Incremental Sequence Labeling: A Tale of Two Shifts

Anonymous ACL submission

Abstract

The incremental sequence labeling task involves continuously learning new classes over time while retaining knowledge of the previous ones. Our investigation identifies two significant semantic shifts: E2O (where the model mislabels an old entity as a non-entity) and O2E (where the model labels a non-entity or old entity as a new entity). Previous research has predominantly focused on addressing the E2O problem, neglecting the O2E issue. This negligence results in a model bias towards classifying new data samples as belonging to the new class during the learning process. To address these challenges, we propose a novel framework, **I**ncremental **S**equential **L**abeling without **S**emantic **S**hifts (IS3). Motivated by the identified semantic shifts (E2O and O2E), IS3 aims to mitigate catastrophic forgetting in models. As for the E2O problem, we use knowledge distillation to maintain the model's discriminative ability for old entities. Simultaneously, to tackle the O2E problem, we alleviate the model's bias towards new entities through debiased loss and optimization levels. Our experimental evaluation, conducted on three datasets with various incremental settings, demonstrates the superior performance of IS3 compared to the previous state-of-the-art method by a significant margin. [1].

1 Introduction

The conventional sequence labeling task typically involves categorizing data into a predetermined set of fixed categories (Lample et al., 2016). However, this approach may need to be revised in natural language processing scenarios, such as the named entity recognition task, where new types of entities continuously emerge. Adapting a fixed set of categories becomes challenging when faced with the dynamic nature of new entity classification requirements. Consequently, continuous model updates are essential to accommodating evolving entity types. Previous studies have advocated for adopting continual learning (Parisi et al., 2019; Monaikul et al., 2021), also known as lifelong learning or incremental learning. Continual learning is a paradigm designed to train models capable of adapting to the continual addition of new categories in real-world scenarios while ensuring that knowledge of old categories is retained. For instance, voice assistants like Siri frequently encounter new event types, such as pandemics, to better understand users' latest intentions and provide information on health protection (Monaikul et al., 2021).

Due to constraints imposed by storage limitations and privacy concerns, there exists a shortage of training data about the older categories (He and Zhu, 2022). Additionally, the manual relabeling of all categories within the new training dataset would incur substantial costs and time investment (De Lange et al., 2021; Bang et al., 2021). Consequently, the model undergoes continuous updates

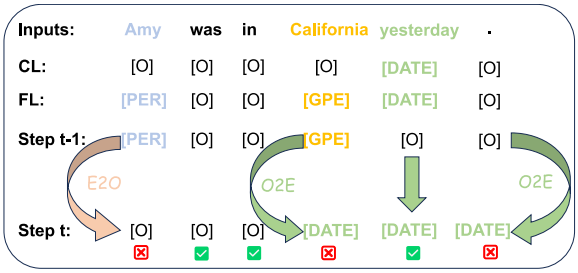


Figure 1: A sample shows two shifts in incremental sequence labeling. E2O denotes the semantic shift of an old entity (such as [PER]) to a non-entity ([O]), and O2E denotes the semantic shift of a non-entity ([O]) or an old entity(such as [GPE]) to a new entity (such as [DATE]). **Inputs** means input sentence. **CL** means *current ground-truth label* at step t. **FL** means the *full ground-truth label* for all steps. **Step** $t-1$ **and Step** t means the predictions in step $t-1$ and t.

[1]Anonymized URL: https://anonymous.4open.science/r/IS3-7CCA/

using a freshly acquired dataset comprising the new categories. As depicted in Fig.1, the model undergoes training based on the *current ground-truth label* and undergoes testing using the *full ground-truth label*.

The incremental sequence labeling task faces a significant challenge known as the catastrophic forgetting problem, as extensively discussed in previous studies (McCloskey and Cohen, 1989; Robins, 1995; Goodfellow et al., 2013; Kirkpatrick et al., 2017). This issue manifests as semantic shifts, leading to a decrease in the discriminative power of entity classes (Zhang et al., 2023b; Ma et al., 2023). In this paper, we decompose the problem into two primary semantic shifts in the incremental sequence task: E2O and O2E. The first semantic shift, E2O, arises from the presence of non-entities, potential old entities (mislabelled as non-entities), and new entities in the new dataset. Progress has been made in addressing E2O through methods falling into three categories: (1) Methods based on knowledge distillation: For instance, RDP proposes a knowledge distillation loss incorporating inter-task relations (Zhang et al., 2023b). At the same time, CFPD introduces a pooled feature distillation loss to alleviate catastrophic forgetting (Zhang et al., 2023a). (2) Methods based on pseudo-labels: OCILNER utilizes class prototypes to label new data (Ma et al., 2023), and CPFD employs old models to label predictions of new data. (3) Methods based on freezing models: Examples include ICE (Liu and Huang, 2023), which freezes the backbone model and old classifiers to maintain the stability of the old classes at the expense of learning new classes.

Existing methods primarily focus on addressing the E2O shift, neglecting the bias towards the emergence of new classes and the consequential second semantic shift, O2E. To address both semantic shifts, we propose a novel framework called **I**ncremental **S**equential **L**abeling without **S**emantic **S**hifts (IS3). IS3 consists of two key components: First, we apply the knowledge distillation method to tackle the E2O shift. Second, we address the O2E shift on two fronts. At the loss function level, we introduce a debiased cross-entropy loss function to mitigate the model's impact on old class distributions, reducing its inclination towards new entities. At the optimization level, we introduce a prototype-based approach to balance the imbalanced contributions of old and new entities during batch updates, which aims to increase the involvement of old entities in the optimization process. Importantly, IS3 adopts a storage-efficient approach, maintaining only one prototype per class with minimal storage costs. Class feature centers serve as prototypes, ensuring no direct correspondence to actual sample information and mitigating privacy leakage concerns.

The contribution of our work can be summarized as follows:

- We propose a novel perspective on the semantic shift problem in incremental sequence labeling task by categorizing the catastrophic forgetting problem into E2O and O2E.
- We propose a novel framework, **I**ncremental **S**equential Labeling without **S**emantic **S**hifts (IS3), to solve the two semantic shifts simultaneously.
- We conduct experiments under nine CIL settings on three datasets, and our method outperforms the previous state-of-the-art methods.

2 Related Work

Incremental Learning The model continually acquires new tasks intending to achieve optimal performance on tasks previously learned (Gepperth and Hammer, 2016; Wu et al., 2019; van de Ven et al., 2022). There are three main categories of current incremental learning methods: regularization-based, rehearsal-based, and architecture-based. Regularization-based methods place constraints on model weights (Kirkpatrick et al., 2017; Zenke et al., 2017), representations of intermediate layer features (Hou et al., 2019; Douillard et al., 2020), and output probabilities (Li and Hoiem, 2017). Rehearsal-based methods overcome forgetting by saving some of the data containing the old classes for learning with the new classes (Lopez-Paz and Ranzato, 2017; Shin et al., 2017). Alternatively, architecture-based approaches involve dynamically expanding the network structure to allow for more data as new classes are added (Hou et al., 2018; Yan et al., 2021).

