The Devil is in the Details: On Models and Training Regimes for Few-Shot Intent Classification

Mohsen Mesgar[1*] **Thy Thy Tran**[1*] **Goran Glavaš**[2] **Iryna Gurevych**[1]

[1]**Ubiquitous Knowledge Processing Lab (UKP)**
Department of Computer Science and Hessian Center for AI (hessian.AI)
Technical University of Darmstadt
[2]CAIDAS, University of Würzburg, Germany
www.ukp.tu-darmstadt.de

Abstract

In task-oriented dialog (ToD) new intents emerge on regular basis, with a handful of available utterances at best. This renders effective Few-Shot Intent Classification (FSIC) a central challenge for modular ToD systems. Recent FSIC methods appear to be similar: they use pretrained language models (PLMs) to encode utterances and predominantly resort to nearest-neighbor-based inference. However, they also differ in major components: they start from different PLMs, use different encoding architectures and utterance similarity functions, and adopt different training regimes. Coupling of these vital components together with the lack of informative ablations prevents the identification of factors that drive the (reported) FSIC performance. We propose a unified framework to evaluate these components along the following key dimensions: (1) *Encoding architectures*: Cross-Encoder vs Bi-Encoders; (2) *Similarity function*: Parameterized (i.e., trainable) vs non-parameterized; (3) *Training regimes*: Episodic meta-learning vs conventional (i.e., non-episodic) training. Our experimental results on seven FSIC benchmarks reveal three new important findings. First, the unexplored combination of cross-encoder architecture and episodic meta-learning consistently yields the best FSIC performance. Second, episodic training substantially outperforms its non-episodic counterpart. Finally, we show that splitting episodes into support and query sets has a limited and inconsistent effect on performance. Our findings show the importance of ablations and fair comparisons in FSIC. We publicly release our code and data[1].

1 Introduction

Intent classification deals with assigning one label from a predefined set of classes or *intents* to user utterances. This task is vital for task-oriented dialog (ToD) systems since the predicted intent of an utterance is an essential input to other modules (i.e., dialog management) in these systems (Ma et al., 2022; Louvan and Magnini, 2020; Razumovskaia et al., 2021). Although intent classification has been widely studied, it still represents a challenge in settings where dialogue systems, including their intent classifiers, need to have the ability to be quickly adjusted to new domains and intent classes. The main challenges in training intent classifiers in such settings lies in the costly labeling of utterances (Zhang et al., 2022a; Wen et al., 2017; Budzianowski et al., 2018; Rastogi et al., 2020; Hung et al., 2022; Mueller et al., 2022). Few-shot intent classification (FSIC), which deals with adjusting intent classifiers to new intents given only a handful of labeled instances, is thus of paramount importance for ToD systems.

Various methods (§2) for FSIC have been proposed (Larson et al., 2019a; Casanueva et al., 2020a; Zhang et al., 2020; Mehri et al., 2020; Krone et al., 2020; Casanueva et al., 2020b; Nguyen et al., 2020; Zhang et al., 2021; Dopierre et al., 2021; Vulić et al., 2021; Zhang et al., 2022b). These methods are generally similar in that they utilize pretrained language models (PLMs) to encode utterances and resort to k nearest neighbors (kNN) inference: the label of a new instance is determined based on the labels of instances with which it has the highest representational similarity, as encoded by the PLM. Despite these general similarities, FSIC methods differ in design choices across several crucial dimensions, including encoding architectures, utterance similarity scoring, and training regimes. These methods tie together what are, in principle, independent design decisions across these dimensions, hindering ablations and insights into what drives the (reported) FSIC performance.

In this work, we (1) induce a framework (PLM-based utterance encoding, utterance similarity scor-

[*] Equal contribution.
[1] https://github.com/UKPLab/eacl2023-few-shot-intent-classification

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 1846–1857
May 2-6, 2023

ing, and nearest-neighbor-based inference) that unifies most of existing FSIC approaches (§3); and (2) focus on three key design decisions within this framework: **(1)** *model architecture for encoding utterances (or utterance pairs)*, where we contrast the less frequently adopted Cross-Encoder architecture (e.g., (Vulić et al., 2021)) against the more common Bi-Encoder architecture[2] (Zhang et al., 2020; Krone et al., 2020; Zhang et al., 2021); **(2)** *similarity function* for scoring utterance pairs based on their joint or separate representations, contrasting the parameterized (i.e., trainable) neural scoring components against cosine similarity as the simple non-parameterized scoring function; and **(3)** *training regimes*, comparing the standard non-episodic training (adopted, e.g., by Zhang et al. (2021) or Vulić et al. (2021)) against the episodic meta-learning training (implemented, e.g., by Nguyen et al. (2020) or Krone et al. (2020)). Our framework lets us evaluate impacts of these three dimensions for different text encoders (e.g., BERT (Devlin et al., 2019) as a vanilla PLM and SimCSE (Gao et al., 2021) as the state-of-the-art sentence encoder) under the same evaluation setup (datasets, intent splits, evaluation protocols and measures) while controlling for confounding factors that impede direct comparison between the FSIC methods.

Our extensive experimental results on seven intent classification datasets reveal three new important findings. First, a Cross-Encoder coupled with episodic training, a previously unexplored FSIC combination, consistently yields best performance across all the datasets. Second, episodic meta-learning yields robust FSIC classifiers across the board: our results demonstrate that it is much more effective for FSIC than the conventional non-episodic training. Finally, although episodic meta-learning entails splitting utterances of an episode into a support and query set during training, we show, for the first time, that this does not generally have a positive effect on the FSIC performance.

