On Multimodal Few-shot Learning for Visually-rich Document Entity Retrieval

Anonymous ACL submission

Abstract

Visually-rich document entity retrieval (VDER), which extracts key information (e.g. date, address, name) from document images (e.g., invoice, receipt) has become an increasingly important topic for NLP in the industrial settings. As many of these document images come from document types that are highly specified to their industry, annotating these documents usually requires extensive amount of training and is often costly. The fact that new document types come out at a constant pace and that each of them have a unique set of entity types leave us a challenging setting where we have a large amount of documents with unseen entity types that occur only a couple of time. Such a setting requires models to have the capability of learning entities in a few-shot manner, while recent works in the field can only handle few-shot learning in the document level. We propose an N-way K-shot setting for VDER that operates on the *entity level* and a new dataset to tackle such a problem. We formulate the problem as a meta learning one and propose a few new algorithms that helps the model to distinguish between in-task-distribution (ITD) entities while being aware of out-of-task-distribution (OTD) ones. To the best of our knowledge, our work is the first systematic study on the N-way K-shot entity-level setting for VDER.

1 Introduction

Visually-rich document understanding (VrDU) aims to analyze scanned documents composed of structured and organized information. As a sub-problem of VrDU, the goal of visually-rich document entity retrieval (VDER) is to extract key information (e.g., date, shipping address, signatures) from the document images such as invoices and receipts with complementary multimodal information. (Xu et al., 2020a; Garncarek et al., 2021; Lee et al., 2022). One unique challenge when modeling the document entity retrieval problem is the fact that their *entity space* (i.e., the set of entity categories that we are going to extract) changes from one document type to the other. However, as the cost of labeling is expensive, we are left with a very large amount of documents with little amount of annotations to each of the new entity types (i.e., *few shots*). Such a scenario makes it difficult to transfer knowledge learned from different documents or entity types without techniques that can deal with few-shot entities. To deal with such scenario in real-world VrDU systems, *few-shot visually-rich document entity retrieval* (FVDER) has become a crucial research topic.

Despite the importance of the FVDER, there has been very limited amount of prior works in this area. Most recent efforts have employed pre-trained language model (Wang and Shang, 2022) or prompt mechanism (Wang et al., 2022) to obtain transferable knowledge from the source domain and apply it to the target domain, where a small number of document images are provided for fine-tuning. However, the settings of few-shot learning, which is often borrowed from other domains such as image classification, may not fit well into the problem of document entity extraction. For example, in these prior works, models are operated in the granularity of the document level rather than entity level. In practice, entity occurrence varies dramatically from one document to the other, and the few-shot setting operated on document level might end up with a lot of instances of a particular entity type, which goes against the purpose of few-shot learning. Another notable limitation is that some of these prior works are not capable of working on unseen entities. Additionally, in these works, there is no way to quantify the size of the entity space and the occurrence of each entity type. We summarized the differences between prior works and ours in Table 1.

In this work, we formalize a novel task setting for few-shot visually-rich document entity retrieval

Method	Instance Granularity	Unseen Entities?	Entity Space Size	Entity Occurrence
(Wang and Shang, 2022)	document level	yes	unspecified	unspecified
(Wang et al., 2022)	document level	no	unspecified	unspecified
Ours	entity level	yes	N-way	K-shot

Table 1: Comparison on Task Formulated and Application Scenarios.

from practical needs, which is operated on entity level with unseen entities and on a N-way K-shot specification. We aim at building a coherent setting for FVDER tasks, which pays attention to extracting *novel* and *rarely-present* entity types from documents. Upon the task setting, we also create a systematic way for learning the knowledge that *fast adapts* to entirely *novel* information or entity types. The key properties of the proposed setting are two folds, following the practical requirements: **1)**, We allow target entities to be scarcely scattered over documents. The few-labelled documents should be selected in a way such that they cooperatively contains certain number of entity occurrences per information type. **2)**, We split entity types into two group such that one for training and the other is used as novel entity types.

To tackle the proposed task setting, we propose a meta-learning based framework. The key idea is to employ the learning-to-learn mechanism for two objectives: 1) making the learning experience to be transferable from the base entity types to the novel ones; and 2) quicker adaptation on novel entity types by reducing the domain gap and task gap between the pre-trained model and our novel tasks through meta-learning. Comparing with the general FSL (Finn et al., 2017; Snell et al., 2017; Chen et al., 2021) and existing FVDER settings (Wang and Shang, 2022; Wang et al., 2022), our unique setting brings new challenges. Specifically, the existence of noisy out-of-task information, as part of the contextual information for in-task information, cannot be shared by different documents and tasks. Thus we also propose several techniques to improve the existing meta-learning approaches for this new task.

Our contributions are summarized as follows:

- We propose a novel entity-level few-shot visually-rich document entity retrieval (FVDER) formulation, where the number of labelled entity occurrences for each entity type is limited. To the best of our knowledge, this is the first work in VDER focusing on the entity retrieval for rare and novel entities.
- We present a new dataset for the meta-learning on FVDER, namely FewVEX, consisting of hundreds of testing FVDER tasks with novel entity types and thousands of training tasks with a held-out set of entity types.
- We propose a few algorithms under meta learning that work for FVDER. We address the specific challenges in our FVDER task and propose strategies to improve several popular gradient-based and metric-based meta-learning baselines.

2 Problem Formulation

We present the novel entity-level few-shot visually-rich document entity retrieval (FVDER) task. Given a document image that consists of structured contents (e.g., textual, visual and layout contents), the goal is to tackle the localization and classification of rare and novel entities from the given image. We formalize such a task below.

2.1 Entity-level N-way K-shot Formulation

In VDER tasks, a document image is often processed through Optical Character Recognition (OCR) (Chaudhuri et al., 2017) to form a sequence of tokens $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_L]$, where L is the sequence length and each token $\mathbf{x}_l$ is composed of multiple modalities $\mathbf{x}_l = \{\mathbf{x}_l^{(v)}, \mathbf{x}_l^{(p)}, \mathbf{x}_l^{(b)}, ...\}$ such as the token id (v), the 1d position (p) of the token in the sequence, the bounding box (b) representing the token's relative 2d position, scale in the image, and so on. The goal is to predict $Y = [y_1, y_2, \ldots, y_L]$, which assigns each token $\mathbf{x}_l$ a label y_l to indicate either the token is one of entities in a set of predefined entity types or does not belong to any entity (denoted as `O` class).

We propose the entity-level few-shot VDER that focuses on the real-world scenario when some types of entities *rarely occur* in documents. Here, an *entity occurrence* is defined as a contiguous subsequence of OCR-parsed tokens with the same entity type as labels. Formally, an entity-level N-way K-shot FVDER task $\mathcal{T} = \{S, Q, \mathcal{E}\}$ consists of a train (*support*) set S containing M_s documents, a test (*query*) set Q containing M_q documents, and

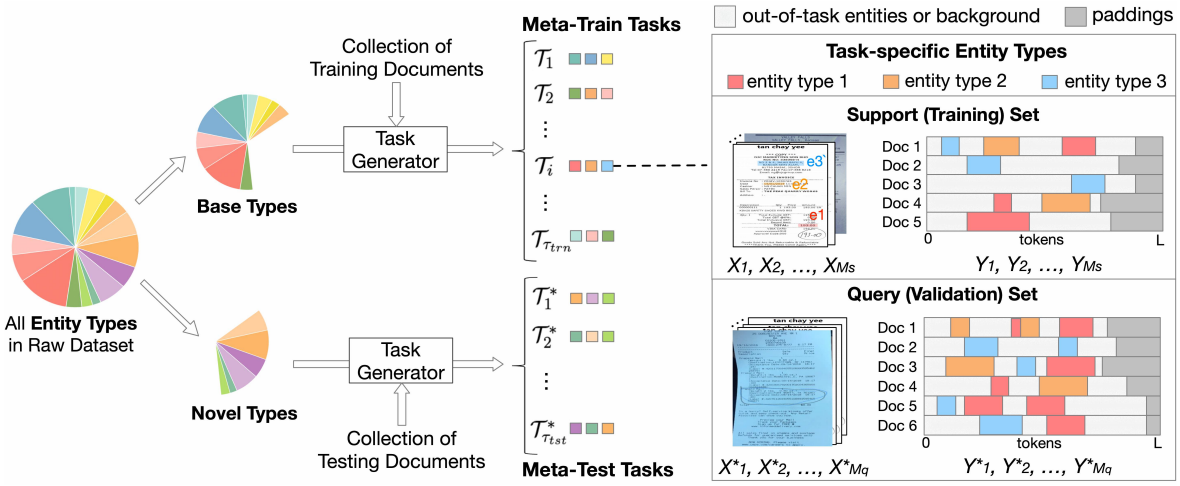


Figure 1: Proposed task setting and problem formulation. Here, $N = 3$ ways, $K = 2$ shots, and $\rho = 2$. Different colors represent different entity types. The pie charts on the left indicates that the target classes in testing tasks are not seen in training tasks. On the right, we show the inputs and labels of an example 3-way 2-shot task.

