AI Open 5 (2024) 39–45

Contents lists available at ScienceDirect

AI Open

journal homepage: www.keaipublishing.com/en/journals/ai-open

Few-shot Named Entity Recognition via encoder and class intervention

Long Ding[a], Chunping Ouyang[a,*], Yongbin Liu[a], Zhihua Tao[a], Yaping Wan[a], Zheng Gao[b]

[a] School of Computer, University of South China, Hengyang City, Hunan Province, Changsheng West Road, ZhengXiang District, 421001, China
[b] Department of Information and Library Science, Indiana University Bloomington, Luddy Hall Suite 2999C 700 N. Woodlawn Avenue
Bloomington, Bloomington, IN 47408, United States of America

ARTICLE INFO

Keywords:
Few-shot learning
Named Entity Recognition
Causal inference

ABSTRACT

In the real world, the large and complex nature of text increases the difficulty of tagging and results in a limited amount of tagged text. Few-shot Named Entity Recognition(NER) only uses a small amount of annotation data to identify and classify entities. It avoids the above problems. Few-shot learning methods usually use prior knowledge to achieve good results. However, prior knowledge may become a confounding factor affecting the relation between sample features and real labels. This problem leads to bias and difficulty accurately capturing class. To solve this problem, a new model, Few-shot Named Entity Recognition via Encoder and Class Intervention, is proposed based on causality. We show that we can steer the model to manufacture interventions on encoder and class, and reduce the interference of confounding factors. Specifically, while cross-sample attention perturbation is used in the encoder layer, a practical causal relation between feature and classification label is developed in the class layer. This way is an attempt of causal methodology in the Few-shot Named Entity Recognition task, which improves the discrimination ability of the NER classifier. Experimental results demonstrate that our model outperforms baseline models in both 5-way and 10-way on two NER datasets.

1. Introduction

In natural language learning, named entity recognition is essential in information extraction, aiming to identify and classify entities in text. In common corpus, named entity recognition tasks have obtained quite high-quality solutions (Li et al., 2019a). However, a rich corpus is not common in practical application. There are often insufficient resources, such as a lack of data annotation and a small number of entities in a class in the corpus. Sparse entity numbers increase the difficulty of obtaining accurate entity classes through training. Given the above situation, a task of learning entity class based on a small amount of labeled data is developed: Few-shot NER. With the popularity of Few-shot, more and more people realize the significance of studying Few-shot NER and applying it to specialized fields.

A common Few-shot task solution is metric learning (Bellet et al., 2013), including matching networks (Vinyals et al., 2016) and prototype networks (Snell et al., 2017). Such a model can learn classes from a few samples without retraining to learn new classes. This dramatically solves the problem caused by insufficient samples and shows strong potential on image classification. In different from images, studies on Natural Language Processing (NLP) in a Few-shot task seem difficult. With the development of Few-shot methods, the relevant methods have been successfully applied to text classification (Sun et al., 2019) and relation extraction (Gao et al., 2019). Prototype network (Fritzler et al., 2019), initially, the Few-shot learning of named entity recognition is a process of learning the prototype of the class and classifying the samples according to the closest prototype. However, in the face of a similar class, the basic prototype network can obtain the general semantic information, but it is difficult to accurately capture the differences between classes, making it difficult to learn the entity-related features accurately.

Most existing models focus on pre-training; the stronger the pre-training model, the better the task effect. Although the model performance is significantly improved, the relation between query set and support set cannot be utilized to improve the model's generalization ability better. In Few-shot learning: firstly, the support set and query set characteristics cannot be accurately learned due to the limited number of samples. Secondly, the same word in the text has different meanings in different contexts. Discriminating entity attributes in the sample by similarity only based on sentence attributes in the support set will cause bias. As shown in Fig. 1, sentences will focus on new feature vectors after data training in the support set. In sentence 1, the entity belongs to the type of 'event-attack/ war/military conflict', and in sentence 2, it belongs to the type of 'event-contest'. For query samples, there are similar characteristics between the two classes. The entity will be divided

