Few-shot Out-of-Scope Intent Classification: Analyzing the Robustness of Prompt-based Learning

Yiwei Jiang, Maarten De Raedt, Johannes Deleu, Thomas Demeester and Chris Develder

IDLab, Ghent University – imec, Technologiepark Zwijnaarde 126, 9052 Ghent, Belgium.

*Corresponding author(s). E-mail(s): yiwei.jiang@ugent.be;
Contributing authors: maarten.deraedt@ugent.be;
johannes.deleu@ugent.be; thomas.demeester@ugent.be;
chris.develder@ugent.be;

Abstract

Out-of-scope (OOS) intent classification is an emerging field in conversational AI research. The goal is to detect out-of-scope user intents that do not belong to a predefined intent ontology. However, establishing a reliable OOS detection system is challenging due to limited data availability. This situation necessitates solutions rooted in few-shot learning techniques. For such few-shot text classification tasks, prompt-based learning has been shown more effective than conventionally finetuned large language models with a classification layer on top. Thus, we advocate for exploring prompt-based approaches for OOS intent detection. Additionally, we propose a new evaluation metric, the Area Under the In-scope and Out-of-Scope Characteristic curve (AU-IOC) This metric addresses the shortcomings of current evaluation standards for OOS intent detection. AU-IOC provides a comprehensive assessment of a model's dual performance capacities: in-scope classification accuracy and OOS recall. Under this new evaluation method, we compare our prompt-based OOS detector against 3 strong baseline models by exploiting the metadata of intent annotations, i.e., intent description. Our study found that our prompt-based model achieved the highest AU-IOC score across different data regimes. Further experiments showed that our detector is insensitive to a variety of intent descriptions. An intriguing finding shows that for extremely low data settings (1- or 5-shot),

employing a naturally phrased prompt template boosts the detector's performance compared to rather artificially structured template patterns.

Keywords: few-shot learning, prompt-based models, outlier/novelty detection, dialogue intent classification

1 Introduction

When deploying a machine learning based model in the wild, it risks facing real-world input data that differs from what it was trained on. To avoid the model producing undesirable output/decisions in such cases, and thus achieve a robust system, out-of-scope (OOS) detection is crucial. Therefore, for classification tasks, OOS detection has been widely studied, e.g., in vision [1–4], text [5–9] and audio [10, 11] domains.

In this work, we focus on OOS detection for task-oriented dialogue systems, more specifically for a crucial building block, namely intent classification. OOS intents pertain to types of user requests that are not supported by the trained dialogue system. It is challenging to design a robust system that can maintain a high accuracy of predicting in-scope intents while also having decent OOS detection performance. Such OOS detection is essential to reduce the risk of misunderstanding. For instance, a bank service chatbot should not treat an OOS intent as a block_account intent. Many approaches have been proposed to tackle the OOS intent detection task [5–7, 9, 12, 13]. These methods assume a large amount of in-scope training data, which is not always available due to the high cost of collecting high-quality labeled data in real-world applications. Thus, our current work specifically focuses on the low-data regime, which makes it even more challenging to build an OOS intent detection model: we coin our task as few-shot OOS intent detection.

Regardless of whether an OOS detection system is trained under a few-shot or full-shot setting, the performance evaluation of such system remains non-trivial. A main issue is to find a balance between in-scope classification performance and OOS detection. For instance, Table 1 shows the key results from [14], where different pretrained encoders are evaluated with 5-shot training data. Among those results, the RoBERTa model obtains the highest in-scope accuracy, yet performs poorly in terms of OOS recall. Similar conflicts of model rankings for different metrics are prevalent in other experiments, as shown in Table 4 in [14]. This complexity makes it challenging to select the best model. One way to achieve a balance between in- and out-of-scope detection metrics is by tuning the threshold of the classification score. Given a test sample, an OOS detection system typically outputs a score that indicates its confidence on whether that sample is in- or out-of-scope. Thus, the score threshold of in-scope classifiers (below which an input would be then classified as out-of-scope) could be tuned to maximize the sum of in-scope accuracy Acc_{in} and OOS recall

R_{oos}, as in [14, 15]. However, it is hard to intuitively interpret this sum of two complementary metrics.

Table 1 Demonstration of the difficulty in ranking models by relying on either in-scope accuracy Acc_{in} or OOS recall R_{oos} alone. †: Numbers taken from Table 4 in [14]. ‡: Numbers computed by us. AU-IOC denotes Area Under the In-scope and Out-of-scope Characteristic curve, a new metric introduced in Section 3.1. Different pretrained encoders are compared on the 5-shot experiment on BANKING dataset. Numbers in parentheses are standard deviations.

5-shot	Acc_{in} †	R_{oos} †	AU-IOC ‡
ALBERT	20.3 (± 2.4)	89.5 (± 1.5)	28.05 (± 3.68)
BERT	25.4 (± 3.6)	**90.9 (± 0.6)**	48.39 (± 1.10)
ELECTRA	30.9 (± 2.3)	87.5 (± 2.4)	44.17 (± 2.79)
RoBERTa	**43.0 (± 2.9)**	83.1 (± 4.3)	**51.46 (± 2.62)**
ToD-BERT	35.5 (± 1.5)	82.7 (± 1.8)	46.69 (± 0.98)

To give a more holistic picture of a classification model's performance, beyond that at a specific model parameter value (e.g., the aforementioned score threshold), metrics such as Area Under the Receiver Operation Characteristic Curve (AUC or AUROC) or Area Under the Precision Recall Curve (AUPR) have been used in [5–7, 16, 17]. Yet, these AUROC and AUPR metrics have been designed for binary classification. To adopt them for OOS detection, we would need to lump all in-scope classes together (typically in the "positive" class). This means we would no longer have any information on in-scope classification performance. Therefore, we propose to use another curve, plotting in-scope accuracy (Acc_{in}) against OOS recall (R_{oos}), from which we define a new metric named "Area Under In-scope and Out-of-scope Characteristic curve (AU-IOC)". Our AU-IOC is designed with three advantages over the aforementioned methods: (i) it clearly indicates the performance of different OOS intent detection models (as shown in the last column of Table 1); (ii) it is threshold-free; (iii) it simultaneously covers the performance of in-scope multi-class classification and OOS detection.

As just mentioned, an ideal intent classifier for dialogue systems should simultaneously achieve both high Acc_{in} and R_{oos}. With the objective of achieving high Acc_{in}, recent studies on prompt-based learning (PBL) have demonstrated state-of-the-art or competitive performance on few-shot language understanding tasks such as text classification [18, 19], named entity recognition [20] and relation extraction [21]. These works reformulate the classification task as cloze form questions. For example, for binary good/bad classification of statements, a prompting input would be: "Hundreds of lives were saved. It is a [mask] thing.", where "good" would need to be predicted for the [mask]. Such prompt-based learning allows for better alignment between pretraining and finetuning stages. However, one overlooked limitation of this promising prompting approach is that the evaluated tasks are restricted to in-scope classification, i.e., the held-out test set has no out-of-scope samples. In the context of intent classification

tasks, one research question naturally arises: *is prompt-based learning robust in identifying out-of-scope (OOS) intents while maintaining in-scope classification performance?* To answer this question, we investigate a simple yet effective prompt-based model as a strategy to tackle the few-shot OOS intent detection problem. Our empirical investigation demonstrates that, by exploiting the meta-data (i.e., descriptions of intent labels), our prompt-based model outperforms strong baseline models by a large margin.

In summary, our contribution is threefold:

- We point out limitations of existing evaluation metrics for OOS detection and propose a new evaluation metric AU-IOC that incorporates two important and complementary aspects of an OOS detector's performance, i.e., in-scope accuracy and OOS recall;
- To the best of our knowledge, we are the first to adopt prompt-based learning for the few-shot OOS detection task, incorporating an textual intent description as part of the prompt. Extensive experiments on 6 datasets for few-shot OOS detection in dialogue intent classification show that a prompt-based model outperforms strong baseline models across various data regimes (from 1- to 50-shot learning), indicating strong robustness and data efficiency of prompt-based learning;
- We further show that our prompt-based OOS detector benefits from textual intent descriptions, yet is insensitive to their variety (e.g., among descriptions provided by multiple annotators). Indeed, empirical results of the few-shot OOS detection task illustrate that providing plain language descriptions of the intents in the prompt, rather than just a short label (e.g., block account), significantly boosts performance, without being sensitive to the exact phrasing of that description. Interestingly, we find that for extremely low data settings (1- or 5-shot), using a naturally phrased prompt template boosts the detector's performance compared to using artificially structured prompt patterns.

2 Preliminaries

We first introduce the formal task definition of few-shot OOS intent detection. Subsequently, by going through two existing evaluation methods, we point out their limitations (I, II III) which motivates our proposed evaluation method in Section 3.1. Table 2 lists frequently used notations in this paper.

2.1 Few-shot Out-of-Scope Intent Detection

Our work tackles the problem of OOS detection in a dialogue system under a few-shot setting, where for each intent class only k positive examples are given, i.e., balanced k-shot learning. Note that we purposefully use the term "out-of-scope" (OOS) instead of "out-of-domain" (OOD), since we focus on identifying utterances that are within the same domain as the predefined intents and thus more challenging to distinguish than totally unrelated OOD utterances. For example, if a chatbot is trained only for booking restaurants (with two intents,

Table 2 Notations used in this paper.

OOS	out-of-scope	TP	number of true positive predictions
in	in-scope	FN	number of false negative predictions
X	an utterance	TN	number of true negative predictions
ϕ	a model's confidence score	TPR	true positive rate
τ	a threshold to gauge ϕ	FPR	false positive rate
Acc	accuracy	L	total number of in-scope intents
P	precision	$\mathcal{V}$	a verbalizer mapping label(s) to textual token(s)
R	recall	$\mathcal{T}$	a template function
C	number of correctly predicted examples	h	a vector encoded by a neural network
N	total number of examples	W, b	trainable parameters of a linear layer
		d	the hidden dimension of a neural network

restaurant_availability and restaurant_address), any questions from other domains (e.g., banking, e-commerce), or even random texts crawled from the web, would be classified as OOD. Only questions with related but not specifically predefined intents, such as restaurant_phone_number, would be flagged as OOS. Like other works, we assume that the OOS data is inaccessible during the training stage [9, 12, 14–16, 22, 23]. Our reason to follow this restriction is that in most cases, the OOS data distribution is unknown or its vast underlying space makes it hard and expensive to collect a sufficient amount of OOS data. The formal task definition is:

> *Given only in-scope training data, learn to determine for a given input utterance X (i) whether X is in- or out-of-scope, and (ii) which intent class it belongs to, in case X is in-scope.*

A common approach for (i) to decide whether X is OOS, is to compare a classifier's confidence value ϕ against a predefined threshold τ, i.e., X is classified as OOS if $\phi < \tau$. An example of ϕ is the normalized output (i.e., pseudo probability) of a softmax layer. Next, we discuss evaluation methods by tuning such a threshold τ and without tuning it.