a class set $\mathcal{E}$ containing N target entity types

$$\begin{aligned} S &= \{(X_1, Y_1), \ldots, (X_{M_s}, Y_{M_s})\} \\ Q &= \{X_1^*, X_2^*, \ldots, X_{M_q}^*\} \\ \mathcal{E} &= \{e_1, e_2, \ldots, e_N\}, \end{aligned} \tag{1}$$

where $X_j = [\mathbf{x}_{j1}, \mathbf{x}_{j2}, ..., \mathbf{x}_{ijl}]$ is the sequence of multimodal token features of document j, $Y_j = [y_{j1}, y_{j2}, ..., y_{ijl}]$ is the sequence of token labels of document j, and e_c denotes the entity type c of $\mathcal{T}$.

The "N-way" means that S and Q do not contain any other entity types unseen in $\mathcal{E}$. Those out-of-task entity types $e' \notin \mathcal{E}$, although exist in these documents, are treated as the background O class.

The "K-shot" specification means that, among the M_s documents in S, the total number of *occurrences* of each entity type is restricted (i.e., entity-level few shots). Considering the condition that entity occurrence varies dramatically from one document to the other, it is very uncommon that a document contains every entity type in $\mathcal{E}$; it is also not guaranteed a entity type occurs only one time in an document. It is not the fact that different entity types strictly have the same occurrences among a few document. Thus, we adopt a *soft* K-shot setting. For each entity type in $\mathcal{E}$, the cumulative number of times it occurs among the M_s support documents should be in the range between $K \sim \rho K$, where $\rho > 1$ is the softening hyperparameter.

The goal of task $\mathcal{T}$ is to learn a *task-specific* model for the class distribution over $\mathcal{E}$ based on the few labeled entity occurrences in S, in order to achieve high performance on Q.

2.2 Meta-learning Formulation

Based on the above formulation for a single task, we can further formulate our problem under the meta-learning setting (Chen et al., 2021). We consider a task distribution $P(\mathcal{T})$ over FVDER tasks, associated with a large pool of entity types $\mathcal{C}$ corresponding to the domain of $P(\mathcal{T})$. For any task $\mathcal{T}_i = \{S_i, Q_i, \mathcal{E}_i\} \sim P(\mathcal{T})$, its target entity types come from the class pool $\mathcal{E}_i \subset \mathcal{C}$. With this assumption, our final goal turns out training a meta-learner such that any task $\mathcal{T}_i \sim P(\mathcal{T})$ can take advantage of it and then obtain a better task-specific model.

Following (Finn et al., 2017; Snell et al., 2017; Chen et al., 2021), a meta-learner for FVDER can be learned by exploiting the experiences on solving a set of meta-training tasks $\mathcal{D}_{meta}^{trn} = \{\mathcal{T}_1, \mathcal{T}_2 ... \mathcal{T}_{\tau_{trn}}\}$ over a set of base classes $\mathcal{C}_{base} \subset \mathcal{C}$, where each training task is from the base classes $\mathcal{E}_i \subset \mathcal{C}_{base}$. The experiences are given in the form of the ground truth labels of query sets. That is, the query sets of training tasks are treated as validation sets, $Q_i = \{(X_j^*, Y_j^*)\}_{j=1}^{M_{qi}}$ for $\forall \mathcal{T}_i \in \mathcal{D}_{meta}^{trn}$. Then, to evaluate the performance of the meta-learner θ on solving few-shot FVDER tasks that focus on *novel* entity types $\mathcal{C}_{novel} = C \setminus \mathcal{C}_{base}$, we will *individually* train a set of meta-testing tasks $\mathcal{D}_{meta}^{test} = \{\mathcal{T}_1^*, \mathcal{T}_2^* ..., \mathcal{T}_{\tau_{tst}}^*\}$, where each testing task $\mathcal{E}_i^* \subset \mathcal{C}_{novel}$. The query sets of meta-testing tasks are unlabelled, treated as the testing data.

3 Dataset

As far as we know, there is no existing benchmark specifically designed for the Entity-level N-way

Datasets	Meta Training (from $\mathcal{C}_{base}$)			Meta Testing (from $\mathcal{C}_{novel}$)			Range
	Domains	# Entity Types	# Tasks	Domains	# Entity Types	# Tasks	of N
FewVEX(S)	CORD	18	3000	CORD	5	128	[1, 5]
FewVEX(M)	CORD+FUNSD	20	3000	CORD+FUNSD	6	256	[1, 6]

Table 2: Statistics of two variants of FewVEX. From each dataset, we can test different N-way K-shot settings.

K-shot FVDER defined in Section 2. To support future research on this problem, we construct a new dataset, *FewVEX*.

We consider two source datasets: **FUNSD** (Jaume et al., 2019) consists of images of forms annotated by the bounding boxes of 3 types of entities; **CORD** (Park et al., 2019) consists of scanned receipts annotated by 6 superclasses which are divided into 30 fine-grained subclasses. We removed 7 entity types that occur in less than $\max_i(M_{si} + M_{qi})$ images [1]. Finally, we collect a set of 26 entity types and a set of 1199 unique document images ($\mathcal{D}_{orig}$) annotated by these entity types, which will be used to construct FewVEX (represented as $\mathcal{D}_{meta} = \{\mathcal{D}^{trn}_{meta}, \mathcal{D}^{tst}_{meta}\}$).

3.1 Entity Type Split

Suppose we have a pool of entity types $\mathcal{C}$. To ensure that testing tasks in $\mathcal{D}^{tst}_{meta}$ focus on novel classes that are unseen in $\mathcal{D}^{trn}_{meta}$ during meta-training, we should split $\mathcal{C}$ into two separate sets $\mathcal{C} = \mathcal{C}_{base} \cup \mathcal{C}_{novel}, \mathcal{C}_{base} \cap \mathcal{C}_{novel} = \emptyset$ such that $\mathcal{C}_{base}$ is used for meta-training and $\mathcal{C}_{novel}$ for meta-testing.

Proposed Datasets. Based on how we conduct the split, we construct two variants of FewVEX: **FewVEX(S)** focuses on single-domain receipt understanding, where $\mathcal{C}_{base}$ and $\mathcal{C}_{novel}$ are split from the 23 entity types in CORD. **FewVEX(M)** focuses on a combination of receipt and form domains, where $\mathcal{C}_{base}$ consists of 18 classes from CORD and 2 from FUNSD, while $\mathcal{C}_{novel}$ contains the other 5 classes in CORD and 1 in FUNSD.

3.2 Single Task Generation

Each individual entity-level N-way K-shot FVDER task $\mathcal{T} = \{S, Q, \mathcal{E}\}$ in either $\mathcal{D}^{trn}_{meta}$ or $\mathcal{D}^{tst}_{meta}$ can be generated through the following steps. **1) Class sampling.** The task's target classes $\mathcal{E}$ are generated by randomly sampling N entity types from either $\mathcal{C}_{base}$ (for the training task) or $\mathcal{C}_{novel}$ (for the testing task). **2) Document sampling.** Given the N target classes, we then collect document images that satisfies the N-way, soft K-shot entity occurrences (as defined in Section 2.1). To promote sampling efficiency, we only look at a subset of original documents $\mathcal{D}^{\mathcal{E}}_{orig} = \{\mathcal{D}^{e}_{orig} | \forall e \in \mathcal{E}\}$, where $\mathcal{D}^{e}_{orig} = \{(X, Y) | \forall (X, Y) \in \mathcal{D}_{orig} \text{ if } e \in Y\}$ is a dataset storing all the candidate documents that contain at least one entity of type e. $\mathcal{D}^{e}_{orig}$ can be generated in advance. Then, we design an Algorithm 1 (see Appendix B) to sample $(M_s + M_q)$ unique documents from $\mathcal{D}^{\mathcal{E}}_{orig}$ such that the first M_s documents have $K \sim \rho K$ shots per entity type (as S) and the remaining M_q ones have $K_q \sim \rho K_q$ shots per entity type (as Q). **3) Annotation Conversion.** A task only focuses on its specific N rarely-present entity types. The entities in the original annotated documents, whose class do not belong to $\mathcal{E}$, are replaced with the background `O` class.

