Adversity-aware Few-shot Named Entity Recognition via Augmentation Learning

Li Huang[1,2], **Haowen Liu**[1], **Qiang Gao**[1,2*], **Jiajing Yu**[1], **Guisong Liu**[1,2,3], **Xueqin Chen**[3]

[1]School of Computing and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu, China
[2]Engineering Research Center of Intelligent Finance, Ministry of Education, Chengdu, China
[3]Kash Institute of Electronics and Information Industry, Kashgar, China
lihuang@swufe.edu.cn, 223081200021@smail.swufe.edu.cn, qianggao@swufe.edu.cn, 223081200001@smail.swufe.edu.cn, gliu@swufe.edu.cn, nedchen0728@gmail.com

Abstract

Few-shot Named Entity Recognition (NER) spotlights the tag of novel entity types in data-limited scenarios or lower-resource settings. Advances with Pre-trained Language Models (PLMs), including BERT, GPT, and their variants, have driven tremendous strategies to leverage context-dependent representations and exploit predefined relational cues, yielding significant gains in witnessing unseen entities. Nevertheless, a fundamental issue exists in prior efforts regarding their susceptibility to adversarial attacks in the intricate semantic environment. This vulnerability undermines the robustness of semantic representations, exacerbating the challenge of accurate entity identification, especially when transitioning across domains. To this end, we propose an **A**dversity-aware **A**ugment **L**earning (**AAL**) solution for the few-shot NER task, dedicated to retrieving and reinforcing entity prototypes resilient to adversarial inference, thereby enhancing cross-domain semantic coherence. In particular, AAL employs a two-stage paradigm consisting of training and fine-tuning. The process initiates with augmentation learning by leveraging two kinds of prompt learning schemes, then identifies prototypes under the guidance of a variational manner. Furthermore, we devise a domain-oriented prototype refinement to optimize prototype learning under conditions of uncertainty attack, facilitating the effective transfer of common knowledge from source to target domains. The experimental results, encompassing the few-shot NER datasets under both certainty and uncertainty conditions, affirm the superiority of the proposed AAL over several representative baselines, particularly its capability against adversarial attacks.

1 Introduction

Named Entity Recognition (NER) constitutes a fundamental task in natural language processing (NLP), which aims at identifying named entities, such as personal names, organizations, and geographical locations, from unstructured texts (Lample et al. 2016; Ma and Hovy 2016). Despite the significant advances achieved by existing NER efforts, their reliance on extensively annotated datasets poses a significant constraint when encountering scenarios with scarce manually tagged samples and the emergence of knowledge from diverse domains, as well as new entity types (Chiu and Nichols 2016; Devlin et al. 2018). This drawback has inspired interest in the realm of few-shot learning (FSL), which empowers NER models with the capacity to transfer knowledge from existing examples to scarcity like humans (Ding et al. 2021; Huang et al. 2021; Ma et al. 2022).

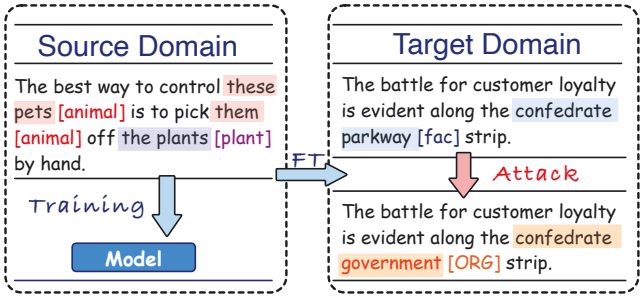

Figure 1: Exhibition of few-shot NER associated with uncertain attack condition. "[·]" refers to entity types.

The dominant solutions with deep neural architectures, e.g., convolutional networks, recurrent networks, and attention mechanisms, have emerged as the preferred alternatives in handling few-shot NER tasks. In contrast to traditional NER efforts, they adaptively seek the diverse dependencies and interactive relations underlying knowledge correlations, facilitating the discovery of potentially foundational but contextual collaborations across various domains.

To uncover knowledge correlations in a few-shot manner, the majority of previous studies attempted to consider token-level dependencies, predominantly striving to discover new entities by measuring the distance between tokens in the target domain corresponding to the source. For example, (Lin et al. 2021) used a pre-trained language model to replace the target entity with other entities that have the same semantic type. (Wang et al. 2022b) proposed two mutual information-based training goals to prevent the model from over-relying on entity-mention information. More recently, (Fang et al. 2023) acknowledged the strength of prototype representations in generalized effectiveness to the few-shot conditions. Besides, (Wang et al. 2022a) utilized prototype classification to capture the semantic representation of each label. In a nutshell, the current few-shot NER solutions usually concentrate on similarity-based methods (Fritzler, Logacheva, and

*Corresponding author (Qiang Gao)

Kretov 2019; Huisman, Van Rijn, and Plaat 2021), which are straight-forward to conduct the classification in the target domain according to its similarity with the representation of each class in the source domain, yielding promising results under lower-resource settings.

Despite the remarkable achievements in deep learning-based paradigms, previous arts endeavor to understand semantic dependencies, particularly through token-level distance measurements. More importantly, there are still two significant challenges: *scarce examples* is one of the inherent issues that hinder models from generalizing effectively beyond very limited tagged datasets, resulting in unsatisfactory outcomes when encountering new or unseen entities. Another is *resilience against adversarial attacks* particularly those leveraging synonyms to create ambiguity. In the real-world scenario, language is context-dependent, with many words or phrases that can be used interchangeably and subjectively. Adversaries, objective or subjective, exploit synonyms or other subtle variations to deliberately obscure entities, leading to potential misclassification or failure of the model to recognize the entity altogether, as illustrated in Fig. 1. Therefore, a robust few-shot NER should not only handle the constraints of limited annotated data but effectively counteract adversarial strategies that aim to introduce ambiguity and undermine its performance.

To address the aforementioned concerns, this study introduces an **A**dversity-aware **A**ugmentation **L**earning (or **AAL**) solution for handling the few-shot NER task. Specifically, AAL is designed as a two-stage paradigm consisting of training and fine-tuning, which thoroughly comprises three primary steps: (1) We first implement augmentation learning using two different prompt templates, *context-enhancing prompt* and *oriented-inducing prompt*, to enrich the source domain dataset and introduce variations that broaden the exposure of AAL to diverse scenarios. (2) To mitigate the influence of irrelevant knowledge, we employ prototype learning on the source domain to capture the common knowledge of entities, thereby facilitating the identification of tokens in the target domain. (3) To effectively harness pertinent knowledge from a novel domain, we introduce a domain-oriented prototype to optimize prototype learning within the context of uncertain noise injection, thereby augmenting the adaptability and stability of AAL. In the end, we adapt concepts from NER to jointly optimize position tags and entity types, thereby reducing the computational complexity caused by category combination explosion. In sum, our main contributions can be outlined as follows:

- We present AAL, a novel adversity-aware augmentation learning solution for the adversarial challenge of few-shot NER. In particular, it capitalizes on inter-domain knowledge transfer to reinforce common dependencies and interactive relation learning by augmenting pivotal prototypes from a source domain to the target.
- To enhance the resilience of prototypes across diverse entities, we employ a domain-oriented adaption prototype learning and further enhance with uncertainty-guided adversarial training, thereby enabling AAL to yield efficacious yet adversity-awareness prototype representations.
- Experimental results on both certain and uncertain datasets, including few-shot and cross-domain conditions, demonstrate the superiority and robustness of the proposed AAL compared to state-of-the-art baselines.