In sum, our comparative evaluation over various design choices for key components of modern FSIC approaches raise the awareness about the importance of ablations and apple-to-apple comparison between complex FSIC systems that conflate several key design decisions. We hope that our findings pave the way for more deliberation in research (and in particular evaluation) for this crucial ToD task.

[2]Also known as Dual Encoder or Siamese Network.

2 Related Work

We focus on few-shot intent classification (FSIC) methods, which perform class inference for utterances based on the labels of nearest neighbor (kNN), either directly in the representation space of the PLM or according to a trained scorer of utterance pairs. We first describe the existing FSIC inference paradigms and explain why we focus on kNN-based methods. We then categorize the literature on FSIC approaches based on kNN-inference along the three key design dimensions.

Inference algorithms for FSIC. Classical methods (Xu and Sarikaya, 2013; Meng and Huang, 2018; Wang et al., 2019; Gupta et al., 2019) for FSIC use the maximum likelihood inference, where a vector representation of an utterance is projected by the classifier into a probability distribution over the intent classes. Training such probability distribution functions, in particular when they are modeled by neural networks, mostly requires a large number of utterances annotated with intent labels, which are infamously expensive to collect in scenarios where new intents emerge on regular basis. By relying on pretrained language models, more recent FSIC methods leverage the language competences they posses (i.e., encode) to alleviate the need for learning to produce probability distributions for a large number intent classes, commonly with a few instances. These recent FSIC methods (Krone et al., 2020; Casanueva et al., 2020b; Nguyen et al., 2020; Zhang et al., 2021; Dopierre et al., 2021; Vulić et al., 2021; Zhang et al., 2022b) instead exploit the similarities between utterance embeddings in the representation space of the (fine-tuned) PLM and infer the intents for new utterances from the labels of nearest neighbors (kNN-based). Since kNN-based methods in general report state-of-the-art performance for FSIC, our comparative empirical evaluation focuses on the design choices for models that adopt this inference algorithm.

Model architectures for encoding utterance pairs. A central design decision within the kNN-based FSIC framework is the choice of the model architecture for encoding utterances. The majority of the approaches (Zhang et al., 2020; Krone et al., 2020; Zhang et al., 2021; Xia et al., 2021) leverage the Bi-Encoder architecture (Bromley et al., 1993; Reimers and Gurevych, 2019a; Zhang et al., 2022a). The core idea of Bi-Encoders is that, given a collection of utterances, each utterance is inde-

pendently encoded by the PLM and mapped into a dense representation space. In such a space, similarities between pairs of utterances can be computed, with a parameterized (i.e., trainable) scoring function or a non-parameterized function such as dot product or cosine similarity. In contrast, some FSIC methods (Vulić et al., 2021; Zhang et al., 2020; Wang et al., 2021; Zhang et al., 2021) use the Cross-Encoder architecture, in which the two utterances are concatenated and encoded jointly by a pretrained text encoder, e.g., BERT (Devlin et al., 2019). The idea is to represent a pair of utterances together using a PLM, where each utterance becomes a context for the other. A Cross-Encoder thus does not produce an embedding for a single utterance but for a pair of utterances. In general, Bi-Encoders are more computationally efficient than Cross-Encoders because of the Bi-Encoder's ability to cache the representations of the candidates. In return, Cross-Encoders, by allowing tokens of one utterance to attend over the tokens of the other (and vice versa), capture better the semantic associations between utterances.

Similarity scoring function. A crucial component in nearest neighbor-based methods for FSIC is the function that produces a similarity score for a pair of utterances. Concerning this dimension of analysis, we categorize FSIC methods into two groups: (1) FSIC approaches that use parameterized (i.e., trainable) neural layers to estimate the similarity score between utterances (Zhou et al., 2022; Zhang et al., 2020; Xia et al., 2021); and (2) methods that rely on non-parameterized similarity metrics such as dot product, cosine similarity, and Euclidean distance (Sauer et al., 2022; Zhang et al., 2022a; Krone et al., 2020; Vulić et al., 2021; Zhang et al., 2022b; Xu et al., 2021; Zhang et al., 2021). Note that the Bi-Encoder architecture can be coupled with both, whereas the Cross-Encoder requires a parametrized scoring module.

Training strategy. To simulate FSIC, the best practice is to split an intent classification corpus into two disjoint sets of intent classes. In this way, one set includes high-resource intents for training of an FSIC classifier, and the other set includes low-resource intents for evaluating the classifier. Concerning the training strategy on the high-resource intents, FSIC methods can be divided into two clusters. One cluster of methods adopts meta-learning or episodic training (Zhang et al., 2022a; Nguyen et al., 2020; Krone et al., 2020). Under this training regime, the goal is to train a meta-learner that could be used to quickly adapt to any few-shot intent classification task with very few labeled examples. To do so, the set of high-resource intents are split to construct many episodes, where each episode is a few-shot intent classification task for a small number of intents. The other cluster includes methods (Zhang et al., 2021; Vulić et al., 2021; Xu et al., 2021; Xia et al., 2021; Zhang et al., 2020, 2021) that use conventional supervised (i.e., non-episodic) training. The non-episodic training simply fine-tunes the FSIC model using all samples from the high-resource intents of the training set.