The task generation algorithm is summarized in Algorithm 1 (in Appendix B). The statistics of FedVEX is summarized in Table 2.

4 Approaches

To solve the proposed FVDER tasks, we employ the meta-learning (i.e., learning-to-learn). Different from the recent advancement based on pre-training or prompts (Wang and Shang, 2022; Wang et al., 2022), meta-learning helps to significantly promote *quick* adaptation on *novel* few-shot entity types.

Figure 2 is an overview of our framework. It consists of three components: the *multimodal encoding* network (Section 4.1), the decoder for *token labelling* (Section 4.3), and a *meta-learner* built upon the encoder-decoder model, where we propose two task-aware meta-learning methods (Section 4.4).

4.1 Multimodal Encoding

We consider an encoder network represented by a parameterized function f^{enc}_{ϕ} with parameters ϕ. The encoder aims to capture the cross-modal semantic relationships between tokens in a document image. To achieve this, we employ a BERT-like Transformer (Devlin et al., 2018) with an additional positional embedding layer for the 2d position of each input token, through which the complex spatial structure of the input document can be incorpo-

[1] For page limit, details are moved to Appendix B.

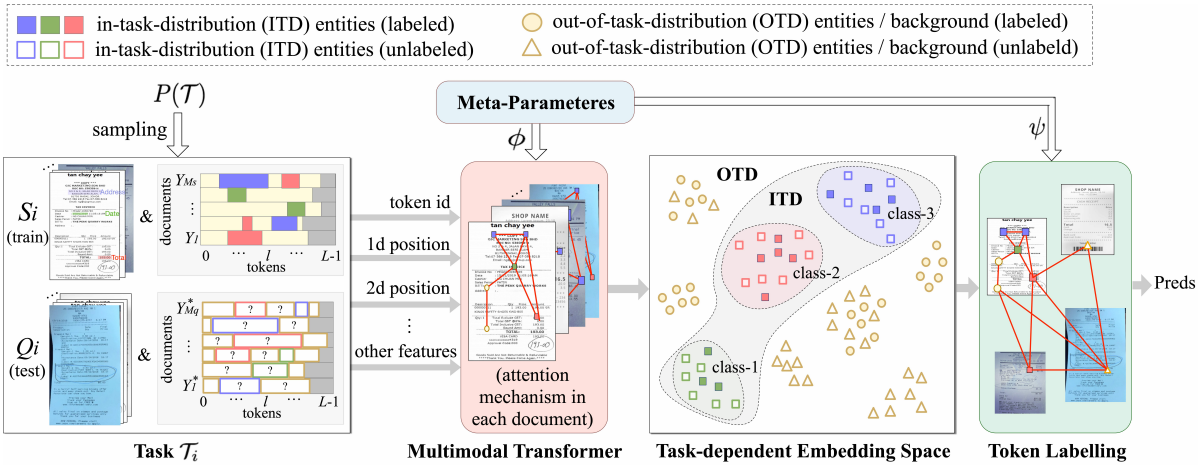


Figure 2: An overview of our meta-learning framework. The framework is applicable to both the metric-based method (aiming to learn ϕ) and gradient-based method (aiming to learn $\{\phi, \psi\}$) .

rated and then interacted with the textual contents via attention mechanisms. The embedding of token l in the document image j of task $\mathcal{T}_i$ is computed as $\mathbf{h}_{ijl} = f_\phi^{enc}(\mathbf{x}_{ijl}|X_{ij})$. In practice, before meta-training, we pre-train the multimodal Transformer on the IIT-CDIP dataset (Harley et al., 2015). Details can be found in Appendix C.1.

4.2 Task-dependent Embedding Space

Through the multimodal encoder, each task $\mathcal{T}_i$ is encoded to a *task-dependent embedding space*. As illustrated in Figure 2, on the task-dependent embedding space, there are all the token embeddings in the task: $H_i = \{\mathbf{h}_{ijl} | l \in [L], (X_j, Y_j) \in S_i \cup Q_i\}$.

There are several properties on the task's embedding space: **1)** *First*, in addition to *in-task distribution* (ITD) entities from the target classes, these exists a large portion (nearly 90% as observed in our dataset FewVEX) of *out-of-task distribution* (OTD) entities or background, which serve as the context for target ITD entities but dominate the task's embedding space. **2)** *Second*, the background tokens follows a multi-mode distribution P_i^{OTD} that consists of several unimodal distributions, each of which represents an outlier entity type aside from ITD. **3)** *Finally*, it is not guaranteed that each unimodal component of P_i^{OTD} is observable in the train set S_i–in many cases, an outlier OTD entity type could occur in the query documents but is absent in the support documents. To sum up, the background distribution in our N-way K-shot FVDER tasks is more complex and noisy, dominates the entire task, and may vary between documents.

4.3 Token Labelling

On the basis of the task-dependent embedding space, the token labelling or decoding process can either leverage a *parameterized* decoder f_ψ^{dec} that acts as the classification head, or rely on *non-parametric* methods, like nearest neighbors.

4.4 Proposed Meta Learners

We consider two main categories of the meta-learning approaches: the gradient-based and the metric-based meta-learning, on each of which we propose our own methods. We specifically pay attention to two properties when solving the entity-level N-way K-shot FVDER tasks: **1) Few-shot out-of-task distribution detection**, which aims to distinguish the ITD (i.e., the target N entity types) against the OTD (i.e., background or any outlier entity type). **2) Few-shot token labelling for in-task distribution tokens**, which assigns each ITD token to one of the N in-task entity types.

4.4.1 Task-aware ContrastProtoNet

We first focus on *metric-based* meta-learning (Snell et al., 2017; Oreshkin et al., 2018). The goal is to learn a set of *meta-parameters* ϕ for the encoder network, generally shared by each task $\mathcal{T}_i \sim P(\mathcal{T})$, such that, on each task's specific embedding space, the distances between token points in both S_i and Q_i are measured using some metrics, e.g., Euclidean distances.

ProtoNet with or without Estimated OTD. One of the most popular and effective metric-based meta-learning methods is the Prototypical Network (ProtoNet) (Snell et al., 2017). For each FVDER

task $\mathcal{T}_i = \{S_i, Q_i, \mathcal{E}_i\}$, the prototype for each entity type $e \in \mathcal{E}_i$ can be computed as the mean embedding of the tokens from S_i belonging to that entity type, that is, $\boldsymbol{\mu}_{i,e} = 1/|I_e^{\mathsf{trn}}| \sum_{(j,l)\in I_e^{\mathsf{trn}}} \mathbf{h}_{ijl}$, where I_e^{trn} is a collection of the token indices for the type-e tokens in the support set. For the out-of-task distribution (OTD), one may consider to estimate its mean embedding as an extra 0-type prototype: $\overline{\boldsymbol{\mu}}_i = 1/|I_{\mathsf{OTD}}^{\mathsf{trn}}| \sum_{(j,l)\in I_{\mathsf{OTD}}^{\mathsf{trn}}} \mathbf{h}_{ijl}$.

A problem of the vanilla methods is that there is no specific mechanism distinguishing the IND entities against the OTD entities, which are weakly-supervised and partially observed from a multi-mode distribution P_i^{OTD}. The prototype $\overline{\boldsymbol{\mu}}_i$ is a biased estimation of the mean of P_i^{OTD} and the covariance of P_i^{OTD} can be larger than any of the ITD classes. In consequence, the task-specific ITD classes may not be clearly distinguished from the OTD classes on the task-dependent embedding space and most of tokens will be misclassified.

