ProtoInfoMax: Prototypical Networks with Mutual Information Maximization for Out-of-Domain Detection

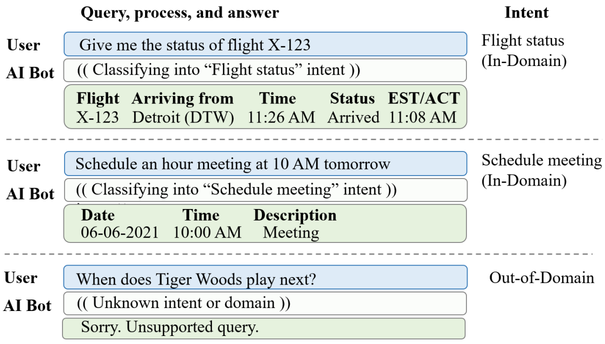
Iftitahu Ni'mah[1,2], **Meng Fang**[1], **Vlado Menkovski**[1], **Mykola Pechenizkiy**[1]
Eindhoven University of Technology (TU/e)[1]
Indonesian Institute of Science (LIPI)[2]
{i.nimah, m.fang, v.menkovski, m.pechenizkiy}@tue.nl[1]

Abstract

The ability to detect Out-of-Domain (OOD) inputs has been a critical requirement in many real-world NLP applications since the inclusion of unsupported OOD inputs may lead to catastrophic failure of systems. However, it remains an empirical question whether current algorithms can tackle such problem reliably in a realistic scenario where zero OOD training data is available. In this study, we propose ProtoInfoMax, a new architecture that extends Prototypical Networks to simultaneously process In-Domain (ID) and OOD sentences via Mutual Information Maximization (InfoMax) objective. Experimental results show that our proposed method can substantially improve performance up to 20% for OOD detection in low resource settings of text classification. We also show that ProtoInfoMax is less prone to typical over-confidence Error of Neural Networks, leading to more reliable ID and OOD prediction outcomes.

1 Introduction

Many real-world applications imply an open world assumption (Scheirer et al., 2013; Fei and Liu, 2016) [1], requiring intelligent system to be aware of novel OOD examples, given limited ID and zero OOD training data. Intent classification for conversational AI services, for instance, often have to deal with unseen OOD utterances (Tan et al., 2019; Kim and Kim, 2018a; Larson et al., 2019; Zheng et al., 2020). Question answering system is also preferred to have a certain degree of language understanding via its ability to contrast between relevant and irrelevant sentences (Yeh and Chen, 2019). Likewise, a classifier trained on past topics of social media posts is often expected to be aware of future social media streams with new unseen topics (Fei and Liu, 2016; Fei et al., 2016). An example of AI system with OOD awareness is illustrated in Figure 1. When user inputs unknown query with OOD intent, instead of providing random feedback, a system that is aware of OOD inputs can better respond informatively.


Query, process, and answer | **Intent**

User Give me the status of flight X-123 | Flight status (In-Domain)
AI Bot ((Classifying into "Flight status" intent))

Flight	Arriving from	Time	Status	EST/ACT
X-123	Detroit (DTW)	11:26 AM	Arrived	11:08 AM

User Schedule an hour meeting at 10 AM tomorrow | Schedule meeting (In-Domain)
AI Bot ((Classifying into "Schedule meeting" intent))

Date	Time	Description
06-06-2021	10:00 AM	Meeting

User When does Tiger Woods play next? | Out-of-Domain
AI Bot ((Unknown intent or domain))
Sorry. Unsupported query.

Figure 1: An example of OOD detection in task-oriented dialog system.

To develop a reliable intelligent system that can correctly process ID inputs and detect unclassified inputs from different distribution (OOD), existing approaches often formulate OOD detection as anomaly detection (Ryu et al., 2017, 2018; Hendrycks et al., 2019). The concept of learning ID classification and OOD detection tasks simultaneously is also incorporated in diverse applications, including open text classification (Shu et al., 2017) and OOD detection in task-oriented dialog system (Kim and Kim, 2018b; Zheng et al., 2020). These methods rely on large-scale ID and OOD labeled training data or well-defined data distributions.

Unfortunately, large data setting makes the methods unrealistic for many real world applications with limited ID and zero OOD training data. As a result, current research introduces few-shot and zero-shot learning frameworks for OOD detection problem in low resource scenario of text classification (Tan et al., 2019). Their objective is to learn a metric space for ID and OOD prediction given prototype representation of ID sentences and target sentences sampled from ID and OOD distri-

[1] System built under this assumption should be able to not only correctly analyze ID inputs but also reliably reject OOD inputs that are not supported by the system.

bution. However, the current method neglects an over-confidence issue of the trained Prototypical Networks in inference stage where both novel ID and OOD inputs occur. For example, OOD samples are likely to be classified as ID with a high similarity score (e.g. $d \approx 1.0$) (Liang et al., 2018; Shafaei et al., 2019), especially if they share common patterns or semantics with ID samples (e.g. common phrases, sentence topicality, sentiment polarity) (Lewis and Fan, 2019).

To mitigate the above problems, we adopt Mutual Information Maximization (InfoMax) objective (Belghazi et al., 2018; Hjelm et al., 2019) for regularizing Prototypical Networks (Section 3.1). We extend Prototypical Networks (Snell et al., 2017) to learn multiple prototype representations by maximizing Mutual Information (MI) estimate between sentences that share a relevant context, such as keywords (Section 3.2). We demonstrate that our proposed method is less prone to typical Over-confidence Error of Neural Networks (Lakshminarayanan et al., 2017; Guo et al., 2017; Liang et al., 2018; Shafaei et al., 2019). This result leads to more reliable prediction outcomes, specifically in inference stage where the model has to deal with both novel ID and OOD examples. Overall, experimental results on real-world low-resource sentiment and intent classification (Section 5) show that the proposed method can substantially improve performance of the existing approach up to 20%.