3 Framework

We first unify formulations of the components we need for our framework. We then present their alternative configurations along our three central dimensions of comparison: (i) model architecture for encoding utterance pairs, (ii) functions for similarity scoring, and (iii) training regimes.

3.1 Nearest Neighbors Inference

Following previous work on FSIC (Zhang et al., 2020; Vulić et al., 2021), we cast the FSIC task as a sentence similarity task in which each intent is an implicit semantic class, captured by the representations of all the utterances associated with that intent. The task is then to find the most similar labeled utterances for the given query. During inference, the FSIC approach should deal with an N-way k-shot intent classification, where N is the number of intents and k is the number of labeled utterances given for each intent label.

Let q be a query utterance and $C = \{c_1, ..., c_n\}$ be a set of its labeled neighbors. The nearest neighbor inference relies on a similarity function, non-parameterized or trainable (which is learned on high-resource intents), to estimate the similarity score s_i between q and any c_i. The query's label $\hat{y}_q$ is inferred as the ground-truth label of the neighbor with the maximum similarity score (i.e., $k = 1$ in k-NN inference): $\hat{y} = y_k, k = \text{argmax}(\{s_1, ..., s_n\})$.

3.2 Model Architectures for Encoding Utterance Pairs

An encoder in an FSIC model represents a pair of a query and a neighbor (i.e., a labeled utterance) into vector $\mathbf{h}_{(q,c_i)} \in \mathbb{R}^d$. We formulate recently used encoders: Bi-Encoder and Cross-Encoder.

Bi-Encoder (BE). BE encodes a pair of utterances independently, deriving independent representations of the query and the neighbor utterance. In particular, for each utterance x in a pair, we pass, "[CLS] x", to a BERT-like PLM and use the representation of "[CLS]" to represent x. Worth noting that the parameters of the PLM are shared in BE.

Cross-Encoder (CE). Different from BE, CE encodes a pair of query q and neighbor c_i *jointly*. We concatenate q with each of its neighbors to form a set of query–neighbor pairs $P = \{(q, c_1), ..., (q, c_n)\}$. We then pass each pair from P as a sequence of tokens to a language model, which is pre-trained to represent the semantic relation between utterances. More formally, we feed a pair of utterances, "[CLS] q [SEP] c_i", to a BERT-like PLM and then use the representation of the "[CLS]" token as the representation of the pair.

3.3 Similarity Scoring Function

Given the pair representation, we compute the similarity between a query and a neighbor utterance by a *parameterized* or *non-parameterized* function.

PArameterized (PA). A neural-based parametric scoring function consists of a fully connected feed-forward network (FF) that transforms a pair representation into a score, $s_i = \sigma\left(\mathbf{W}^{1\times d}\mathbf{h}_{(q,c_i)} + b\right)$, where the weight $\mathbf{W}$ and bias b are trainable parameters, d is the size of the vector $\mathbf{h}_{(q,c_i)}$, and $\sigma(.)$ denotes the `sigmoid` activation function.

Non-Parameterized (NP). In contrast to PA, NP often uses vector-based similarity metrics as scoring functions, e.g., cosine similarity or Euclidean distance. Following Vulić et al. (2021), in this work we adopt the cosine similarity between h_q and h_{c_i}.

3.4 Model Configurations

Given the aforementioned components, we illustrate (Figure 1) three possible model configurations: (i) CE+PA; (ii) BE+PA, and (iii) BE+NP.

CE +PA. In this configuration, we feed the joint encoding of the utterance pair to a parameterized similarity scoring function. We note again, due to a single representation vector for both utterances, CE cannot be coupled with a non-parameterized scoring (NP).

BE +PA. In this configuration, we represent the pair by concatenating the representations of each utterance with the vectors of difference and element-wise product between those representations:

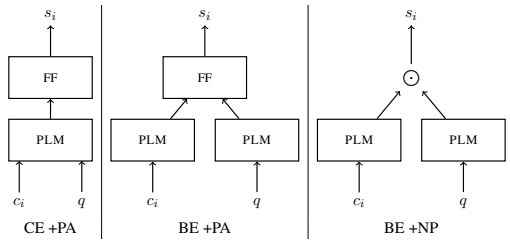

Figure 1: A demonstration of possible model configurations of encoder architectures and similarity functions to estimate the similarity score s_i between a query q and neighbor c_i. CE and BE show Cross-Encoder and Bi-Encoder architectures using a BERT-like PLM, respectively. PA and NP show parametric and non-parametric similarity functions, respectively. PA is modeled by feedforward (FF) layers and NP by the dot product $\odot$.

$$h_{(q,c_i)} = (h_q \oplus h_{c_i} \oplus |h_q - h_{c_i}| \oplus h_q \odot h_{c_i}), \quad (1)$$

where $\oplus$ is the concatenation operation and $\odot$ is the dot product. We motivate Equation 1 by the findings reported in Reimers and Gurevych (2019b). Similar to CE +PA, we use the sigmoid activation function on top of the feed-forward layer. The size of $\mathbf{W}$ is then $1 \times 4d$.

BE +NP. We use cosine similarity to estimate the similarity between input utterances during prediction. During training, we compute the dot product between the query and each neighbor representation vector to directly estimate their similarity scores $s_i = \sigma\left(h_q \odot h_{c_i}\right)$, where $\odot$ indicates the dot product, and σ is the `sigmoid` function. We apply σ to scale s_i to a value between 0 and 1.