Regarding the above challenges, we propose a task-aware method that adopts two techniques to boost the performance.

Meta Contrastive Loss. During meta-training, we encourage the N ITD entity types to be distinguished from each other as well as far away from any unimodal component of OTD. To achieve this, we adopt the idea from supervised contrastive learning (Khosla et al., 2020) to compute a *meta contrastive loss* (MCON) from each task, which will be further used to compute meta-gradients for updating the meta-parameters ϕ. Intuitively, our meta-objective is that the query tokens from the ITD type-e should be pushed away from any OTD tokens and other types of ITD tokens within the same task, and should be pulled towards the prototype $\boldsymbol{\mu}_{i,e}$ of support tokens and the other query tokens belonging to the same entity type. Formally, let $I_{\mathsf{ITD}}^{\mathsf{val}} = \{(j,l)|l \in [L], (X_j^*, Y_j^*) \in Q_i, y_{ijl}^* \in \mathcal{E}_i\}$ denote a collection of ITD validation tokens. The meta contrastive loss computed from $\mathcal{T}_i$ is

$$\mathcal{L}_i^{\mathsf{MCON}} = \sum_{(j,l)\in I_{\mathsf{ITD}}^{\mathsf{val}}} \frac{-1}{|A^+(j,l)|} \sum_{\mathbf{u}\in A^+(j,l)} a_{ijl}(\mathbf{u})$$
$$a_{ijl}(\mathbf{u}) = \log \frac{\exp(\mathbf{h}_{ijl}^\top \mathbf{u})}{\sum_{\mathbf{v}\in A(j,l)} \exp(\mathbf{h}_{ijl}^\top \mathbf{v})}. \quad (2)$$

For each *anchor*, i.e., the ITD validation token l in document j, we let $A^+(j,l) = \{\mathbf{h}_{irm}|(r,m) \in I_{\mathsf{ITD}}^{\mathsf{val}} \setminus \{(j,l)\}, y_{ijl}^* = y_{irm}^*\} \cup \{\boldsymbol{\mu}_{i,e}|e \in \mathcal{E}_i, y_{ijl}^* = e\}$ denote a collection of the *positive* embeddings/prototype for the anchor and let $A(j,l) = \{\mathbf{h}_{irm}|(r,m) \in I_{\mathsf{ALL}} \setminus \{(j,l)\}\} \cup \{\boldsymbol{\mu}_{i,e}\}_{e\in\mathcal{E}_i}$ contain all the ITD/OTD embeddings and prototypes ($I_{\mathsf{ALL}} = \{(j,l)|l \in [L], (X_j, Y_j) \in S_i \cup Q_i\}$) in $\mathcal{T}_i$.

Unsupervised OTD Detector. During the testing time for novel entity types, we adopt the non-parametric token-level nearest neighbor classifier, which assigns $\mathbf{x}_{ijl}$ the same label as the support token that is nearest in the task's embedding space:

$$\hat{y}_{ijl}^{\mathsf{nn}} = \text{argmax}_{y_{irm} \text{ where } (r,m)\in I_{\mathsf{ALL}}^{\mathsf{trn}}} \mathbf{h}_{ijl}^\top \mathbf{h}_{irm}, \quad (3)$$

where $I_{\mathsf{ALL}}^{\mathsf{trn}} = \{(r,m)|m \in [L], (X_r, Y_r) \in S_i\}$. The ITD or OTD entity tokens in Q_i should be closer to the corresponding ITD or OTD tokens in S_i that belong to the same entity type. However, since the embedding space dependent on the support set is not sufficiently rich, the network may be blind to properties of the out-of-task distribution P_i^{OTD} that turn out to be necessary for accurate entity retrieval. To tackle this, we exploit an unsupervised out-of-distribution detector (Ren et al., 2021) operating on the task-dependent embedding space, in assistance with the classifier. Specifically, we define an OTD detector: $\hat{y}_{ijl} = \mathsf{O}$ if $r(\mathbf{h}_{ijl}) \geq R_i$; otherwise, $\hat{y}_{ijl} = \hat{y}_{ijl}^{\mathsf{nn}}$, where R_i is the task-dependent uncertainty threshold and $r(\mathbf{h}_{ijl})$ is defined as the OTD score of each token computed as its minimum Mahalanobis distance among the N ITD classes: $r(\mathbf{h}_{ijl}) = \min_{e\in\mathcal{E}_i} (\mathbf{h}_{ijl} - \boldsymbol{\mu}_{i,e})^\top \Omega_{i,e}^{-1} (\mathbf{h}_{ijl} - \boldsymbol{\mu}_{i,e})$. Here, $\Omega_{i,e} = \sum_{(j,l)\in I_e^{\mathsf{trn}}} (\mathbf{h}_{ijl} - \boldsymbol{\mu}_{i,e})^\top (\mathbf{h}_{ijl} - \boldsymbol{\mu}_{i,e})$ is the covariance matrix for entity type e computed from the type-e tokens in the support set (I_e^{trn}). The higher OTD score indicates the more likely the token belongs to the background.

4.4.2 Computation-efficient Gradient-based Meta-learning with OTD Detection

For *gradient-based meta learning*, the goal is to learn the meta-parameters $\theta = \{\phi, \psi\}$ globally shared over the task distribution $P(\mathcal{T})$, which can be fast fine-tuned for any given individual task $\mathcal{T}_i$.

Computation-efficient Meta Optimization. Although MAML (Finn et al., 2017) is the most widely adopted approach, the fact that it needs to differentiate through the fine-tuning optimization process makes it a bad candidate for Transformer-based encoder-decoder model, where we need to save a large number of high-order gradients for the encoder. Instead, we consider two alternatives

which require less computing resources and more efficient. **ANIL** (Raghu et al., 2019) employs the same bilevel optimization framework as MAML but the encoder is not fine-tuned during the inner loop. The features from the encoder are reused in different tasks, to enable the rapid fine tuning of the decoder. **Reptile** (Nichol et al., 2018) is a first-order gradient based approach that avoids the high-order meta-gradients. To further boost training efficiency, we exploit Federated Learning (Lin et al., 2022) for meta-optimization of Transformer.

Task-aware Hierarchical Classification (HC). A vanilla classifier can achieve high performance in the label-sufficient VDER. However, it turns out to be not robust in few-shot FVDER tasks because of the existence of the complicated out-of-task entities–the models usually either get over-confident on the N IID entity types or fail to distinguish target entities from the OTD background. For this reason, we incorporate OTD detection into the decoder and propose a hierarchical classifier, which has two classifiers $\psi = \{\psi_1, \psi_2\}$: 1) *binary* classifier $f^{bin}_{\psi_1}$, so that all ITD tokens are classified against OTD ones, and 2) *entity* classifier $f^{ent}_{\psi_2}$, so that ITD tokens are classified to one of the N entity types of the task. Specifically, suppose $P_i^{\texttt{OTD}}$ and $P_i^{\texttt{ITD}}$ denotes the OTD and ITD of the task $\mathcal{T}_i$, respectively. The probability that the token $\mathbf{h}_{ijl}$ is from OTD is denoted as $P(y_{ijl} = \texttt{O}) = f^{ent}_{\psi'_{i1}}(\mathbf{h}_l)$, which is used as the OTD score to weight the entity prediction. The probability that the token is the entity type-e is computed as $P(y_{ijl} = e | \mathbf{x}_{ijl} \in P_i^{\texttt{ITD}}) = (1 - P(y_{ijl} = \texttt{O})) f^{ent}_{\psi'_{i2}}(\mathbf{h}_{ijl})_e$.

5 Experiments

We experimented the methods implemented using JAX on 16 TPUs. We use the Adam optimizer to update the meta-parameters. For gradient based methods, we use vanilla SGD for the inner-loop optimization and fix 15 SGD updates with a constant learning rate of 0.015. Other hyperparameters are available in the Appendix D.

We consider on two types of performance: **Overall**, which is the precision (P), recall (R) and micro F1-score over meta-testing tasks; **Task Specificity (TS)**, which is the AUROC (Xiao et al., 2020) using the negative OTD scores over meta-testing tasks.