2 Preliminaries

Few-shot NER on Episode Learning. Given the source domain $\mathcal{D}_s = \{(\mathcal{S}_s, \mathcal{Q}_s)\}$, the task of few-shot NER should adapt to the target domain of $\mathcal{D}_t = \{(\mathcal{S}_t, \mathcal{Q}_t)\}$. Under episode learning, each episode consists of a *support set* $\mathcal{S}_{s/t} = \{(x^{(i)}_{s/t}, y^{(i)}_{s/t})\}^{N\times K}_{i=1}$ for adaption, and a *query set* $\mathcal{Q}_{s/t} = \{(x^{(j)}_{s/t}, y^{(j)}_{s/t})\}^{N\times K'}_{j=1}$ for evaluation. Here, N denotes the number of entity types in an episode, K and K' denote the number of examples per entity type in *support set* and *query set*, respectively, commonly referred to N-way K-shot setting (Ding et al. 2021). Typically, K is very small, often $K = 1$ or 5. The goal is to learn a model parameterized by Θ, such that it can accurately predict the named entities for a novel query instance x_t under a new semantic situation, as $\Theta(x_t) = y_t$, where $(x_t, y_t) \notin \mathcal{D}_s$.

Adversarial Attack. We utilize the widely used ambiguous situation of synonym institution as our adversarial attack setting (Li et al. 2020), where a word in the original text is randomly replaced by its synonyms (Ren et al. 2019). This substituted operation is imperceptible to humans but sensitive to the model. In detail, adversarial example $\mathcal{A}_p = (x_p, x'_p)$ are constructed by selecting out entities in the sentence that may mislead the target model and then replacing them using synonyms in the synonym set, where x_p is the specifical entity, x'_p is the corresponding attack sample. In this work, we randomly selected adversarial examples to "fool" the few-shot NER to simulate the adversarial attack scenarios.

Fine-tuning of PLMs. Fine-tuning, inspired by PLMs, is a widely used practice to address the discrepancy between pre-training objectives and downstream task requirements. In a nutshell, the process commences with an initialized PLM $\mathcal{M}$, where an input sequence $X = \{x_1, \ldots, x_n\}$ is transformed into $\hat{X} = (X, T)$, T is a task-oriented template. Subsequently, $\mathcal{M}$ along with extra fine-tuning layers (OPT) encodes $\hat{X}$ to comprehend semantic dependencies associated with the guidance of template T.

3 Methodology

Drawing upon prior arts, the reason for model performance decline for cross-domain learning in few-shot NER is manifested in two aspects: 1) imprecise classification of unseen entity types within limited training data; 2) inaccuracies in entity classification owing to ambiguous entity descriptions or intended attacking. In our proposed AAL, we introduce a novel two-stage training and fine-tuning framework for few-shot NER, incorporating augmented learning with respect to data augment and prototype augment to enhance robustness understanding. The overview of the proposed model is illustrated in Fig. 2. Now we turn to elaborate on the design of each module.

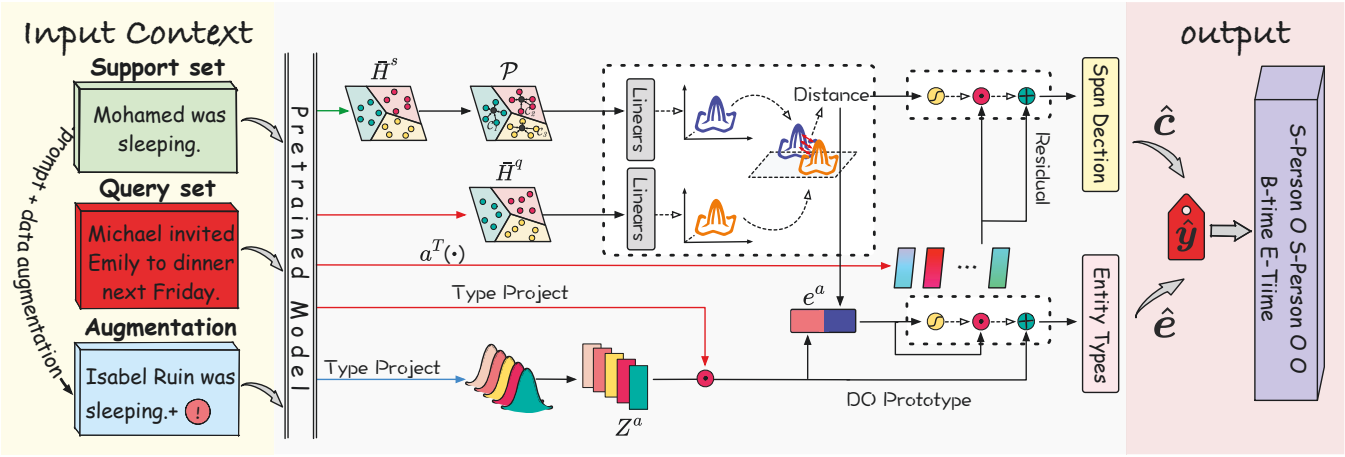

Figure 2: The illustration of the inference phase via our proposed AAL, where the distance is calculated between prototype $\mathcal{P}$ and entity representation $\bar{H}^q$. During training, distances are computed based on prototype $\mathcal{P}$ and augmentation set entities $\bar{H}^a$.

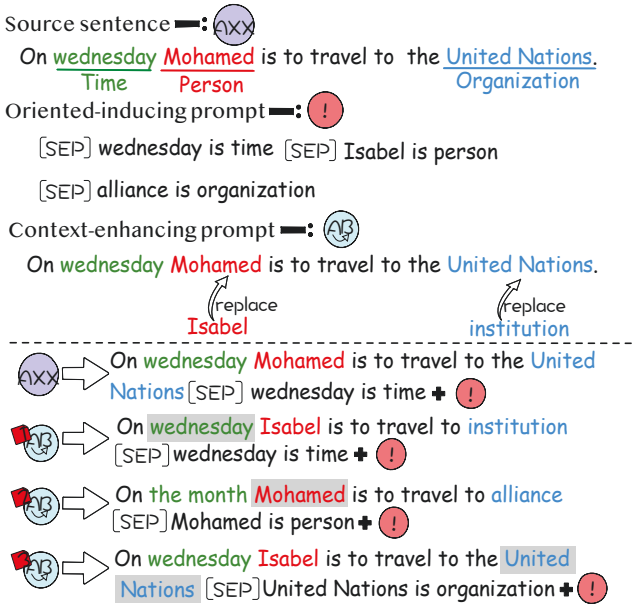

Figure 3: Illustration of prompt schemes for support set.