3.5 Training Regimes

To train the aforementioned model configurations, we formulate three training techniques as follows (Figure 2): *Non-Episodic Training (NE)*, *Episodic Training (EP)* and *Episodic Training with Support and Query splits (EPSQ)*. The training strategies rely on an identical loss function for each query.

Loss per query sample. We use the loss function defined by Zhang et al. (2020) for FSIC. In particular, we define a ground-truth binary vector y_q for a query q given a set of neighbors $C = \{c_1, ..., c_n\}$. If the query and its i-th neighbor belong to the same intent class, the corresponding label for the pair is $y_{q,i} = 1$, otherwise $y_{q,i} = 0$. Given such ground-truth label vector in consideration of the n

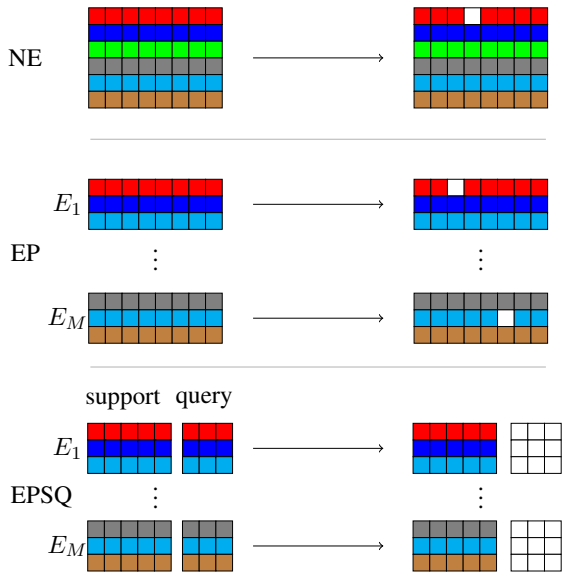

Figure 2: A illustration of training techniques, i.e., Non-episodic (NE), episodic (EP) and episodic with a support and query split (EPSQ). Each cell shows an utterance in a dataset. Each color depicts an intent class. Each row shows labeled utterances with an identical intent. In NE, the training instances are used all together and the loss is the average loss for each sample (white cell) when the other samples are neighbors. In EP, the training instances are divided into M episodes (E) and the loss is computed similar to NE for each episode. In EPSQ, each episode is split into fixed sets of of support and query instances, and the loss is computed for only the samples in the query set.

neighbors, $\mathbf{y}_q = [y_{q,t}|t = 1, ..., n]$ and similarity scores estimated by a model configuration for all pairs $\mathbf{s}_q = [s_{q,t}|t = 1, ..., n]$, we compute the binary cross-entropy loss for the query q as follows:

$$l_q(\mathbf{y}_q, s_q|C) = -\frac{1}{n}\sum_{t=1}^{n}\left[y_{q,t}\log(s_{q,t}) + (1-y_{q,t})\log(1-s_{q,t})\right]. \quad (2)$$

NE. For the NE training, the classifier learns the semantic relation between all high-resource intent classes altogether. Let D represent a batch of utterances for high-resource intent classes. Therefore, we take each utterance in D as a query q and predict its label concerning the rest of the utterances as neighbors. More formally, we estimate the loss for the NE training as follows:

$$\mathcal{L} = \frac{1}{|D|}\sum_{q\in D} l_{q|D-q}(\mathbf{y}_q, \mathbf{s}_q), \quad (3)$$

where l_q is the loss defined in Equation 2 between ground truth label vector $\mathbf{y}_q$ and a vector of scores $\mathbf{s}_q$ estimated by a model configuration.

EP. An episode is a set of utterances for several intent classes. An episode formulates an N-way intent classification task, where N is the number of intent classes in the episode. The core idea behind meta-learning is to learn from a large set of high-resource intent classes by chunking the set into many episodes (Lee et al., 2022). These episodes are known as training episodes (a.k.a meta-training episodes). If set $\mathcal{I}$ denotes the intent labels of a benchmark corpus, any N randomly selected intents from $\mathcal{I}$ can be used to construct a training episode. Let's refer to these selected intents for episode E by $\mathcal{I}_E$. Then, episode E contains utterances whose intent labels are in I_E. It is worth noting that intent classes in training episodes may overlap to let a classifier learn the semantic relations between all intent labels of the benchmark. In EP, we construct M episodes from the set of utterances for high-resource intent classes D. We define the following loss function:

$$\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{|E_i|}\sum_{q\in E_i} l_{q|E_i}(\mathbf{y}_q, \mathbf{s}_q), \quad (4)$$

where E_i is the ith episode, $\mathbf{y}_q$ is the ground-truth labels for the query given neighbors in the episode E_i, and $\mathbf{s}_q$ is the similarity scores between the query and any neighbor in the episode.

EPSQ. The common practice in meta-learning is to imitate the few-shot setup, an episode is split into two disjoint sets: a support and a query set (Lee et al., 2022). An episode's support set includes only a few utterances from each intent class in I_E. An episode's query set includes the rest of the utterances in the episode. A classifier should classify utterances in the query set using the utterances and intent labels in the support set. Given the kNN terminology, the support set is the set of neighbors and the query set is a set of query utterances. Therefore, the main difference between EPSQ and EP is that the number of neighbors in EPSQ is limited to only a few examples of each intent in the support set. The loss function in EPSQ is defined as follows:

$$\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{|Q_i|}\sum_{q\in Q_i} l_{q|S_i}(\mathbf{y}_q, \mathbf{s}_q), \quad (5)$$

where Q_i is the query set and S_i is the support set of the ith episode.