2.2 Threshold-tuning Evaluation

Prior works [14, 15, 24, 25] tune the threshold based on two major metrics, in-scope accuracy Acc_{in} and OOS recall R_{oos}.[1] The two metrics are defined as follows: $Acc_{in} = C_{in}/N_{in}, R_{oos} = C_{oos}/N_{oos}$, where C_{in} is the number of correctly predicted in-scope examples, N_{in} is the total number of in-scope

[1]The reason for using recall instead of precision to evaluate the OOS detection is that a recall error (an OOS question is wrongly classified as an in-scope intent) would generate a completely wrong response, while a precision error (i.e., an in-scope question is misclassified as OOS) is rather safe since it usually triggers a fallback response that asks the user to rephrase the question.

examples, C_{oos} is the number of correctly predicted OOS examples, and N_{oos} is the total number of OOS examples. There is a trade-off between Acc_{in} and R_{oos} when tuning the threshold. For example, let's assume a model that simulates a probability distribution of all the utterance examples, thus its output spans the range [0, 1] and the two extreme cases are $\tau = 0$ and $\tau = 1$. When $\tau = 0$, all examples are classified as in-scope intents. Consequently, the in-scope accuracy will be high (if the model is properly trained), while $R_{oos} = 0$ (which would only be acceptable if the system is deployed in a friendly environment where professional or trained users never ask OOS questions). On the other hand, when $\tau = 1$, all examples are classified as OOS, i.e., $Acc_{in} = 0$ and $R_{oos} = 1$, which clearly is useless.

Prior works [14, 15, 24, 25] pursue the best trade-off (i.e., set the threshold τ) by optimizing an objective function $f(Acc_{in}, R_{oos}) = g(\tau)$. A simple objective adopted by [14, 15] is $f = Acc_{in} + R_{oos}$, giving the same weight to both metrics. The threshold is tuned by maximizing f on the dev set. Table 1 shows partial results from [14], where performance of five different pretrained encoders is compared against each other. The ALBERT model has the second highest R_{oos} however owns the lowest Acc_{in}. The RoBERTa model obtains the highest Acc_{in}, while it is inferior to the BERT model in terms of R_{oos}. Such inconsistency prevails on the other experiment results (see the full results in [14]), making it difficult to clearly distinguish the model under comparison (**limitation** I). Another drawback is that Acc_{in} and R_{oos} are calculated at a specific threshold. This means that these metrics at a specific model parameter setting (for τ) do not offer a nuanced and holistic view of the performance trade-off between in-scope classification and OOS detection, e.g., how fast Acc_{in} drops when R_{oos} escalates by varying the threshold (**limitation** II).

2.3 Threshold-free Evaluation

To avoid the limitations incurred by tuning the threshold, [3, 16] propose to use the Area Under the Receiver Operating Characteristic curve (AUROC) and Area Under the Precision-Recall curve (AUPR), which are threshold-independent performance evaluation metrics [26]. Neither of these two threshold-free methods can reflect the multi-class classification accuracy because all in-scope classes are merged into one class, normally the positive, and the OOS class is regarded as the negative. The ROC curve (conceptually visualized in Fig. 1(b)) plots the true positive rate (TPR, equivalent to the recall of in-scope examples, R_{in}) against the false positive rate (FPR) calculated at a series of thresholds. By computing the integral of a ROC curve, AUROC does not rely on any specific threshold, and it can be interpreted as the probability that a positive example has a greater confidence score than a negative example [27]. A random binary classifier obtains merely 50% AUROC and a "perfect" classifier reaches 100%, which makes it easier to rank the model performance than using the threshold-tuning method. The relation of TPR and FPR to the threshold-tuning metrics (i.e., Acc_{in} and R_{oos}) is shown in Eqs. (1)–(2), where N_{in} and N_{oos} denote the total number of in-scope examples and OOS examples respectively.

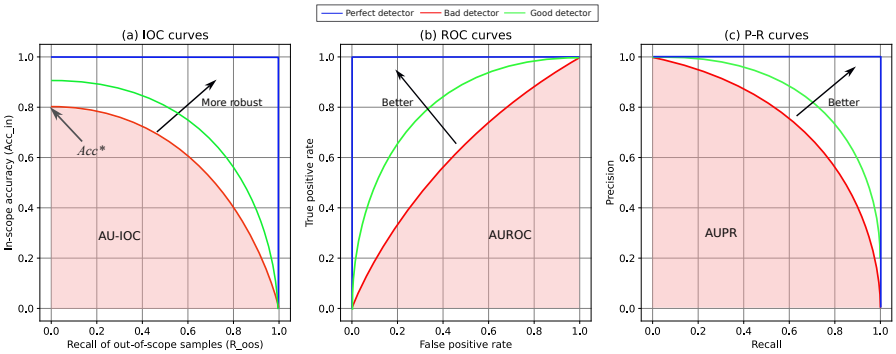

Fig. 1 Conceptual illustration of (a) In-scope and Out-of-scope Characteristic (IOC) curves, (b) Receiver Operating Characteristic (ROC) curves and (c) Precision-Recall (P-R) curves. Acc^* denotes the maximal Acc_{in} that a detector can obtain. Shaded areas denote the area under each curve. Best viewed in color.

$$TPR = TP/(TP + FN) = TP/N_{in} \geq C_{in}/N_{in} \tag{1}$$
$$\Rightarrow \quad TPR \geq Acc_{in}$$
$$FPR = FP/(FP + TN) = (N_{oos} - C_{oos})/N_{oos} \tag{2}$$
$$\Rightarrow \quad FPR = 1 - R_{oos}$$

More specifically, when computing the number of true positive (TP) predictions for TPR, an example is only required to have a confidence score greater than the threshold. In other words, it does not matter if an utterance's intent class is correctly predicted or not, which by contrast, is considered in calculating the number of correct in-scope predictions (C_{in}). This is why TP is larger or equal than C_{in} in Eq. (1). Hence, TPR may overestimate a classifier's performance for in-scope classification. In the worst case, despite attaining a high TPR, a model might in fact have a low Acc_{in}, especially when the data has a rather large number of in-scope classes (e.g., ≥ 10). To the authors' best knowledge, this problem of being overly optimistic (**limitation** III) has not yet been discussed in prior OOS research works.

Similarly, the PR curve (shown in Fig. 1(c)) is a graph displaying the precision ($P_{in} = TP/(TP + FP)$) and recall ($R_{in} = TP/(TP + FN) = TPR$) against each other. The inequality in Eq. (1) also holds in computing R_{in}, making it always higher than Acc_{in}. Therefore, AUPR also faces the overestimation problem as AUROC does.

3 Proposed Methodology

3.1 Proposed Evaluation Method

To overcome the limitations (I, II, III) discussed in Section 2.2 and Section 2.3, we propose a new evaluation metric designed to meet the following requirements: (i) it should allow for clearly distinguishing models; (ii) it simultaneously covers the two most relevant perspectives of a classifier with OOS detection capabilities, i.e., in-scope accuracy Acc_{in} and OOS recall R_{oos}; (iii) it is preferably threshold-free to offer a holistic view. Motivated by these requirements, we propose to evaluate OOS models by the *Area Under the In-scope and Out-of-Scope Characteristic curve* (AU-IOC), which reflects a model's performance in both of in-scope accuracy and OOS recall. The In-scope and Out-of-Scope Characteristic curve (IOC) plots Acc_{in} and R_{oos} against each other, parameterized over the full range of possible OOS detection thresholds. An IOC curve monotonically decreases as R_{oos} increases (see Fig. 1(a)). The intercept at the y-axis (Acc^*) is the maximal Acc_{in} that a detector can obtain. Tweaking the threshold τ varies R_{oos} within [0, 1]. Hence, being the integral of an IOC curve, the AU-IOC is never greater than Acc^*, i.e., AU-IOC $\leq Acc^*$, where the equality occurs if and only if a detector can "perfectly" distinguish OOS examples from in-scope examples. This fact also implies that a classifier with high Acc^* would have high AU-IOC as well. Note that a perfect detector should attain AU-IOC =100%, i.e., $Acc^* = 100\%$, i.e., its perfect in-scope accuracy does not drop when moving the OOS detection threshold towards perfect OOS recall.

3.2 Prompt-Based Model

As pointed out in Section 3.1, an OOS intent detector should ideally achieve both high Acc_{in} and high R_{oos} to obtain a high AU-IOC score. To achieve high Acc_{in}, recent works based on prompt-based learning (PBL) have demonstrated state-of-the-art performance on few-shot text classification tasks [18, 19, 28], as well as competitive results on few-shot named entity recognition [20] and few-shot relation extraction [21]. However, to the best of our knowledge, there have been no studies applying PBL on the few-shot OOS intent detection task, let alone investigating its robustness for this task. Given PBL's excellent performance in few-shot in-scope language understanding tasks, we hypothesize it would also outperform existing models for the few-shot OOS intent detection task.

The key concept of prompt-learning is to formulate inputs in a natural language format, i.e., phrase the task following a certain input template, as proposed in the PET model [18]. Text classification tasks are thus transformed into cloze-style questions, similar to the input format used for masked language modeling (e.g., BERT [29]). In this work, we frame intent detection as a text entailment task rather than a multi-class classification task. The original PET model [18] maps each class label to a single token or multiple tokens to achieve the multi-class classification directly. However, in our study, an intent label

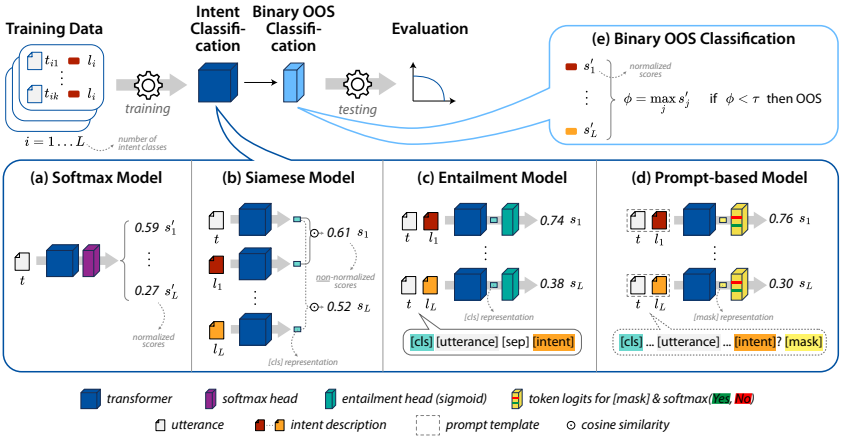

Fig. 2 Overview of our approach. We compare different architectures including (a) Softmax, (b) Siamese, (c) Entailment and (d) Prompt-based model. L, s, ϕ, τ respectively denote the number of in-scope classes, predicted score, confidence value and threshold. (e) illustrates how we identify the out-of-scope utterances by comparing ϕ against τ.

cannot be easily mapped to only one token, making it hard to follow the PET methodology.[2] Therefore, for each intent, we consider a one-vs-all classifier, enabling us to exploit the intent description with richer semantic information compared to the intent label.[3] As stated in [28], when handling datasets (i.e., BANKING) with multiple labels longer than a single token, they also convert the multi-class classification task to a binary classification task, leading to more efficient training and inference.

To this end, our model treats an utterance and an intent as "premise" and "hypothesis" respectively. In our case, the entailment relation is binary, either positive (+) or negative (−). Following prior works [18, 30], we use a verbalizer $\mathcal{V}$ mapping the relation labels to tokens from the vocabulary of a language model, i.e., $\mathcal{V}(+)$ ="*yes*" and $\mathcal{V}(-)$ ="*no*". The logits of "*yes*" and "*no*" output at the mask position of a template input are normalized through a softmax layer to generate a probability distribution over these two possible entailment outcomes. The resulting probability of the "*yes*" token is taken as the entailment score. The model architecture is depicted in Fig. 2(d). For the final intent classification, the intent with the highest entailment score is chosen and we use that score as confidence value. The training objective is to maximize the probability of true labels for each training instance, phrased according to a predefined input template (see Section 5.4).