3.1 Augmentation Learning

Motivated by existing augmentation efforts and advancements in prompt-based techniques (Lee et al. 2021), we integrate prompt learning into data augmentation, thereby establishing a dynamic mechanism incorporating task-specific prompts for enhancing *support set*. We propose two distinct types of prompts: *context-enhancing prompt* and *oriented-inducing prompt*, as depicted in Fig. 3. These prompts are designed to introduce controlled variations that maintain consistency with the original entity types while allowing for diverse semantics context and broadening the exposure of AAL to diverse scenarios, thereby improving performance in low-resource settings.

Context-enhancing Prompt (E_c). To enhance the diversity of entity examples, we employ a context-enhancing prompting scheme that entails substituting a particular entity within a sentence attribute consistency, generating new, diverse examples while preserving the original sentence structure and context. Formally, this procedure involves a binomial distribution to determine the replacement of each entity probabilistically. During replacement, we choose to replace a word from the top-N word sets corresponding to this entity type. The employment of *context-enhancing prompt* directs the model towards the specific labels to anticipate, facilitating semantic comprehension by metric learning in diverse contexts. Notably, augmented sentences retain an unchanged entity, such as words with gray underlining in Fig. 3, and we give a template to describe it. For instance, the entity "`Mohamed`" vs. "`Mohamed is person`".

Oriented-inducing Prompt (E_o). Subsequently, to mitigate the influence of entity token replacement, we devise an oriented-inducing prompt strategy for our augmentation learning. This strategy extends the concept of entity replacement by explicitly linking the most prevalent entity tokens with their corresponding positional tags, i.e., $\{B, I, O, E, S\}$, thereby contextualizing replacements. The entities and their tags are subsequently concatenated using the special token "`[SEP]`" to maintain semantic coherence. For each sentence in the *support set*, we sequentially select each entity and append it to the end of the sentence using a tailored prompt template. This strategy enhances dataset diversity, ensuring that AAL generalizes to unseen entities and refines its recognition efficacy under uncertain scenarios.

3.2 Semantic Encoding

We initially leverage the pre-trained Language Model (PLM), such as BERT, to encode the input text into contextualized tensors. To comprehend semantic dependency, we feed the concatenate inputs of text associated with its prompt template to PLM to acquire the contextualized intermediate representation. Specifically, AAL employs two semantic encoding schemes, $H^a = PLM(X, E_c, E_o)$ de-

notes the *support set* fed to PLM are augmented, while $H = PLM(X)$ refers to the inputs are come from the original *support set*, where $\{H^a, H\} \in \mathbb{R}^{|X| \times d_m}$. Particularly, the remains where symbols with a superscript a denote AAL utilized the augmentation *support set*.

3.3 Prototype Learning

Inspired by MANNER (Fang et al. 2023), we employ prototype learning for entities, which encapsulates the essential representation of entities of the same type. Compared with MANNER which relies on a key-value pairs memory to form prototypes, AAL leverages dynamic representations that adapt to their semantic conditions and emphasizes capturing common knowledge in uncertain semantic contexts.

We introduce $\mathcal{P} = \{P_1, \cdots, P_\epsilon\} \in \mathbb{R}^{\epsilon \times d_m}$ to encapsulate the prototypical representations of the entity types from source domain $\mathcal{D}_s$, where ϵ refers to the number of entity types and P_i denotes the prototype of i-th entity type. In particular, we categorize non-typing as an entity type, indicating that tokens are not considered entities. Each prototype P_i represents a specific entity type i and can be expressed as:

$$P_i = \frac{1}{n_j} \sum_{j=1}^{n_j} (\bar{h}_i)_j, \tag{1}$$

where $\bar{h}_i \in \mathbb{R}^{d_m}$ signifies the representative token for the i-th entity type, extracted from the tensor $\bar{H} = PLM_4(\mathcal{S}_s)$, which is computed as the average outcomes from the last four layers of PLM, applied to $\mathcal{S}_s \subset \mathcal{D}_s$, without augmentations. n_j quantifies the number of tokens classified to the i-th entity type within $\mathcal{D}_s$. Notably, we only created prototypes on the *support set*.

Whereafter obtaining the prototypical representation $\mathcal{P}$, we operate the metric learning on the augmentation *support set* and *query set*. Specifically, we conduct prototype learning under the guidance of Jenson-Shannon divergence (Fuglede and Topsoe 2004) between prototypes P_i and tokens $\bar{h}^a_j$ which is retrieved from $\bar{H}^a = PLM_4(\mathcal{S}^a_s, \mathcal{Q}_s)$ as:

$$\begin{aligned} \boldsymbol{\pi}(P_i, \bar{h}^a_j) &= D_{JS}(\mathcal{N}_{P_i}, \mathcal{N}_{\bar{h}^a_j}), \\ &= \frac{1}{2}\big(D_{KL}(\mathcal{N}_{(\mu_{P_i}, \sigma_{P_i})} || \mathcal{N}_{(\mu_{\bar{h}^a_j}, \sigma_{\bar{h}^a_j})}) \\ &+ D_{KL}(\mathcal{N}_{(\mu_{\bar{h}^a_j}, \sigma_{\bar{h}^a_j})} || \mathcal{N}_{(\mu_{P_i}, \sigma_{P_i})})\big), \end{aligned} \tag{2}$$

where $\mathcal{N}_{(\mu_*, \sigma_*)}$ denotes a d-dimensional Gaussian distribution projected by a linear layer, D_{KL} refers to the Kullback-Leibler divergence, thereby $\boldsymbol{\pi}(\cdot)$ measures the distance between tokens and prototypes. For all tokens in $\{\mathcal{S}^a_s, \mathcal{Q}_s\}$, the prototype learning can be conducted as:

$$\mathcal{L}_p = \frac{1}{m} \sum_j^m \big(- \log \frac{\frac{1}{|\psi(i)|} \sum_{(P_i, \bar{h}^a_j) \in \psi(i)} \exp\big(-\boldsymbol{\pi}(P_i, \bar{h}^a_j)\big)}{\sum_i^\epsilon \exp\big(-\boldsymbol{\pi}(P_i, \bar{h}^a_j)\big)}\big), \tag{3}$$

where $m = |\{\mathcal{S}^a_s, \mathcal{Q}_s\}|$ denotes the amount of tokens in $\{\mathcal{S}^a_s, \mathcal{Q}_s\}$, $(P_i, \bar{h}^a_j) \in \psi(i)$ refers to token $\bar{h}^a_j$ and prototype P_i are attributes the same entity type, i.e., $\psi(i)$.