5.1 Main Results

Table 3 compares different meta-learning methods. Under the same N-way K-shot setting (columns), the traditional meta-learning methods fail to balance the precision and recall performances: ANIL and Reptile using vanilla decoders can achieve high precision but tend to perform low recall; the vanilla Prototypical Networks tend to be opposite: low precision but high recall. In contrast, ANIL+HC, Reptile+HC and ContrastProtoNet, which employ several strategies to detect and alleviate the influence of out-of-task distributions, achieve better precision-recall balance and thus can obtain high F1 scores and high task specificity. In Figure 4 (in Appendix F), we show ROC curves and visualization of embedding space, comparing ANIL+HC against ANIL, from 4-way 1-shot to 4-shot setting. W we observe the increase of TS and the more accurate boundary between background embeddings and different entity types.

The reasons are as follows. (1) ANIL and Reptile treat the dominant OTD instances as an extra class as well. The problem turns out the imbalanced classification in meta-learning, one of the challenges in FVDER tasks. By using an OTD detector, ANIL+HC and Reptile+HC can faster adapt to the task-specific boundary between OTD and ITD. Overall, this potentially increase the recall and task specificity score and the overall F1 score. (2) For the vanilla metric-based methods, where OTD instances are treated as one extra class, the ITD testing instances tend to be close to ITD class centers so that we have high recall. However, OTD instances dominate the task. It is possible that some OTD testing instances are closer to ITD centers than the OTD class center (the average center of multiple OTD classes) so that most of them are misclassified as one of ITD classes, i.e., low precision. In opposite, ContrastProtoNet does not make any assumption on the OTD distribution; instead, we enforce OTD to be far away from ITD classes and classify via token-level similarities while considering probabilistic uncertainty.

5.2 Class Structure Disentanglement

We examine the explanability and disentanglement of the learned representations (generated by the meta-parameters of encoder). Figure 3 shows tSNE visualizations of the learned embedding space of a selected task. Overall, by comparing Figure 3 to Table 3, the higher performance appears to be consistent with more disentangled clusters. Moreover, from the first column containing IND (*red*) tokens and OTD (*blue*) tokens, we observe that

Methods	4-way 1-shot				4-way 4-shot				5-way 2-shot			
	Overall			TS	Overall			TS	Overall			TS
	P	R	F1	AUROC	P	R	F1	AUROC	P	R	F1	AUROC
ProtoNet	0.02	0.10	0.03	N/A	0.02	0.09	0.03	N/A	0.02	0.09	0.03	N/A
ProtoNet+EOD	0.13	0.47	0.21	N/A	0.11	0.58	0.23	N/A	0.11	0.35	0.17	N/A
ContrastProtoNet	0.54	0.43	**0.47**	**0.59**	0.61	0.59	**0.60**	**0.89**	0.49	0.41	**0.44**	**0.62**
Reptile	0.48	0.10	0.15	0.58	0.62	0.44	0.51	0.67	0.39	0.09	0.14	0.59
ANIL	0.39	0.19	0.25	0.56	0.54	0.44	0.50	0.87	0.35	0.13	0.19	0.61
Reptile+HC	0.35	0.13	0.20	0.63	0.63	0.65	**0.64**	**0.98**	0.34	0.12	0.18	0.65
ANIL+HC	0.40	0.58	**0.50**	**0.95**	0.47	0.59	0.51	0.98	0.38	0.56	**0.46**	**0.92**

Table 3: Performance on 4-way 1-shot, 4-way 4-shot, and 5-way 2-shot settings of FewVEX(S).

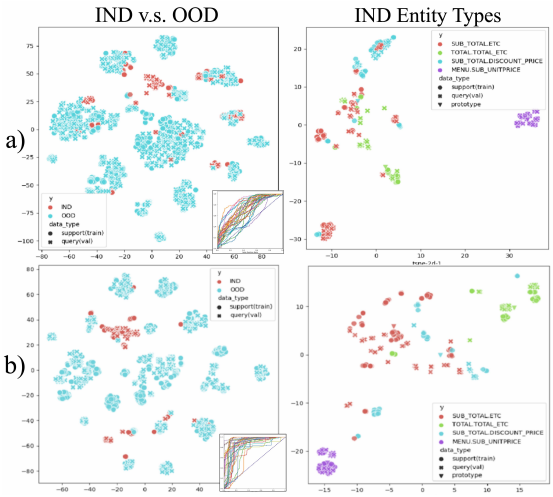


Figure 3: Embedding space visualization for a randomly-selected meta-testing task, comparing a) vanilla ProtoNet and b) ContrastProtoNet methods, under the 4-way 4-shot setting of FewVEX(S).

the blue points dominate the embedding space and comprises multiple clusters, which demonstrates the out-of-task distribution is multimodal, making it hard to identify in-task entities. Further, we try to understand the disentangled structure of classes from the clusters. We zoom into the 4 IND classes, which are represented by different colors in the right column in Figure 3, We observe that "`menu (sub_uniprice)`" (violet) is far away from the other three classes, while the other three classes are slightly entangled. Such class structure represents the relationships between these entity types, which is explainable: the red and blue classes belong to the same superclass `sub_total`; the green and red are both `etc`-related information.

5.3 Multi-domain Few-shot VDER

Table 4 reports the 4-way 2-shot results on the mixed-domain FewVEX(M), which combines receipts with forms for few-shot learning. The results slightly underperform those under the single-domain setting. A reason could be that the structure of forms is different from that of receipts and it is challenging to find the good meta-parameters for both domains. Moreover, the number of classes in the form domain is much smaller than that in the receipt domain. Such imbalanced class combination would push the meta-parameters to adapt to the relative prominent domain.

Methods	P	R	F1	AUROC
ProtoNet	0.02	0.10	0.03	N/A
ProtoNet+EOD	0.18	0.46	0.26	N/A
ContrastProtoNet	0.54	0.46	**0.50**	**0.85**
Reptile	0.45	0.17	0.25	0.57
ANIL	0.39	0.19	0.26	0.56
Reptile+HC	0.42	0.23	0.30	0.88
ANIL+HC	0.44	0.56	**0.49**	**0.97**

Table 4: Performance on 4-way 2-shot FewVEX(M).

6 Conclusions

In this paper, we studied the multimodal few-shot learning problem of VDER. We started by proposing a new formulation of the FVDER problem to be an entity-level, N-way K-shot learning under the framework of meta learning as well as a new dataset, which is designed to reflect the practical problems. We exploited a wide range of approaches, including metrics based and gradient based meta learning methods, along with a few new techniques we came up with for this new setting. The proposed methods achieves major improvements over the baselines for FVDER. We believed our approaches can be further improved in the following directions: 1) A better algorithm that distinguishes between the OTD and ITD that goes between the proposed ones. and 2) A formulation that considers the correlations between entity instance within each meta learning tasks.

Limitations

There exists a few limitations to this work. Firstly, the derived dataset is based on the current open source ones for document understanding, which are small in their size and has very limited amount of classes. A dedicated dataset that is built specifically for the purpose of studying few-shot learning for document entity retrieval is needed. Secondly, the scope of our current studies is limited to non-overlapping entities. The performance of the models under nested and entities with overlapping ground truth is yet to be examined.

Ethics Statements

The dataset created in this paper was derived from public datasets (i.e., FUNSD, CORD) which are publicly available for academic research. No data collection was made during the process of making this work. The FUNSD and CORD datasets themselves are a collection of receipts and forms collected and released by a third party paper which has been widely used in the field of visually rich document entity retrieval research and is not expected to contain any ethnics issues to the best of our knowledge.

References

Ali Furkan Biten, Ron Litman, Yusheng Xie, Srikar Appalaraju, and R Manmatha. 2022. Latr: Layout-aware transformer for scene-text vqa. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16548–16558.

Arindam Chaudhuri, Krupa Mandaviya, Pratixa Badelia, and Soumya K Ghosh. 2017. Optical character recognition systems. In *Optical Character Recognition Systems for Different Languages with Soft Computing*, pages 9–41. Springer.

Yinbo Chen, Zhuang Liu, Huijuan Xu, Trevor Darrell, and Xiaolong Wang. 2021. Meta-baseline: Exploring simple meta-learning for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9062–9071.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR.