[*] Corresponding author.
E-mail address: ouyangcp@126.com (C. Ouyang).

https://doi.org/10.1016/j.aiopen.2024.01.005
Received 18 August 2023; Received in revised form 18 December 2023; Accepted 29 January 2024
Available online 5 February 2024

	Sentence	Entity type
Support	Manhattan home long enough for some 4,000 American Soldiers, fleeing their loss in the Battle of Brooklyn, to reassemble in Washington Heights and join reinforcements to make a successful counterattack.	True Type: event-attack/ war/ military conflict
	The Battle of Brooklyn is an annual basketball game played between the basketball programs of Long Island University and St. Francis College.	True Type: event-contest
Query	The annual Battle of Brooklyn ended with a victory for Long Island University, where the combatants are poised to beat their opponents.	True Type: event-contest False Type: event-attack/ war/ military conflict

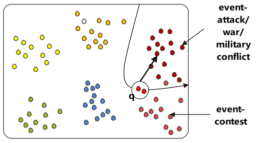

Fig. 1. Case of misclassification of similar entities.

into false type 'event-attack/ war/military conflict' according to feature similarity and prototype distance. In text information, different parts of a sentence have different meanings. Focusing attention on certain words in the sentence can be obtained by taking the sentence encoding as the input of Q and V, called the attention mechanism (Wang et al., 2018a). It is a non-local convolution mode. But there is no relationship dependency annotation in the attention mechanism, and the data bias will mislead the weight.

In this paper, we propose a novel causal mechanism Few-shot Named Entity Recognition via Encoder and Class Intervention(FSECI). We hypothesize the causal relationship between encoder and class labels, respectively. By disturbing features and classification rather than directly affecting them, the differences between similar semantic samples are increased, and the errors caused by confounding factors are alleviated. Our main contributions are as follows:

- We cross-sample features and integrate query features into the supported encoder based on the attention principle. Support features are disturbed rather than directly affected, avoiding worse attention caused by feature bias. The confusion caused by prior knowledge is subtracted to a certain extent.
- We use class adjustment to highlight the weight difference of classes and rework the inter-class prototypes. It reduces the influence of false correlation between labels and makes the classifier more accurate for correlation learning. This approach allows us to identify the true relation and eliminate confounding factors.
- Combining encoder intervention with class adjustment to realize Few-shot named entity recognition. This is an important practice of causal hypothesis theory on this task. Evaluate our model(FSECI) through two datasets in different scenarios. Experimental results show that FSECI achieves better performance compared with existing state-of-the-art baselines.

2. Related work

Few-shot learning: The application of meta learning in supervised learning. As a branch of Few-shot learning, Few-shot NER is used to identify new classes in a few sample data. Under the big standard division, it is usually realized by data augmentation, transfer, feature transformation, and knowledge link.

Fries et al. (2017) adopt the method of knowledge augmentation, uses the generated model to unify supervision, remove noise, and improve the accuracy of NER training with limited data. Some researchers (Tsai and Salakhutdinov, 2017; Ma et al., 2016) also adopt augmentation learning to carry out knowledge transfer and strengthen structural representation. But the above model is limited in a limited space and highly dependent on specialized knowledge. Transfer learning uses domain similarity to share data and build models among domains. Wang et al. (2018b) uses the idea of transfer learning to learn the feature changes of different fields and carry out parameter transfer. Cotterell and Duh (2017) improve and enrich low-resource feature representation by transferring knowledge of high-resource languages to alleviate resource constraints and lack of annotated data. Transfer learning performs well in similar tasks but cannot achieve cross-domain information acquisition for different domains. Knowledge links through structured resources heuristic tag data, using external ontology knowledge base and supplementary tagging entity. Lee et al. (2016) match text sequences with entries in the dictionary and generates a weak markup training corpus under distant supervision. However, this method is easy to causes knowledge noise and strongly depends on conditional assumptions. Feature transformation reduces domain differences by mapping source domain and target domain data features to a unified space (Pan et al., 2010). Kim et al. (2015) convert fine-grained labels to coarse-grained labels and adds attributes to simple labels. This method is also used in Zero-shot learning but is prone to over-fitting for optimization problems.