Span Detection. We follow the NER task to formulate span detection as a sequence labeling task, i.e., to predict the position tag $\{B, O, I, E, S\}$ of tokens. Based on the *semantic encoding* of PLM, we employ a feed-forward network associated with a gated linear layer to obtain the prediction of the position tag as:

$$\mathtt{P}_\theta(\hat{\boldsymbol{c}}|X, \mathcal{P}) = a^T(H^a) \odot g(\boldsymbol{\beta} W_c + b_c), \tag{4}$$

where $\odot$ is the element-wise product, $a^T(\cdot)$ represents a two-layer feed-forward neural network with residual connection, $g(\cdot)$ refers to the $sigmoid$ operation, H^a is the output of PLM with augmentation dataset as input, $\boldsymbol{\beta} \in \mathbb{R}^{|X| \times 2}$ is a 2-d vector, where one dimension records the top metric between each token and its corresponding prototype, while the other dimension captures the metric with a specific prototype representing non-entity. $W_c \in \mathbb{R}^{2 \times d_c}, b_c \in \mathbb{R}^{d_c}$ are learnable parameters. Finally, span detection loss is calculated by prediction $\hat{\boldsymbol{c}}$ associated with its golden label $\boldsymbol{c}$:

$$\mathcal{L}_c = \sum \boldsymbol{c}(\log \mathtt{P}_\theta(\hat{\boldsymbol{c}}|X, \mathcal{P})). \tag{5}$$

Entity Typing. We follow the principle of prototypical networks (Snell, Swersky, and Zemel 2017; Fang et al. 2023) to conduct the entity type prediction. Initially, we calculate domain-oriented entity representation associated with augmented *support set* $\mathcal{S}^a_s$ and *query set* $\mathcal{Q}_s$:

$$\begin{aligned} \mathtt{P}_\theta(Z_s|(\mathcal{S}^a_s, \mathcal{Q}_s)) &= \|_{i \in \varepsilon} \left[\mathtt{P}_\theta(Z^{\psi(i)}_s | S^{(a,\psi(i))}, Q^{\psi(i)})\right] \\ &= \|_{i \in \varepsilon} \left[\mathcal{N}(P^a_i | f_\theta(S^{(a,\psi(i))}, Q^{\psi(i)}), \sigma^2 \boldsymbol{I})\right], \end{aligned} \tag{6}$$

where $\|$ refers to group augmentation prototypes, and the updated prototype distribution, resulting from the function f_θ, is calculated by:

$$\begin{aligned} f_\theta(S^{\psi(i)}, Q^{\psi(i)}) &= \gamma \cdot FFN(r^{\{s;q\}}_i) + (1 - \gamma) \cdot r^s_i, \\ r^*_i &= \frac{1}{|\psi(i)|} \sum_j \hat{h}^{(*,\psi(i))}_j, \text{ where } \hat{h}_j \in \psi(i) \end{aligned} \tag{7}$$

where $\hat{h}^* \in \mathbb{R}^{d_z}$ denotes *semantic encoding* with corresponding input datasets with a linear layer to transform as $\hat{h}^{(s;q)} = PLM(\mathcal{S}^a_s; \mathcal{Q}_s)W_s + b_s$, learnable parameters $W_s \in \mathbb{R}^{d_m \times d_z}, b_s \in \mathbb{R}^{d_z}$, r^*_i is the mean value of the token representations within i-th entity type in the corresponding set, and FFN$(\cdot)$ is the feed-forward neural network with the ReLU activation function. The hyperparameter γ modulates entity information with support and query sets. Thus, we will obtain domain-oriented prototype $Z_s \in \mathbb{R}^{\epsilon \times d_z}$.

Consequently, given a sentence $X = \{x_1, \cdots, x_n\}$, the probability of entity types of X can be expressed as follows:

$$\mathtt{P}_\theta(\tilde{\boldsymbol{e}}|X, Z_s) = \sum_d ([H] W_e + b_e) \odot (Z^a_s)^T, \tag{8}$$

herein, we employ an uncertainty-augmentation strategy through the random injection of diverse noise to Z_s, yielding a modified entity representation to $Z^a_s \in \mathbb{R}^{n_e \times \epsilon \times d_z}$, n_e denotes the sample size for augmentation, thereby $[H] \in \mathbb{R}^{n_e \times |X| \times d_z}$ refers to the expanding output of PLM, $\tilde{\boldsymbol{e}} \in \mathbb{R}^{n_e \times |X| \times \epsilon}$ is the prediction of entity types of sequence X, $W_e \in \mathbb{R}^{d_m \times d_z}, b_e \in \mathbb{R}^{d_z}$ are learnable parameters, $\sum_d$ denotes the addition performed along the last dimension.

Specifically, we further augment prototypes to calibrate entity typing, which can be expressed as:

$$\boldsymbol{e}^a = [\tilde{\boldsymbol{e}}; \boldsymbol{\pi}(\mathcal{P}, \bar{H}^a)], \tag{9}$$

where $\boldsymbol{e}^a \in \mathbb{R}^{n_e \times |X| \times 2\epsilon}$, and the prediction is defined as:

$$\begin{aligned} \mathtt{P}_\theta(\hat{\boldsymbol{e}}|X, \mathcal{P}, Z_s) &= \big((A \odot g(B))W_{\hat{e}} + b_{\hat{e}}\big) + \tilde{\boldsymbol{e}}, \\ \text{where } A &= \boldsymbol{e}^a W_{a1} + b_{a1},\ B = \boldsymbol{e}^a W_{a2} + b_{a2}. \end{aligned} \tag{10}$$

Herein, $\{W_{a1}, W_{a2}\} \in \mathbb{R}^{2\epsilon \times 2\epsilon}, W_{\hat{e}} \in \mathbb{R}^{2\epsilon \times \epsilon}$ are learnable parameters, $\hat{e} \in \mathbb{R}^{n_e \times |X| \times \epsilon}$ is the finally prediction of entity types. Thereby, the entity type loss can be calculated by:

$$\mathcal{L}_e = \sum [\boldsymbol{e}] \log \mathtt{P}_\theta(\hat{\boldsymbol{e}}|(X, \mathcal{P}, Z_s)), \tag{11}$$

where $[e] \in \mathbb{R}^{n_e \times |X| \times \epsilon}$ denotes expanding golden label for matching the shape of prediction $\hat{\boldsymbol{e}}$.