Łukasz Garncarek, Rafał Powalski, Tomasz Stanisławek, Bartosz Topolski, Piotr Halama, Michał Turski, and Filip Graliński. 2021. Lambert: layout-aware language modeling for information extraction. In *International Conference on Document Analysis and Recognition*, pages 532–547. Springer.

Zhangxuan Gu, Changhua Meng, Ke Wang, Jun Lan, Weiqiang Wang, Ming Gu, and Liqing Zhang. 2022. Xylayoutlm: Towards layout-aware multimodal networks for visually-rich document understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4583–4592.

Adam W Harley, Alex Ufkes, and Konstantinos G Derpanis. 2015. Evaluation of deep convolutional nets for document image classification and retrieval. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 991–995. IEEE.

Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. 2019. Funsd: A dataset for form understanding in noisy scanned documents. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 2, pages 1–6. IEEE.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673.

Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille.

Chen-Yu Lee, Chun-Liang Li, Timothy Dozat, Vincent Perot, Guolong Su, Nan Hua, Joshua Ainslie, Renshen Wang, Yasuhisa Fujii, and Tomas Pfister. 2022. Formnet: Structural encoding beyond sequential modeling in form document information extraction. *arXiv preprint arXiv:2203.08411*.

Jing Li, Billy Chiu, Shanshan Feng, and Hao Wang. 2020. Few-shot named entity recognition via meta-learning. *IEEE Transactions on Knowledge and Data Engineering*.

Bill Yuchen Lin, Chaoyang He, Chulin Xie, Fatemehsadat Mireshghallah, Ninareh Mehrabi, Tian Li, Mahdi Soltanolkotabi, and Xiang Ren, editors. 2022. *Proceedings of the First Workshop on Federated Learning for Natural Language Processing (FL4NLP 2022)*. Association for Computational Linguistics, Dublin, Ireland.

Xiaojing Liu, Feiyu Gao, Qiong Zhang, and Huasha Zhao. 2019. Graph convolution for multimodal information extraction from visually rich documents. *arXiv preprint arXiv:1903.11279*.

Alex Nichol, Joshua Achiam, and John Schulman. 2018. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*.

Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. 2018. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*, pages 721–731.

Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. 2019. Cord: a consolidated receipt dataset for post-ocr parsing. In *Workshop on Document Intelligence at NeurIPS 2019*.

Krishna Pillutla, Sham M Kakade, and Zaid Harchaoui. 2019a. Robust aggregation for federated learning. *arXiv preprint arXiv:1912.13445*.

Krishna Pillutla, Sham M Kakade, and Zaid Harchaoui. 2019b. Robust aggregation for federated learning. *arXiv preprint arXiv:1912.13445*.

Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. 2019. Rapid learning or feature reuse? towards understanding the effectiveness of maml. In *International Conference on Learning Representations*.

Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. 2021. A simple fix to mahalanobis distance for improving near-ood detection. *arXiv preprint arXiv:2106.09022*.

Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. 2019. Meta-learning with latent embedding optimization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning. *arXiv preprint arXiv:1606.04080*.

Yaqing Wang, Subhabrata Mukherjee, Haoda Chu, Yuancheng Tu, Ming Wu, Jing Gao, and Ahmed Hassan Awadallah. 2021a. Meta self-training for few-shot neural sequence labeling. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1737–1747.

Zifeng Wang, Zizhao Zhang, Jacob Devlin, Chen-Yu Lee, Guolong Su, Hao Zhang, Jennifer Dy, Vincent Perot, and Tomas Pfister. 2022. Queryform: A simple zero-shot form entity query framework. *arXiv preprint arXiv:2211.07730*.

Zilong Wang and Jingbo Shang. 2022. Towards few-shot entity recognition in document images: A label-aware sequence-to-sequence framework. *arXiv preprint arXiv:2204.05819*.

Zilong Wang, Yiheng Xu, Lei Cui, Jingbo Shang, and Furu Wei. 2021b. Layoutreader: Pre-training of text and layout for reading order detection. *arXiv preprint arXiv:2108.11591*.

Xiongwei Wu, Doyen Sahoo, and Steven Hoi. 2020. Meta-rcnn: Meta learning for few-shot object detection. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1679–1687.

Zhisheng Xiao, Qing Yan, and Yali Amit. 2020. Likelihood regret: An out-of-distribution detection score for variational auto-encoder. *Advances in neural information processing systems*, 33:20685–20696.

Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei Florencio, Cha Zhang, Wanxiang Che, et al. 2020a. Layoutlmv2: Multi-modal pre-training for visually-rich document understanding. *arXiv preprint arXiv:2012.14740*.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020b. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1192–1200.

Xiao Yang, Ersin Yumer, Paul Asente, Mike Kraley, Daniel Kifer, and C Lee Giles. 2017. Learning to extract semantic structure from documents using multimodal fully convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5315–5324.

Jaesik Yoon, Taesup Kim, Ousmane Dia, Sungwoong Kim, Yoshua Bengio, and Sungjin Ahn. 2018. Bayesian model-agnostic meta-learning. In *Advances in Neural Information Processing Systems*, pages 7332–7342.

Martin Zinkevich, Markus Weimer, Lihong Li, and Alex Smola. 2010. Parallelized stochastic gradient descent. *Advances in neural information processing systems*, 23.

A Related Work

Visually-rich Document Entity Retrieval (VDER). Deep neural networks–RNNs, CNNs and Graph Neural Networks (GNNs) have been extensively adopted to solve VDER (Yang et al., 2017; Liu et al., 2019). Most recently, motivated by the advancement of Transformers, researchers have started pre-training models to integrate visual and layout information with the text embeddings (Gu et al., 2022; Xu et al., 2020b,a; Biten et al., 2022; Garncarek et al., 2021; Lee et al., 2022), and then fine-tune the models on VDER tasks in a label-sufficient supervised manner (Xu et al., 2020a; Garncarek et al., 2021; Lee et al., 2022). Different from these lines of work, we tackle the challenging few-shot VDER task without sufficient annotation for rare and novel entity types.

Few-shot Document Entity Retrieval. This has not been a lot of efforts on few-shot VDER. Most recently, researchers have explored multimodal pre-training method (Wang et al., 2021b) that will be fine-tuned on a small number of fully-labelled document. The most recent work have employ the prompt method (Wang and Shang, 2022). Before fine-tuning on a single few-shot task, its model already sees the same entity types in a label-sufficient source domain. In contrast, we address the limitation of the task settings of these methods, i.e., incapability for novel and rare entity types, and propose a novel task setting.

Meta-learning for Few-shot Learning. Meta-learning approaches to few-shot learning problem mainly include gradient-based methods (Finn et al., 2017; Yoon et al., 2018; Rusu et al., 2019) and metric-based methods (Snell et al., 2017; Oreshkin et al., 2018; Koch et al., 2015; Vinyals et al., 2016). There are a variety of meta-learning approaches associated with different few-shot learning tasks in CV and NLP, such as the general few-shot image classification (Chen et al., 2021), few-shot object detection (Wu et al., 2020), few-shot sequence labelling (Wang et al., 2021a), few-shot named entity recognition (Li et al., 2020). Different from these literature, this paper is the first work that explores the meta-learning formulation for few-shot VDER.

B Dataset

Since there is no dataset specifically designed for the FVDER task defined in Section 2, we construct a new dataset, FewVEX, to benchmark and evaluate the meta-learning based FVDER.

B.1 Collection of Entity Types and Documents

First, we collect the entity types $\mathcal{C}$ associated with the task distribution $P(\mathcal{T})$ and a set of document images $\mathcal{D}_{orig}$ annotated by these entity types.

We consider two source datasets that are widely used in normal large-scale document understanding tasks such as entity recognition, parsing, and information extraction. The first one is the Form Understanding in Noisy Scanned Documents (FUNDS) dataset (Jaume et al., 2019) comprises 199 real, fully annotated, scanned forms, with a total of three types of entities (i.e., questions, answers, heads). The second one is the Consolidated Receipt Dataset for post-OCR parsing (CORD) dataset (Park et al., 2019). CORD consists of 1000 receipt images of texts and contains 6 superclasses (menu, void menu, subtotal, void total, total, and etc) which are divided into 30 fine-grained subclasses. For different entity types, the total numbers of entity occurrences over the CORD images are highly imbalanced, ranging from 1 occurrence of entity "`void menu (nm)`" to 997 occurrences of "`menu (price)`".