Incremental Sequence Labeling The traditional sequence labeling task is the task of labeling each token of a one-dimensional linear input sequence, which requires each token to be categorized according to its contextual content(Rei et al., 2016; Akbik et al., 2018). However, previous methods can only recognize classes in a fixed set. Therefore, continuous learning paradigms are introduced

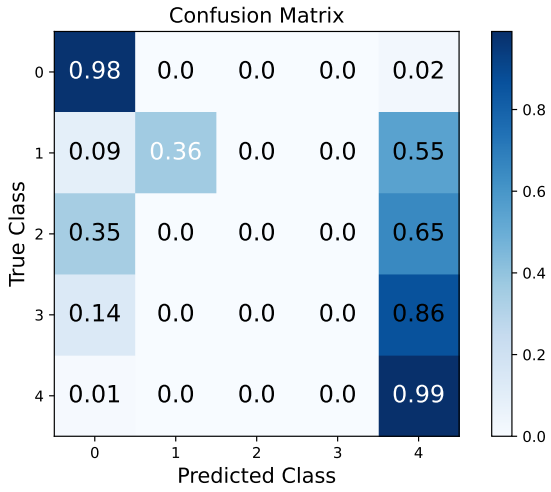


Figure 2: Confusion Matrix of the ExtendNER method in Task 4. It indicates that the model predicts the old entities as new entities with high probability and predicts the old entity as non-entity, with severe O2E semantic shift and E2O semantic shift.

in sequence labeling tasks, including incremental named entities (Monaikul et al., 2021; Zheng et al., 2022; Zhang et al., 2023a), incremental event detection (Cao et al., 2020; Yu et al., 2021), and so on.

Methods for incremental sequence labeling tasks can be categorized into distillation-based, rehearsal-based, and other approaches. Distillation-based methods encompass ExtendNER (Monaikul et al., 2021), which is the pioneer in applying knowledge distillation to incremental sequence labeling task, RDP (Zhang et al., 2023b) with a relational distillation approach, and CPFD (Zhang et al., 2023a) utilizing pooled features distillation loss. CFNER (Zheng et al., 2022) introduces a causal framework for extracting new causal effects in entities and non-entities. Rehearsal-based approaches include KCN (Cao et al., 2020) and KD+R+K (Yu et al., 2021), both employing rehearsal samples to address class imbalance and catastrophic forgetting in incremental event detection. L&R (Xia et al., 2022) proposes a learn-and-review framework by training a new backbone model and a generative model simultaneously, generating synthetic samples of the old class to be trained with new samples. OCILNER (Ma et al., 2023) uses rehearsal samples to compute class feature centers as class prototypes, generates an entity-oriented feature space through comparative learning, and annotates new data with pseudo-labels using class prototypes. Other methods encompass span-based and freezing model-based approaches, among others.

The mentioned methodologies primarily focus on preserving the existing knowledge of the model and do not explicitly consider the implications of transitioning from non-entity to entity semantics. In contrast, our proposed method, IS3, provides a fresh perspective on model forgetting by addressing the model's inclination towards new classes during task adaptation. IS3 not only addresses issues related to model mislabeling, indirectly mitigating the problem of semantic migration from entity to non-entity, but also handles the challenge of semantic migration from non-entity to entity. By recognizing and addressing the model's bias towards new classes during adaptation, our approach offers a comprehensive solution to the dynamic challenges associated with transitioning between different semantic categories.

3 Problem Formulation

Formally, the objective of incremental sequence labeling is to acquire knowledge through a series of tasks $\mathcal{T} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_N\}$. Each task contains its own dataset $\mathcal{D}_t = \{(x^i, y^i) | y^i \in \mathcal{Y}_t\}$ where (x^i, y^i) is a pair formed by the input token sentence and the label corresponding to each token in the sentence and $\mathcal{Y}_t$ stands for the current label set. Notably, y^i only labels the token corresponding to the current task t, and the other tokens are labeled as O class (potential old entities $\mathcal{Y}_{1:t}$, and unseen entities $\mathcal{Y}_{t+1:N}$). At task t ($t > 1$), the new model $\mathcal{M}_t$ learns only from the new dataset and is expected to perform well on the learned classes $\bigcup_{i=1}^{t} \mathcal{Y}_i$.

4 Method

In this section, we systematically address the catastrophic forgetting problem by decomposing it into two distinct semantic shift challenges (Section 4.1). Subsequently, we present a comprehensive framework designed to address these semantic shifts individually, focusing on E2O in Section 4.2 and O2E in Section 4.3. The overarching goal is to effectively mitigate the catastrophic forgetting problem, as illustrated in Fig.4.

4.1 Two semantic shift problems

In the incremental sequence labeling task, semantic shift can be decomposed into entity to non-entity semantic shift and non-entity to entity semantic shift, which are abbreviated as E2O and O2E.

Figure 3: Illustration of E2O and O2E. When "Amy" encounters E2O problem, the label is biased from [PER] to [O]. "California" encounters O2E problem, the label is shifted from [GPE] to [DATE].

E2O refers to the model incorrectly categorizing entities as non-entities during the learning process. This misclassification stems from the incremental sequence labeling task, where only new entities are labeled in the new dataset, potentially causing old entities to be erroneously labeled as non-entities. For instance, in Fig.3, the name "Amy" is mistakenly labeled as a non-entity. This misclassification induces a gradual shift in the semantics of old entities towards non-entities, leading to a blurred boundary between the two classes. Several previous approaches have addressed this bias issue. Methods like RDP focus on designing improved distillation techniques to maintain the stability of the model's old entities. Similarly, OCILNER utilizes comparative learning to obtain a more discriminative feature space, clarifying the classification boundaries between entities and non-entities. These strategies aim to mitigate the impact of E2O, ensuring a more accurate preservation of entity semantics during incremental sequence labeling tasks.

O2E signifies the model incorrectly labeling non-entities or old entities as new entities during the learning process. As seen in Fig.3, our observations indicate that while the model maintains good discrimination between old entities. However, in Fig.2, there is a bias towards new entities in predictions during incremental learning. Our research identifies two key contributing factors to this bias.

The first factor is related to the classifier dimension's predisposition. When learning new entities, the ordinary cross-entropy function induces the model to fit and converge faster on the distribution of new entities by excessively penalizing the classifier dimension associated with old entities. This over-penalization of old entities results in a pronounced bias in significant classification scores towards the new classes.

The second factor involves a tendency at the feature optimization level. The current dataset mainly contains samples of new entities with minimal representation from other entities, including potential old and future new entities, to facilitate effective learning of new entities. As a result, in the same batch, the probability of old entities participating in model optimization is much lower than the probability of new entities' participation . Consequently, there is a predisposition towards new categories at the feature optimization level. Addressing these aspects is crucial for mitigating the O2E semantic shift and achieving more balanced and accurate predictions during incremental sequence labeling tasks.

Notably, the E2O and O2E problems are interconnected. If the O2E problem occurs in the model during incremental sequence labeling, it can gradually blur the boundaries between entities and entities and among entities. It can also indirectly contribute to the E2O problem, ultimately impacting the model's discriminative ability. We will address these two semantic biases separately to mitigate catastrophic forgetting during incremental sequence labeling.