Fig. 2 illustrates the training and inference procedures of our prompt-based model and other baseline models which will be introduced in Section 4.2. To clarify our prompt-based solution for intent detection, assume there are two possible intents: i_1 = stop_account, and i_2 = create_account. Now,

[2]We also experimented on a multi-mask PET model [28] to directly predict labels directly yielding much worse performance and less efficient training compared to our prompt-based model.
[3]Without specification, "intent" represents either "intent label" or "intent description".

consider two utterances to classify: an in-scope utterance u_1 = "*Can you suspend my bank account right now?*" and an out-of-scope u_2 = "*My contactless card does not work. I want to fix it.*" For all the experiments in this work (except Section 5.4), our prompt-based model uses the template $\mathcal{T}$ = Joe said "[utterance]". Does Joe mean [intent]? [mask], in which [utterance] and [intent] are placeholders. A well-trained prompt based model should score u_1 with a higher "*yes*" score for intent i_1 than for i_2, i.e., $p(\text{"}yes\text{"}|\mathcal{T}(u_1, i_1)) > p(\text{"}yes\text{"}|\mathcal{T}(u_1, i_2))$. In addition, the model should score the OOS utterance u_2 with a low score (lower than a threshold τ) for both intents, i.e., $\max_{\ell \in \{0,1\}} p(\text{"}yes\text{"}|\mathcal{T}(u_2, i_\ell)) < \tau$.

4 Experimental Setup

We experiment with our proposed prompt-based model on six intent classification datasets (Section 4.1). We compare it against three strong baseline models, described in Section 4.2.

4.1 Datasets

Numerous intent classification datasets are at our disposal. In choosing the most suitable dataset for our study, we adhered to two primary criteria: (i) the dataset must contain more than two intent classes. This prerequisite is essential as it enables the segregation of an intent as the Out-of-Scope (OOS) category, which would be unfeasible with two or fewer classes.; (ii) it is imperative that the intent classes exhibit a balanced distribution, thereby facilitating k-shot experiments. In essence, each class should possess a minimum of k training samples. We choose 6 benchmark datasets which meet our requirement. Table 3 shows their statistics. To save space, we only present the intent split of CLINC-Banking in Table 4 while the other 5 datasets' intent splits are made available online. [4]

SNIPS [31] has 7 intents annotated for dialogues between a user and a virtual assistant. We randomly selected 5 as in-scope intents.

Facebook [32] has 12 intents across 3 domains including setting alarms, reminders and querying the weather. We randomly picked 8 as in-scope intents.

CLINC-Banking [24] has 15 intents in total from which 10 intents were chosen as in-scope and the others are OOS as shown in Table 4.

Stackoverflow [33] was crawled from the stackoverflow website. It consists of technical question titles covering 20 topics in total. We randomly picked 14 as in-scope topics.

HWU64 [34] has 64 intents in total ranging from home automation, email queries, etc. . We randomly selected 40 as in-scope intents.

[4] https://bit.ly/3r4bDN0

Table 3 Statistics of the 6 datasets used in our experiments

. INS and OOS denote in-scope and out-of-scope respectively.

Dataset		#intents	Train	Dev	Test
SNIPS	INS	5	9,385	500	484
	OOS	-	-	200	216
Facebook	INS	8	19,336	3,029	6,151
	OOS	-	-	866	1,650
CLINC-Banking	INS	10	500	500	500
	OOS	-	-	400	350
Stackoverflow	INS	14	11,191	1,406	1,403
	OOS	-	-	594	597
HWU64	INS	40	5,691	640	684
	OOS	-	-	366	392
BANKING	INS	50	5,905	1,506	2,000
	OOS	-	-	530	1,080

BANKING [35] focuses on a single domain, i.e., banking. Its fine-grained 77 intents makes itself the most challenging intent classification task as the intents are semantically closely-related. From the 77 intents, 50 were selected as in-scope intents.

In a k-shot experiment, we randomly sample k utterances per each in-scope intent to construct a training set. The sampling process is dependent on the random seed used for training a model.

4.2 Models Under Comparison

We introduce three canonical architectures as baselines: (i) a basic softmax-based utterance classifier, (ii) a siamese model as the vanilla bi-encoder architecture, and (iii) a binary entailment classifier as the vanilla cross-encoder. These architectures are commonly used in the literature of few-shot OOS intent detection [14, 16, 24, 25] or sentence representation learning [36, 37]. Fig. 2(a-c) depicts these three architectures.

Softmax model

A straightforward approach for text classification is to finetune a pre-trained language model that feeds a softmax-activated classification layer. As shown in Fig. 2(a), the input to the pre-trained encoder is constructed as "[cls] [utterance]". The last hidden state h_c of the [cls] token ($h_c \in \mathbb{R}^d$, with d the output dimension of the encoder) is passed to a linear layer (indicated as 'Head' in Fig. 2(a)), parameterized by $W \in \mathbb{R}^{d \times L}$ and $b \in \mathbb{R}^L$. The scores are normalized by a softmax layer and used in computing the cross-entropy loss. The maximal output of the softmax layer is used as the confidence score ϕ for the OOS detection [14, 16, 24], i.e., $\phi = \max \text{ softmax}(Wh_c + b)$. If ϕ

Table 4 In-scope and OOS intents of CLINC-Banking dataset. The description for each intent is also listed.

Intent label	Intent description
In-scope	
transactions	a request to review transactions related to a specific type of purchase, a specific time period, or a specific card
pay bill	a request for assistance or a statement to pay a specific type of bill where the payment account is optionally specified
spending history	an enquiry about the total amount of money spent on a particular type of purchase or during a specific time period
routing	asking for the routing number, with the bank or account name optionally specified
pin change	a statement of a forgotten pin code, or a request to create a new pin code or to change an existing pin code for a specified bank account
account blocked	requesting the reason behind a frozen, hold, or blocked bank account
report fraud	a statement of fraudulent activity, unauthorized access, or theft involving a bank account or card
interest rate	a request for obtaining the interest rate for an optionally specified bank account
bill balance	a request to obtain the total amount of outstanding bills, or due bills, of a specific type during a specific time period
order checks	a request to order additional checks for an optionally specified bank account
Out-of-scope	
balance	enquiring about the balance on a specified bank account, or whether the amount of money on a specified bank account is enough for a specific type of purchase
bill due	enquiring about when a specified type of bill is due for payment
min payment	enquiring about the minimum amount of money or payment for an optionally specified bill
freeze account	a request to freeze, block, or stop payments on a specified bank account
transfer	a request to transfer a specified amount of money from one bank account to another

exceeds the OOS score threshold, the intent label attaining score ϕ is used as classification output.

Siamese model

We implement a siamese network, which has shown great power in the representation learning of sentence embeddings [36, 37] and natural language inference tasks [38, 39]. Similar to the entailment classifier discussed next, a siamese classifier predicts a binary relationship between an utterance and an intent label depending on whether they match or not. The difference is that a siamese classifier treats an utterance and an intent label as two sentences and encodes them separately, generating two sentence vectors u, v (i.e., the [cls] encodings for utterance and intent). As illustrated in Fig. 2(b) , we compute the cosine

similarity score between u and v. The mean squared error between the similarity score and the true binary label is used as the training objective. During inference, we rank all the possible combinations of a test utterance and intent labels by their similarity scores. The highest score is used as the confidence score ϕ for OOS detection (where the corresponding intent label is thus ignored if ϕ falls below the OOS score threshold).

Entailment model

To better gauge the semantic relationship between utterances and intents, [25] proposes to concatenate an utterance and an intent label as a combined input to a pre-trained language model, in a cross-encoder setting. In particular, the utterance as premise and label as hypothesis are provided to an encoder in the format "[cls] [utterance] [sep] [intent]". A binary classification layer is added on top of the encoder, as shown in Fig. 2(c). The whole model is finetuned to predict "entailment" if the utterance's true label is the provided intent, and "non-entailment" otherwise. Specifically, the [cls] encoding is passed to a linear layer (with parameters $W \in \mathbb{R}^d$ and $b \in \mathbb{R}$) with a sigmoid activation for the binary entailment prediction. During inference, all the possible combinations of a test utterance and intent labels are evaluated. The highest entailment probability among the combinations is used as the confidence score ϕ for the OOS detection. If ϕ exceeds the OOS score threshold, the utterance is classified as the intent attaining that maximal score ϕ.

4.3 Experimental setup

We employ RoBERTa-base [40] as the backbone encoder for our prompt-based model and all the models used in our comparison. This ensures a fair comparison between the various architectures. However, our primary focus is not to merely pursue a state-of-the-art model. Instead, we aim to investigate the performance disparities of various key *architectures* for the OOS intent detection task under the proposed evaluation method.

For all the models under comparison, the learning rate is set to 1e-5 following common practice [18, 30]. The batch size and maximum text length are set to 64 and 128 respectively, given the memory limit of the GPU we used (a single NVIDIA GTX-1080Ti 12Gb). We use the AdamW optimizer [41] with weight decay of 0.01. The models are trained for 50 epochs, with the optimal epoch selected based on the AU-IOC score on the dev set. For different experiments, we keep using the same set of 5 different random seeds to ensure the sampled training data remains the same.

5 Results and Analysis

We first compare our prompt-based model's performance against baseline models, for limited labeled in-scope data (k-shot settings for $k = 1 \ldots 50$) (Section 5.1). Next, we analyze the robustness of our prompt-based detector

and baseline models (Section 5.2). Finally, as ablations, we study (i) our prompt-based model's sensitivity to different intent descriptions (Section 5.3), and (ii) the influence of choosing either a naturally phrased prompt or an artificially structured template (Section 5.4).

5.1 Few-shot OOS Intent Detection Performance

Table 5 shows the evaluation results of each model on the CLINC-Banking and BANKING datasets. The prompt-based model is found to perform very well. It obtains the highest AU-IOC score on all six datasets in 1/5/10-shot settings (except Stackoverflow), indicating that the prompt-based model is more data-efficient in the OOS intent detection task than the other architectures. Furthermore, in the extreme few-shot case (i.e., 1-shot), the prompt-based model outperforms other discriminative models (i.e., Siamese and Entailment) by a large margin on all six datasets. It is noteworthy that it is unfair to compare the Softmax model to the other models in the 1-shot setting, since the discriminative models receive extra training signal, i.e., the intent description, which can be considered a pseudo utterance. However, even if we compare the Softmax model's score at 2-shot against discriminative models' score at 1-shot (see Fig. 3), the inferiority of Softmax still stands. In fact, to achieve similar performance as the 1-shot prompt-based approach, the Softmax model needs at least 3 training instances per class.

In addition, we conduct a bootstrap analysis [42], to estimate the significance of the higher score of the prompt-based model with respect to each of the other models. To this end, we sampled the test results of all models 5,000 times with replacement. The resulting one-tailed significance levels (p) are indicated in Table 5 by markers $\star$, †, ‡, which denote $p < 0.1$, < 0.05 and < 0.01 with respect to the best model in each column. For the 1-shot setting, the prompt-model significantly outperforms the others, and its advantage becomes less pronounced for larger k.