Training Criterion. Given source domain datasets $D_s = \{S_s, Q_s\}$, the corresponding sample as $\boldsymbol{o} = (X, \boldsymbol{y}, \boldsymbol{c}, \boldsymbol{e})$, where $\boldsymbol{c} = \{B, O, I, E, S\}$ refers to namely position tags, $\boldsymbol{e} = \{e_1, \cdots, e_\epsilon\}$ represents entity types, and $\boldsymbol{y}$ is the ground-truth of jointy-label, such as "`B-person`". We combine the results of span detection and entity classification of our model as the joint probability distribution of $\mathtt{P}_\theta(\hat{\boldsymbol{y}}, \hat{\boldsymbol{c}}, \hat{\boldsymbol{e}}) = \mathtt{P}_\theta(\hat{\boldsymbol{c}}|X, \mathcal{P}) \cdot \mathtt{P}_\theta(\hat{\boldsymbol{e}}|X, \mathcal{P}, Z_s)$. Thus the loss function of joint prediction can be computed as follows:

$$\mathcal{L}_J = \sum_{\boldsymbol{o} \in D_s} \mathbb{E}[-\log \mathtt{P}_\theta(\hat{\boldsymbol{y}}, \hat{\boldsymbol{c}}, \hat{\boldsymbol{e}}, |X, Z_s, \mathcal{P})]. \tag{12}$$

Ultimately, the overall loss can be linearly combined as:

$$\mathcal{L} = \mathcal{L}_s + \mathcal{L}_e + \mathcal{L}_J + \alpha \mathcal{L}_p. \tag{13}$$

3.4 Target Domain Adapting.

Since we do not have access to the *query set* when the model is adapted to the target domain, we follow the previous work (Ma et al. 2022) to fine-tune the model with a few examples (K-shot) in the target domain. After adapting AAL to the target domain by fine-tuning, we use $\mathtt{P}_\theta(\hat{\boldsymbol{y}}, \hat{\boldsymbol{c}}, \hat{\boldsymbol{e}}|X, \mathcal{P}, Z_t)$ to predict the sentence X in the query set of the target domain. Notably, we directly employ $\mathtt{P}_\theta(Z_t|\mathcal{S}_t^a)$ (cf. Eq.(6)) as the entity representation that is conducted on the support set of the target domain.

4 Experiments

4.1 Experimental Setup

Datasets. To align with previous studies, we conduct experiments on the following datasets: (1) *SNIPS* (Coucke et al. 2018): It has 7 domains with different label sets and a small number of samples, with a relatively even number of samples per domain per label set, which makes it easy to simulate a small number of samples. (2) *Cross-Dataset* (Hou et al. 2020): It is constructed from datasets from four different domains: CoNLL-2003 (Tjong Kim Sang 2002), GUM (Zeldes 2017), WNUT-2017 (Derczynski et al. 2017), and Ontonotes (Pradhan et al. 2013). We take two of the datasets as the training set, one as the validation set, and one as the test set. (3) *Adversarial examples.* Following (Xue et al. 2024), we use the textual adversarial attack algorithm Bert-Attack (Li et al. 2020) to perform textual adversarial attacks on samples in the support set $\mathcal{S}_t$ and query set $\mathcal{Q}_t$ of the target domain. The cross-domain migration learning capability is demonstrated by the performance of the Few-shot NER model in the query set $\mathcal{Q}_t$ of the target domain.

Baselines. Our baselines include *TransferBERT*, *M-Network*, and *L-TapNet+CDT* used in (Hou et al. 2020), *ProtoBERT* (Fritzler, Logacheva, and Kretov 2019), *ESD* (Wang et al. 2021), *MANNER* (Fang et al. 2023). Recently related studies *SimBERT* (Hou et al. 2020), *CONTaiNER* (Das et al. 2022), *Decp-MetaNER* (Ma et al. 2022), *SpanProto* (Wang et al. 2022a), *MSDP* (Dong et al. 2023), and *BDCP* (Xue et al. 2024) as additional baselines for cross-dataset validation. We also evaluate five strong baselines, including BDCP and MANNER, under the uncertainty condition using the authors' original code.

Evaluation. To align with previous efforts such as (Hou et al. 2020), we evaluate all methods using *F1-score* with the test episode and average all scores as results. Notably, we used five different random seeds in each experiment and reported the mean and standard deviation of these results.

Implementation Details. We exploit the Uncased-Bert base model as the PLM. $\{d_m, d_z, n_e, \alpha, \gamma\}$ and dropout are set to $\{768, 128, 5, 0.5, 0.5, 0.1\}$. We employ the AdamW optimizer for AAL, accelerated on an NVIDIA A100 GPU. For reproduction, the source code is released at https://github.com/swufe-NiceLab-GeoText/AAL.git.

4.2 Performance Evaluation

Table 1 - Table 4 report the performance of various methods under certain and uncertain conditions. The best gain is highlighted in **bold** while the second best is underline. The findings are summarized as follows.

Performance on Certainty Condition. Table 1 and Table 2 present the performance comparison of our AAL against baselines on *SNIPS* and *Cross-Dataset* under certain conditions. AAL achieves an average enhancement of 6.27% and 3.04% in overall results for the 1-shot and 5-shot scenarios, outperforming the robust baseline MANNER. Specifically, significant F1-score improvements of 12.68% and 8.05% are noted on *SNIPS* (Se, 1-shot) and *Cross-Dataset* (Wiki, 1-shot), respectively, indicating the efficacy of AAL in adapting to novel domains with minimal labeled data. This underscores AAL's strength in leveraging target domain support samples, particularly in low-resource scenarios, affirming its adaptive and resource-efficient nature.

Performance on Uncertainty Condition. Table 3 and Table 4 report the performance of our AAL alongside baselines on *SNIPS* and *Cross-Dataset* following the application of Bert-Attack adversarial algorithm to the target domain data. The findings reveal a substantial decline in the performance of baseline models under adversarial conditions, highlighting the vulnerability of existing few-shot NER. Specifically, MANNER, a leading model, experiences an average F1 score reduction of 7.94% and 9.07% on *SNIPS* and *Cross-Dataset*, respectively. This susceptibility stems from their reliance on specific word features, making them prone to interference from adversarially crafted words. In contrast, our AAL, leveraging oriented-inducing prompt learning, enhances resilience by associating entity types with words vulnerable to such attacks, thereby reducing adversarial interference. Moreover, we augment model robustness by incorporating unaltered support set prototypes, further contributing to defense against adversarial attacks.