To summarize, our contributions are as follows:

- We introduce ProtoInfoMax – Prototypical Networks that learn to distinguish between ID and OOD representations via Mutual Information Maximization (InfoMax) objective (Section 3.1).

- We enhance ProtoInfoMax by incorporating multiple prototype representations (Section 3.2) to further improve the discriminability of the learned metric space.

- We further investigate the reliability of the trained Prototypical Networks in this study (Section 5.3-5.4). Our problem of interest is determining if a target query x is from ID distribution $\mathcal{P}_{id}$ or OOD distribution $\mathcal{P}_{ood}$, the learned metric space shall indicates a well calibrated model $P_d(y|x)$. That is, the model assigns high similarity score ($d \approx 1.0$) for test samples drawn from $\mathcal{P}_{id}$ and assigns low score ($d \approx 0.0$) for samples drawn from $\mathcal{P}_{ood}$.

2 Problem Definition

Similar to the previous setting (Tan et al., 2019), we consider zero-shot OOD detection problem for meta-tasks in this study. In general, there are three main inputs (Figure 2) for prototypical learning in this study: ID support set S^{id}, ID target query Q^{id}, and OOD target query Q^{ood}.

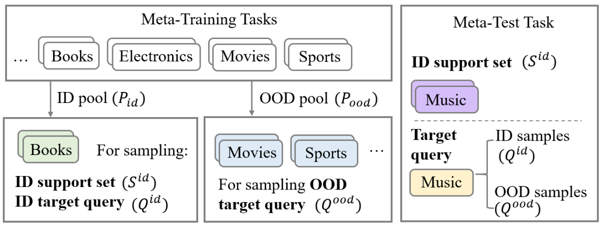


Figure 2: Example of meta-training and meta-test tasks for ID few-shot and OOD zero-shot learning in this study. OOD labels are unknown to the model during training.

Figure 2 illustrates an overview of meta-training and meta-testing task construction in sentiment classification data. For each training episode, ID examples $\mathcal{D}_{id}$ are drawn independently from ID distribution in meta-training tasks $\mathcal{P}_{T_i}, T_i \in \mathcal{T}$ (later refer to as $\mathcal{P}_{id}$), defined as

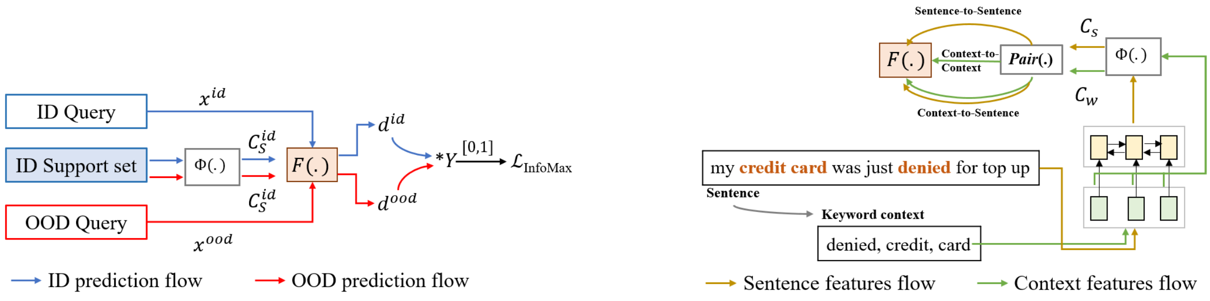
$$\mathcal{D}_{id} = \{(x_1, y_1), \ldots, (x_n, y_n)\} \sim \mathcal{P}_{T_i} \quad (1)$$

ID support set S^{id} and ID target query Q^{id} are drawn from $\mathcal{D}_{id}$, where S^{id} and Q^{id} are mutually exclusive $S^{id}, Q^{id} \in \mathcal{D}_{id}; S^{id} \cap Q^{id} = \emptyset$. In Figure 2, ID domain is exemplified by “Books” domain.

OOD data is drawn from out-of-scope or out-of-episode distribution $\mathcal{P}_{\mathcal{T}_j}, \mathcal{T}_j \in \mathcal{T}, \mathcal{T}_j \neq \mathcal{T}_i$ (later refer to as $\mathcal{P}_{ood}$). In the above example, OOD domain is the remaining domains in data set, apart from “Books” domain.

$$\mathcal{D}_{ood} = \{(x_1, y_1), \ldots, (x_n, y_n)\} \sim \mathcal{P}_{T_j} \quad (2)$$

For each training episode, the model is regularized based on distance metric $d = F(.)$ between prototype vector of ID support set $C^{id} = \Phi(S^{id})$ and target queries $\{Q^{id}, Q^{ood}\} \sim Q$. During meta-validation and meta-testing tasks, given novel ID support set S^{id} and novel target query Q, which is either drawn from P_{id} or P_{ood}, the score $d = F(C^{id}, Q)$ is then compared to some threshold $\tau > 0$ (Section 4.3). Target queries with scores above threshold are then classified as ID examples. The ones with scores below the threshold are classified as OOD.



(a)

(b)

Figure 3: Proposed ProtoInfoMax. (a) ProtoInfoMax with prototype vector based on sentence features C_S. The encoder architecture that projects raw inputs into vectors in metric space is omitted to simplify the illustration. Both ID and OOD queries use the same ID support set C_S^{id}. (b) A partial illustration of ProtoInfoMax++ with multiple prototype vectors (C_S, C_W), correspond to prototype vector based on sentence features and keyword context features respectively. Green boxes represent word embedding layer in encoder module of Prototypical Networks. Yellow boxes represent bidirectional GRU as sentence embedding layer.