Overfitting is the most important problem to be solved in the Few-shot classification. In order to reduce the over-fitting phenomenon due to too little data, a metric-based meta-learning method (prototype network Snell et al., 2017; Lin et al., 2022) is used. Prototype network is used to calculate the Euclidean distance between each query instance and the prototype and is applied to image classification (Li et al., 2019b, 2021), text classification (Sun et al., 2019), relation extraction (Gao et al., 2019; Yang et al., 2020). The research on NER classification has always been a complex problem. Prototype networks put samples into the same metric space, where similar samples are closer to each other and dissimilar samples are farther apart. However, in the case of multiple classes, if two or more classes have similar distances, it will greatly influence the discrimination of classification results. The accuracy of the model encoder affects the result of prototype calculation to a certain extent. In encoder extraction, query features are cross-fused in support. New features are integrated into the information between sample sentences, highlighting similarities and differences between samples.

Attention mechanism: The Attention mechanism succeeds in all sorts of tasks. From image recognition (Yu et al., 2019) to natural language processing tasks, the model can pay attention to the important information in sentences and fully learn. In relation extraction, Zhou et al. (2016) use attention to extract features at the word level of sentences and integrate them into sentence-level feature vectors. Vaswani et al. (2017) use multi-head self-attention to learn text representation and capture word dependence within sentences. We put forward the method of causal inference to fuse the similarity information between sentences on the principle of attention to strengthen the commonness between sentences and highlight the difference and similarities.

Causal Inference: Causal reasoning (Yang et al., 2016) is a process of drawing conclusions about causality based on the conditions for the occurrence of a result. In recent years, Some researchers (Lukas et al., 2018; Bengio et al., 2019; Yang et al., 2023) have applied causal reasoning to machine learning and achieved good performance. In computer vision, Yue et al. (2021) decouple the features of samples and classes, and use counterfactual generation to discriminate samples belonging to known or unknown classes in zero-shot learning. Yang et al. (2021) use the causal and attention mechanism to integrate sentence features into image features and obtain visual features related to sentences. The final prediction is made by integrating the two modes' similarity information. This method is an innovative study in the vision-language field. We introduce the causal theory to Few-shot NER, incorporate new feature extraction methods, discover hidden causal structures, and enhance feature information and class information through intervention. Few-shot NER learning is dependent on prior knowledge, which will become a confounding factor in predicting real labels. The confounding factor will cause harmful bias and mislead the model to learn some wrong information in the data. From the perspective of causality, causal learning is based on the existing Few-shot learning through the front door or back door adjustment to remove confounding.

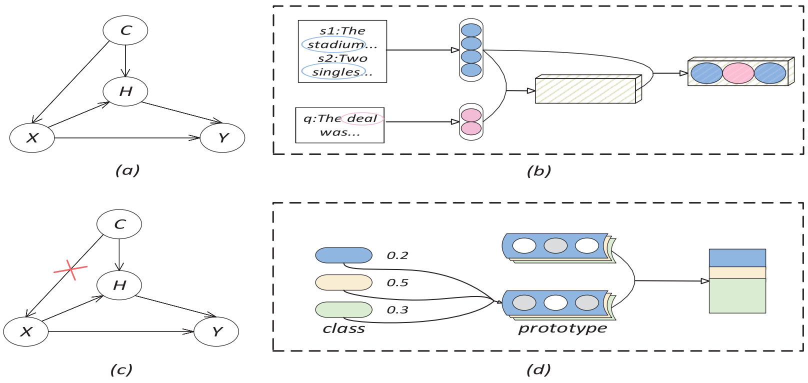

Fig. 2. (a) Causal Graph for FSECI; (b) Interventional model for P(Y|do(X)); (c) Encoder adjustment; (d) Class adjustment.

