to discover intents with inherent hierarchy and automatically generate labels for new intents.

Acknowledgements

The authors wish to thank Matthew Fryer, Zoe Yang, and all members of the Conversational AI Team at Expedia Group for their support and review to this work. We also thank the anonymous reviewers for their valuable feedback.

References

Hirotugu Akaike. 1974. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723.

David Arthur and Sergei Vassilvitskii. 2006. k-means++: The advantages of careful seeding. Technical report, Stanford.

Shai Ben-David, Dávid Pál, and Hans Ulrich Simon. 2007. Stability of k-means clustering. In *International conference on computational learning theory*, pages 20–34. Springer.

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.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

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.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised

learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680.

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, Xiaowei Xu, et al. 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*, volume 96, pages 226–231.

Hongchao Fang, Sicheng Wang, Meng Zhou, Jiayuan Ding, and Pengtao Xie. 2020. Cert: Contrastive self-supervised learning for language understanding. *arXiv preprint arXiv:2005.12766*.

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

John M Giorgi, Osvald Nitski, Gary D. Bader, and Bo Wang. 2020. Declutr: Deep contrastive learning for unsupervised textual representations. *ArXiv*, abs/2006.03659.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, 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)*.

Erel Levine and Eytan Domany. 2001. Resampling method for unsupervised estimation of cluster validity. *Neural computation*, 13(11):2573–2593.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130.

Phuc Luong, Sunil Gupta, Dang Nguyen, Santu Rana, and Svetha Venkatesh. 2019. Bayesian optimization with discrete variables. In *Australasian Joint Conference on Artificial Intelligence*, pages 473–484. Springer.

James MacQueen et al. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

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*. Association for Computational Linguistics.

Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

Saurav Sahay, Eda Okur, Nagib Hakim, and Lama Nachman. 2021. Semi-supervised interactive intent labeling. In *Proceedings of the Second Workshop on Data Science with Human in the Loop: Language Advances*, pages 31–40.

Gideon Schwarz. 1978. Estimating the dimension of a model. *The annals of statistics*, pages 461–464.

Chen Shi, Qi Chen, Lei Sha, Sujian Li, Xu Sun, Houfeng Wang, and Lintao Zhang. 2018. Auto-dialabel: Labeling dialogue data with unsupervised learning. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 684–689.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. 2012. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *arXiv preprint arXiv:2004.09297*.

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

Robert L Thorndike. 1953. Who belongs in the family? *Psychometrika*, 18(4):267–276.

Nikhita Vedula, Rahul Gupta, Aman Alok, and Mukund Sridhar. 2020. Automatic discovery of novel intents & domains from text utterances. *arXiv preprint arXiv:2006.01208*.

Nikhita Vedula, Nedim Lipka, Pranav Maneriker, and Srinivasan Parthasarathy. 2019. Towards open intent discovery for conversational text. *arXiv preprint arXiv:1904.08524*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

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