3 ProtoInfoMax

We propose two models: **ProtoInfoMax** and **ProtoInfoMax++**, briefly illustrated in Figure 3. The main difference between the two models is their prototype generator, further discussed in Section 3.2. **ProtoInfoMax++** merges multiple prototype representations: (i) standard feature averaging prototype vector based on sentence features (referred to as C_S); and (ii) prototype vector based on keyword context features (referred to as C_w). We regularize both models with an InfoMax objective, discussed in Section 3.1.

3.1 InfoMax Objective

We adopt the recently proposed Mutual Information Maximization (InfoMax) training objective for deep learning (Belghazi et al., 2018; Hjelm et al., 2019) as contrastive view of data drawn from ID and OOD distribution $(\mathcal{P}_{id}, \mathcal{P}_{ood})$. The idea is simple, we want to maximize MI estimates for samples drawn from ID distribution $\mathcal{P}_{id}$, while penalizing OOD samples with lower MI estimates.

The InfoMax principle (Linsker, 1988; Bell and Sejnowski, 1995), have been recently adapted for the use with DNNs in diverse applications. For example, the InfoMax objective is used to learn useful representations by maximizing relevant information between local and global features of image data (Hjelm et al., 2019), to learn speaker representations (Ravanelli and Bengio, 2019), and to learn robust question answering system (Yeh and Chen, 2019).

In this study, a multi-objective for simultaneously classifying ID sentences and detecting OOD sentences can also be formulated as a contrastive learning framework via an InfoMax objective. That is, the model is enforced to learn binary reject function $\mathcal{L}$ that partitions the input space $\mathcal{X}$ with respect to $\mathcal{P}_{\text{id}}$ and $\mathcal{P}_{\text{ood}}$. Incorporating a binary reject function for regularizing Prototypical Networks in the current OOD detection problem can simplify the overall training mechanism. Namely, it can be approximated by a simple BCE-loss implementation of InfoMax objective (Hjelm et al., 2019; Yeh and Chen, 2019).

InfoMax based on BCE-loss [2] for the current OOD detection problem is then formulated as the approximation of MI between ID prototype vectors and target queries $I(C^{id}, Q)$ via Jensen-Shannon divergence (JS).

$$I(C^{id}, Q) \geq \mathbb{E}_{\mathbb{P}}[\log F(C^{id}, x^{id})] + \mathbb{E}_{\mathbb{Q}}[\log(1 - F(C^{id}, x^{ood}))] \tag{3}$$

where $\mathbb{E}_{\mathbb{P}}$ and $\mathbb{E}_{\mathbb{Q}}$ denote the expectation over ID and OOD samples respectively; $x^{id} \in Q^{id}, x^{ood} \in Q^{ood}, \{Q^{id}, Q^{ood}\} \sim Q$.

Loss function based on the above InfoMax objective $\mathcal{L}_{\texttt{infomax}}$ is then defined as binary cross-entropy loss between ID and OOD prediction.

$$\mathcal{L}^{\text{BCE}}_{\texttt{infomax}}(F(C^{id}, Q)) = \frac{1}{|Q^{id}|} \sum_{x^{id} \in Q^{id}} \log F(C^{id}, x^{id}) + \frac{1}{|Q^{ood}|} \sum_{x^{ood} \in Q^{ood}} \log(1 - F(C^{id}, x^{ood})) \tag{4}$$

[2] For a theoretical justification on how binary cross-entropy (BCE) loss approximates Mutual Information (MI) between two random variables, including the alternatives, we refer reader to the prior works on investigating InfoMax objective for deep representation learning (Belghazi et al., 2018; Hjelm et al., 2019; Tschannen et al., 2020; Kong et al., 2020).

Figure 3a illustrates the proposed model with an InfoMax objective. Since the prediction outcomes mainly depend on the distance $d = F(.)$ between target queries and class prototypes, the actual label supervision of target queries is by default not included in the training. To further enrich the learned metric representation, especially in the current training tasks that contain multiple ID class representations, the label supervision of target query Y was preserved. This can be done by applying element-wise multiplication between distance metric $d = F(.)$ and one-hot-encoding of class labels $*Y$ before loss function, following strategy in previous study (Tan et al., 2019).

3.2 Prototype Generator $\Phi(.)$

For both proposed models, we use standard prototype generator $\Phi(.)$ based on feature averaging. Given encoded representations of per batch ID support set $S^{id}_{ebd} = Encode(S^{id}), S^{id}_{ebd} \in \mathbb{R}^{b \times k \times d}$, the prototype vector C is described as an averaged representation of those k-sentence features, $C = \frac{1}{k}\sum_{i=1}^{k} S^{id}_{ebd(i)}, C \in \mathbb{R}^{b \times d}$.

Sentence-based Features Given sentence features of ID support set S^{id}_{ebd}, ID class prototype C_S is defined as a mean vector of those sentence features: $C_S = \Phi(S^{id}_{ebd})$. The overall training flow is simple, illustrated as simultaneous ID and OOD prediction flow in Figure 3a.

Given prototype vector based on sentence features C_S and target queries Q drawn from $\mathcal{P}_{id}$ and $\mathcal{P}_{ood}$, the loss function for ProtoInfoMax is described as

$$\mathcal{L}_{\texttt{infomax}} = \mathcal{L}^{\text{BCE}}_{\texttt{infomax}}(F(C_S, Q)) \qquad (5)$$

Keyword-based Features In an extreme low resource setting where training examples are insufficient, the model may not be able to learn meaningful sentence representations. It is thus advantageous for incorporating auxiliary knowledge to better guide the model training. Here, we utilize keywords as auxiliary inputs [3]. for **ProtoInfoMax++** (Figure 3b).