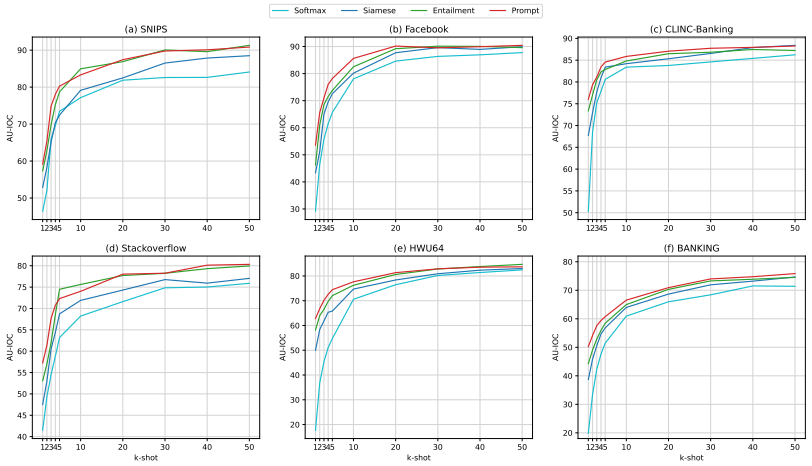

Fig. 3 AU-IOC scores for k-shot experiments (k from 1 to 50) on the six datasets. Results of each model are averaged over 5 runs with different random seeds. Best viewed in color.

We also compare k-shot OOS detection among the included models for a wider range of k, i.e., from 1 to 50, as shown in Fig. 3. For 4 of 6 datasets (i.e., Facebook, CLINC-Banking, HWU64 and BANKING), the prompt-based model outperforms the other 3 models over the full range of k and obtains the highest AU-IOC scores with extremely limited data, i.e., 1- to 4-shots. For SNIPS and Stackoverflow, the prompt-based model's performance is superior to the other models at 1- to 4-shots while attains close performance compared to the Entailment model. On Facebook and CLINC-Banking, the prompt-based model's performance plateaus from $k = 20$ onwards, whereas for HWU64 and BANKING, the score steadily increases up until $k = 50$, but its advantage over the other models is mostly pronounced for k up to 10 only.

Table 5 OOS detection performance (AU-IOC scores, test set) of baselines and the prompt-based model on CLINC-Banking and BANKING datasets with 1/5/10-shot training data. All the models use RoBERTa-base as the encoder. Reported numbers are mean (± std) over 5 runs with different random seeds. The markers ⋆, †, ‡ respectively denote the one-tailed significance levels of the bootstrap-based p-value, i.e., $p < 0.1$, < 0.05 and < 0.01 with respect to the highest scoring model in each column

SNIPS	1-shot	5-shot	10-shot
Softmax	46.42 (± 1.96)‡	73.46 (± 2.49)‡	77.15 (± 1.68)‡
Siamese	52.33 (± 1.87)‡	72.43 (± 0.80)‡	79.14 (± 0.78)‡
Entailment	54.02 (± 0.53)‡	78.73 (± 0.52)†	**84.95 (± 0.92)**
Prompt	**59.17 (± 1.44)**	**80.27 (± 1.34)**	83.31 (± 0.80)⋆
Facebook	1-shot	5-shot	10-shot
Softmax	29.24 (± 0.64)‡	65.67 (± 1.92)‡	78.05 (± 2.52)‡
Siamese	43.33 (± 3.80)‡	72.57 (± 1.47)‡	80.16 (± 0.75)‡
Entailment	46.25 (± 2.31)‡	73.79 (± 0.92)‡	82.52 (± 2.10)†
Prompt	**53.56 (± 2.13)**	**78.18 (± 2.05)**	**85.61 (± 1.89)**
CLINC-Banking	1-shot	5-shot	10-shot
Softmax	50.33 (± 2.16)‡	80.59 (± 1.80)‡	83.40 (± 2.19)⋆
Siamese	67.74 (± 2.69)‡	83.40 (± 3.39)	84.19 (± 1.22)
Entailment	73.37 (± 2.80)⋆	82.86 (± 0.80)‡	84.81 (± 1.32)
Prompt	**76.02 (± 1.64)**	**84.56 (± 1.26)**	**85.86 (± 2.61)**
Stackoverflow	1-shot	5-shot	10-shot
Softmax	41.49 (± 2.96)‡	63.19 (± 1.41)‡	68.22 (± 1.24)‡
Siamese	47.52 (± 1.07)‡	68.73 (± 2.73)‡	71.89 (± 0.47)‡
Entailment	53.08 (± 0.59)‡	**74.49 (± 2.35)**	**75.62 (± 1.60)**
Prompt	**57.25 (± 0.92)**	72.31 (± 1.83)	74.02 (± 1.41)‡
HWU64	1-shot	5-shot	10-shot
Softmax	17.69 (± 1.51)‡	54.76 (± 3.29)‡	70.59 (± 2.56)‡
Siamese	49.95 (± 2.45)‡	65.81 (± 2.46)‡	74.71 (± 0.23)‡
Entailment	58.15 (± 1.55)‡	72.06 (± 1.16)	76.24 (± 0.71)†
Prompt	**62.80 (± 1.06)**	**72.57 (± 0.68)**	**77.64 (± 0.94)**
BANKING	1-shot	5-shot	10-shot
Softmax	19.85 (± 0.81)‡	51.46 (± 2.62)‡	60.93 (± 1.22)‡
Siamese	38.70 (± 5.73)‡	56.81 (± 0.90)‡	63.98 (± 1.33)†
Entailment	44.27 (± 1.36)‡	58.44 (± 1.50)‡	64.97 (± 1.05)†
Prompt	**50.27 (± 0.73)**	**60.76 (± 0.85)**	**66.54 (± 0.58)**

5.2 Robustness of Prompt-Based Models

We analyze the robustness of the prompt-based model by comparing its IOC curves and confidence score distribution against other models.

5.2.1 Analysis of IOC curves

As discussed in Section 2.2-2.3, there is a trade-off between Acc_{in} and R_{oos}, i.e., and increase of R_{oos} typically comes with the cost of sacrificing Acc_{in}. A desired robust model is supposed to have as high Acc_{in} as possible across the full spectrum of R_{oos}. To gain more insight into the trade-off, in Fig. 4 we plot IOC curves (Acc_{in} vs. R_{oos}) at 1/5/10-shot of the prompt-based model and baseline models, for CLINC-Banking, Stackoverflow and BANKING.[5]

For CLINC-Banking, Fig. 4(a-c) shows that the prompt-based model is more robust than the other models, especially at 1-shot (Fig. 4(a)), where the prompt-based model achieves maximal Acc_{in} (of 0.87) and maintains the highest level for larger R_{oos}. The superiority of the prompt-based model remains at 5-shot (Fig. 4(b)). As the number of shots increases to 10 (Fig. 4(c)), the performance gap disappears. Nonetheless, the prompt-based model's Acc_{in} drops more slowly than other models, thus overall being the most robust model.

For Stackoverflow, the prompt-based model is the most robust at 1-shot setting as shown in Fig. 4(d), in which the prompt-based model achieves the highest Acc_{in} (0.69) compared to the other 3 models. Such performance difference is maintained as R_{oos} ranges from 0 to 1. While for the 5-shot setting Fig. 4(e), the Entailment model demonstrates the best robustness as its AU-IOC score is always higher than the other models'. For the 10-shot setting Fig. 4(f), a critical point is at R_{oos}=0.9 where the Siamese model's Acc_{in} starts to dominate. Such transition implies that each model may have its own advantageous range. In other words, with 10-shot training data of Stackoverflow, a trained Siamese model is a better choice for deployment if a very high R_{oos} is desired ($\geq$0.9).

For BANKING, the prompt-based model is consistently the most robust across various k-shot settings, as shown in Fig. 4(g-i). We do note that Acc_{in} is lower than those on CLINC-Banking and Stackoverflow, mainly because of BANKING's highest number (50) of in-scope classes and their close semantic relatedness as they all belong to a single domain. Further, we observe that for 5/10-shot settings, the Siamese model attains a slightly higher Acc_{in} than the prompt-based model does when $R_{oos} < 0.4$. However, the Siamese model fails to maintain this advantage for higher R_{oos} (>0.4), thus indicating its inferiority in terms of robustness.

In summary, according to the findings presented in Fig. 4, with few training examples per class ($k = 5$), the prompt-based model retains an overall higher Acc_{in} than the other models, even when the threshold is lowered to achieve a sufficiently high R_{oos}. In this sense, the prompt-based model turns out to

[5] For completeness, in Appendix A we also plot IOC curves of the 4 architectures for all the 6 datasets from 1- to 50-shot settings in Fig. A1-A6.

be the most robust model in a few-shot setting (even though this effect is no longer visible for $k = 10$, see Fig. 3). Also, for all k-shot settings on the 3 datasets, when R_{oos} approaches 1, the entailment model performs on par with the prompt-based model.

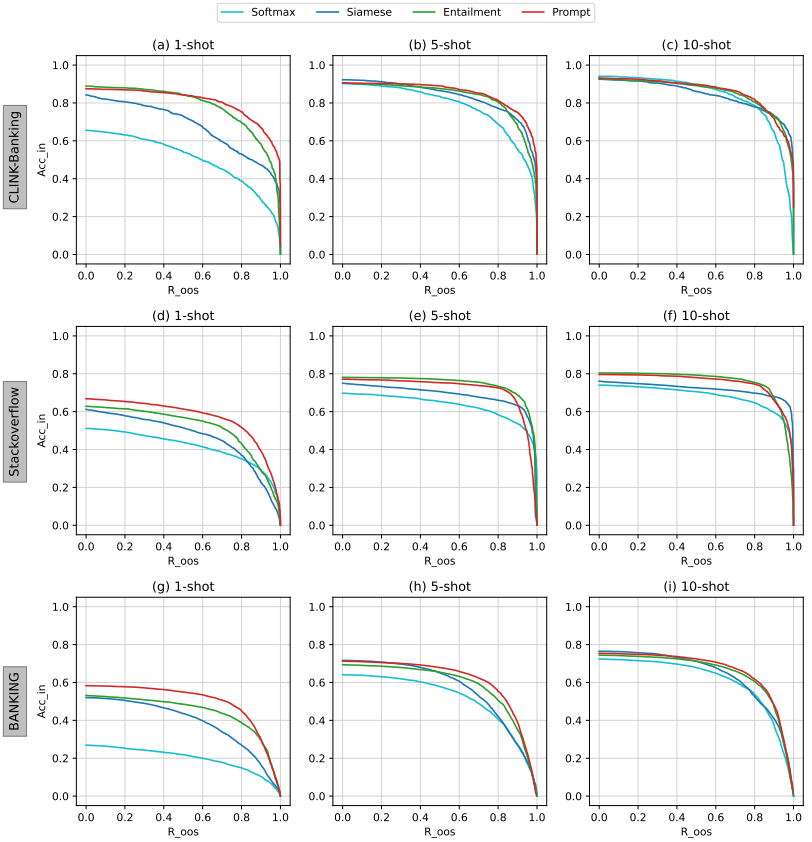

Fig. 4 IOC curves (Acc_{in} vs. R_{oos}) of 4 models in 1/5/10-shot settings evaluated on the test set of (a-c) CLINK-Banking, (d-f) Stackoverflow and (g-i) BANKING datasets. Results of each model are averaged over 5 runs with different random seeds. Best viewed in color.