3. Methodology

This section defines the few-shot Name Entity Recognition task and then systematically introduces our model, which introduces causality and attention network to finish the named entity recognition task.

3.1. Task definition

Few-shot Name Entity Recognition aims to generate reliable classifiers when a small number of sample examples are given. In Few-shot, N classes will be randomly selected in the training stage, and K samples for each class, namely N ∗ K samples, constitute the support set input. A part of quantity sample Q (N ∗ Q sample number) is extracted from the remaining data of N classes as query set. The model is used to learn how to distinguish these classes. This task is usually called N-way K-shot. Entity classes are usually divided into “Person”, “Location”, “event”, “Other” etc., where the “Other” class represents the unmarked entity type. We pre-define the entity class $D_e = \{d_1, d_2, \dots, d_k\}$, support set is defined as Eq. (1), where $L = \{l_1, l_2, \dots, l_n\}$ represents a sentence, and y represents the class label corresponding to each l_i. The query set is defined as Eq. (2),

$$S(L, y) = S((l_1, y_1), (l_2, y_2), \dots, (l_n, y_n)) \tag{1}$$

$$Q(L, y') = Q((l_1, y'_1), (l_2, y'_2), \dots, (l_n, y'_n)) \tag{2}$$

3.2. Causal Graph

Our model uses encoder and class causal intervention to capture similarities between samples. Fig. 2(a) shows a complete causal structure diagram, indicating causal function relations through directed edges.

We formulate the causalities among text information X, priori knowledge C, the representation after feature transformation H, and the classification result Y.

$C \rightarrow H \leftarrow X$: The related feature H is determined by the text information X and prior knowledge C.

$H \rightarrow Y \leftarrow X$: Y can be directly affected by X or indirectly affected by H, where H acts as the intermediate of $X \rightarrow Y$. Cut off $X \rightarrow Y$ can also obtain the classification result of Y, but the existence of H medium is inevitable. It acts as a bridge between X and Y, and the goal cannot be achieved by blocking the link of $H \rightarrow Y$.

For Few-shot learning, while pre-training brings rich prior knowledge, it also becomes a confounding factor in the learning process, making it difficult for the classifier to find the true causal relationship between sample features and sample annotations.

Through Causal Graph for FSECI, we can identify that prior knowledge C acts as a confounding factor within the model, leading to the misinterpretation of the label Y due to the presence of extraneous information between X and Y during the NER classification process. To mitigate the impact of confounding factors, we employ the do-calculus to intervene on X, specifically, by blocking $C \rightarrow X$ (Fig. 2(b)) to complete the intervention. Encoder adjustment: By performing feature fusion based on feature dimensions,encoder level adjustment is obtained by incorporating query features. Class adjustment: Following the pre-trained model predict the probabilities associated with X belonging to different class, then calculating a weighted average to derive a new feature representation, thereby updating the class information.

The model only takes $P(Y|X)$ as the measurement standard without considering the influence of other factors. It is difficult to identify the causal relationship from X to Y accurately. To pursue the true causal relationship between X and Y, we need to use the causal intervention $P(Y|do(X))$ (as shown in Fig. 2(b)), rather than $P(Y|X)$.

3.3. Model

FSECI is designed to achieve $P(Y|do(X))$ classification prediction. The model is divided into two parts, the first part carries out causal intervention for the encoder, and the second part carries out causal intervention for class. The framework of the FSECI model is shown in Fig. 3. The model achieves classification prediction by calculating the distances between each query and prototypes.