Intuitively, relevant sentences drawn from the same domain or intent distribution may share relevant context or keywords. Therefore, extracted keywords from a sentence can be viewed as local context representation of the corresponding sentence. The more keywords that two sentences share in common, the more similar or related the two sentences are. Likewise, keywords that are close together with respect to their angular distance or orientation in embedding space are expected to carry similar semantic meaning. Sentences containing those similar subset of keywords is considered to carry similar or related semantics.

Given ID support set S^{id}, ID target Q^{id}, and OOD target Q^{ood} and their corresponding extracted keywords as model's raw inputs, C_w is defined as a mean vector representation of sentence's keywords $\{w_1, w_2, \ldots, w_n\}$ weighted by their corresponding Idf value: $C_w = \frac{1}{i}\sum_{i=1}^{n}(w_i * Idf_i), C_w \in \mathbb{R}^{b \times d}$. For ID support set containing k-sentences, C_w is averaged over n-keywords and k-sentence features: $C_w = \frac{1}{k}\frac{1}{n}\sum_{j=1}^{k}\sum_{i=1}^{n}(w_i * Idf_i)^j$.

We use multiple perspectives of pairwise similarity to compute distance metric $d = F()$.

- sentence-to-sentence similarity $F(C_S, Q)$

  This similarity function is used as the default measure for Prototypical Networks in this study. Here, C_S denotes prototype vector of ID support set (referred to as C^{id} in another section) and Q is sentence embedding projection of target queries.

- context-to-context similarity $F(C_w^{\text{sup}}, C_w^{\text{Q}})$

  We want to maximize MI between prototype representation of keywords in support set and target queries. C_w^{sup} is prototype vector computed from keyword contexts in support set, while C_w^{Q} is computed from keywords in target queries.

- context-to-sentence similarity $F(C_w^{\text{sup}*}, C_w^{\text{Q}*})$ We want to maximize MI between sentences that share relevant context representations. Since the two features come from different embedding space (sentence vs. word embedding), context-based prototype vectors are first projected into sentence embedding space by an element-wise matrix multiplication with features from support set and target query: $C_w^{\text{sup}*} = C_w^{\text{sup}} * C_S$; $C_w^{\text{Q}*} = C_w^{\text{Q}} * Q$.

The total loss for ProtoInfoMax++ is then described as cumulative losses based on the above distance measures $F(.)$, given target samples from

[3] We employ Tf-Idf feature extractor (Sparck Jones, 1972; Salton and Buckley, 1988) to automatically extract keywords from sentences. Each sentence is entitled to maximum 10 keywords

both ID and OOD distribution.

$$\begin{aligned}\mathcal{L}_{\texttt{infomax++}} = {} & \mathcal{L}^{\text{BCE}}_{\texttt{infomax}}(F(C_S, Q)) + \\ & \mathcal{L}^{\text{BCE}}_{\texttt{infomax}}(F(C_w^{\text{sup}}, C_w^{\text{Q}})) + \\ & \mathcal{L}^{\text{BCE}}_{\texttt{infomax}}(F(C_w^{\text{sup}*}, C_w^{\text{Q}*}))\end{aligned} \tag{6}$$

Note that during meta-validation and meta-testing, ProtoInfoMax++ is only given raw sentences (no keywords) as source inputs for the model.

4 Experiments

Code and datasets in this study will be made available publicly.

4.1 Dataset

Amazon Product Reviews For structuring Amazon review data into meta-tasks, we followed strategy from previous works on few-shot classification (Yu et al., 2018; Tan et al., 2019).

AI Conversational Data For constructing intent classification meta-tasks, we use two data sets that share contexts: AI Conversational Data (Chatterjee and Sengupta, 2020); and (CLINC150) (Larson et al., 2019; Casanueva et al., 2020) [4]. The preprocessed data contains disjoint classes across tasks, introducing a more challenging ID and OOD prediction task for Prototypical Networks in this study. In meta-training, each task (domain) is composed of 10 intent category labels ($N = 10$) [5]. Meta-validation and meta-testing are constructed from CLINC150. We use $N = 1$ and $N = 2$ set up to inspect model performance on one ID class and multiple ID classes prediction respectively.

4.2 Model and Hyper-parameters

Baselines We use two baselines: 1) **Proto-Net** (Snell et al., 2017; Yu et al., 2018), a native Prototypical Network with entropy-based loss function; 2) **O-Proto** (Tan et al., 2019), state-of-the-art approach for simultaneously learning ID classification and OOD detection. We constructed all models, including our proposed approaches, based on Bidirectional GRU and Attention network as encoder's main architecture.

Hyper-parameters For all models, we initialized word representation from pretrained fastText [6]. We updated fastText representation by further training it on benchmark data set separately from model training. This strategy is necessary since the model is required to capture essential context (word) representations from limited number ID training data and unknown OOD context. For all models, we use one layer Bidirectional-GRU (output dim=200); and one layer Attention Network that is initialized based on $r = 5$ context query representations sampled from uniform distribution $\mathcal{U}[.1, .1]$. Cosine similarity is used as distance metric $d = F(.) \sim cos(\theta)$ to compute (1-angular distance) between target queries and class prototypes (ID support set). All models were trained up to 60 epochs. Experiments were done in two computing nodes of HPC cluster with varying specifications [7].