5.2.2 Analysis of confidence score distributions

Fig. 5 presents the confidence score output by different models on 5-shot experiments of CLINC-Banking, Stackoverflow and BANKING.[6] Ideally, there would be no overlap between the in-scope and OOS confidence score distributions. Compared to the other three models, the prompt-based model is better at distinguishing in-scope and OOS samples as the overlap between the two sets is

[6]To save space, the score distributions of the other 3 datasets are plotted in Fig. B7, Appendix B.

minimal as shown in Fig. 5(d, h, l). The entailment model demonstrates comparable ability given the polarized distribution of confidence scores in Fig. 5(c, k). However, we observe a more scattered score distribution of the entailment model for in-scope examples (particularly for the BANKING dataset), as well as mistakenly higher scored OOS samples. Consequently, the entailment model has lower Acc_{in} compared to the prompt-based model when R_{oos} is fixed. However, there is an exception in Fig. 5(g) where the Entailment model exhibits a more polarized score distribution compared to that of the prompt-based model Fig. 5(h). The clearer differentiation between in-scope and OOS samples leads to a higher AU-IOC score (see Fig. 3(d)) and a more robust curve (see Fig. 4(e)). The drawback of the Softmax and Siamese models is similar: they are overly confident in assigning high scores to OOS samples. This causes both models to have lower R_{oos} compared to the prompt-based model when Acc_{in} is fixed.

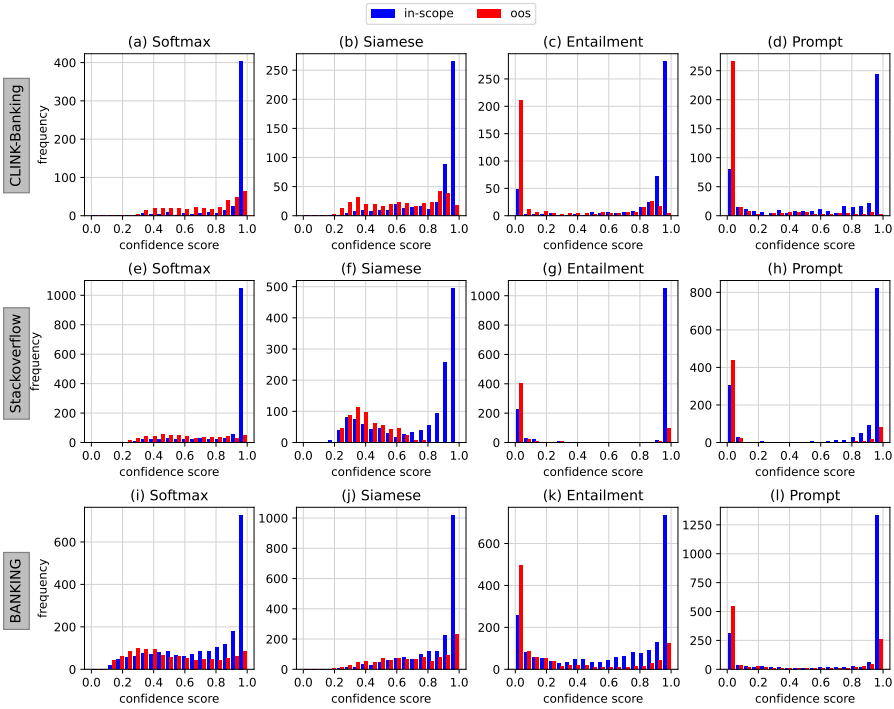

Fig. 5 Confidence score histogram at the 5-shot setting on the test set of (a-d) CLINK-Banking, (e-h) Stackoverflow and (i-l) BANKING. Best viewed in color.

5.3 Effects of Intent Description

As explained in Table 4, we provide a description for each intent, which enables the prompt-based model to better strike a balance between Acc_{in} and R_{oos}. However, the description in Table 4 is created by just a single annotator,

which may incur a certain bias in the label phrasing. To assess the impact of such annotation bias, we independently collected intent descriptions from 4 extra annotators. We assume that all annotators reach a similar level of understanding on an intent, even though they may phrase their descriptions differently based on their personal writing habits. Thus, their annotations' contents are expected to closely resemble each other.[7] An ideal OOS detection model should be insensitive to different descriptions, assuming they convey the same content. To this end, we experiment with descriptions of the 5 annotators and average the results labeled as **annotator** in Fig. 6.

Aside from annotator impacts, we investigate two additional questions: (1) Does the prompt-based model really "understand" the semantic relations between utterances and their corresponding intent descriptions? In other words, the model might classify utterances by only matching text patterns between utterances and intent descriptions. (2) Is it an overkill solution to use the intent descriptions? Why not simply use any utterance as the corresponding intent description, assuming it covers the key aspects of its intent class? The textual intent label itself might also serve as a useful intent description.

Motivated by these questions, we also experiment with three other settings to validate the effectiveness of intent descriptions/labels:

1) (**shuffle**) To investigate the effect of uninformative intent descriptions, we shuffle the mapping between intent labels and intent descriptions in Table 4; the shuffled descriptions thus serve as an artificial worst case scenario of unsuitable intent descriptions. For example, after shuffling, the description of account blocked is used to represent pay bill;
2) (**utterance**) We randomly sample an utterance for each intent from the largest train set (i.e., the 50-shot) as a "prototypical" utterance, which we in general expect to be less comprehensive than the annotator's description. We repeat the sampling process 5 times, obtaining different sets of "prototypical" utterances and average their results;
3) (**label**) Instead of using the description, we use the short intent label itself (e.g., pay bill, account blocked), which we a priori expect to be less informative than the longer description.

Fig. 6 summarizes the experimental results of the various utterance description/label settings for CLINC-Banking. The overall small standard deviation (across the different annotations) of AU-IOC in the **annotator** setting, indicates that the prompt-based model is rather insensitive to the exact phrasing of the intent descriptions. (Note that the standard deviation slightly larger for $k = 1$, which is likely caused by the random choice of the single training utterance.) Not surprisingly, the **shuffle** setting gets the worst scores across all k, as the model can hardly learn any meaningful relation between utterances and another label's descriptions, and apparently cannot learn to sufficiently ignore the incorrect intent description even for higher k. Interestingly, the **label** and **utterance** settings demonstrate comparable performance, with the latter being marginally better, except for the 20-shot scenario. This can be attributed

[7] All annotations are available from https://bit.ly/3Xo5BAR

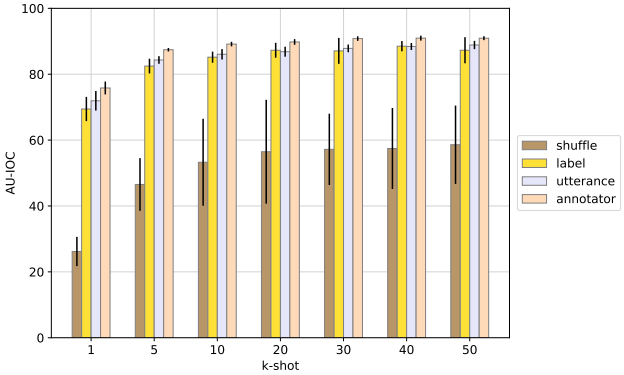

Fig. 6 Effects of different intent description on AU-IOC scores on CLINC-Banking (dev set) for the prompt-based model. 4 settings are compared: **shuffle**, **label**, **utterance** and **annotator**. Results of each description type are mean (± std) over 5 runs with different random seeds. Better viewed in color.

to the selected representative utterance not necessarily being more informative than a short label, as opposed to a description which is more carefully crafted by an annotator.

Overall, we conclude from these experiments that OOS detection performance is improved by using well-phrased descriptions to represent intents rather than using a random utterance or even a more informative but still short intent label in the prompt.

5.4 Effects of Prompt Templates

In Section 3.2, the prompt template is introduced as part of the prompt-based model. As suggested by prior research [43], the quality of a prompt template might influence the performance of text classification. Therefore, we investigate to what extent the design of prompt templates impacts the robustness of the prompt-based model in OOS detection. The templates we experiment with are listed in Table 6. The "unnatural templates" comprise an artificial template structure that only contains placeholders for an utterance, intent description and a mask token. We alter the position of [mask] among $\mathcal{T}_1$–$\mathcal{T}_3$ and always place [utterance] before [intent]. In contrast, the "natural templates" have better readability as we add phrases between [utterance] and [intent]. Take $\mathcal{T}_5$ as an example, where we assume a dummy person Joe is posing a question. The prompt-based model needs to answer "Does Joe mean [intent]?" with yes or no. For all the templates, we use RoBERTa-base as the encoder and use intent descriptions listed in Table 4 to fill in the [intent] placeholder.

Fig. 7 shows the AU-IOC scores of all the templates for the CLINC-Banking dataset. In general, $\mathcal{T}_5$ achieves the overall best performance, particularly on the 1-shot setting. This is likely due to the fact that compared to the other templates, $\mathcal{T}_5$ is composed in a more natural style that is better aligned to the corpus (e.g., book texts, Wikipedia) used for pretraining large language models,

Table 6 List of prompt templates. [mask] and [sep] are special tokens used by a pretrained language model (e.g., RoBERTa). Texts are fixed in a template.

Unnatural templates	
$\mathcal{T}_1$	[utterance] [mask] [intent]
$\mathcal{T}_2$	[utterance] [intent] [mask]
$\mathcal{T}_3$	[mask] [utterance] [intent]
Natural templates	
$\mathcal{T}_4$	[utterance] implies [intent] ? [mask]
$\mathcal{T}_5$	Joe said “ [utterance] ”. Does Joe mean [intent] ? [mask]
$\mathcal{T}_6$	“ [utterance] ” is asked by Joe. I think Joe means [intent]. Am I right? [mask]

including RoBERTa, which was used in our experiments. In the 1-shot setting, the OOS detection performance varies significantly among different templates. Apart from the template design, the high variance can be attributed to the inherent randomness associated with training on only one example. However, as the number of shots increases beyond 5, the variance of AU-IOC scores decreases, implying that prompt-based models are less sensitive to prompt templates when a sufficient amount of data is available. Nonetheless, on average, natural templates result in improved OOS detection performance compared to unnatural templates.

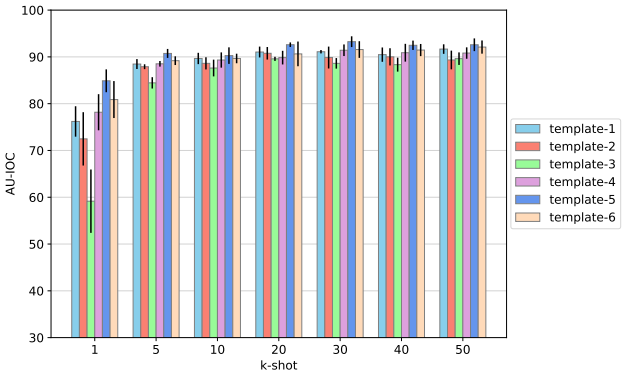

Fig. 7 Effects of different prompt templates on AU-IOC scores on CLINC-Banking (dev set). Templates 1-3 are unnatural templates and templates 4-6 are natural templates. Results of each template are mean (±std) on 5 runs with different random seeds. Better viewed in color.

6 Related work

Out-of-Scope Intent Detection. Out-of-scope (OOS) detection in text classification is an emerging field [9, 12, 13, 16, 22, 23, 44, 45]. Prior studies tackle this problem mainly using either of two approaches: (i) adding real or synthetic OOS samples to the training data as the $(L{+}1)^{\text{th}}$ class and learning a $(L{+}1)$-way classifier, where L denotes the number of in-scope classes [13, 44, 45],

or (ii) training a classifier without any OOS data [9, 12, 16, 22, 23]. OOS data can be expensive and difficult to acquire, particularly for specialized domains with large and uncertain data spaces. Despite synthetic OOS data being cheaper to obtain, it is often hard to interpret its textual meaning, as it is typically synthesized in a high-dimensional embedding space.