4.2 Solving E2O problem via knowledge distillation

When learning the current task t, the model $\mathcal{M}^t$ is trained on the training examples with the current entities, which often leads to catastrophic forgetting for the old entities. To alleviate the E2O problem, we use knowledge distillation (Hinton et al., 2015). This method preserves the prior knowledge by distilling the output probabilities from the old $\mathcal{M}^{t-1}$ to the current model $\mathcal{M}^t$. Therefore, the objective function for solving the E2O problem can be expressed as:

$$\mathcal{L}_{kd} = \frac{1}{|\mathcal{D}_t|} \sum_{i=1}^{|\mathcal{D}_t|} \hat{y}_i^{t-1} \log \hat{y}_i^t, \qquad (1)$$

where $\hat{y}_i^{t-1}$ and $\hat{y}_i^t$ represent the output probabilities of the current model and the old model respectively.

Through Eq.1, "Amy" in Fig.4 corrects the current model's incorrect labeling via the output proba-

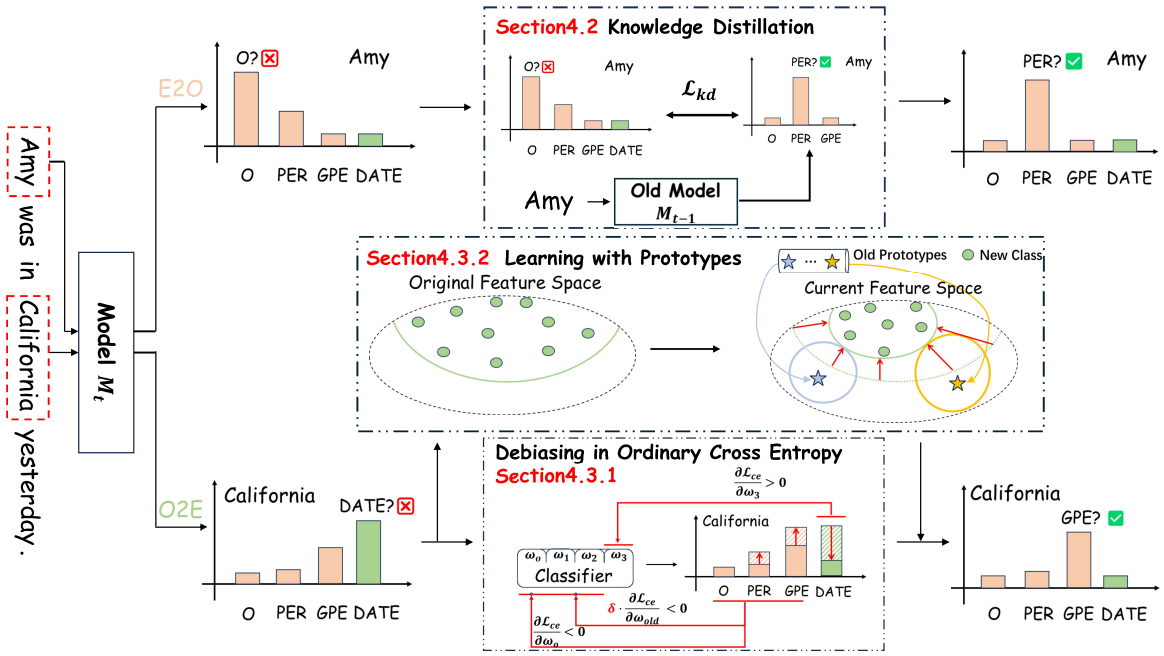


Figure 4: Overview of our framework IS3 for incremental sequence labeling. We solve the O2E problem by distillation loss L_{kd}. Besides, we use two modules: debiased cross-entropy loss L_{ce}^{Debias} and prototype learning to solve the E2O problem.

bilities provided by the old model, thus maintaining discriminative properties between old entities.

4.3 Solving O2E problem

In this section, we address the O2E problem at the debiased loss and feature optimization levels.

4.3.1 Debiasing in Ordinary Cross Entropy

The overall model parameters are defined as $\Theta = \{\theta, \omega\}$. The model's backbone $f_\theta : X \to \mathbb{R}^d$ extracts feature embeddings of dimension d from the inputs. Following the backbone, a linear classifier produces logits $\Phi(\cdot) = \omega^T \cdot f_\theta(\cdot) : X \to \mathbb{R}^{|\mathcal{Y}_t|}$, where ω represents the classifier weights for the corresponding dimensions. As the number of classes of recognizable entities increases as well, the dimension of the classifier increases. The model is trained by a cross-entropy loss function, which is defined as:

$$\begin{aligned}\mathcal{L}_{ce} &= -\frac{1}{|\mathcal{D}_t|}\sum_{i=1}^{|\mathcal{D}_t|} y_i \log\left(\frac{e^{\Phi_{y_i}(x_i)}}{\sum_{y'} \in \mathcal{Y}_t e^{\Phi_{y'}(x_i)}}\right) \\ &= \frac{1}{|\mathcal{D}_t|}\sum_{i=1}^{|\mathcal{D}_t|} \log[1 + \sum_{y' \neq y_i} e^{\Phi_{y'}(x_i) - \Phi_{y_i}(x_i)}],\end{aligned} \tag{2}$$

where y_i denotes the label of the new entity for the current incremental step t. Fig.2 shows that confusion matrix of previous method at incremental step 4. It clearly shows that most predictions are biased towards the recent entity (class 4). We find that such a bias can be found in the cross-entropy loss function. When learning new entities, the model's gradient update for old entities is defined as:

$$\frac{\partial \mathcal{L}_{ce}}{\partial \omega_{y'}} \propto e^{\Phi_{y'}(x_i)}(y' \neq y_i), \tag{3}$$

where the gradient update for old entities is proportional to the classification score for that entity. During the incremental sequence labeling process, this gradient update exhibits an overly penalizing effect on the old entity probability distributions. It shows up as an excessive reduction in the output probability score of the old entity. We provide a more detailed explanation and derivation in Appendix A.

We assume the old model has learned the optimal representation of old entities. Therefore, the new entities should have a smaller impact on the knowledge of old entities. Otherwise, because of the absence of rehearsal samples of the old entities, the model will face catastrophic forgetting of the old entities. In addition, the new entity was not in the predefined set, and a change from a non-entity to a new entity occurs during learning. Therefore, it is reasonable to have a penalizing effect on non-entities, and the debiased cross-entropy loss func-

tion is defined as follows:

$$\mathcal{L}_{ce}^{Debias} = \frac{1}{|\mathcal{D}_t|}\sum_{i=1}^{|\mathcal{D}_t|}\log[1+\sum_{y\prime\neq y_i} e^{\delta\Phi_{y\prime}(x_i)-\Phi_{y_i}(x_i)}], \tag{4}$$

where δ is the correction factor for the gradient update of the old entity weights (excluding non-entity weights), $\delta \in [0, 1]$. When $\delta \to 0$, the model will no longer penalize the learning of old entities. When $\delta \to 1$, Eq.4 degenerates to the traditional cross-entropy loss function.

4.3.2 Learning with Prototypes

In Section 4.1, we elucidate the reasons behind the emergence of O2E at the feature optimization level. In this section, we introduce the utilization of class centers of old entities as class prototypes during the learning process of new entities. Following each task training, we compute prototypes using feature representations from the training set and store them. These prototypes then participate in training the model classifier for the subsequent task alongside the feature representations of new entities.

The class prototypes of old entities serve two essential purposes: firstly, they participate in optimization alongside new entities in each batch, ensuring a balanced optimization process among entities. Secondly, these class prototypes act as anchors in the feature space, mitigating the issue of over-labeling new entities. As depicted in Fig.4, the introduction of old prototypes reduces the potential over-labeling of new entities, enhancing the precision of new entity learning.