Dataset	#Classes			#Episodes		
	Train	Valid	Test	Train	Valid	Test
Balanced N-way k-shot						
Clinc (150)	50	50	50	10k	10k	600
Banking (77)	25	25	27	10k	10k	600
Liu (54)	18	18	18	10k	10k	600
Hwu (64)	23	16.4	24.6	10k	10k	600
Imbalanced Support Sets						
ATIS (19)	5	7	7	1,372	213	119
SNIPS (7)	4	-	3	240	-	210
TOP (18)	7	5	6	10,095	1,286	292

Table 1: The examined datasets and their main statistics. The numbers in parenthesis show the total number of intent classes for each dataset. For HWU64, each split's number of classes varies at each run to ensure there is no cross-split domain, hence the decimal number.

4 Experiments

We conduct our experiments in two different setups: (i) balabced N-way k-shot and (ii) imbalanced classes in the support sets. The former refers to the typical few-shot learning setup, where the numbers of classes and examples per class are balanced. In contrast, the imbalanced setup randomly defines the numbers of classes and examples, imitating the imbalance nature of some benchmarks for intent classification. While arguably some utterances can be annotated to transform imbalanced episodes into balanced ones, imbalanced few-shot learning is still a huge practical challenge for various expensive domains, e.g., those that require experts for annotation (Krone et al., 2020).

Datasets, splits, and episodes. Table 1 summarizes the main statistics (e.g., the number of classes per data split for each datasets) of the datasets and their splits as we use in our experiments. For the balanced N-way k-shot setup, we use Clinc (Larson et al., 2019b), Banking (Casanueva et al., 2020b), and Hwu (Liu et al., 2021) from DialoGLUE (Mehri et al., 2020) as well as Liu (Liu et al., 2021). For the sake of fair comparisons, we use the exact splits and episodes as used by Dopierre et al. (2021) for FSIC. For 5 folds, we randomly split intents of each dataset into three sets to construct training, valid and test episodes. We then generate 5-way k-shot episodes for each split in each fold, where $k \in \{1, 5\}$. For the imbalanced setup, we use ATIS (Hemphill et al., 1990), SNIPS (Coucke et al., 2018), and TOP (Gupta et al., 2018). We follow Krone et al. (2020) to construct episodes for these datasets.

Settings. We use BERT-based-uncased and SimCSE as PLMs. We fine-tune them using the AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of $2e-5$. Both batch size and maximum sequence length are set to 64. See the Appendix for the full list of hyperparameters. For experiments on each fold of balanced datasets, we train a FSIC classifier for a maximum of 10,000 5-way K-shots episodes. We evaluate the classifier on the validation set after every 100 updates, and stop the training if validation performance does not improve over 5 consecutive evaluation steps. To alleviate the impact of random selection of few-shot samples, we report the average performance of a classifier for 600 test episodes, compatible with Dopierre et al. (2021). For the experiments on the imbalanced datasets, similar to Krone et al. (2020), we conduct the experiments over 1 fold due to the limited number of intents. The average number of shots per intent used in episodes of ATIS, SNIPS, and TOP is about 4, 5, and 4, respectively (see Appendix for details). For both N-way k-shot and imbalanced setups, the number of examples in query sets is identical for all intents in the query sets. So, for all experiments we report the accuracy metric averaged over all runs and folds.

Models in comparison. Alongside the results of the model configurations (§3), we report the results of the following FSIC methods to put our results in context. **Random** assigns a random intent class from the support set to each query utterance. **BE (fixed)+NP** represents a generic configuration employed by the majority of PLM-based FSIC baselines, e.g., ConvBERT (Mehri and Eric, 2021), TOD-BERT (Wu et al., 2020), and DNNC-BERT (Zhang et al., 2020), inter alia. These methods use pretrained BERT and further fine-tune on other NLP tasks (e.g., NLI) or other dialogue datasets. **ProtoNet** (Dopierre et al., 2021) is inspired by prototypical network method (Snell et al., 2017), which has been shown to achieve the state-of-the-art accuracy among meta-learning methods for few-shot learning tasks including FSIC (Krone et al., 2020). This method is not based on instance-level similarity. It encodes an intent class by a prototype vector, which is the mean of vector representations of a few utterances given for the intent. In any given episode, the prototype vector is computed for each intent. The probabilities of intents are then estimated based on the distances between a query vector and respective prototypes.