In our work, we focus on the second approach (ii), i.e., we assume training without access to OOS data, which is more appealing in the early stages of developing a dialogue system where no OOS data is available. Several studies have focused on OOS intent detection in few-shot learning scenarios, the most relevant to our work being [14, 15, 25, 46]. In [46], the Prototypical Network [47] is adapted to the OOS detection task. Their meta-learning framework independently samples meta-tasks that treat each other as simulated OOS data to obtain prototypical embeddings for each label as well as an OOS class, and eventually using cosine similarity to perform classification (similar to the Siamese architecture we include among our baseline models). The work of [15] is the first to explicitly define how to tune the threshold of the confidence score based on the sum of in-scope accuracy Acc_{in} and OOS recall R_{oos}. Additionally, they further pretrain BERT on a large amount of NLI data, which improves few-shot OOS detection performance compared to using the vanilla BERT model. Our work adopts the same threshold-based idea, but we propose a new evaluation method (cf. our AU-IOC metric) that is threshold-free and offers more holistic view of Acc_{in} and R_{oos}. Furthermore, we propose a prompt-based model that can use recent pretrained language models (PLMs) (either multi-lingual [48] or mono-lingual[8]) as is, without any further finetuning. We believe this is particularly useful for (low-resource) languages other than English (e.g., Arabic, Dutch) where suitable finetuning data (e.g., for NLI) may not be readily available.

Following the threshold-tuning evaluation method, [25] fuses the semantic information of intent labels into an entailment architecture in order to gauge if an utterance is in-scope or OOS. Conceptually, we are inspired by their finding that semantic information in the intent label name can be exploited to boost classification performance, and carry it further by adopting even more informative intent *descriptions* or sample *utterances*. Besides the entailment model, we further extend the idea to prompt-based models.

In [14], the robustness of various PLMs was evaluated using the Softmax architecture on the few-shot OOS intent detection task and RoBERTa was found the most robust among them. Based on their findings, we also choose RoBERTa as our text encoder. As stated before, we include the Softmax architecture idea among the baselines we compare our proposed prompt-based model against.

Prompt-based Learning. Our idea to adapt prompt-based learning (PBL) on the OOS intent detection task is mainly inspired by [18, 19, 28], in which masked language models are exploited for few-shot in-scope text classification, called Pattern Exploiting Training (PET). Our method differs from PET

[8] See [49, Tables 5–6] for a list of mono-lingual PLMs.

in 2 aspects: (1) PET focuses on in-scope classification while our research emphasizes OOS detection; (2) PET utilizes large task-specific unlabeled data and ensemble learning. Our method focuses on few-shot learning without any unlabeled data. We propose a single-model solution instead of training multiple models in order to lower the training cost. This technique has also been applied to other few-shot in-scope language understanding tasks such as named entity recognition [20] and relation extraction [21], where PBL achieved competitive results compared to strong baseline models.

As [50] has shown, another type of prompt-based learning based on causal language models (e.g., GPT-2) also has promising performance in few-shot in-scope text classification. However, our pilot experiments showed that employing GPT-2 Small as the base encoder (with comparable number of parameters to that of RoBERTa-base) yields much worse results in OOS intent detection compared to those from using RoBERTa-base. The suboptimal OOS detection performance of using GPT-2 Small could be attributed to its rather limited capacity, which is supported by the ablation studies of [50] that revealed that on average, GPT-2 Large yielded significantly better classification accuracy compared to GPT-2 Small. Unfortunately, our current computational resources disallows us to train or finetune larger models, such as RoBERTa-large or GPT-2 Medium/Large. However, this presents a promising avenue for future research.

7 Conclusion and Future Work

In this work, we investigate the robustness of prompt-based learning in the few-shot out-of-scope (OOS) intent classification task. Inspired by the recent success of prompt-based learning in few-shot in-scope language understanding tasks, we propose a simple yet effective prompt-based OOS detector by leveraging a masked language model.

Furthermore, to overcome the limitations of existing evaluation metrics for OOS detection, we propose a more comprehensive metric, the Area Under In-scope and Out-of-Scope Characteristic curve (AU-IOC), which offers a holistic view of in-scope accuracy and OOS recall and clearly distinguishes different OOS detection models.

Under this new evaluation method, we compare our prompt-based detector against 3 strong baseline models by performing extensive experiments on 6 datasets. Our study found that by exploiting the meta-data of intent annotation (i.e., the intent description), the prompt-based detector achieved the highest AU-IOC score across various data regimes (from 1- to 50-shot) and significantly outperforms all the baseline models at extremely low data settings (1/5-shot). Additionally, further experiments showed that our prompt-based detector is insensitive to intent descriptions phrased in different formats. Also, we found that a prompt template formulated in a natural style is a key ingredient to the high robustness of our prompt-based detector.

As for our future work, we will extend the scope of prompt-based models by adapting causal language models (e.g., GPT-3, ChatGPT) on the few-shot OOS intent classification task. Our work mainly relies on discrete prompt templates, which need manual design and cannot be parameterized. To this end, we are also interested in investigating the effects of their continuous counterparts.

Declarations

Scientific assessment

We thank the reviewers for their useful feedback, which helped us to improve the manuscript, including with their suggestion to add more datasets.

Funding

The first author is supported by *China Scholarship Council* (No. 201906020194) and *Ghent University Special Research Fund (BOF)* (No. 01SC0618). This research also received funding from the Flemish Government under the *"Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen"* programme.

Author Contribution

Yiwei Jiang: Conceptualization, Methodology, Software, Investigation, Writing - original draft preparation. **Maarten De Raedt**: Conceptualization, Investigation, Writing - review and editing. **Johannes Deleu**: Conceptualization, Investigation, Writing - review and editing. **Thomas Demeester**: Conceptualization, Investigation, Writing - review and editing, Supervision. **Chris Develder**: Conceptualization, Investigation, Writing - review and editing, Supervision.

Competing Interests

The authors have no competing interests to disclose in any material discussed in this article.

Ethical Compliance

This study does not involve any human participant or animal. All the data used in this article are sourced from open and publicly accessible platforms. No proprietary, confidential, or private data has been used.

Data Availability

The original datasets used in this study come from multiple studies [24, 31–35]. Our work adapted these datasets for training our models and experiment analysis. The adapted versions are available from the corresponding author on reasonable request.

Appendix A IOC curves at 1-50 shots

Fig. A1-A6 plot IOC curves of different models in 1-50 shot settings for SNIPS, Facebook, CLINC-Banking, Stackoverflow, HWU64 and BANKING dataset respectively.

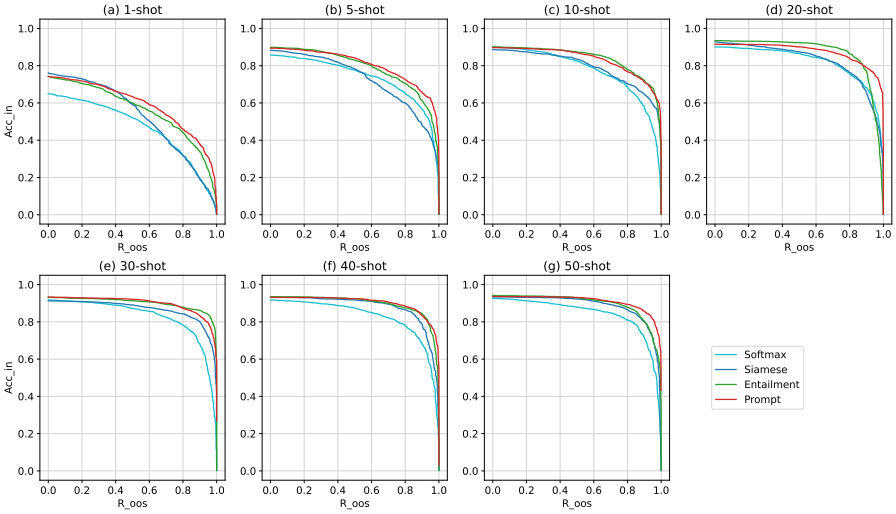

Fig. A1 IOC curves (Acc_{in} vs. R_{oos}) of models in 1-50 shot settings evaluated on SNIPS (test set). Better viewed in color.

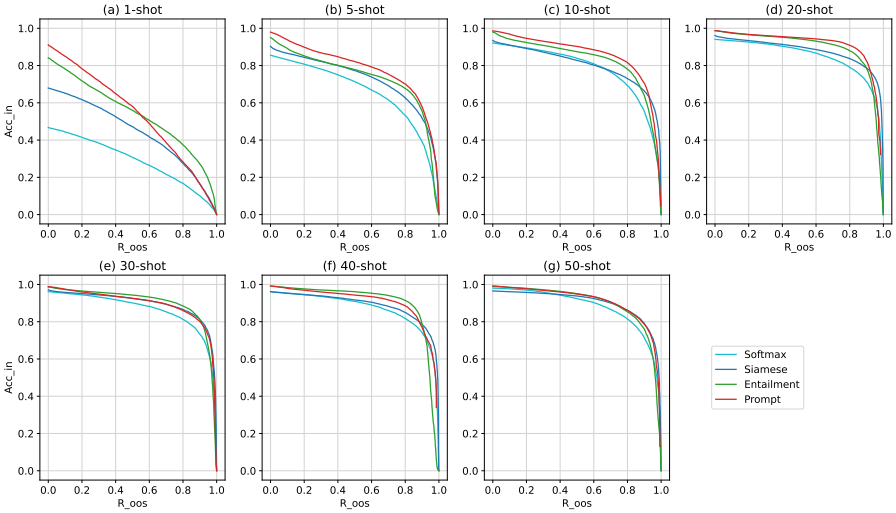

Fig. A2 IOC curves (Acc_{in} vs. R_{oos}) of models in 1-50 shot settings evaluated on Facebook (test set). Better viewed in color.

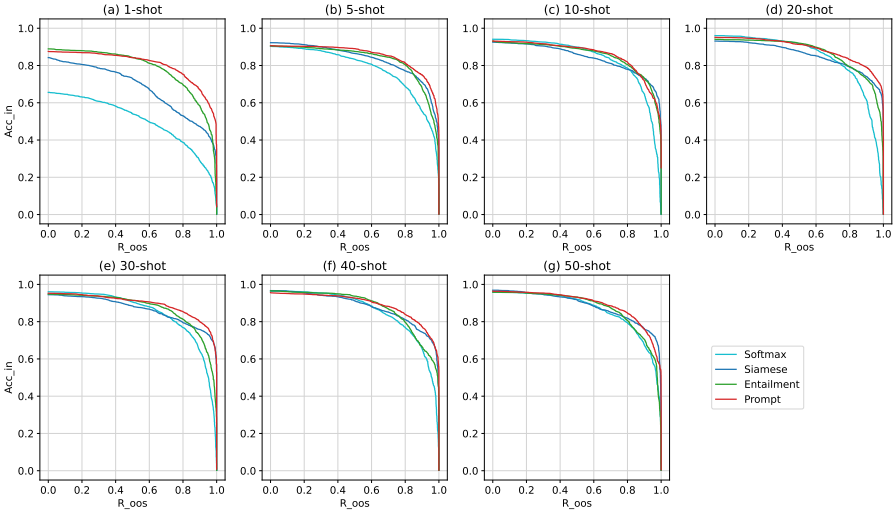

Fig. A3 IOC curves (Acc_{in} vs. R_{oos}) of models in 1-50 shot settings evaluated on CLINC-Banking (test set). Better viewed in color.