To this end, we defined the loss function of prototypes as follows:

$$\mathcal{L}_{pro} = -\sum_{i=1}^{t-1}\widetilde{y_i}\log\left(\frac{e^{\omega^T\mathcal{P}_i}}{\sum_{j=0}^{|\mathcal{Y}_t|}e^{\omega^T\mathcal{P}_j}}\right), \tag{5}$$

where $\widetilde{y_i}$ stands for the label of the old prototype and $\mathcal{P}_i$, $\mathcal{P}_j$ stand for old prototypes, defined as follows:

$$\mathcal{P}_t = \frac{1}{|\mathcal{D}_t|}\sum_{i=1}^{|\mathcal{D}_t|} f_\theta(x_i). \tag{6}$$

Our approach differs from OCILNER's approach which uses prototypes in the following two ways:(1) OCILNER's approach stores old samples for calculating prototypes. Yet in this paper, we only use the training data in each incremental step for calculating prototypes, and do not introduce replay samples. (2) OCILNER uses prototypes to label new datasets and adopts a cosine similarity as the threshold for entity labeling. However, in this paper, we found that some of the real non-entities also have a high cosine similarity with entities, which can easily produce wrong labeling for real non-entities and exacerbate semantic migration from entities to non-entities.

In summary, the objective function of our method is defined as follows:

$$\mathcal{L} = \underbrace{\mathcal{L}_{ce}^{Debias} + \alpha\mathcal{L}_{pro}}_{\mathcal{L}_{O2E}} + \underbrace{\beta\mathcal{L}_{kd}}_{\mathcal{L}_{E2O}}. \tag{7}$$

5 Experiments

5.1 Experimental Setup

Datasets We conducted experiments on three widely used datasets: i2b2 (Murphy et al., 2010), OntoNotes5 (Hovy et al., 2006), and MAVEN (Wang et al., 2020). We divide the dataset into disjoint slices according to categories. In each slice, we keep only the category labels visible to the current task, and the rest of the labels are labeled as non-entities.

Settings We sort the above slices according to initial letter and train them in a FG-*a*-PG-*b* manner. FG means that the pre-trained model is trained with *a* entity types as the initial model and PG means that the initial model is trained with *b* entity types at each following incremental step.

Baselines We consider the following state-of-the-art methods for incremental sequence labeling: Self-Training (Rosenberg et al., 2005; De Lange et al., 2019), ExtendNER (Monaikul et al., 2021), CFNER (Zheng et al., 2022), DLD (Zhang et al., 2023c), RDP (Zhang et al., 2023b), OCILNER (Ma et al., 2023), ICE (Liu and Huang, 2023), CFPD (Zhang et al., 2023a). Detailed descriptions of the baselines and their experimental setup are provided in Appendix C.

Implementation Details We use *bert-base-cased* model from HuggingFace (Wolf et al., 2019) as backbone, with a hidden dimension of $d = 768$. We use the AdamW (Loshchilov and Hutter, 2018) optimizer, with learning rate $1e^{-6}$ and $1e^{-3}$ for backbone and classifier. We report the mean and standard deviation results over five runs.

Metrics Considering that each of the categories should have a comparable degree of contribution in the test, we use Macro F1 to evaluate the performance of the model. We use the last step Macro F1 result in $\mathcal{A}_T$, and the average Macro F1 result in $\bar{\mathcal{A}}$,

Table 1: Comparisons with state-of-the-art methods on i2b2. The best results are highlighted in **bold** and the second best results are underlined. The average of each incremental step is provided in Fig.10.

Methods	FG-1-PG-1		FG-2-PG-2		FG-8-PG-1		FG-8-PG-2	
	$\mathcal{A}_T$	$\bar{\mathcal{A}}$	$\mathcal{A}_T$	$\bar{\mathcal{A}}$	$\mathcal{A}_T$	$\bar{\mathcal{A}}$	$\mathcal{A}_T$	$\bar{\mathcal{A}}$
FT	2.16 ± 0.18	14.98 ± 0.47	7.38 ± 1.10	25.00 ± 0.74	2.41 ±0.17	16.14 ± 1.81	6.38 ± 1.23	25.82 ± 1.36
SelfTrain	17.76 ± 1.75	37.32 ± 2.28	36.63 ± 6.27	54.07 ± 3.12	7.01 ± 3.51	27.27 ± 3.47	24.05 ± 6.61	47.81 ± 2.81
ExtendNER	19.54 ± 1.59	39.10 ± 3.17	29.20 ± 5.86	48.26 ± 4.05	7.83 ± 1.42	29.03 ± 1.15	24.00 ± 6.40	42.53 ± 2.92
CFNER	34.15 ± 4.79	50.15 ± 2.18	47.21 ± 2.99	58.03 ± 2.28	21.50 ± 1.49	38.53 ± 1.01	23.91 ± 3.91	46.31 ± 3.39
DLD	23.03 ± 4.08	42.87 ± 4.35	41.05 ± 2.79	57.28 ± 1.37	13.10 ± 3.05	35.12 ± 2.24	32.01 ± 4.47	51.66 ± 1.71
RDP	28.05 ± 1.85	47.61 ± 2.03	44.53 ± 2.79	59.75 ± 1.25	26.83 ± 3.01	42.02 ± 1.57	41.43 ± 5.32	56.92 ± 4.07
OCILNER	9.30 ± 1.79	27.75 ± 2.82	18.45 ± 3.18	42.43 ± 1.90	19.76 ± 3.56	41.01 ± 2.77	24.86 ± 2.12	46.75 ± 2.14
ICE_PLO	35.45 ± 0.91	45.65 ± 1.32	40.32 ± 0.58	50.25 ± 0.93	44.79 ± 0.93	50.61 ± 0.72	44.23 ± 2.22	51.05 ± 1.83
ICE_O	36.96 ± 1.17	46.93 ± 1.07	43.29 ± 1.79	51.24 ± 1.70	46.24 ± 1.36	51.70 ± 0.85	49.10 ± 1.33	53.56 ± 1.22
CPFD	17.72 ± 3.95	46.11 ± 1.45	31.44 ± 5.19	53.84 ± 2.39	5.0 ± 3.97	32.86 ± 3.49	23.03 ± 7.47	50.26 ± 3.38
IS3 (Ours)	**43.88** ± 2.05	**56.87** ± 0.56	**54.84** ± 1.35	**61.83** ± 0.87	**50.75** ± 1.28	**58.38** ± 1.35	**56.96** ± 0.68	**63.03** ± 1.07

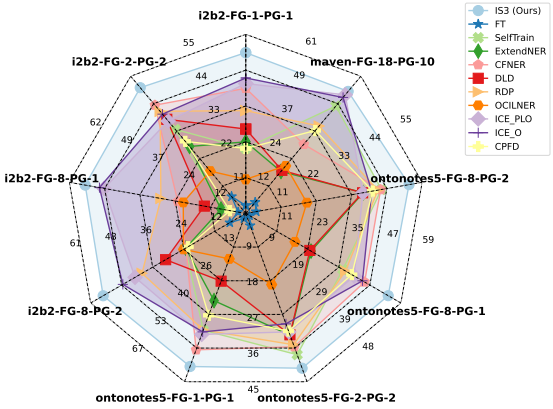
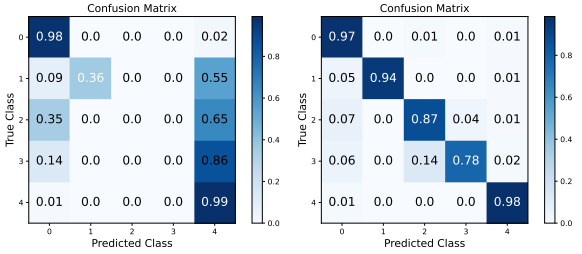


Figure 5: The comparison between our method and previous state-of-the-art methods on nine incremental learning settings. We report the MacroF1 score after learning the final task. The detailed results are provided in Table 1, Table 7 and Table 5.