4.3 Evaluation Metrics

ID and OOD Detection Error We use **(i) Equal Error Rate (EER)** for measuring error in predicting OOD; **(ii) Class Error Rate (CER$^{\text{id}}$)** for measuring error in predicting ID examples; and **(iii) CER$^{\text{all}}$** for measuring error in ID prediction given both ID and OOD subsets, following the previous work on OOD detection (Ryu et al., 2018; Tan et al., 2019). Except for CER$^{\text{id}}$, the EER and CER$^{\text{all}}$ metrics are calculated based on heuristically selected threshold value τ – prediction score at which False Acceptance Rate (FAR) and False Rejection Rate (FRR) are equal.

$$\text{FAR} = \frac{\text{FN}}{\text{\# OOD examples}} \tag{7}$$

$$\text{FRR} = \frac{\text{FP}}{\text{\# ID examples}} \tag{8}$$

$$\text{EER} = \frac{1 - (\text{TP} + \text{TN})}{\text{\# Examples}} \tag{9}$$

$$\text{CER}^{\text{id}} = \frac{\text{TP}^{\text{id}}}{\text{\# ID examples}} \tag{10}$$

$$\text{CER}^{\text{all}} = \frac{\text{TP}}{\text{\# ID examples}} \tag{11}$$

Since OOD class labels are unknown to the trained model, TN is calculated based on the actual OOD examples at which prediction score is below threshold τ: the model predicts OOD samples as OOD. TP is based on the actual ID examples at

[4] We use different benchmarks for intent classification task because the footage of preprocessed data from previous work (Tan et al., 2019) is unavailable publicly.

[5] We follow the structure exemplified in sentiment classification tasks. More explanation on how we construct data is provided in Appendix.

[6] https://fasttext.cc/

[7] provided in Appendix

Method	Sentiment Cls ($N = 2$)						Intent Cls ($N = 1$)		Intent Cls ($N = 2$)		
	EER		CER$^{\text{id}}$		CER$^{\text{all}}$		EER		EER	CER$^{\text{id}}$	CER$^{\text{all}}$
	$\mathcal{T}^{\text{val}}$	$\mathcal{T}^{\text{test}}$	$\mathcal{T}^{\text{val}}$	$\mathcal{T}^{\text{test}}$	$\mathcal{T}^{\text{val}}$	$\mathcal{T}^{\text{test}}$	$\mathcal{T}^{\text{val}}$	$\mathcal{T}^{\text{test}}$			
<u>Baselines</u>											
Proto-Net ($\mathcal{L}_{id}$)	0.398	0.387	**0.266**	**0.285**	**0.445**	0.536	0.456	0.420	0.525	0.316	0.634
O-Proto ($\mathcal{L}_{id}^{ent} + \mathcal{L}_{id}^{hinge} + \mathcal{L}_{ood}^{hinge}$)	0.348	0.375	0.411	0.409	0.631	0.643	0.404	0.390	0.482	0.373	0.683
<u>This study</u>											
ProtoInfoMax	0.373	0.278	0.351	0.365	0.592	0.521	0.398	**0.368**	0.398	0.256	0.549
ProtoInfoMax++	**0.335**	**0.245**	0.301	0.315	0.532	**0.469**	**0.369**	0.382	**0.388**	**0.225**	**0.519**

Table 1: Performance for $K = 100$ [8]. The lower the better. Scores are based on top-3 the highest accuracy score for ID prediction (1-CER$^{\text{id}}$) across meta-validation and meta-testing episodes (epochs). For one class prediction of intent classification ($N = 1$), EER and (1-CER$^{\text{all}}$) are equal, and CER$^{\text{id}} = 1.0$ because the number of ID class within the subset is 1. Evaluation for both $N = 1$ and $N = 2$ intent classification use the same model trained on $N = 10, K = 100$.

which the prediction score is above threshold: ID samples are predicted as ID. FN: the model predicts OOD as ID. FP: ID samples are predicted as OOD. TP$^{\text{id}}$ is the number of correctly classified ID examples, excluding OOD samples.

Reliability Diagram Reliability diagram (Niculescu-Mizil and Caruana, 2005; Guo et al., 2017) depicts gaps between accuracy and model confidence. The larger the gap, the less calibrated the model is. That is, either the model is being under-confident or over-confident on estimating the winning predicted class labels. We use Expected **Calibration Error (ECE)** (Naeini et al., 2015; Guo et al., 2017) to summarize the difference in expectation between confidence and accuracy (gaps) across all bins. We use distance metric $d = F(.)$ as model confidence measure, following relevant work on distance-based prototypical learning (Xing et al., 2020).

5 Results and Analysis

We demonstrate the effectiveness of our proposed methods (**ProtoInfoMax** and **ProtoInfoMax++**) on two benchmarks for OOD detection (Table 1). Notice that native Prototypical Networks (**Proto-Net**) performs reasonably well, specifically for ID prediction (see scores based on **CER**^{id} and **CER**^{all}). However, this result can occur to models that always output prediction with a high score (e.g. high similarity score based on d in the current work), regardless whether the prediction is correct. The insight into this over-confidence behaviour is provided in Section 5.3 and 5.4.

5.1 Performance in different K-shot

Our **ProtoInfoMax** and **ProtoInfoMax++** also show a considerably consistent performance on meta-testing tasks under different K-shot values (Table 2 and 3), outperforming O-Proto.