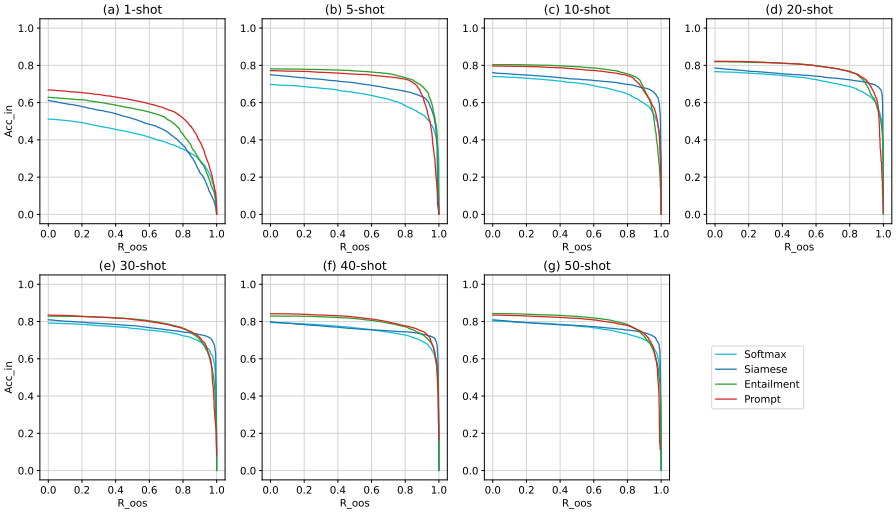

Fig. A4 IOC curves (Acc_{in} vs. R_{oos}) of models in 1-50 shot settings evaluated on Stack-overflow (test set). Better viewed in color.

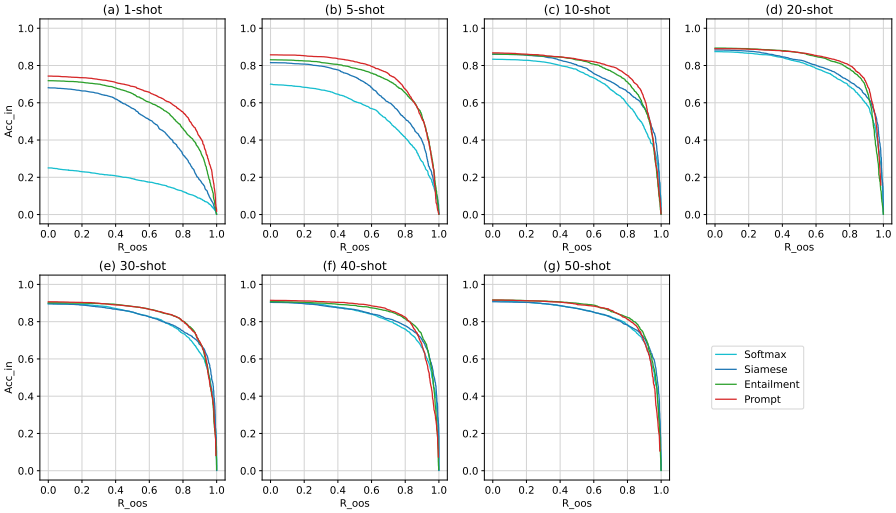

Fig. A5 IOC curves (Acc_{in} vs. R_{oos}) of models in 1-50 shot settings evaluated on HWU64 (test set). Better viewed in color.

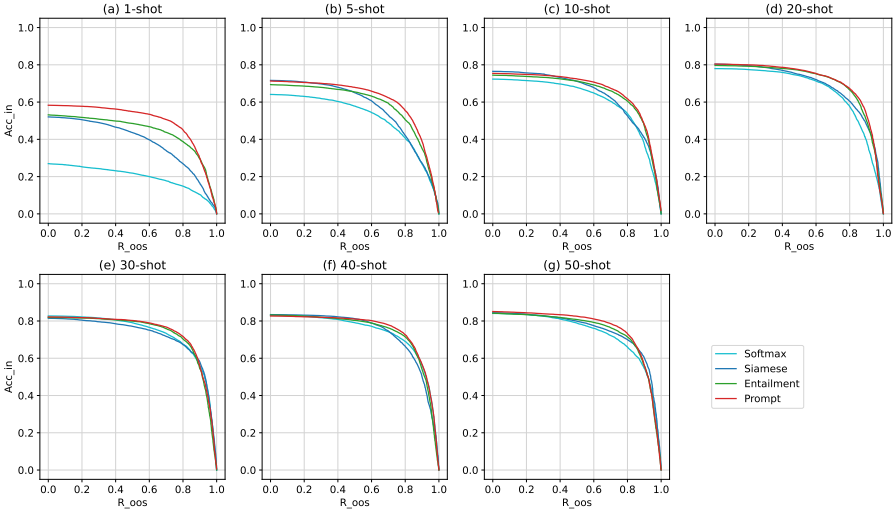

Fig. A6 IOC curves (Acc_{in} vs. R_{oos}) of models in 1-50 shot settings evaluated on BANKING (test set). Better viewed in color.

Appendix B Confidence score distributions of the other 3 datasets at 5-shot

Fig. B7 shows the confidence score distributions of the 4 architectures on 3 datasets (SNIPS, Facebook and HWU64) at 5-shot.

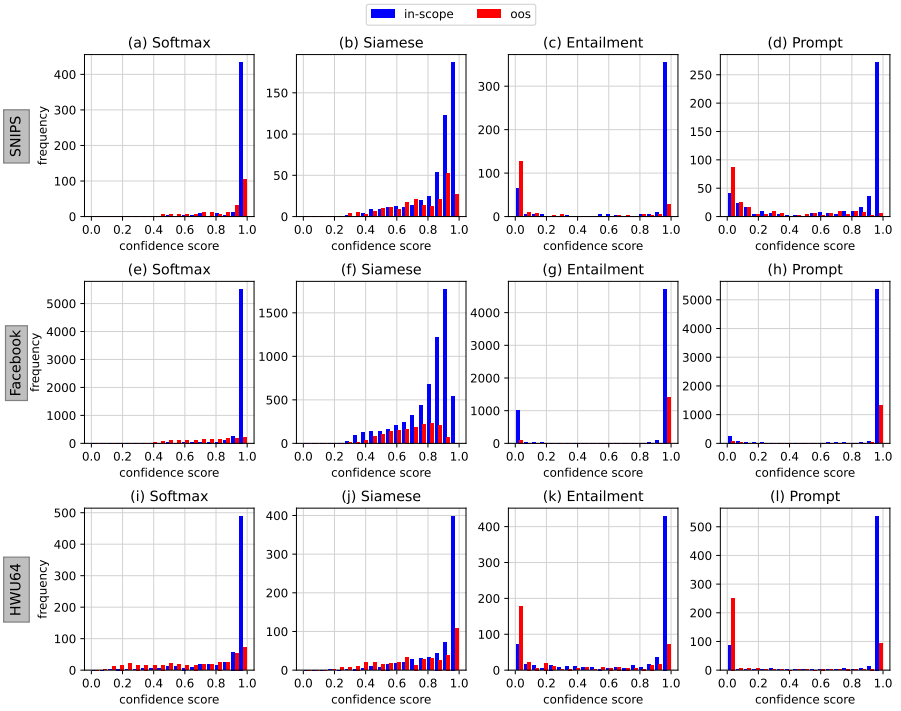

Fig. B7 Confidence score histogram at the 5-shot setting on the test set of (a-d) SNIPS, (e-h) Facebook and (i-l) HWU64. Best viewed in color.

Appendix C Inference speed

Fig. C8 illustrates the inference throughput against the number of in-scope classes (denoted as L). To ensure a fair comparison between the models and to aptly simulate the online evaluation setting, we standardized the input batch size to 1 across all models. This means that each batch contains only a single user question. We observed that the throughput of the Softmax model remains relatively stable (approximately 62 instances/s) irrespective of the variations in L. The Softmax model bypasses the one-vs-all binary classification, thereby exhibiting speed insensitivity. In the case of the Siamese model, we implemented a strategy to cache the intent label embedding to foster efficiency. However, despite this optimization, we found that the computational demand for the cosine similarity operation escalates as L increases. In contrast, the throughput for the other two models is much smaller when L increases over 14. Notably, the prompt-based model surpasses others in achieving higher AU-IOC scores, albeit at the expense of reduced inference throughput, particularly when L exceeds 10. A significant factor contributing to this reduced speed is the tensor extraction operations involved in the prompt-based model, requiring data transfer between the GPU and CPU, which is time-consuming. While our primary focus in this study remains on scrutinizing the robustness of different models in handling

the Out-of-Scope (OOS) intent detection task, we acknowledge that optimizing the inference speed is a critical aspect that warrants attention in future work. It is also pertinent to note that the inference time is contingent upon the hardware utilized during the evaluation, implying that a change in hardware could potentially alter the throughput numbers reported.

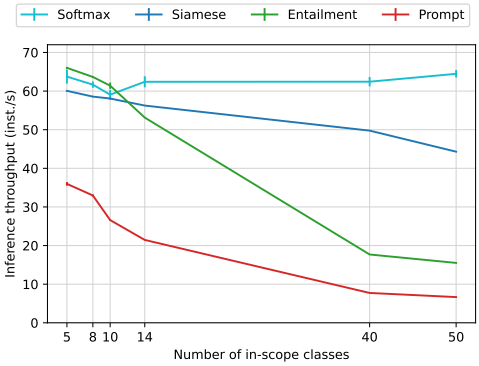

Fig. C8 Inference throughput v.s. number of in-scope classes. All the throughput numbers are computed with a single NVIDIA GTX-1080Ti GPU.

References

[1] Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. In: Proceedings of ICLR, Vancouver, BC, Canada (2018). https://openreview.net/forum?id=H1VGkIxRZ

[2] Hsu, Y., Shen, Y., Jin, H., Kira, Z.: Generalized ODIN: detecting out-of-distribution image without learning from out-of-distribution data. In: Proceedings of CVPR, Seattle, WA, USA, pp. 10948–10957 (2020). https://doi.org/10.1109/CVPR42600.2020.01096

[3] Ren, J., Liu, P.J., Fertig, E., Snoek, J., Poplin, R., DePristo, M.A., Dillon, J.V., Lakshminarayanan, B.: Likelihood ratios for out-of-distribution detection. In: Proceedings of NeurIPS, vol. 32. Vancouver, BC, Canada, pp. 14680–14691 (2019). https://proceedings.neurips.cc/paper/2019/hash/1e79596878b2320cac26dd792a6c51c9-Abstract.html

[4] Lee, K., Lee, K., Lee, H., Shin, J.: A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In: Proceedings of NeurIPS, vol. 31. Montréal, Canada, pp. 7167–7177 (2018). https://proceedings.neurips.cc/paper/2018/hash/abdeb6f575ac5c6676b747bca8d09cc2-Abstract.html

[5] Zheng, Y., Chen, G., Huang, M.: Out-of-domain detection for natural language understanding in dialog systems. IEEE/ACM Trans. Audio, Speech and Lang. Proc. **28**, 1198–1209 (2020). https://doi.org/10.1109/TASLP.2020.2983593

[6] Jin, D., Gao, S., Kim, S., Liu, Y., Hakkani-Tür, D.: Towards textual out-of-domain detection without in-domain labels. IEEE/ACM Trans. Audio, Speech and Lang. Proc. **30**, 1386–1395 (2022). https://doi.org/10.1109/TASLP.2022.3162081