From the two datasets, we obtain a combined source dataset denoted as $\mathcal{D}_{orig}$, which contains 1199 unique document images with original annotations on 33 classes. However, we observe that some fine-grained classes in CORD occurs in less than $max_i(M_{si}+M_{qi})$ images, the maximum number of documents within individual tasks. This will result in a large amount of repetitive usage of the same documents within one task and between different tasks. Therefore, we further sort the 33 classes by the number of unique document images where they occur and then discard three entity types that occurs in low frequency.

To sum up, we finally have a total of $|\mathcal{C}|$ =30 entity types and $|\mathcal{D}_{orig}| = 1199$ unique document images annotated by these entity types. The pie chart (on the left) in Figure 1 illustrates the number of occurrences of the final entity types.

B.2 Collection of Training and Testing Tasks

Second, we create a meta-learning dataset $\mathcal{D}_{meta} = \{\mathcal{D}^{trn}_{meta}, \mathcal{D}^{tst}_{meta}\}$, consisting of a meta-training set $\mathcal{D}^{trn}_{meta}$ containing τ_{trn} training tasks and a meta-testing set $\mathcal{D}^{tst}_{meta}$ containing τ_{tst} testing tasks. Each task instance follows the N-way K-shot FVDER task setting. An overview of dataset construction is in Figure 1.

B.2.1 Entity Type Split

To ensure that testing tasks in $\mathcal{D}^{tst}_{meta}$ focus on novel classes that are unseen during meta-training $\mathcal{D}^{trn}_{meta}$, we should split the total entity types $\mathcal{C}$ into two separate sets $\mathcal{C} = C_{base} \cup \mathcal{C}_{novel}, \mathcal{C}_{base} \cap \mathcal{C}_{novel} = \emptyset$ such that $\mathcal{C}_{base}$ is used for meta-training and $\mathcal{C}_{novel}$ for meta-testing.

Specifically, we use a split ratio γ to control the number of novel classes and randomly choose $\gamma|C|$ entity types from $\mathcal{C}$ as $\mathcal{C}_{novel}$. Then, $\mathcal{C}_{base} = \mathcal{C} \setminus \mathcal{C}_{novel}$. Note that for the cases that some entity types occurs in less number of documents than the others, we set a threshold U and any entity type that occurs in less than U documents are forced to be one of the novel classes.

B.2.2 Single N-way K-shot Task Generation

Each individual task $\mathcal{T} = \{S, Q, \mathcal{E}\}$ in either $\mathcal{D}^{trn}_{meta}$ or $\mathcal{D}^{tst}_{meta}$ can be generated by the following steps (summarized in Algorithm 1).

Class sampling. The target classes of task $\mathcal{E}$ is generated by randomly sampling N entity types from either $\mathcal{C}_{base}$ (for the training task) or $\mathcal{C}_{novel}$ (for the testing task).

Document sampling. Given the N target classes, we then collect document images that satisfies the few-shot setting defined in Section 2.1. However, one problem of document sampling from the original corpus is the *inefficiency*. It is because, for each task, only a small number of documents that contain the corresponding classes can be the candidate documents of the task. For example, if each document contains only a small number of entity types, the majority of documents would be rejected. To improve sampling efficiency, one strategy is to count entities in each document in advance and, for each entity type, all the candidate documents that contain this type are temporally stored in a new dataset. Then, we only look at the task-specific candidate datasets $\mathcal{D}^{\mathcal{E}}_{orig} = \{\mathcal{D}^{e}_{orig} | \forall e \in \mathcal{E}\}$, where $\mathcal{D}^{e}_{orig} = \{(X, Y) | \forall (X, Y) \in \mathcal{D}_{orig} \text{ if } e \in Y\}$. Specifically, we randomly sample M_s documents such that the total number of entity instances is satisfied–that is, $K \sim \rho K$ shots per entity type. Likewise, we sample M_q documents for Q, such that there are $K_q \sim \rho K_q$ shots per entity type. We keep track a table to record the current count of occurrences of each type of entity types in the task.

Label Conversion. In the few-shot setting, the majority region of an document does not follow the *in-task distribution* (ITD) of $\mathcal{E}$. These regions' tokens are treated as either background or the other types of entities from the *out-of-task distribution* (OTD), whose original labels should be arbitrarily converted into `O` label. In addition, we map the original labels of ITD tokens to relative labels. For example, if we use I/O schema, the relative labels should range from label id 0 to label id $(N-1)$.

Algorithm 1 Single FVDER Task Generation

Require: $N, K, K_q, \rho, \mathcal{C}_{base}, \mathcal{C}_{novel}, \mathcal{D}_{orig}$.
Randomly sample N entity types from either $\mathcal{C}_{base}$ or $\mathcal{C}_{novel}$ and obtain $\mathcal{E}$.
Initialize: $S = \emptyset, Q = \emptyset$
Initialize: $\mathcal{D}^{\mathcal{E}}_{orig} = \{\mathcal{D}^{e}_{orig} | \forall e \in \mathcal{E}\}$ from $\mathcal{D}_{orig}$.
Initialize: N integers $train\_count[e] = 0$ for $\forall e \in \mathcal{E}$.
Initialize: N integers $test\_count[e] = 0$ for $\forall e \in \mathcal{E}$.
// Document sampling for S
while $\min_{e \in \mathcal{E}} train\_count[e] < K$ **do**
  Find the least frequent entity type in the current task, i.e., $\hat{e} = \text{argmin}_{e \in \mathcal{E}} train\_count[e]$.
  Sample a document (X_j, Y_j) from $\mathcal{D}^{\hat{e}}_{orig}$
  Add (X_j, Y_j) to S
  for $e \in \mathcal{E}$ **do**
    Remove the selected document from candidate dataset $\mathcal{D}^{e}_{orig} \leftarrow \mathcal{D}^{e}_{orig} \setminus \{(X, Y)\}$
    Update $train\_count[e]$ if Y contains entity type e.
    if $train\_count[e] > \rho K$ **then**
      Mask $(train\_count[e] - \rho K)$ instances of type-e by setting token labels to -1
    end if
  end for
end while
// Document sampling for Q
while $\min_{e \in \mathcal{E}} test\_count[e] < K_q$ **do**
  Find the least frequent entity type in the current task, i.e., $\hat{e} = \text{argmin}_{e \in \mathcal{E}} test\_count[e]$.
  Sample a document (X_j, Y_j) from $\mathcal{D}^{\hat{e}}_{orig}$
  Add (X_j, Y_j) to Q
  for $e \in \mathcal{E}$ **do**
    Remove the selected document from candidate dataset $\mathcal{D}^{e}_{orig} \leftarrow \mathcal{D}^{e}_{orig} \setminus \{(X, Y)\}$
    Update $test\_count[e]$ if Y contains entity type e.
    if $test\_count[e] > \rho K_q$ **then**
      Mask $(test\_count[e] - \rho K_q)$ instances of type-e by setting token labels to -1
    end if
  end for
end while
Label conversion for $\forall (X_j, Y_j) \in S \cup Q$.
return: $\mathcal{T} = \{S, Q, \mathcal{E}\}$

B.3 Dataset Variants

We fix the testing shot as K_q=4. We propose two variants of meta-dataset, each of which pay attention to different challenges in few-shot learning. The statistics is summarized in Table 2: **FewVEX(S)** focuses on single-domain receipt understanding under N-way K-shot setting. The training and testing classes are both from CORD. The goal is to learn domain-invariant meta-parameters.

FewVEX(M) focuses on learning domain-agnostic meta-parameters from a combination of receipt and form understanding. Receipt and form documents may appear in the same task.

C Experiment Details

C.1 Multimodal Encoder

We pre-train the multimodal Transformer on the IIT-CDIP dataset (Harley et al., 2015). It should be noting that this paper does not focus on the pre-training technique. In fact, our framework does not require a well pre-trained encoder, since the meta-learning will further meta-tune the pre-trained encoder to capture the domain knowledge of $P(\mathcal{T})$. Thus, we stop the pre-training until an 81.5% token classification accuracy.