(a) ExtendNER (b) IS3 (Ours)

Figure 6: Visualization of prediction of previous method and IS3 approach in task 4. Our approach greatly mitigates the E2O and O2E shift problems and balances the old and new classes well on the model predictions.

on all incremental steps as evaluation metrics. $\mathcal{A}_T$ and $\bar{\mathcal{A}}$ are defined in Appendix D.

5.2 Results and Analysis

Comparisons with State-Of-The-Art To validate the effectiveness of our approach, we conducted exhaustive experiments on the i2b2, OntoNotes5, and MAVEN datasets. We used the Finetune Only (FT) approach as a lower bound for comparison. Table 1 displays the results of the experiments conducted on i2b2. Due to space limitations, we provide the results on MAVEN in Table 5 and OntoNotes5 in Table 7. We show the experimental results under nine incremental learning settings in detail through Fig 5. Our method consistently outperforms the previous state-of-the-art method in multiple settings, from FG-1-PG-1 to FG-8-PG-2. The poor performance of the previous method may be attributed to the ignorance of O2E. As shown in Fig.6, during the learning process, the previous method, ExtendNER, confuses new entities with non-entities due to O2E and old entities with non-entities due to E2O. Both of them together lead to poor prediction results of the model. We have effectively mitigated the above problems through our framework IS3, which strikes a good balance between maintaining old entities and learning new ones.

To further demonstrate the effectiveness of our method, we visualize the feature representation through T-SNE (Van der Maaten and Hinton, 2008). As shown in Fig.11 in appendix E, the ExtendNER method faces serious E2O and O2E problems, with new entities and non-entities overwriting old ones, leading to catastrophic forgetting. Our method successfully addresses the issue of semantic bias that

Figure 7: A sample from OntoNotes5. EVENT, DATE, GPE are old entites. ORG is a new entity. The previous method had the issue of mislabeling non-entities as old entities and overwriting old entities as new ones. In contrast, our method accurately labels old entities when learning the new entity, demonstrating its effectiveness and superiority.

Table 2: The ablation study of our method on i2b2 and OntoNotes5 under the setting FG-1-PG-1, MAVEN under the setting FG-18-PG-10. The ablation of each component resulted in a significant decrease in model performance, proving the effectiveness of all our components.

Methods	i2b2		OntoNotes5		MAVEN	
	$\mathcal{A}_T$	$\bar{\mathcal{A}}$	$\mathcal{A}_T$	$\bar{\mathcal{A}}$	$\mathcal{A}_T$	$\bar{\mathcal{A}}$
IS3 (Ours)	43.88 $\pm$ 2.05	56.87 $\pm$ 0.56	50.23 $\pm$ 0.94	54.65 $\pm$ 0.84	40.15 $\pm$ 0.38	48.16 $\pm$ 0.16
w/o $\mathcal{L}_{ce}^{Debias}$	40.79 $\pm$ 0.89	54.39 $\pm$ 0.19	47.89 $\pm$ 0.91	52.77 $\pm$ 1.21	38.19 $\pm$ 0.98	46.56 $\pm$ 0.58
w/o $\mathcal{L}_{pro}$	25.88 $\pm$ 2.78	45.95 $\pm$ 2.53	44.26 $\pm$ 1.33	50.07 $\pm$ 1.08	34.64 $\pm$ 0.78	45.15 $\pm$ 0.39
w/o Both	23.22 $\pm$ 2.12	37.81 $\pm$ 3.81	42.77 $\pm$ 0.22	49.11 $\pm$ 0.49	31.03 $\pm$ 0.34	42.61 $\pm$ 0.87

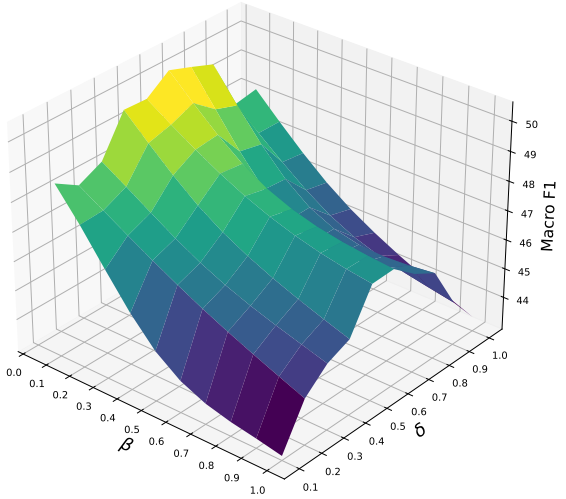


Figure 8: The results of different hyper-parameter choices on i2b2 with the setting FG-1-PG-1. We show the results are $\delta \in (0, 1]$ and $\beta \in (0, 1]$.

arises when the model learns a new task.

Ablation Study We explored the validity of the components of our approach through ablation experiments, and the results are shown in Table 2. We removed the debiased cross-entropy loss $\mathcal{L}_{ce}^{Debias}$ and prototype loss $\mathcal{L}_{pro}$ modules, respectively. These results demonstrate the essential roles played by both $\mathcal{L}_{ce}^{Debias}$ and $\mathcal{L}_{pro}$ modules. The $\mathcal{L}_{ce}^{Debias}$ reduces the penalizing effect of the new entity on the old entity and enhances the discrimination between the old and new entities by improving the prediction confidence of the old entity. The $\mathcal{L}_{pro}$ corrects the bias of modeling new entities by shrinking the scope of over-labeling new entities through old prototypes.

Hyper-Parameter Analysis Fig.8 shows the results of different hyper-parameter choices on OntoNotes5 with the setting FG-1-PG-1. We consider two hyper-parameters: the correction factor in the debiased cross-entropy loss δ and the weight of the prototype loss β. The results show that δ around 0.5 reaches the best result, indicating that a moderate penalty effect reduction favors model performance. As β keeps increasing, it makes the model overfit for the old prototype, leading to a decrease in model performance.

Case Study We provide an example in Fig.7 to demonstrate that the previous method suffers from an O2E offset when learning a new entity ORG, overwriting the old entity EVENT as a new entity. Simultaneously, the model inherits past O2E issues (labeling [O] as [DATE]). Additionally, it suffers from E2O, which fails to recognize the old entity accurately. Our method effectively balances these two types of offset problems and is more conducive to model learning.

6 Conclusion

In this paper, we introduce a novel perspective on the catastrophic forgetting problem in incremental sequence annotation, identifying and addressing both E2O and O2E semantic shifts. Bridging gaps in previous research, we propose the IS3 framework to tackle both issues. Comprehensive experiments on three datasets demonstrate that our IS3 method significantly outperforms previous state-of-the-art approaches. This work provides a fresh outlook on the incremental sequence labeling task and offers effective solutions to mitigate the catastrophic forgetting problem.