Model	**EER**	**CER$^{\text{id}}$**	**CER$^{\text{all}}$**
<u>K=1</u>			
O-Proto	0.381	0.450	0.676
ProtoInfoMax	**0.313**	0.432	0.616
ProtoInfoMax++	0.335	**0.430**	**0.615**
<u>K=10</u>			
O-Proto	0.311	0.425	0.606
ProtoInfoMax	0.286	0.419	0.578
ProtoInfoMax++	**0.254**	**0.375**	**0.537**
<u>K=100</u>			
O-Proto	0.375	0.409	0.643
ProtoInfoMax	0.278	0.365	0.521
ProtoInfoMax++	**0.245**	**0.315**	**0.469**

Table 2: Performance under different K-shot values in sentiment classification ($N = 2$). Scores are based on the highest accuracy ($1 - \text{CER}^{\text{id}}$) on $\mathcal{T}^{\text{test}}$.

Model	**EER**	**CER$^{\text{id}}$**	**CER$^{\text{all}}$**
<u>K=1</u>			
O-Proto	0.515	0.391	0.698
ProtoInfoMax	0.480	0.397	0.674
ProtoInfoMax++	**0.452**	**0.384**	**0.638**
<u>K=10</u>			
O-Proto	0.493	0.402	0.694
ProtoInfoMax	0.451	0.400	0.686
ProtoInfoMax++	**0.401**	**0.329**	**0.598**
<u>K=100</u>			
O-Proto	0.482	0.373	0.683
ProtoInfoMax	0.398	0.256	0.549
ProtoInfoMax++	**0.388**	**0.225**	**0.519**

Table 3: Performance under different K-shot values in intent classification ($N = 2$).

[8] Notice that our results (O-Proto performance) is different with those reported in (Tan et al., 2019). This might due to: different implementation framework (our PyTorch vs. their native Tensorflow implementation), different hyper-parameters (we use less training epoch due to our computational constraints), or different computing resources (GPU/CPU capacity used to train the models).

5.2 On Threshold Score, FAR, and FRR

For all Prototypical Networks in this study, the performance on ID classification and OOD detection mainly depend on the learned distance metric $d = F(.) \sim cos(\theta)$ to compute prediction outcomes and the heuristically selected threshold τ at which FAR=FRR. In the current work, lower prediction and threshold score indicate prediction uncertainty. When model is being less certain, novel ID and OOD samples are likely to receive low score with respect to their distance to ID class prototypes. When model is over-confident, OOD samples are likely to be predicted with high score.

Figure 4 shows the selected threshold score across models in intent classification task. It can be observed that **O-Proto** has a tendency to be over-confident, suggested by a considerably high threshold score ($\tau = 0.97$ at epoch 0 and $\tau = 0.93$ at epoch 40). Both **ProtoInfoMax** and **ProtoInfoMax++** are being less confident after several epochs, yielding lower thresholds ($\tau = 0.87$ and $\tau = 0.74$ respectively). Specific to **ProtoInfoMax++**, the model converges faster in early episode (epoch= 0), yielding lower threshold score ($\tau = 79$). Notice that the gaps between FAR and FRR for both **ProtoInfoMax** and **ProtoInfoMax++** at epoch = 40 are smaller. This indicates that both models underestimate ID and OOD samples, assigning them with low similarity scores ($d \leq 0.0$) with respect to ID class prototypes [9].

5.3 Reliability in ID Prediction

Figure 5 compares the reliability of models in sentiment classification [11]. In general, all models in this study tend to be over-confident, suggesting that future work focusing on directly tackling and investigating such problem is essential [12].

Compared to the baselines, our proposed **ProtoInfoMax** and **ProtoInfoMax++** are shown to be less prone to typical over-confidence problem with respect to smaller gaps between their confidence score and the prediction accuracy. **Proto-Net**, however, suffers greatly from such over-confidence problem. It can be observed that Proto-Net assigns high similarity scores ($d \geq 0.9$) for all prediction points (see accuracy is lower than confidence score in Figure 5a).

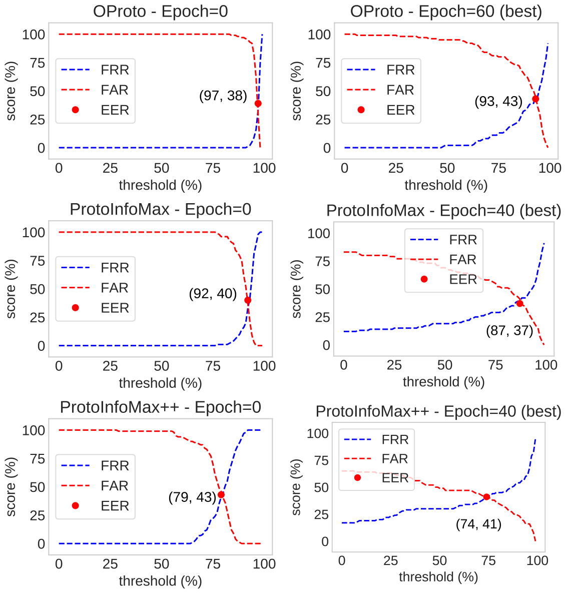
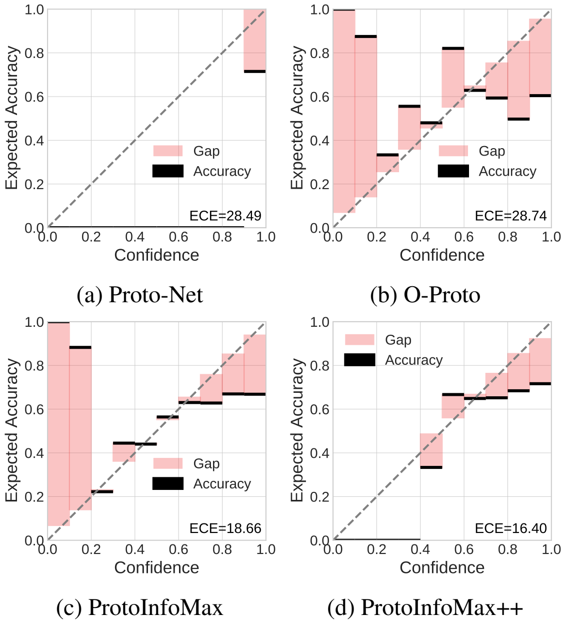


Figure 4: EER, FRR, FAR in intent classification meta-testing. Score (%) denotes proportion of samples that are either rejected (ID) or accepted (OOD) based on the selected threshold. To plot the above FAR and FRR, 200 prediction points correspond to ID and OOD test samples were drawn randomly from 6 domains in $N = 2$ meta-testing episodes.