	1-shot					5-shot				
	Clinc	Banking	Liu	Hwu	Avg	Clinc	Banking	Liu	Hwu	Avg
Random	20.17	20.17	20.17	20.17	20.17	19.71	19.71	19.71	19.71	19.71
BE (fixed)+NP	30.88	27.75	30.83	29.49	29.74	48.57	38.01	45.79	41.15	43.38
ProtoNet	94.29	82.20	80.06	74.37	82.73	98.10	91.57	89.62	86.48	91.44
CE +PA										
NE	58.45	48.88	48.98	50.12	51.61	66.93	64.46	55.83	59.35	61.64
EP	93.60	79.46	77.36	72.13	80.64	98.26	**92.38**	**88.33**	84.43	**90.85**
EPSQ	**94.65**	**79.82**	**78.13**	**72.64**	**81.31**	**98.49**	92.15	88.18	**84.59**	**90.85**
BE +PA										
NE	79.48	60.26	59.15	52.04	62.73	88.04	70.28	70.49	60.47	72.32
EP	82.66	66.43	59.76	50.53	64.85	92.87	77.99	70.60	61.18	75.66
EPSQ	83.26	66.53	60.41	51.40	65.40	92.51	78.59	70.82	64.13	76.51
BE +NP										
NE	58.04	45.24	53.18	42.57	49.76	78.10	68.57	61.52	54.86	65.76
EP	67.58	52.85	52.39	41.73	53.64	76.28	67.69	63.33	51.37	64.67
EPSQ	67.80	53.83	53.17	40.96	53.94	81.31	64.58	65.32	50.41	65.41

Table 2: **BERT-based** results for the **balanced 5-way k-shot** setup, $k \in \{1, 5\}$.

5 Results and Discussion

We compare the the configurations described in (§3) and baselines (§4) for balanced and imbalanced FSIC setups using BERT, as the most widely used pretrained language model, and SimCSE, the state-of-the-art model for encoding the meaning of sentences. Our main experimental findings are as follows

- The Cross-Encoder architecture with parameterized similarity function and episodic training consistently yields the best FSIC accuracy.
- Episodic training yields more robust FSIC classifiers than non-episodic training for most of the examined setups and datasets.
- Splitting episode utterances into support and query (sub)sets, a commonly adopted practice in episodic training, does not give consistent performance gains.

5.1 Balanced FSIC

Table 2 shows accuracy of the examined FSIC approaches under comparison – based on BERT as PLM – in 1-shot and 5-shots settings. All model configurations consistently outperform the "BE (fixed)+NP" baseline. This demonstrates that fine-tuning BERT's parameters for intent classification using high-resource intent classes is paramount for generalization to unseen intents.

For both 1-shot and 5-shots, CE +PA trained with either of the two episodic training regimes (EP and EPSQ, without and with support-query splitting, respectively), achieves a higher accuracy (29% on average) than when trained in non-episodic fashion (NE), reaching, on average, the performance of ProtoNet as the state-of-art FSIC method. Both episodic training regimes are more effective than the non-episodic training across the board, not just in combination with the CE architecture. BE +PA trained via EP achieves about 2% higher accuracy for 1-shot and 3% for 5-shots than when trained with NE. For BE +NP, episodic learning (EP) results in 3.8% higher accuracy than NE for 1-shot. The only exception to this trend is BE +NP with 5-shot where EP trails NE by 1%.

EPSQ tends to exhibit a similar average accuracy as EP (less than 1% difference for average across all CE +PA, BE +PA, and BE +NP setups). This leads a conclusion that splitting utterances of an episode into a support and a query set – a common practices in episodic (FSIC) learning (Dopierre et al., 2021; Krone et al., 2020) – does not really have a pronounced (positive) effect on performance. So, it does not seem to increase the capability to generalize to unseen intent classes, as has been commonly believed but until now, to the best of our knowledge, empirically untested.

As expected, more shots (5-shots vs 1-shot) lead to consistently better FSIC accuracy: BE +NP trained with NE performs 16% better (and the other FSIC about 10% better on average). This makes intuitive sense: more shots help classifiers better refine the boundaries between the new intents.

Given that utterances in task-oriented dialogue systems are short texts, we next investigate how intermediate training for sentence representations (Phang et al., 2018; Reimers and Gurevych, 2019a; Gao et al., 2021) changes the performance of FSIC

	1-shot					5-shot				
	Clinc	Banking	Liu	Hwu	Avg	Clinic	Banking	Liu	Hwu	Avg
BE (fixed) +NP	91.33	75.48	78.75	74.58	80.03	97.89	90.33	89.61	86.93	91.19
CE +PA										
NE	60.51	54.87	49.99	46.41	52.95	78.33	72.71	68.66	67.99	71.92
EP	94.33	83.64	79.24	77.03	83.56	**98.80**	**94.22**	**90.13**	**88.54**	**92.92**
EPSQ	**95.01**	**83.83**	79.40	**77.49**	**83.93**	98.77	94.04	90.10	88.40	92.83
BE +PA										
NE	90.69	76.21	68.76	66.05	75.43	96.71	88.12	80.76	79.26	86.21
EP	90.93	76.81	71.32	65.72	76.19	96.74	88.18	83.83	80.64	87.35
EPSQ	90.95	76.43	71.33	65.71	76.11	96.83	87.95	84.10	80.90	87.45
BE +NP										
NE	93.69	81.60	79.51	75.54	82.58	98.08	91.56	89.61	87.82	91.77
EP	93.24	80.15	79.82	76.49	82.43	98.01	91.91	89.77	87.62	91.83
EPSQ	93.44	80.46	**80.21**	76.68	82.70	98.02	91.95	89.83	87.65	91.86

Table 3: **SimCSE-based** results for the **balanced 5-way k-shot** setup, $k \in \{1, 5\}$.