C.2 Training Parallelism

Both meta-training and meta-testing were run in a multi-process manner. We employ episodic training pipeline to learn the meta-parameters from training tasks (i.e., episodes). At each meta-training step, a total of τ episodes are trained and then validated to obtain the meta-gradients used for updating meta-parameters. Suppose B is the total number of available TPU devices on each process. Since the parameter size of the Transformer-based encoder is large, we use B devices to train each episode in parallel, that is, the documents of one task are equally assigned to different devices. A problem is that the prototypes, the nearest neighbors of data points, or the adapted parameters trained on the support set, are only computed on each local device. For validation on the query set, however, we should consider, the scope of the entire task over different local devices. Therefore, we employ federated learning techniques (Zinkevich et al., 2010; Pillutla et al., 2019a,b) for a distributed optimization, where we collect the locally trained parameters or prototypes from the B devices of a single episode and average their parameters.

D Hyperparameters

We summarize the hyperparameters for constructing FewVEX in Table 5.

E Evaluation Metrics

We consider on two types of performance: **Overall**, which is the precision (P), recall (R) and micro F1-score over meta-testing tasks. We use the I/O tagging schema and the "seqeval" tool to compute the P/R/F1. **Task Specificity (TS)**, which is the AUROC (Xiao et al., 2020) using the negative OTD scores over meta-testing tasks. To evaluate how well the learned meta-learners can distinguish in-task distribution (ITD) from the out-of-task distribution (OTD), we propose to solve the out-of-distribution (OOD) detection as a subtask. OOD calculates a ITD score for each data point representing how likely it belongs to the task-specific distribution. For measuring task specificity, we calculate AUROC (Xiao et al., 2020) using the ITD scores over all test episodes. A higher AUROC value indicate better TS performance, and a random guessing detector corresponds to an AUROC of 50%. We use the "sklearn.metrics" tool to compute the AUROC and plot ROC curves.

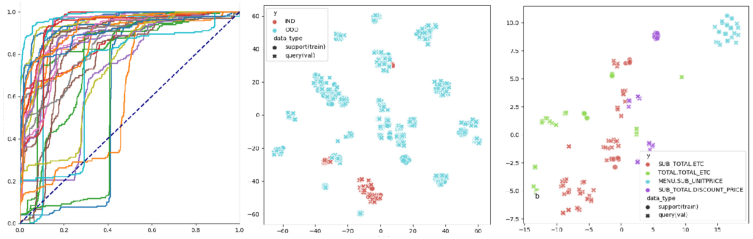
Hyperparameters	Value
ρ	3
γ	0.6
U	20
K_q	4

Table 5: Hyperparameters.

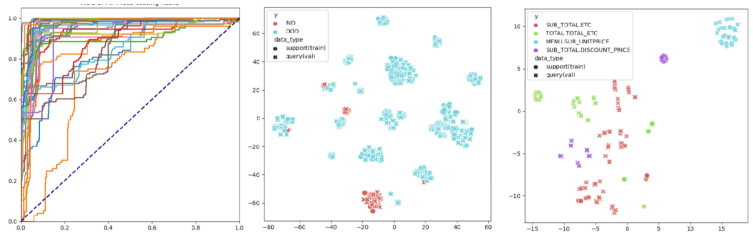
F Visualization

Vo visualize the TS, we plot the ROC curves of all the meta-testing tasks, where each curve represent one task. Another visualization for TS is to show how ITD and OOD are distinguished against each other. We randomly select a testing task and exploit tSNE (Van der Maaten and Hinton, 2008) to visualize the learned embeddings of all the tokens in the task, where ITD tokens are denoted as red points and OTD tokens are blue points. Finally, we use tSNE to visualize the learned embeddings of only the ITD token instances in the task, where different colors represent different entity types.

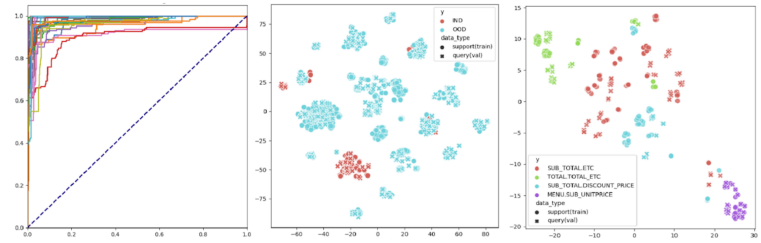
More visualization results are reported in Figure 4, Figure 5, Figure 6.



(a) ANIL (4-way 1-shot)

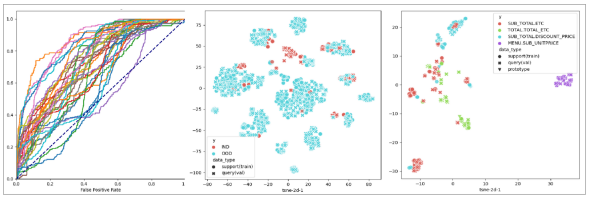


(b) ANIL + HC (4-way 1-shot)

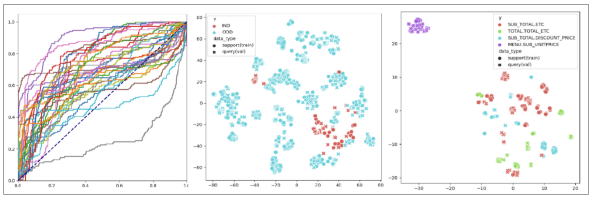


(c) ANIL + HC (4-way 4-shot)

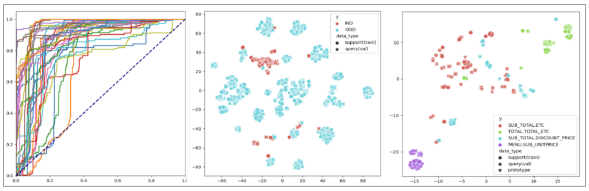
Figure 4: Visualization under 4-way 4-shot and 4-way 1-shot settings of FewVEX(S), for ANIL and ANIL+HC.



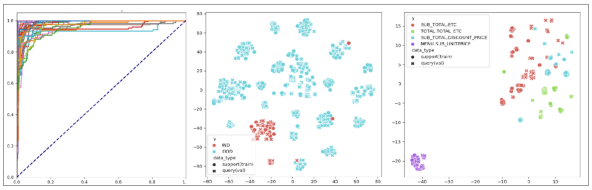
(a) ProtoNet (4-way 4-shot)



(c) Reptile (4-way 4-shot)

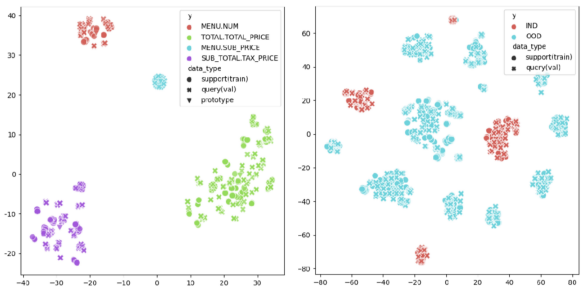


(b) ContrastProtoNet (4-way 4-shot)

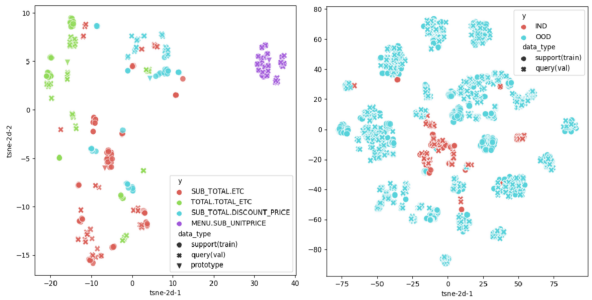


(d) Reptile + HC (4-way 4-shot)

Figure 5: Visualization and ROC curves of different methods on the 4-way 4-shot setting of FewVEX(S).



(a) token embeddings of a training task (4 base classes)



(b) token embeddings of a testing task (4 novel classes)

Figure 6: Task-specific Class distribution of a training task and a testing task of 4-way 4-shot setting. The meta-parameters trained using ContrastProtoNet on FewVEX(M). Solid points represent train (support) tokens, cross points represent val/test (query) tokens, and the triangle points represent prototypes.