[7] Shen, Y., Hsu, Y.-C., Ray, A., Jin, H.: Enhancing the generalization for intent classification and out-of-domain detection in SLU. In: Proceedings of ACL, pp. 2443–2453. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.acl-long.190

[8] Zhou, W., Liu, F., Chen, M.: Contrastive out-of-distribution detection for pretrained transformers. In: Proceedings of EMNLP, pp. 1100–1111. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (2021). https://doi.org/10.18653/v1/2021.emnlp-main.84

[9] Zhang, H., Xu, H., Lin, T.-E.: Deep open intent classification with adaptive decision boundary. In: Proceedings of AAAI, vol. 35, pp. 14374–14382. AAAI Press, Online (2021). https://ojs.aaai.org/index.php/AAAI/article/view/17690

[10] Lane, I., Kawahara, T., Matsui, T., Nakamura, S.: Out-of-domain utterance detection using classification confidences of multiple topics. IEEE/ACM Trans. Audio, Speech and Lang. Proc. **15**(1), 150–161 (2007). https://doi.org/10.1109/TASL.2006.876727

[11] Iqbal, T., Cao, Y., Kong, Q., Plumbley, M.D., Wang, W.: Learning with out-of-distribution data for audio classification. In: Proceedings of ICASSP, pp. 636–640. IEEE, Barcelona, Spain (2020). https://doi.org/10.1109/ICASSP40776.2020.9054444

[12] Lin, T.-E., Xu, H.: Deep unknown intent detection with margin loss. In: Proceedings of ACL, pp. 5491–5496. Association for Computational Linguistics, Florence, Italy (2019). https://doi.org/10.18653/v1/P19-1548

[13] Zhan, L.-M., Liang, H., Liu, B., Fan, L., Wu, X.-M., Lam, A.Y.S.: Out-of-scope intent detection with self-supervision and discriminative training. In: Proceedings of ACL, pp. 3521–3532. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.acl-long.273

[14] Zhang, J., Hashimoto, K., Wan, Y., Liu, Z., Liu, Y., Xiong, C., Yu, P.: Are pre-trained transformers robust in intent classification: A missing ingredient in evaluation of out-of-scope intent detection. In: Proceedings

of the 4th Workshop on NLP for ConvAI, pp. 12–20. Association for Computational Linguistics, Dublin, Ireland (2022). https://doi.org/10.18653/v1/2022.nlp4convai-1.2

[15] Zhang, J., Hashimoto, K., Liu, W., Wu, C.-S., Wan, Y., Yu, P., Socher, R., Xiong, C.: Discriminative nearest neighbor few-shot intent detection by transferring natural language inference. In: Proceedings of EMNLP, pp. 5064–5082. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.emnlp-main.411

[16] Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: Proceedings of ICLR, Toulon, France (2017). https://openreview.net/forum?id=Hkg4TI9xl

[17] Liu, J., Lin, Z., Padhy, S., Tran, D., Bedrax Weiss, T., Lakshminarayanan, B.: Simple and principled uncertainty estimation with deterministic deep learning via distance awareness, vol. 33. Online, pp. 7498–7512 (2020). https://proceedings.neurips.cc/paper/2020/hash/543e83748234f7cbab21aa0ade66565f-Abstract.html

[18] Schick, T., Schütze, H.: Exploiting cloze-questions for few-shot text classification and natural language inference. In: Proceedings of EACL, pp. 255–269. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.eacl-main.20

[19] Schick, T., Schütze, H.: It's not just size that matters: Small language models are also few-shot learners. In: Proceedings of NAACL, pp. 2339–2352. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.naacl-main.185

[20] Li, D., Hu, B., Chen, Q.: Prompt-based text entailment for low-resource named entity recognition. In: Proceedings of ICCL, pp. 1896–1903. International Committee on Computational Linguistics, Gyeongju, Republic of Korea (2022). https://aclanthology.org/2022.coling-1.164

[21] Chen, Y., Harbecke, D., Hennig, L.: Multilingual relation classification via efficient and effective prompting. In: Proceedings of EMNLP, pp. 1059–1075. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (2022). https://aclanthology.org/2022.emnlp-main.69

[22] Shu, L., Xu, H., Liu, B.: DOC: Deep open classification of text documents. In: Proceedings of EMNLP, pp. 2911–2916. Association for Computational Linguistics, Copenhagen, Denmark (2017). https://doi.org/10.18653/v1/D17-1314

[23] Yan, G., Fan, L., Li, Q., Liu, H., Zhang, X., Wu, X.-M., Lam, A.Y.S.:

Unknown intent detection using Gaussian mixture model with an application to zero-shot intent classification. In: Proceedings of ACL, pp. 1050–1060. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.acl-main.99

[24] Larson, S., Mahendran, A., Peper, J.J., Clarke, C., Lee, A., Hill, P., Kummerfeld, J.K., Leach, K., Laurenzano, M.A., Tang, L., Mars, J.: An evaluation dataset for intent classification and out-of-scope prediction. In: Proceedings of EMNLP, pp. 1311–1316. Association for Computational Linguistics, Hong Kong, China (2019). https://doi.org/10.18653/v1/D19-1131

[25] Qu, J., Hashimoto, K., Liu, W., Xiong, C., Zhou, Y.: Few-shot intent classification by gauging entailment relationship between utterance and semantic label. In: Proceedings of the 3rd Workshop on NLP for ConvAI, pp. 8–15. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.nlp4convai-1.2

[26] Davis, J., Goadrich, M.: The relationship between precision-recall and roc curves. In: Proceedings of ICML, pp. 233–240. Association for Computing Machinery, Pittsburgh, Pennsylvania, USA (2006). https://doi.org/10.1145/1143844.1143874

[27] Fawcett, T.: An introduction to roc analysis. Pattern Recognition Letters **27**(8), 861–874 (2006). https://doi.org/10.1016/j.patrec.2005.10.010

[28] Schick, T., Schütze, H.: True Few-Shot Learning with Prompts—A Real-World Perspective. Transactions of the Association for Computational Linguistics **10**, 716–731 (2022). https://doi.org/10.1162/tacl_a_00485

[29] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL, pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/N19-1423

[30] Tam, D., R. Menon, R., Bansal, M., Srivastava, S., Raffel, C.: Improving and simplifying pattern exploiting training. In: Proceedings of EMNLP, pp. 4980–4991. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (2021). https://doi.org/10.18653/v1/2021.emnlp-main.407

[31] Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., Doumouro, C., Gisselbrecht, T., Caltagirone, F., Lavril, T., Primet, M., Dureau, J.: Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. ArXiv **abs/1805.10190** (2018)

[32] Schuster, S., Gupta, S., Shah, R., Lewis, M.: Cross-lingual transfer learning for multilingual task oriented dialog. 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), pp. 3795–3805. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/N19-1380. https://aclanthology.org/N19-1380

[33] Xu, J., Wang, P., Tian, G., Xu, B., Zhao, J., Wang, F., Hao, H.: Short text clustering via convolutional neural networks. In: Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, pp. 62–69. Association for Computational Linguistics, Denver, Colorado (2015). https://doi.org/10.3115/v1/W15-1509. https://aclanthology.org/W15-1509

[34] Liu, X., Eshghi, A., Swietojanski, P., Rieser, V.: Benchmarking natural language understanding services for building conversational agents. In: Increasing Naturalness and Flexibility in Spoken Dialogue Interaction: 10th International Workshop on Spoken Dialogue Systems, pp. 165–183 (2021). https://doi.org/10.1007/978-981-15-9323-9_15. Springer

[35] Casanueva, I., Temcinas, T., Gerz, D., Henderson, M., Vulic, I.: Efficient intent detection with dual sentence encoders. In: Proceedings of the 2nd Workshop on NLP for ConvAI - ACL 2020 (2020). Data available at https://github.com/PolyAI-LDN/task-specific-datasets. https://arxiv.org/abs/2003.04807

[36] Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: Proceedings of EMNLP, pp. 3982–3992. Association for Computational Linguistics, Hong Kong, China (2019). https://doi.org/10.18653/v1/D19-1410

[37] Thakur, N., Reimers, N., Daxenberger, J., Gurevych, I.: Augmented SBERT: Data augmentation method for improving Bi-encoders for pairwise sentence scoring tasks. In: Proceedings of NAACL, pp. 296–310. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.naacl-main.28

[38] Chen, Q., Zhu, X., Ling, Z.-H., Wei, S., Jiang, H., Inkpen, D.: Enhanced LSTM for natural language inference. In: Proceedings of ACL, pp. 1657–1668. Association for Computational Linguistics, Vancouver, Canada (2017). https://doi.org/10.18653/v1/P17-1152

[39] Williams, A., Nangia, N., Bowman, S.: A broad-coverage challenge corpus for sentence understanding through inference. In: Proceedings of NAACL, pp. 1112–1122. Association for Computational Linguistics, New Orleans, Louisiana (2018). https://doi.org/10.18653/v1/N18-1101

[40] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: RoBERTa: A robustly optimized BERT pretraining approach. ArXiv **abs/1907.11692** (2019)

[41] Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: Proceedings of ICLR, Vancouver, BC, Canada (2019). https://openreview.net/forum?id=Bkg6RiCqY7

[42] Efron, B., Tibshirani, R.J.: An Introduction to the Bootstrap. CRC press, New York, USA (1994)

[43] Gao, T., Fisch, A., Chen, D.: Making pre-trained language models better few-shot learners. In: Proceedings of ACL, pp. 3816–3830. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.acl-long.295

[44] Chen, D., Yu, Z.: GOLD: Improving out-of-scope detection in dialogues using data augmentation. In: Proceedings of EMNLP, pp. 429–442. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (2021). https://doi.org/10.18653/v1/2021.emnlp-main.35

[45] Cheng, Z., Jiang, Z., Yin, Y., Wang, C., Gu, Q.: Learning to classify open intent via soft labeling and manifold mixup. IEEE/ACM Trans. Audio, Speech and Lang. Proc. **30**, 635–645 (2022). https://doi.org/10.1109/TASLP.2022.3145308

[46] Tan, M., Yu, Y., Wang, H., Wang, D., Potdar, S., Chang, S., Yu, M.: Out-of-domain detection for low-resource text classification tasks. In: Proceedings of EMNLP, pp. 3566–3572. Association for Computational Linguistics, Hong Kong, China (2019). https://doi.org/10.18653/v1/D19-1364

[47] Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: Proceedings of NeurIPS, vol. 30. Long Beach, CA, USA, pp. 4077–4087 (2017). https://proceedings.neurips.cc/paper/2017/file/cb8da6767461f2812ae4290eac7cbc42-Paper.pdf

[48] Conneau, A., Lample, G.: Cross-lingual language model pretraining. In: Proceddings of NeurIPS, vol. 32. Vancouver, BC, Canada (2019). https://proceedings.neurips.cc/paper/2019/file/c04c19c2c2474dbf5f7ac4372c5b9af1-Paper.pdf

[49] Kalyan, K.S., Rajasekharan, A., Sangeetha, S.: Ammu: A survey of transformer-based biomedical pretrained language models. J. of Biomedical Informatics **126**(C) (2022). https://doi.org/10.1016/j.jbi.2021.103982

[50] Min, S., Lewis, M., Hajishirzi, H., Zettlemoyer, L.: Noisy channel language

model prompting for few-shot text classification. In: Proceedings of ACL, pp. 5316–5330. Association for Computational Linguistics, Dublin, Ireland (2022). https://doi.org/10.18653/v1/2022.acl-long.365