	ATIS	SNIPS	TOP	Avg
Random	21.34	33.70	23.99	26.34
BE (fixed) + NP	53.80	51.62	33.03	46.15
CE + PA				
NE	62.86	65.03	49.41	59.10
EP	**79.71**	**93.94**	**68.04**	**80.56**
EPSQ	71.58	92.98	62.84	75.80
BE + PA				
NE	42.91	80.22	53.48	58.87
EP	69.52	60.54	51.46	60.51
EPSQ	66.44	62.21	56.05	61.57
BE + NP				
NE	65.86	77.92	45.52	63.10
EP	65.09	79.16	47.85	64.03
EPSQ	55.67	80.08	42.97	59.57

Table 4: Results for the imbalanced setup using BERT.

models. To this end, we substitute BERT with SimCSE. Table 3 shows the results. Our three main findings hold for SimCSE-based FSIC models too. Importantly, unlike with BERT, now only CE +NP trained episodically outperforms the "BE (fixed)+NP" baseline (where PLM is not fine-tuned for intent detection). This confirms the effectiveness of coupling CE and episodic training for FSIC. It also indicates that intent detection fine-tuning is well-aligned with learning sentence representations, which is why it generally brings lower gains (or no gains) over "BE (fixed) + NP", when we start from SimCSE, pretrained exactly for encoding the meaning of sentences.

5.2 Imbalanced FSIC

Table 4 shows the results on the three imbalanced datasets. CE +PA with EP again substantially outperforms all its counterparts, confirming this never-investigated FSIC configuration as a very effective approach for the FSIC task. On average, episodic training (EP) again outperforms non-episodic (NE) training. The CE + PA and BE + NP configurations generally yield higher performance when trained without splitting the support utterances from query utterances (EP vs EPSQ). This questions the common belief in episodic meta-learning that splitting episodes into support and query sets is (always) beneficial. Overall, the findings from the imbalanced datasets align well with the main findings from central experiments on balanced datasets, as reported in Table 2 and Table 3.

6 Conclusions

We shed light on factors that contribute to performance of models for few-shot intent classification (FSIC), a crucial task in modular dialogue systems. We categorize FSIC approaches across three essential dimensions: (1) the Cross-Encoder vs. Bi-Encoder encoder architectures; (2) the parameterized (i.e., trainable) vs non-parameterized utterance similarity scoring; and (3) episodic vs non-episodic training. Our extensive evaluation, encompassing seven standard FSIC datasets, reveals that the previously unexplored combination of Cross-Encoder architecture (with parameterized utterance similarity scoring) and episodic training consistently yields the best FSIC performance. We additionally find that (i) episodic meta-learning generally outperforms the non-episodic training and (ii) that the widely adopted hypothesis in meta-learning that splitting episodes into support and query sets helps generalization and boost performance may not hold for FSIC. We hope that our findings lead to more deliberation on FSIC evaluation protocols and more insightful "apple-to-apple" comparisons between competing models and model variants.

Limitations and Ethical Concerns.

In this paper, we shed light to few-shot intent classification tasks in modular (task-oriented) dialogue systems. Dialog systems, given their direct interaction with human users, must be devoid of any negative stereotypes and must not exhibit any behaviour that could be potentially harmful to humans. That said, our work does not address the generation component of dialog systems, but merely the intent classification. As such, we do not believe it raises any ethical concerns.

The main limitation of the work – conditioned primarily by the available computational resources – is the scope of our empirical comparison: we focus on FSIC methods that subscribe to pairwise similarity scoring of utterances and nearest neighbours inference. While this subsumes much of the best performing approaches in the literature, there is a fair body of recent work that does not fall in this group. Another limitation of the work is the monolingual focus on English only. We intend to extend our work to cross-lingual transfer to other languages, for which fewer labeled intent classification datasets exist.

Acknowledgements

This work has been funded by the German Research Foundation (DFG) as part of the UKP-SQuARE project (grant GU 798/29-1), by the Collaboration Lab with Nexplore "AI in Construction" (AICO) and by the European Union under the Horizon Europe grant No 101070351 (SERMAS). We also thank our anonymous reviewers for their constructive comments on this paper.

References

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. 1993. Signature verification using a "siamese" time delay neural network. In *Advances in Neural Information Processing Systems*, volume 6. Morgan-Kaufmann.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020a. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45, Online. Association for Computational Linguistics.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020b. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45, Online. Association for Computational Linguistics.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *ArXiv preprint*, abs/1805.10190.

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.

Thomas Dopierre, Christophe Gravier, and Wilfried Logerais. 2021. PROTAUGMENT: Unsupervised diverse short-texts paraphrasing for intent detection meta-learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2454–2466, Online. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Abhirut Gupta, Anupama Ray, Gargi Dasgupta, Gautam Singh, Pooja Aggarwal, and Prateeti Mohapatra. 2018. Semantic parsing for technical support questions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3251–3259, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Arshit Gupta, John Hewitt, and Katrin Kirchhoff. 2019. Simple, fast, accurate intent classification and slot labeling for goal-oriented dialogue systems. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 46–55, Stockholm, Sweden. Association for Computational Linguistics.

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

Chia-Chien Hung, Anne Lauscher, Ivan Vulić, Simone Ponzetto, and Goran Glavaš. 2022. Multi2WOZ: A robust multilingual dataset and conversational pretraining for task-oriented dialog. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3687–3703, Seattle, United States. Association for Computational Linguistics.