	Models	We	Mu	Pl	Bo	Se	Re	Cr
1-SHOT	TransferBERT	55.82±2.75	38.01±1.74	45.65±2.02	31.63±5.32	21.96±3.98	41.79±3.81	38.53±7.42
	M-Network	21.74±4.60	10.68±1.07	39.71±1.81	58.15±0.68	24.21±1.20	32.88±0.64	69.66±1.68
	L-TapNet+CDT	71.53±4.04	60.56±0.77	66.27±2.71	84.54±1.08	76.27±1.72	70.79±1.60	62.89±1.88
	ProtoBERT	46.72±1.03	40.07±0.48	50.78±2.09	68.73±1.87	60.81±1.70	55.58±3.56	67.67±1.16
	ESD	78.25±1.50	54.74±1.02	71.15±1.55	71.45±1.38	67.85±0.75	71.52±0.98	78.14±1.46
	MANNER	75.41±0.52	60.93±0.14	66.65±0.70	72.80±0.32	68.35±1.00	74.99±0.11	59.20±2.64
	Ours	**80.87±0.35**	**65.50±0.97**	**78.06±0.77**	**88.41±0.35**	**81.03±0.12**	**81.02±0.24**	**79.04±0.75**
5-SHOT	TransferBERT	59.41±0.30	42.00±2.83	46.07±4.32	20.74±3.36	28.20±0.29	67.75±1.28	58.61±3.67
	Matching Network	36.67±3.64	33.67±6.12	52.62±2.84	69.09±2.36	38.42±4.06	33.28±2.99	72.10±1.48
	L-TapNet+CDT	71.64±3.62	67.16±2.97	75.88±1.51	84.38±2.81	82.58±2.12	70.05±1.61	73.41±2.61
	ProtoBERT	67.82±4.11	55.99±2.24	46.02±3.19	72.17±1.75	73.59±1.60	60.18±6.96	66.89±2.88
	ESD	84.50±1.06	66.61±2.00	79.69±1.35	82.57±1.37	82.22±0.81	80.44±0.80	**81.13±1.84**
	MANNER	86.53±0.21	74.93±0.55	80.24±0.22	83.91±0.57	83.78±0.56	84.72±0.48	72.07±0.34
	Ours	**87.00±1.06**	**75.36±0.11**	**85.88±0.32**	**90.22±0.57**	**87.12±0.21**	**85.09±0.37**	80.24±0.39

Table 1: Overall performance on the SNIPS.

Models	1-shot				5-shot			
	News	Wiki	Social	Mixed	News	Wiki	Social	Mixed
TransferBERT	4.75±1.42	0.57±0.32	2.71±0.72	3.46±0.54	15.36±2.81	3.62±0.57	11.08±0.57	35.49±7.60
M-Network	19.50±0.35	4.73±0.16	17.23±2.75	15.06±1.61	19.85±0.74	5.58±0.23	6.61±1.75	8.08±0.47
L-TapNet+CDT	44.30±3.15	12.04±0.65	20.80±1.06	15.17±1.25	45.35±2.67	11.65±2.34	23.30±2.80	20.95±2.81
ProtoBERT	32.49±2.01	3.89±0.24	10.68±1.40	6.67±0.46	50.06±1.57	9.54±0.44	17.26±2.65	13.59±1.61
SimBERT	19.22±0.00	6.91±0.00	5.18±0.00	13.99±0.00	32.01±0.00	10.63±0.00	8.20±0.00	21.12±0.00
CONTaiNER	34.09±0.94	10.81±0.45	16.45±0.92	32.96±0.91	58.63±1.56	24.31±0.66	27.50±0.58	48.62±2.81
Decp-MetaNER	46.09±0.44	17.54±0.98	25.14±0.24	34.13±0.92	58.18±0.87	31.36±0.91	31.02±1.28	45.55±0.90
SpanProto	47.71±0.51	20.16±0.80	30.19±0.94	37.91±0.79	61.61±1.03	43.75±0.50	31.37±0.94	49.04±0.93
MSDP	49.14±0.52	21.88±0.29	30.10±0.56	38.05±0.88	63.98±0.80	36.53±0.81	35.61±0.72	49.99±0.95
MANNER	49.06±0.48	23.17±0.20	28.54±0.69	43.61±0.48	64.84±0.51	40.86±0.96	35.86±1.28	58.37±0.62
BDCP	43.88±0.00	11.85±0.00	25.90±0.00	28.16±0.00	58.76±0.00	32.17±0.00	31.84±0.00	46.29±0.00
Ours	**49.21±0.89**	**31.22±0.94**	**32.05±1.04**	**46.51±0.35**	**66.18±0.97**	**44.19±0.45**	**39.91±0.24**	**59.06±0.28**

Table 2: Overall performance on the Cross-Dataset.

Decoupling Study. We explore the effectiveness of different components of AAL by comparing with its four decoupling variants on *Cross-Dataset*: *AAL w/o DO proto* removes the domain-oriented prototype residual connections; *AAL w/o base proto* removes the prototype module; *AAL w/o aug* removes the context-enhancing prompt in augmentation learning; *AAL w/o T & aug* removes the whole augmentation learning. The results of the decoupling experiments are shown in Fig. 4. Overall, AAL outperforms its variant models, validating the positive contribution of various components. A significant performance decline is evident in *AAL w/o DO proto*, underscoring the efficacy of domain-oriented prototypes and showcasing that incorporating uncertainty injection enhances AAL's stability. The superiority of AAL over *AAL w/o proto* confirms the efficacy of capturing prototypes representing common knowledge among entities. Moreover, the degradation in performance when the AAL removes data augmentation (*AAL w/o aug* and *AAL w/o T & aug*) further validates our statement, emphasizing the crucial role of data enhancement.

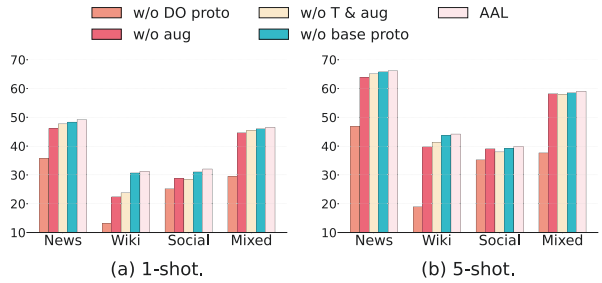

Figure 4: Decopling study.

Entity Representation Visualization. To illustrate the efficacy of AAL on entity representation, we employ the test set in the 1-shot scenario from *SNIPS Bo* and utilize the t-SNE (Van der Maaten and Hinton 2008) toolkit to visualize them. As depicted in Fig. 5, these embeddings demonstrate clustering around type prototype regions for distinct entity classes. Notably, even after exposure to textual adversarial attacks, the majority of span representations maintain this clustering pattern, indicating the robustness of our proposed AAL in discerning and assigning entity types accurately in both original and adversarial conditions.

Interpretability of Dependency Learning. We randomly sampled instances from the *query set* of *SNIPS Bo* under a 1-shot learning scenario, alongside its adversarial counterparts, to depict semantic dependencies in Fig. 6. The alignment of AAL's correct entity type prediction for both original and attack tokens highlights its adversarial awareness. Examining attention heatmaps reveals that despite substituting the entity "book" with "title", the token dependencies exhibit minimal shifts in semantic correlation, highlighting the cooperation of augmentation and prototype learning. Notably, the similarity in heatmap color distribution underscores AAL's robustness in capturing token semantics, further emphasizing its resilience to adversarial attacks.