7 Limitations

While the proposed method effectively mitigates catastrophic forgetting to some extent, its reliance on the predictions of old models for preserving existing knowledge can result in accumulated prediction errors, which may lead to poor model performance in more incremental steps. Moreover, the current method does not thoroughly explore the relationship between the penalty effect and the dataset, leaving potential avenues for future research.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th international conference on computational linguistics*, pages 1638–1649.

Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. 2021. Rainbow memory: Continual learning with a memory of diverse samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8218–8227.

Pengfei Cao, Yubo Chen, Jun Zhao, and Taifeng Wang. 2020. Incremental event detection via knowledge consolidation networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 707–717.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. 2019. Continual learning: A comparative study on how to defy forgetting in classification tasks. *arXiv preprint arXiv:1909.08383*, 2(6):2.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. 2021. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385.

Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. 2020. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 86–102. Springer.

Alexander Gepperth and Barbara Hammer. 2016. Incremental learning algorithms and applications. In *European symposium on artificial neural networks (ESANN)*.

Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. 2013. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*.

Jiangpeng He and Fengqing Zhu. 2022. Exemplar-free online continual learning. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 541–545. IEEE.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. 2018. Lifelong learning via progressive distillation and retrospection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 437–452.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. 2019. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 831–839.

Eduard Hovy, Mitch Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: the 90% solution. In *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, pages 57–60.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Zhizhong Li and Derek Hoiem. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947.

Minqian Liu and Lifu Huang. 2023. Teamwork is not always good: An empirical study of classifier drift in class-incremental information extraction. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2241–2257, Toronto, Canada. Association for Computational Linguistics.

David Lopez-Paz and Marc'Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Ruotian Ma, Xuanting Chen, Zhang Lin, Xin Zhou, Junzhe Wang, Tao Gui, Qi Zhang, Xiang Gao, and Yun Wen Chen. 2023. Learning "O" helps for learning more: Handling the unlabeled entity problem

for class-incremental NER. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5959–5979, Toronto, Canada. Association for Computational Linguistics.

Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier.

Natawut Monaikul, Giuseppe Castellucci, Simone Filice, and Oleg Rokhlenko. 2021. Continual learning for named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13570–13577.

Shawn N Murphy, Griffin Weber, Michael Mendis, Vivian Gainer, Henry C Chueh, Susanne Churchill, and Isaac Kohane. 2010. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). *Journal of the American Medical Informatics Association*, 17(2):124–130.

German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. 2019. Continual lifelong learning with neural networks: A review. *Neural networks*, 113:54–71.

Marek Rei, Gamal KO Crichton, and Sampo Pyysalo. 2016. Attending to characters in neural sequence labeling models. *arXiv preprint arXiv:1611.04361*.

Anthony Robins. 1995. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2):123–146.

Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. 2005. Semi-supervised self-training of object detection models.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. 2017. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30.

Gido M van de Ven, Tinne Tuytelaars, and Andreas S Tolias. 2022. Three types of incremental learning. *Nature Machine Intelligence*, 4(12):1185–1197.

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

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020. Maven: A massive general domain event detection dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1652–1671.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. 2019. Large scale incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 374–382.

Yu Xia, Quan Wang, Yajuan Lyu, Yong Zhu, Wenhao Wu, Sujian Li, and Dai Dai. 2022. Learn and review: Enhancing continual named entity recognition via reviewing synthetic samples. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2291–2300.

Shipeng Yan, Jiangwei Xie, and Xuming He. 2021. Der: Dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023.

Pengfei Yu, Heng Ji, and Prem Natarajan. 2021. Lifelong event detection with knowledge transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5278–5290.

Friedemann Zenke, Ben Poole, and Surya Ganguli. 2017. Continual learning through synaptic intelligence. In *International conference on machine learning*, pages 3987–3995. PMLR.

Duzhen Zhang, Wei Cong, Jiahua Dong, Yahan Yu, Xiuyi Chen, Yonggang Zhang, and Zhen Fang. 2023a. Continual named entity recognition without catastrophic forgetting. *arXiv preprint arXiv:2310.14541*.

Duzhen Zhang, Hongliu Li, Wei Cong, Rongtao Xu, Jiahua Dong, and Xiuyi Chen. 2023b. Task relation distillation and prototypical pseudo label for incremental named entity recognition. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 3319–3329.

Duzhen Zhang, Yahan Yu, Feilong Chen, and Xiuyi Chen. 2023c. Decomposing logits distillation for incremental named entity recognition. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1919–1923.

Junhao Zheng, Zhanxian Liang, Haibin Chen, and Qianli Ma. 2022. Distilling causal effect from miscellaneous other-class for continual named entity recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3602–3615, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Derivation of Debiased Cross-entropy Loss Function

The overall model parameters are defined as $\Theta = \{\theta, \omega\}$. The model's backbone $f_\theta : X \rightarrow \mathbb{R}^d$ extracts feature embeddings of dimension d from the inputs. Following the backbone, a linear classifier

produces logits $\Phi(\cdot) = \omega^T \cdot f_\theta(\cdot) : X \to \mathbb{R}^{|\mathcal{Y}_t|}$, where ω represents the classifier weights for the corresponding dimensions and $\mathcal{Y}_t$ represents the current label set. The softmax probability of new entity is defined as: $p_{y_i} = \frac{e^{\Phi_{y_i}(x_i)}}{\sum_{y' \in \mathcal{Y}_t}} e^{\Phi_{y'}}(x_i)$. The derivation of Debiased Cross-entropy Loss Function is proved as follows:

$$\begin{aligned} \frac{\partial \mathcal{L}_{ce}}{\partial \omega_{y\prime}} &= \frac{\partial \mathcal{L}_{ce}}{\partial p_{y_i}} \cdot \frac{\partial p_{y_i}}{\partial \Phi_{y'}(x_i)} \cdot \frac{\partial \Phi_{y'}(x_i)}{\partial \omega_{y\prime}} \\ &= -\frac{1}{\ln 2 \cdot p_{y_i}} \cdot f_\theta(x_i) \cdot \frac{\partial p_{y_i}}{\partial \Phi_{y'}(x_i)} \\ &= \frac{f_\theta(x_i)}{\ln 2} \cdot p_{y'} \propto e^{\Phi_{y'}(x_i)} (y' \neq y_i) \end{aligned} \tag{8}$$

for the same input x_i, $\frac{f_\theta(x_i)}{\ln 2}$ can be viewed as a constant. Therefore, the gradient penalty of the new entity over the old entity is proportional to the probability value of the old entity.

B Datasets

Table 3: Examples of inputs and outputs for each dataset.

Dataset	Entity Type	Sample	Entity Type Sequence (Alphabetical Order)
i2b2	16	141k	AGE, CITY, COUNTRY, DATE, DOCTOR, HOSPITAL, IDNUM, MEDICALRECORD, ORGANIZATION, PATIENT, PHONE, PROFESSION, STATE, STREET, USERNAME, ZIP
OntoNotes5	18	77k	CARDINAL, DATE, EVENT, FAC, GPE, LANGUAGE, LAW, LOC, MONEY, NORP, ORDINAL, ORG, PERCENT, PERSON, PRODUCT, QUANTITY, TIME, WORK_OF_ART
MAVEN	178	124k	ACTION, ARREST, BRINGING, CONTROL, EXPANSION, INCIDENT, INFLUENCE, VIOLENCE etc.