The first step is encoder intervention: The encoder intervention section is highlighted in the model Fig. 4, and the process of output target Y through the mediation T from the input set X is shown. Fig. 4(left) shows the traditional way: X is taken as the input of the support, the prototype is calculated, and the prediction is realized through the process T. Process T represents the feature representation after mapping through self-attention. The whole process is $X \rightarrow T \rightarrow Y$, and method typically use $P(Y|X)$ as the ultimate objective for learning and training the model.

$$P(Y|X) = \sum_t P(T = t|X)P(Y|T = t) \tag{3}$$

For the first encoder fusion intervention in FSECI, as shown in Fig. 4(right): T selects appropriate knowledge from X and uses T to predict the classification of Y. However, the bias between the large amount of pre-training data and the Few-shot data will cause a certain error bias in the classification results. As shown in Fig. 1: the Battle of Brooklyn is often related to war, although the Few-shot data appears in the form of a competition, the Few-shot classification related to war. To solve this problem, we introduce the first step of feature fusion to make the adjusted features have more unified information. We adjust the input mode of X to perform front-door intervention,

$$\begin{aligned} P(Y|do(X)) &= \sum_t P(T = t|X)P(Y|do(T)) \\ &= \sum_t P(T = t|X) \sum_x P(X = x)P(Y|T = t, X = x) \end{aligned} \tag{4}$$

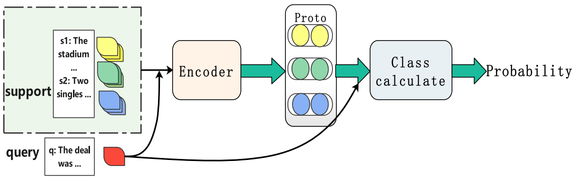

Fig. 3. The framework of FSECI model. ‘support’ is the support set. The model learn from ‘support’ to predict the label of ‘query’. The model consist of two parts: encoder module and class module.

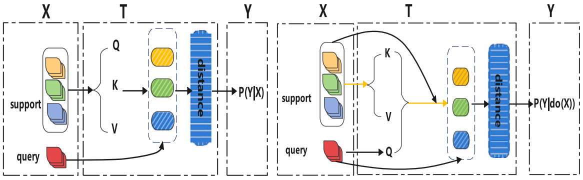

Fig. 4. Left: Classification is achieved by traditional attention method; right: Encoder intervention implements classification.

The x here represents different input cases. For our model, it represents a cross-sample of query and support. By adjusting the traditional attention method, the first step is to intervene the input features and reduce the interference of false information caused by other factors. As shown in Fig. 1, it is possible to learn that query-related entities belong to false types through the support sample set alone. Because after learning the false correlation of the support set, the features of the sample in the query are highly similar to sentence1, it is easy to infer that entity belongs to false type, leading to error discrimination. Through input intervention in the first step, the query's related attributes indirectly affect support's strong guiding feature discrimination, which will provide a legal basis for correct selection.

The second step is class adjustment intervention: Suppose there are $M = \{m_1, \ldots, m_m\}$ training classes and the average feature of each class is denoted as $C = \{c_1, c_2, \ldots, c_m\}$, $P(m_i|q)$ as the sample belongs to probability of m_i class.

We adjust between classes, redefine the probability output of the classifier, reassign the new weight of the feature mean from the pre-training according to the classification probability, and calculate the classification results. In this process, the pre-established prototypes for each class are used to predict the probability belongs to each class. The probabilities are subjected to a weighted average to obtain new feature representation between classes. Subsequently, a new round of probability prediction is carried out.

Then, we use $P(m_i|x)$ to represent the classifier and x_i' to represent the average feature of a certain lass. $P(m_i|x)x_i'$ to represent real vector. The classification adjustment formula is as follows,

$$P(Y|do(X)) = \frac{1}{m}\sum_{i=1}^{m} P(Y|x \bigoplus P(m_i|x)x_i') \tag{5}$$

We combine the input feature intervention of the first layer with the class adjustment intervention of the second layer to make the hierarchical adjustment mechanism more refined. In general, we make a secondary adjustment to the classification results based on the adjustment of the input features. Thus, we can get:

$$\begin{aligned} &P(Y|do(X = x)) \approx \\ &\sum_t P(T = t|X) \sum_x P(X = x)P(Y|T = t, X = x) \\ &+ \frac{1}{m}\sum_{i=1}^{m} P(Y|x \bigoplus P(m_i|x)x_i') \end{aligned} \tag{6}$$