Jason Krone, Yi Zhang, and Mona Diab. 2020. Learning to classify intents and slot labels given a handful of examples. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 96–108, Online. Association for Computational Linguistics.

Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019a. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1311–1316, Hong Kong, China. Association for Computational Linguistics.

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. 2019b. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1311–1316, Hong Kong, China. Association for Computational Linguistics.

Hung-yi Lee, Shang-Wen Li, and Thang Vu. 2022. Meta learning for natural language processing: A survey. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 666–684, Seattle, United States. Association for Computational Linguistics.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2021. *Benchmarking Natural Language Understanding Services for Building Conversational Agents*, pages 165–183. Springer Singapore, Singapore.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Samuel Louvan and Bernardo Magnini. 2020. Recent neural methods on slot filling and intent classification for task-oriented dialogue systems: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 480–496, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Tingting Ma, Qianhui Wu, Zhiwei Yu, Tiejun Zhao, and Chin-Yew Lin. 2022. On the effectiveness of sentence encoding for intent detection meta-learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3806–3818, Seattle, United States. Association for Computational Linguistics.

S. Mehri, M. Eric, and D. Hakkani-Tur. 2020. Dialoglue: A natural language understanding benchmark for task-oriented dialogue. *ArXiv*, abs/2009.13570.

Shikib Mehri and Mihail Eric. 2021. Example-driven intent prediction with observers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2979–2992, Online. Association for Computational Linguistics.

Lian Meng and Minlie Huang. 2018. Dialogue intent classification with long short-term memory networks. In *Natural Language Processing and Chinese Computing*, pages 42–50, Cham. Springer International Publishing.

Aaron Mueller, Jason Krone, Salvatore Romeo, Saab Mansour, Elman Mansimov, Yi Zhang, and Dan Roth. 2022. Label semantic aware pre-training for few-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8318–8334, Dublin, Ireland. Association for Computational Linguistics.

Hoang Nguyen, Chenwei Zhang, Congying Xia, and Philip Yu. 2020. Dynamic semantic matching and aggregation network for few-shot intent detection. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1209–1218, Online. Association for Computational Linguistics.

Jason Phang, Thibault Févry, and Samuel R Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *ArXiv preprint*, abs/1811.01088.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY,*

USA, February 7-12, 2020, pages 8689–8696. AAAI Press.

Evgeniia Razumovskaia, Goran Glavaš, Olga Majewska, Edoardo M Ponti, Anna Korhonen, and Ivan Vulić. 2021. Crossing the conversational chasm: A primer on natural language processing for multilingual task-oriented dialogue systems. *ArXiv preprint*, abs/2104.08570.

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

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

Anna Sauer, Shima Asaadi, and Fabian Küch. 2022. Knowledge distillation meets few-shot learning: An approach for few-shot intent classification within and across domains. In *Proceedings of the 4th Workshop on NLP for Conversational AI*, pages 108–119, Dublin, Ireland. Association for Computational Linguistics.

Jake Snell, Kevin Swersky, and Richard S. Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4077–4087.

Ivan Vulić, Pei-Hao Su, Samuel Coope, Daniela Gerz, Paweł Budzianowski, Iñigo Casanueva, Nikola Mrkšić, and Tsung-Hsien Wen. 2021. ConvFiT: Conversational fine-tuning of pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1151–1168, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chengyu Wang, Haojie Pan, Yuan Liu, Kehan Chen, Minghui Qiu, Wei Zhou, Jun Huang, Haiqing Chen, Wei Lin, and Deng Cai. 2021. Mell: Large-scale extensible user intent classification for dialogue systems with meta lifelong learning. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*.

Yufan Wang, Jiawei Huang, Tingting He, and Xinhui Tu. 2019. Dialogue intent classification with character-cnn-bgru networks. *Multimedia Tools and Applications*, 79:4553–4572.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

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

Congying Xia, Caiming Xiong, and Philip Yu. 2021. Pseudo siamese network for few-shot intent generation. SIGIR '21, page 2005–2009, New York, NY, USA. Association for Computing Machinery.

Puyang Xu and Ruhi Sarikaya. 2013. Convolutional neural network based triangular crf for joint intent detection and slot filling. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 78–83.

Weiyuan Xu, Peilin Zhou, Chenyu You, and Yuexian Zou. 2021. Semantic transportation prototypical network for few-shot intent detection. In *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, pages 251–255. ISCA.

Haode Zhang, Haowen Liang, Yuwei Zhang, Li-Ming Zhan, Xiao-Ming Wu, Xiaolei Lu, and Albert Lam. 2022a. Fine-tuning pre-trained language models for few-shot intent detection: Supervised pre-training and isotropization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 532–542, Seattle, United States. Association for Computational Linguistics.

Haode Zhang, Haowen Liang, Yuwei Zhang, Li-Ming Zhan, Xiao-Ming Wu, Xiaolei Lu, and Albert Lam. 2022b. Fine-tuning pre-trained language models for few-shot intent detection: Supervised pre-training and isotropization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 532–542, Seattle, United States. Association for Computational Linguistics.

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

Jianguo Zhang, Kazuma Hashimoto, Wenhao Liu, Chien-Sheng Wu, Yao Wan, Philip Yu, Richard Socher, and Caiming Xiong. 2020. Discriminative nearest neighbor few-shot intent detection by transferring natural language inference. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5064–5082, Online. Association for Computational Linguistics.

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