	Models	We	Mu	Pl	Bo	Se	Re	Cr
1-SHOT	L-TapNet+CDT	61.13±0.21	52.37±0.59	60.08±4.18	78.15±0.92	61.90±2.03	66.76±0.75	55.92±2.00
	ESD	64.48±0.94	43.65±0.19	60.81±0.33	62.26±0.68	49.32±0.75	57.17±1.03	48.87±0.46
	Decp-MetaNER	29.74±0.13	24.54±0.08	47.48±0.70	51.37±0.32	28.17±0.29	42.37±0.75	17.71±0.79
	MANNER	62.11±1.85	50.45±0.19	62.17±0.23	72.62±0.21	53.32±0.18	62.17±0.14	52.32±0.35
	BDCP	29.13±0.14	23.88±0.92	51.34±0.23	51.05±0.81	29.06±0.48	41.87±0.11	18.00±0.15
	Ours	**67.37±0.16**	**52.52±0.74**	**69.69±1.14**	**79.21±0.19**	**66.47±1.41**	**69.25±0.43**	**57.32±1.50**
5-SHOT	L-TapNet+CDT	65.43±3.47	62.37±0.16	63.65±1.03	77.49±2.66	70.78±2.64	63.92±2.11	58.61±1.94
	ESD	73.36±0.24	55.00±0.48	68.00±0.18	70.78±0.86	64.84±0.29	68.90±0.44	58.87±0.31
	Decp-MetaNER	37.56±0.40	32.21±0.24	61.60±0.43	56.07±0.24	40.48±0.32	50.99±0.35	22.20±0.14
	MANNER	75.51±0.15	62.31±0.53	75.18±0.33	79.42±0.81	74.87±0.28	74.05±0.58	60.95±0.52
	BDCP	37.62±0.26	31.64±0.21	62.31±0.14	55.46±0.21	40.48±0.66	51.28±0.69	22.33±0.39
	Ours	**76.20±0.99**	**62.76±0.55**	**76.02±1.03**	**80.47±0.53**	**75.32±0.78**	**75.17±0.40**	**61.31±0.35**

Table 3: Overall performance on the attacked SNIPS.

Models	1-shot				5-shot			
	News	Wiki	Social	Mixed	News	Wiki	Social	Mixed
L-TapNet+CDT	3.37±0.14	4.78±1.96	19.47±0.76	0.19±0.05	2.66±0.15	0.43±0.05	0.96±0.25	0.29±0.02
ESD	25.57±0.36	6.54±0.33	13.28±0.35	17.52±0.09	27.90±1.47	9.96±1.57	16.31±0.59	20.83±2.07
Decp-MetaNER	34.89±0.33	9.05±0.12	16.16±0.32	21.74±0.93	46.62±0.28	18.61±2.51	18.54±0.75	29.90±0.91
MANNER	38.57±0.17	20.11±0.59	18.46±0.40	35.59±0.69	53.51±0.66	36.01±0.44	22.77±0.53	46.72±1.21
BDCP	34.95±0.99	12.64±0.29	16.18±0.44	25.62±0.68	45.52±0.28	24.68±0.33	18.36±0.53	32.78±1.39
Ours	**41.25±0.36**	**26.51±0.25**	**20.55±0.37**	**37.39±0.35**	**55.20±0.47**	**37.66±0.49**	**23.99±0.52**	**47.42±0.46**

Table 4: Overall performance on the attacked Cross-Dataset.

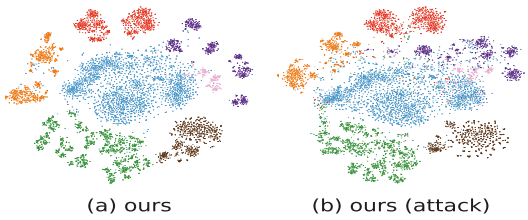

Figure 5: Visualization of entity representation.

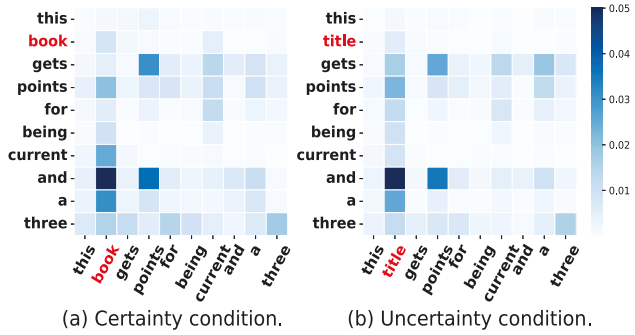

Figure 6: Attention heatmap for two conditions.

5 Related Work

Few-shot NER. Recent arts have introduced comprehensive benchmarks under the unified N-way K-shot framework (Ding et al. 2021; Hou et al. 2020). Part of these works center on developing models by common knowledge learning (Huang et al. 2021; Tong et al. 2021; Ji et al. 2022), leveraging prototype networks (Snell, Swersky, and Zemel 2017; Wang et al. 2022a) or Model-Agnostic Meta-Learning (Finn, Abbeel, and Levine 2017) to adapt to low-resource scenarios via internal support set adjustments. Alternatively, transfer learning involves pre-training feature extractors on a resource-rich source domain before adapting to the target domain (Yang and Katiyar 2020). An emerging trend is two-stage models (Ziyadi et al. 2020; Ma et al. 2022), which improve localization and classification by addressing span detection and entity typing sequentially.

Adversarial Robustness on Texts. Extensive research has been dedicated to devising textual adversarial attacks, including synonym substitution, word embedding perturbation, and phrase-level manipulations that preserve sentence coherence but cause erroneous predictions (Zeng et al. 2023; Liu et al. 2022). These endeavors highlight the susceptibility of models to adversarial attacks. To evaluate few-shot NER resilience, Xue et al. employed Bert-attack, mitigating adversarial disturbances through interference minimization. Zeng et al. enhanced robustness by creating input variants through random word masking, while Lin et al. utilized Wikidata entities for contextual substitutions leveraging a pre-trained language model. Despite this progress, research on cross-domain transfer learning's adversarial robustness remains limited, especially on cross-domain transfer in the context of few-shot NER.

6 Conclusion

This work presents AAL for few-shot NER task, addressing challenges through two key strategies. First, it employs prompt-based augmentation learning to enhance data diversity and semantic understanding. Second, it introduces augmented prototype learning, leveraging general entity knowledge and domain-oriented adaptations, further refined by uncertainty-guided adversarial attacks for prototype calibration. Experimental results demonstrate AAL's superiority across conditions, highlighting its potential for advancing augmentation learning, semantic dependency modeling, and broader few-shot NLP tasks in open-world settings.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant No.62102326 and No.62376228, Sichuan Science and Technology Program under Grant No.2023ZYD0145, the Natural Science Foundation of Sichuan Province under Grant No.2023NSFSC1411 and No.25QNJJ0627.