Table 4: Detailed description of each dataset.

Inputs	Xinhua news agency , Beijing , August 31st
FL	B-ORG I-ORG I-ORG O B-GPE O B-DATE I-DATE
Inputs	There were no direct effects of the earthquake ' s
FL	O O O O B-Influence O O B-Catastrophe O O
	shaking due to its low intensity.
	B-Motion O O O O O

FL means the *full ground-truth label* for all steps. During the learning process, we will label unseen entities as non-entities [O].

C Baselines

The introduction about the baselines in the experiment and their settings are as follows:

- SelfTrain (Rosenberg et al., 2005; De Lange et al., 2019): SelfTrain utilizes the labels generated by the predictions of the old model on the new dataset, combined with the labels of the new entities, to guide the training of the new model.
- ExtendNER (Monaikul et al., 2021): ExtendNER introduces knowledge distillation to review the knowledge of old entities, aiming to align the outputs of the old and new models for old entities using KL divergence. In contrast to SelfTrain, ExtendNER retains specific structural information through the probability distribution of the model output. The coefficient of the distillation loss $\lambda = 2$.
- CFNER (Zheng et al., 2022): CFNER proposes a unified causal framework to extract causality from both new entity types and the Other-Class and employs curriculum learning to alleviate the impact of label noise and introduce a self-adaptive weight to balance the causal effects between new entity types and the Other-Class. The number of matched tokens $K = 3$, the initial value of balancing weight $\lambda_{base} = 2$ and the initial value of confidence threshold $\delta_1 = 1$.
- DLD (Zhang et al., 2023c): DLD decomposes a prediction logit into two terms, measuring the probability of an input token belonging to a specific entity type or not. The coefficient of the distillation loss $\lambda = 2$.
- RDP (Zhang et al., 2023b): RDP introduces a task relation distillation scheme with two aims: ensuring inter-task semantic consistency by minimizing inter-task relation distillation loss and enhancing model prediction confidence by minimizing intra-task self-entropy loss. The coefficient of inter-task relation distillation loss $\lambda_1 = 0.3$ and the coefficient of intra-task self-entropy loss $\lambda_2 = 0.1$.
- OCILNER (Ma et al., 2023): OCILNER introduces a novel representation learning method aimed at acquiring discriminative representations for entities and non-entities, which can dynamically identify entity clusters within non-entities. The threshold for relabeling samples $\beta_i = 0.98 - 0.05 * (t - i)$, where t is the current step, and i is the id of the old task.
- ICE (Liu and Huang, 2023): ICE freezes the backbone model and the old entity classifiers, focusing solely on training new entity classifiers. This approach includes two methods: ICE_O and ICE_PLO. The former combines logits of non-entity with logits of new entities for output probability computation during

training, while the latter combines all previous logits with new entity logits.

- CFPD (Zhang et al., 2023a): CPFD introduces a pooled feature distillation loss that adeptly balances the trade-off between retaining knowledge of old entity types and acquiring new ones and a confidence-based pseudo-labeling method for the non-entity type. The balancing weight $\lambda = 2$.

D Metrics

The last step Macro F1 result $\mathcal{A}_T$ and the average Macro F1 result $\bar{\mathcal{A}}$ are defined as follows:

$$a_t = \frac{1}{|\mathcal{D}_t|} \sum_{i=1}^{|\mathcal{D}_t|} \mathbb{1}(\operatorname{argmax}_{y' \in \mathcal{Y}_i} \Phi_{t,y'}(x_i) = y_i), \quad (9)$$

where a_t represents the F1 score of the t^{th} entity, $|\mathcal{D}_t|$ repesents the number of entities and $\mathbb{1}(\cdot)$ is the indicator for $\Phi_{t,y'}(x_i) = y_i$.

$$\mathcal{A}_T = \frac{1}{N} \sum_{j=1}^{N} a_j, \quad (10)$$

where $\mathcal{A}_T$ stands for the MacroF1 score at incremental step t.

$$\bar{\mathcal{A}} = \frac{1}{N} \sum_{k=1}^{N} A_{T_k}, \quad (11)$$

where $\bar{\mathcal{A}}$ stands for the average MacroF1 score for all incremental steps.

E Additional Experimental Results

Table 5 highlights the superiority of our method IS3 over previous methods on MAVEN. However, due to its larger number of classes, the performance of the model decreases in subsequent incremental steps. Table 7 shows the results of the experiments conducted on OntoNotes5. Our method IS3 achieves improvements over the previous SOTA ranging from 5.47% to 10.52% in MarcoF1 score, and 3.89% to 6.53%, under four settings (FG-1-PG-1, FG-2-PG-2, FG-8-PG-1, and FG-8-PG- 2) of the OntoNotes5 dataset.

The previous method exhibits a rapid decrease in probability distribution with increasing incremental steps, coinciding with a decline in the F1 score. In contrast, our approach IS3 effectively mitigates the model's penalization of old entities, thereby maintaining good performance.

Table 5: Comparisons with state-of-the-art methods on MAVEN. The best results are highlighted in **bold** and the second best results are underlined.

Methods	MAVEN	
	$\mathcal{A}_T$	$\bar{\mathcal{A}}$
SEQ	3.69 ± 0.17	11.75 ± 0.13
SelfTrain	35.33 ± 0.41	45.42 ± 0.76
ExtendNER	13.81 ± 0.56	24.92 ± 0.74
CFNER	22.74 ± 1.52	34.77 ± 1.38
DLD	14.18 ± 0.37	24.98 ± 0.43
RDP	28.76 ± 2.44	38.01 ± 1.09
OCILNER	21.70 ± 1.77	30.13 ± 0.75
ICE_PLO	39.01 ± 0.51	44.02 ± 0.96
ICE_O	38.16 ± 1.26	43.43 ± 1.31
CPFD	27.28 ± 1.39	41.31 ± 1.31
IS3 (Ours)	**40.15** ± 0.38	**48.16** ± 0.16

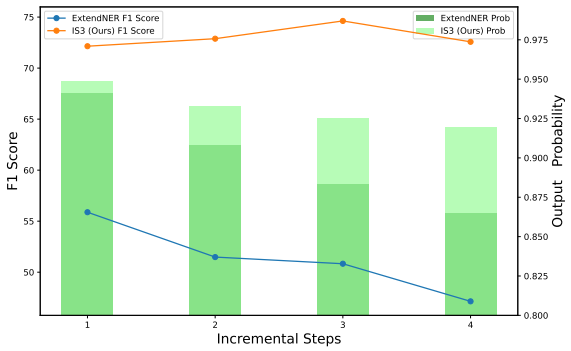


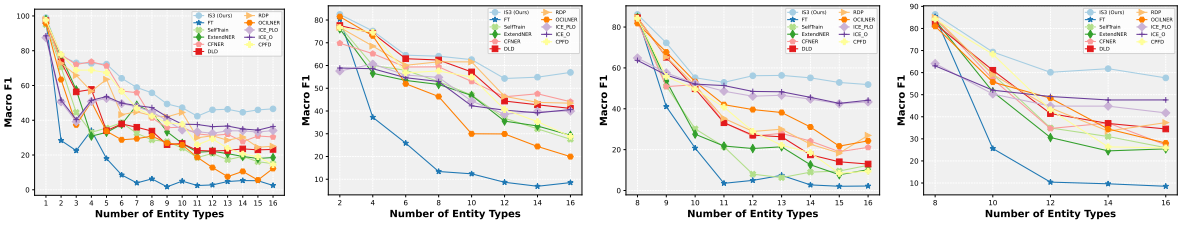
Figure 9: The F1 score and probability distributions of class "DATE" in OntoNotes5 with incremental steps.