Next, we will explain the implementation process: Model input: BERT (Devlin et al., 2018) was used as a mapping function to obtain the initially hidden representation. BERT was able to capture the longer distance dependencies more efficiently. At the same time, BERT uses a multi-attention mechanism as a benchmark to capture bidirectional context information. After entering the sentence into BERT, the initial hidden feature w_0 of the support set and query set is obtained. Here, given a sequence $l = \{l_1, l_2, \ldots, l_n\}$, for each token l_i, the encoder produces contextualized representations as

Table 1
Statistics of Few-NERD.

Dataset	Train	Dev	Test
Few-NERD_INTRA	99 519	19 358	44 059
Few-NERD_INTER	130 112	18 817	14 007

$$W_0 = f(l) \tag{7}$$

The self-features of support set and query set have been obtained by BERT. The traditional attention mechanism is as follows, where $Q \in R^{n\times d_k}, K \in R^{m\times d_k}, V \in R^{m\times d_v}$, n and m represents the number of tokens,

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{8}$$

In the Q-K-V mechanism, we replace Q with query sample features to extend the original features. $W \in R^{n\times d} \rightarrow W' \in R^{n\times n\times d}$ where d is the dimension. we calculate cosine distance of Q and K to obtain the similarity matrix between query sentences and support sentences as follows,

$$A_{i,j} = softmax(Cos\_Sim(Q_i, K_j)) \tag{9}$$

$$X' = \sum_{i\in I, j\in J} V_j A_{i,j} \tag{10}$$

where V_j, K_j come from the support set samples, Q_i comes from the query set. And i, j represent the ith and jth sentence, respectively. The output X' represents the representation information of support based on the query. In order to obtain regional information, the normalization function is used to process the results. The normalization function maps the sample eigenvalues to [0,1] and gives new weights to the samples to obtain the information. F(*) represents the normalization function. Overall new feature adjustment is,

$$\mathbf{X} = X' \bigoplus (X \cdot F(X')) \tag{11}$$

The classifier takes the adjusted feature X as input and then calculates the prototype. At the beginning of training, the prototype $proto_i$ under class i is randomly initialized. Different from the traditional prototype calculation (Qi et al., 2018; Yang and Katiyar, 2020), we calculate the distance in token,

$$proto_i = \frac{1}{S_i}\sum_{x\in S_i} g(x) \tag{12}$$

$$d_i(x) = d(g(q_x), proto_i) = \|g(q_x) - proto_i\|_2^2 \tag{13}$$

where support set is defined as S_i. After the distance between q_x and prototype is obtained, the probability $p = \{p_1, p_2, \ldots, p_m\}$ of sample q_x belonging to M classes is obtained. At the same time, the distance value is converted into probability weight and class features and then combined with the original features to obtain the new class features. The distance $d'(x)$ between the sample and the prototype is recalculated. For the type of y and a query x, we get

$$y^* = \{y_1, y_2, \ldots, y_m\} = argmin \prod_{i=1}^{I} p(y_i|x) \times p(y_i|y_{i-1}) \tag{14}$$

Table 2
Performance of different models on Few-NERD_INTER.

Few-NERD_INTER (%)												
Model	5-way 1∼2-shot			5-way 5∼10-shot			10-way 1∼2-shot			10-way 5∼10-shot		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
ProtoBert	38.82	54.66	45.40	50.53	64.05	56.49	33.47	49.61	39.98	45.77	59.82	51.86
NNShot	49.84	58.65	53.89	49.27	60.38	54.26	41.17	51.01	45.57	47.57	57.35	52.01
Struct	56.98	55.66	56.31	61.63	53.38	57.21	51.67	46.83	49.13	59.26	45.50	51.47
ProtoShot	48.78	55.04	51.72	54.87	59.58	57.13	41.71	49.02	45.07	52.18	54.06	53.11
FSECI	**59.84**	**62.49**	**61.14**	**66.27**	**68.04**	**67.15**	**53.09**	**59.54**	**56.13**	**63.88**	**64.06**	**63.96**

Table 3
Performance of different models on Few-NERD_INTRA.