References

Chiu, J. P.; and Nichols, E. 2016. Named entity recognition with bidirectional LSTM-CNNs. *Transactions of the association for computational linguistics*, 4: 357–370.

Coucke, A.; Saade, A.; Ball, A.; Bluche, T.; Caulier, A.; Leroy, D.; Doumouro, C.; Gisselbrecht, T.; Caltagirone, F.; Lavril, T.; et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*.

Das, S. S. S.; Katiyar, A.; Passonneau, R.; and Zhang, R. 2022. CONTaiNER: Few-Shot Named Entity Recognition via Contrastive Learning. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 6338–6353. Dublin, Ireland: Association for Computational Linguistics.

Derczynski, L.; Nichols, E.; Van Erp, M.; and Limsopatham, N. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, 140–147.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ding, N.; Xu, G.; Chen, Y.; Wang, X.; Han, X.; Xie, P.; Zheng, H.-T.; and Liu, Z. 2021. Few-nerd: A few-shot named entity recognition dataset. *arXiv preprint arXiv:2105.07464*.

Dong, G.; Wang, Z.; Zhao, J.; Zhao, G.; Guo, D.; Fu, D.; Hui, T.; Zeng, C.; He, K.; Li, X.; et al. 2023. A multi-task semantic decomposition framework with task-specific pre-training for few-shot ner. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 430–440.

Fang, J.; Wang, X.; Meng, Z.; Xie, P.; Huang, F.; and Jiang, Y. 2023. MANNER: A Variational Memory-Augmented Model for Cross Domain Few-Shot Named Entity Recognition. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 4261–4276. Toronto, Canada: Association for Computational Linguistics.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, 1126–1135. PMLR.

Fritzler, A.; Logacheva, V.; and Kretov, M. 2019. Few-shot classification in named entity recognition task. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, 993–1000.

Fuglede, B.; and Topsoe, F. 2004. Jensen-Shannon divergence and Hilbert space embedding. In *International symposium onInformation theory, 2004. ISIT 2004. Proceedings.*, 31. IEEE.

Hou, Y.; Che, W.; Lai, Y.; Zhou, Z.; Liu, Y.; Liu, H.; and Liu, T. 2020. Few-shot slot tagging with collapsed dependency transfer and label-enhanced task-adaptive projection network. *arXiv preprint arXiv:2006.05702*.

Huang, J.; Li, C.; Subudhi, K.; Jose, D.; Balakrishnan, S.; Chen, W.; Peng, B.; Gao, J.; and Han, J. 2021. Few-shot named entity recognition: An empirical baseline study. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, 10408–10423.

Huisman, M.; Van Rijn, J. N.; and Plaat, A. 2021. A survey of deep meta-learning. *Artificial Intelligence Review*, 54(6): 4483–4541.

Ji, B.; Li, S.; Gan, S.; Yu, J.; Ma, J.; and Liu, H. 2022. Few-shot named entity recognition with entity-level prototypical network enhanced by dispersedly distributed prototypes. *arXiv preprint arXiv:2208.08023*.

Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; and Dyer, C. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Lee, D.-H.; Kadakia, A.; Tan, K.; Agarwal, M.; Feng, X.; Shibuya, T.; Mitani, R.; Sekiya, T.; Pujara, J.; and Ren, X. 2021. Good examples make a faster learner: Simple demonstration-based learning for low-resource NER. *arXiv preprint arXiv:2110.08454*.

Li, L.; Ma, R.; Guo, Q.; Xue, X.; and Qiu, X. 2020. Bert-attack: Adversarial attack against bert using bert. *arXiv preprint arXiv:2004.09984*.

Lin, B. Y.; Gao, W.; Yan, J.; Moreno, R.; and Ren, X. 2021. RockNER: A simple method to create adversarial examples for evaluating the robustness of named entity recognition models. *arXiv preprint arXiv:2109.05620*.

Liu, Q.; Zheng, R.; Rong, B.; Liu, J.; Liu, Z.; Cheng, Z.; Qiao, L.; Gui, T.; Zhang, Q.; and Huang, X.-J. 2022. Flooding-X: Improving BERT's resistance to adversarial attacks via loss-restricted fine-tuning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5634–5644.

Ma, T.; Jiang, H.; Wu, Q.; Zhao, T.; and Lin, C.-Y. 2022. Decomposed meta-learning for few-shot named entity recognition. *arXiv preprint arXiv:2204.05751*.

Ma, X.; and Hovy, E. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Pradhan, S.; Moschitti, A.; Xue, N.; Ng, H. T.; Björkelund, A.; Uryupina, O.; Zhang, Y.; and Zhong, Z. 2013. Towards robust linguistic analysis using ontonotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, 143–152.

Ren, S.; Deng, Y.; He, K.; and Che, W. 2019. Generating Natural Language Adversarial Examples through Probability Weighted Word Saliency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1085–1097. Association for Computational Linguistics.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30.

Tjong Kim Sang, E. F. 2002. Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Tong, M.; Wang, S.; Xu, B.; Cao, Y.; Liu, M.; Hou, L.; and Li, J. 2021. Learning from miscellaneous other-class words for few-shot named entity recognition. *arXiv preprint arXiv:2106.15167*.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Wang, J.; Han, C.; Wang, C.; Tan, C.; Qiu, M.; Huang, S.; Huang, J.; and Gao, M. 2022a. Spanproto: A two-stage span-based prototypical network for few-shot named entity recognition. *arXiv preprint arXiv:2210.09049*.

Wang, P.; Xu, R.; Liu, T.; Zhou, Q.; Cao, Y.; Chang, B.; and Sui, Z. 2021. An enhanced span-based decomposition method for few-shot sequence labeling. *arXiv preprint arXiv:2109.13023*.

Wang, X.; Dou, S.; Xiong, L.; Zou, Y.; Zhang, Q.; Gui, T.; Qiao, L.; Cheng, Z.; and Huang, X. 2022b. MINER: Improving out-of-vocabulary named entity recognition from an information theoretic perspective. *arXiv preprint arXiv:2204.04391*.

Xue, X.; Zhang, C.; Xu, T.; and Niu, Z. 2024. Robust Few-Shot Named Entity Recognition with Boundary Discrimination and Correlation Purification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 17, 19341–19349.

Yang, Y.; and Katiyar, A. 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. *arXiv preprint arXiv:2010.02405*.

Zeldes, A. 2017. The GUM corpus: Creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3): 581–612.

Zeng, J.; Xu, J.; Zheng, X.; and Huang, X. 2023. Certified robustness to text adversarial attacks by randomized [mask]. *Computational Linguistics*, 49(2): 395–427.

Ziyadi, M.; Sun, Y.; Goswami, A.; Huang, J.; and Chen, W. 2020. Example-based named entity recognition. *arXiv preprint arXiv:2008.10570*.