Table 6: The comparison of training time and trainable parameters for each task on OntoNotes5.

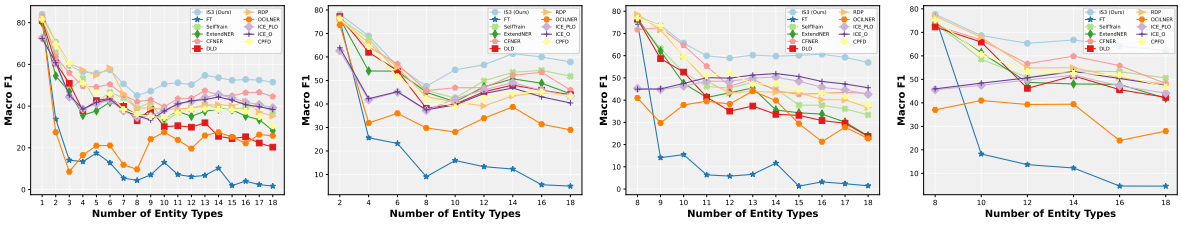
	# Time (Min)	# Trainable Params each Task
SEQ	150	109M
SelfTrain	276	109M
ExtendNER	182	109M
CFNER	512	109M
DLD	158	109M
RDP	188	109M
OCILNER	420	109M
ICE	126	28K
CPFD	282	109M
Ours	155	109M

Table 7: Comparisons with state-of-the-art methods on OntoNotes5. The best results are highlighted in **bold** and the second best results are underlined.

Dataset	Methods	FG-1-PG-1		FG-2-PG-2		FG-8-PG-1		FG-8-PG-2	
		$\mathcal{A}_T$	$\bar{\mathcal{A}}$	$\mathcal{A}_T$	$\bar{\mathcal{A}}$	$\mathcal{A}_T$	$\bar{\mathcal{A}}$	$\mathcal{A}_T$	$\bar{\mathcal{A}}$
OntoNotes5	FT	1.65 ± 0.11	12.91 ± 0.41	4.49 ± 0.44	20.69 ± 0.25	1.42 ± 0.08	12.41 ± 0.38	3.97 ± 0.37	21.45 ± 0.28
	SelfTrain	38.32 ± 5.29	47.07 ± 1.67	52.23 ± 0.43	56.14 ± 0.88	38.26 ± 3.44	49.31 ± 2.92	51.71 ± 1.39	58.51 ± 1.04
	ExtendNER	28.62 ± 2.42	42.20 ± 2.16	45.05 ± 0.61	52.30 ± 1.03	25.71 ± 5.67	40.34 ± 3.64	44.82 ± 2.42	55.25 ± 1.58
	CFNER	44.76 ± 0.28	50.76 ± 1.61	49.29 ± 2.25	55.94 ± 1.37	46.81 ± 0.99	54.91 ± 0.69	51.41 ± 2.21	60.41 ± 0.43
	DLD	22.22 ± 5.38	38.47 ± 4.73	44.88 ± 0.78	51.91 ± 1.15	25.25 ± 1.69	41.43 ± 1.01	44.53 ± 1.66	55.17 ± 1.18
	RDP	38.25 ± 5.02	48.14 ± 2.60	48.55 ± 3.54	54.81 ± 2.57	39.31 ± 4.29	52.28 ± 3.11	50.34 ± 1.86	59.89 ± 0.83
	OCILNER	14.91 ± 4.39	24.72 ± 3.21	26.31 ± 2.38	35.96 ± 1.76	19.39 ± 2.98	30.41 ± 2.98	23.28 ± 4.21	30.27 ± 4.46
	ICE_PLO	39.69 ± 0.36	43.76 ± 0.16	43.81 ± 0.34	46.38 ± 0.36	42.69 ± 0.09	46.95 ± 0.21	44.66 ± 0.61	47.72 ± 0.61
	ICE_O	38.87 ± 0.37	43.51 ± 0.23	40.82 ± 0.35	44.71 ± 0.28	45.98 ± 0.28	49.11 ± 0.49	48.01 ± 0.49	49.91 ± 0.57
	CPFD	33.44 ± 1.18	44.73 ± 0.69	43.48 ± 0.72	50.79 ± 1.05	41.77 ± 2.79	52.46 ± 1.02	48.36 ± 2.35	58.60 ± 1.99
	IS3 (Ours)	**50.23** ± 0.94	**54.65** ± 0.84	**57.23** ± 1.19	**58.25** ± 0.56	**56.11** ± 1.15	**61.44** ± 0.11	**62.23** ± 0.10	**66.01** ± 0.74

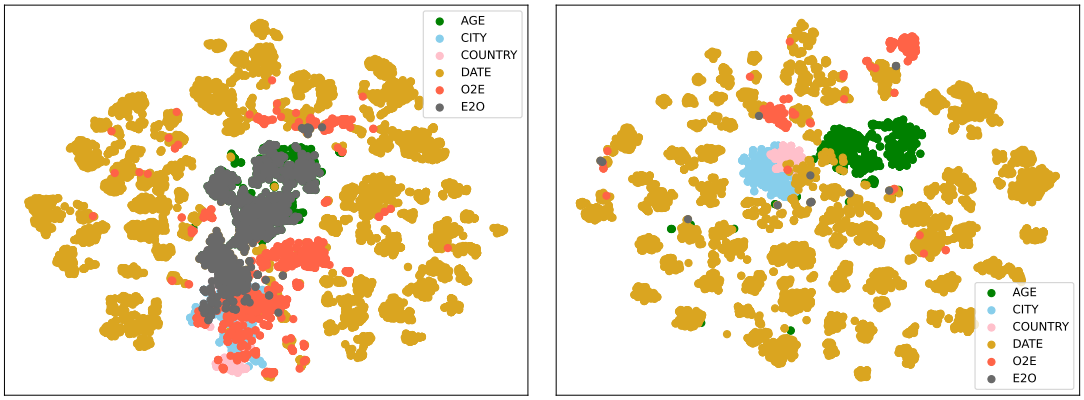


(a) i2b2 (FG-1-PG-1) (b) i2b2 (FG-2-PG-2) (c) i2b2 (FG-8-PG-1) (d) i2b2 (FG-8-PG-2)



(e) OntoNotes5 (FG-1-PG-1) (f) OntoNotes5 (FG-2-PG-2) (g) OntoNotes5 (FG-8-PG-1) (h) OntoNotes5 (FG-8-PG-2)

Figure 10: Comparison of the step-wise Macro F1 score on i2b2 and OntoNotes5.



(a) ExtendNER (b) IS3 (Ours)

Figure 11: The T-SNE visualization of the feature representations on ExtendNER and our method. Our approach IS3 greatly mitigates O2E and E2O, resulting in good discrimination between old and new entities.