(a) Proto-Net

(b) O-Proto

(c) ProtoInfoMax

(d) ProtoInfoMax++

Figure 5: Reliability Diagram for ID prediction. Confidence scores were taken from $\mathcal{T}^{test}$ in sentiment classification ($N = 2, K = 100$) based on the highest $1 - \mathrm{CER}^{\mathrm{id}}$ [10].

Our methods achieve the lowest ECE scores (ECE ProtoInfoMax = 18.66 and

[9] We do not normalize $d, d \in [-1, 1]$ here to inspect whether the model penalizes OOD samples severely with similarity score $d \leq 0.0$.

[11] Since OOD labels are unknown during training, this evaluation only include the prediction outcomes from ID target queries as test samples.

[12] In current work, we abuse terminology of "confidence score" to refer to the output of distance metric d, following relevant work on distance-based prototypical learning (Xing et al., 2020).

ECE ProtoInfoMax++$=$ 16.40), suggesting a better reliability with respect to smaller gaps between model's confidence score and prediction accuracy. **O-Proto** (Figure 5c) and **ProtoInfoMax** (Figure 5b) are also shown to be both under-confident and over-confident. The models under-estimate correct ID target queries (large gaps with high accuracy for $d \in (0.0, 0.2)$) and over-estimate incorrect ID examples (large gaps with lower accuracy for $d \in (0.7, 1.0)$).

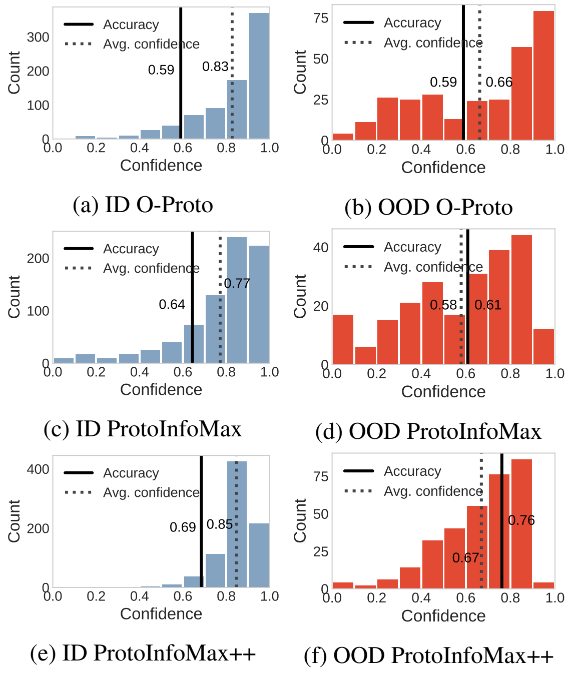
5.4 Reliability in OOD Prediction

The reliability based on confidence histogram for ID and OOD prediction is provided in Figure 6 [13]. In general, all models over-estimate their prediction given ID target queries (see that the average confidence is higher than accuracy in Figure 6a, 6c, 6e). However, compared to **O-Proto**, our **ProtoInfoMax** and **ProtoInfoMax++** have a higher accuracy in ID classification task given their reasonably high confidence. Notice that for ID prediction task, **ProtoInfoMax++** is more confident than the other two models ($d \in (0.4, 1.0)$ in Figure 6e).

For OOD detection [14], our **ProtoInfoMax** and **ProtoInfoMax++** are shown to be well calibrated than **O-Proto**. See that the average confidence scores of both models are lower than their prediction accuracy (ProtoInfoMax avg. confidence: 0.58 in Figure 6d and ProtoInfoMax++: 0.67 in Figure 6f). In contrast, the average of confidence score of **O-Proto** is higher than its prediction accuracy (Avg. Confidence $= 0.66$, Accuracy$= 0.59$ in Figure 6b), indicating the model prediction with an over-confidence issue.

6 Conclusion

In this study, we aim at effectively training Prototypical Networks to simultaneously learn ID classification and OOD detection tasks. We propose Prototypical Networks with Mutual Information Maximization objective, named **ProtoInfoMax**. Experiments on two recent benchmarks for OOD recognition demonstrate the effectiveness and reliability of the proposed method. Although we do not specifically tackle over-confidence problem of Neural Networks by calibrating models during training and evaluation stage, we observe that the proposed ProtoInfoMax and ProtoInfoMax++ are less prone to typical over-confidence problem in the current OOD detection domain than the existing approach. Overall, we improve performance of existing approach up to 20% for OOD detection in low resource text classification.



(a) ID O-Proto (b) OOD O-Proto

(c) ID ProtoInfoMax (d) OOD ProtoInfoMax

(e) ID ProtoInfoMax++ (f) OOD ProtoInfoMax++

Figure 6: Confidence histogram for ID and OOD prediction. Confidence scores were taken from $\mathcal{T}^{test}$ in sentiment classification ($N = 2, K = 100$). Note that the value of accuracy and average confidence here are not as precise as 1-CER$^{\text{id}}$ score, since they were averaged across normalized bin scores.

Our work primarily focuses on sentiment and intent classification tasks. However, our proposed approach is applicable for a realistic setting of OOD detection in real world NLP applications. It is also interesting future direction for many practical applications to further investigate the over-confidence issue of Neural Networks, which occurs in the current work and affects the reliability of model prediction in inference stage.

References

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. 2018. Mutual information neural estimation. In *International Conference on Machine Learning*, pages 531–540.

[12] Reliability diagram and confidence histogram on intent classification tasks is provided in Appendix.

[13] Since OOD labels are unavailable during training, the reliability diagram is not applicable for evaluating OOD prediction.

[14] Here, we view the task as one class OOD prediction where test samples contain OOD target queries only. Values below threshold τ are classified as OOD. Values above threshold are classified as ID.

Anthony J. Bell and Terrence J. Sejnowski. 1995. An Information-Maximization Approach to Blind Separation and Blind Deconvolution. *Neural Computation*, 7(6):1129–1159.

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

Ajay Chatterjee and Shubhashis Sengupta. 2020. Intent mining from past conversations for conversational agent. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4140–4152, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Geli Fei and Bing Liu. 2016. Breaking the closed world assumption in text classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 506–514, San Diego, California. Association for Computational Linguistics.

Geli Fei, Shuai Wang, and Bing Liu. 2016. Learning cumulatively to become more knowledgeable. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1565–1574, New York, NY, USA. Association for Computing Machinery.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1321–1330. JMLR.org.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. 2019. Deep anomaly detection with outlier exposure. *Proceedings of the International Conference on Learning Representations*.

Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Philip Bachman, Adam Trischler, and Yoshua Bengio. 2019. Learning deep representations by mutual information estimation and maximization. In *ICLR 2019*. ICLR.

Joo-Kyung Kim and Young-Bum Kim. 2018a. Joint learning of domain classification and out-of-domain detection with dynamic class weighting for satisficing false acceptance rates. *Proc. Interspeech 2018*, pages 556–560.

Joo-Kyung Kim and Young-Bum Kim. 2018b. Joint learning of domain classification and out-of-domain detection with dynamic class weighting for satisficing false acceptance rates. In *Proc. Interspeech 2018*, pages 556–560.

Lingpeng Kong, Cyprien de Masson d'Autume, Lei Yu, Wang Ling, Zihang Dai, and Dani Yogatama. 2020. A mutual information maximization perspective of language representation learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6405–6416, Red Hook, NY, USA. Curran Associates Inc.

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

Mike Lewis and Angela Fan. 2019. Generative question answering: Learning to answer the whole question. In *International Conference on Learning Representations*.

Shiyu Liang, Yixuan Li, and R Srikant. 2018. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*.

R. Linsker. 1988. Self-organization in a perceptual network. *Computer*, 21(3):105–117.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

Alexandru Niculescu-Mizil and Rich Caruana. 2005. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632.

Mirco Ravanelli and Yoshua Bengio. 2019. Learning speaker representations with mutual information. *Proc. Interspeech 2019*, pages 1153–1157.

Seonghan Ryu, Seokhwan Kim, Junhwi Choi, Hwanjo Yu, and Gary Geunbae Lee. 2017. Neural sentence embedding using only in-domain sentences for out-of-domain sentence detection in dialog systems. *Pattern Recognition Letters*, 88:26 – 32.

Seonghan Ryu, Sangjun Koo, Hwanjo Yu, and Gary Geunbae Lee. 2018. Out-of-domain detection based on generative adversarial network. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 714–718, Brussels, Belgium. Association for Computational Linguistics.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523.

W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult. 2013. Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1757–1772.

Alireza Shafaei, Mark Schmidt, and James Little. 2019. A Less Biased Evaluation of Out-of-distribution Sample Detectors. In *British Machine Vision Conference (BMVC)*.

Lei Shu, Hu Xu, and Bing Liu. 2017. DOC: Deep open classification of text documents. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2911–2916, Copenhagen, Denmark. Association for Computational Linguistics.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

Ming Tan, Yang Yu, Haoyu Wang, Dakuo Wang, Saloni Potdar, Shiyu Chang, and Mo Yu. 2019. Out-of-domain detection for low-resource text classification tasks. 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 3566–3572, Hong Kong, China. Association for Computational Linguistics.

Michael Tobias Tschannen, Josip Djolonga, Paul Kishan Rubenstein, Sylvain Gelly, and Mario Lučić. 2020. On mutual information maximization for representation learning. In *International Conference on Learning Representations*. Michael Tschannen and Josip Djolonga contributed equally.

Chen Xing, Sercan Arik, Zizhao Zhang, and Tomas Pfister. 2020. Distance-based learning from errors for confidence calibration. In *International Conference on Learning Representations*.

Yi-Ting Yeh and Yun-Nung Chen. 2019. QAInfomax: Learning robust question answering system by mutual information maximization. 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 3370–3375, Hong Kong, China. Association for Computational Linguistics.

Mo Yu, Xiaoxiao Guo, Jinfeng Yi, Shiyu Chang, Saloni Potdar, Yu Cheng, Gerald Tesauro, Haoyu Wang, and Bowen Zhou. 2018. Diverse few-shot text classification with multiple metrics. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1206–1215, New Orleans, Louisiana. Association for Computational Linguistics.

Y. Zheng, G. Chen, and M. Huang. 2020. Out-of-domain detection for natural language understanding in dialog systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1198–1209.

Yinhe Zheng, Guanyi Chen, and Minlie Huang. 2020. Out-of-domain detection for natural language understanding in dialog systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1198–1209.