Few-NERD_INTRA (%)												
Model	5-way 1∼2-shot			5-way 5∼10-shot			10-way 1∼2-shot			10-way 5∼10-shot		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
ProtoBert	19.20	33.34	24.37	32.14	**50.71**	39.34	13.45	24.45	17.35	25.02	**40.00**	30.78
NNShot	29.92	34.53	32.06	32.85	39.18	35.74	20.34	24.34	22.16	24.48	30.72	27.25
Struct	40.05	32.77	36.04	48.82	30.20	37.32	30.78	21.58	25.38	42.65	19.63	26.89
ProtoShot	26.35	32.91	29.27	40.31	48.51	44.03	17.59	23.80	20.23	32.44	33.03	32.73
FSECI	**41.03**	**37.52**	**39.20**	**55.41**	49.11	**52.07**	**33.26**	**30.30**	**31.71**	**54.29**	36.03	**43.32**

Table 4
Error analysis with different models under 5-way 1∼2-shot on Few-NERD_INTER.

5-way 1∼2-shot					
Dataset	Models	Span error (%)		Type error (%)	
		FP	FN	Within	Outer
Few-NERD_INTER	ProtoBert	6.65	1.98	5.95	13.23
	NNShot	3.43	3.79	5.66	7.89
	Struct	2.98	4.12	6.07	8.41
	ProtoShot	5.98	1.98	6.48	13.70
	FSECI	3.03	2.87	5.29	8.10

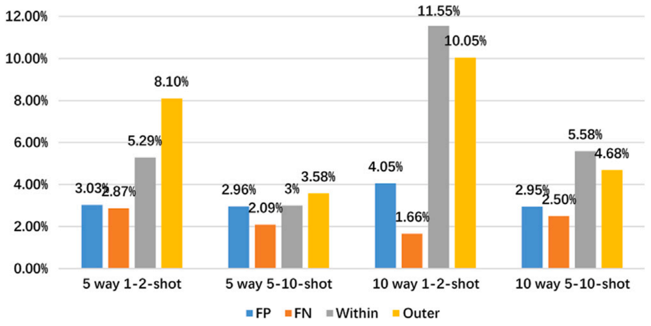

Fig. 5. Error analysis under N-way K-shot of FSECI on Few-NERD_INTER.

4. Experiment

4.1. Datasets and settings

We conducted experiments on datasets: Few-NERD_INTRA and Few-NERD_INTER (Ding et al., 2021). Its original corpus, Wikipedia, contains 188 238 sentences in 66 fine-grained entity types. Table 1 completely statistics the data distribution of the two datasets.

Few-NERD_INTRA: All entities in different collections belong to different coarse-grained types. The train set contains entity class {People, MISC, Art, Product}, the dev set contains entity class {Event, Building}, and the test set contains entity class {ORG, LOC}. Fine-grained entity types share little knowledge in the train, test, and dev set.

Few-NERD_INTER: Coarse-grained entity sharing. Allocate 60% of the fine-granularity of the eight coarse-grained type entities to train set, 20% to dev set, and 20% to test set.

The effect of the NER model under different granularity is explored by using the biases of two datasets. We use BERT to calculate context embedding for all models in line with Baselines. We set the learning rate in 1e−4 and the batch size in 2. We evaluate models by calculating the Precision(P), Recall(R), and Micro F1-score(F1).

4.2. Baselines

ProtoBert: Snell et al. (2017) use the prototypical network with the BERT encoder. A prototypical network learns a metric space for each class by calculating the token embeddings average for the same entity type. NNShot: Yang and Katiyar (2020) uses a structured neighbor method to obtain the token-level prototype closest to the token, which is a simple and efficient learning method. StructShot: Ding et al. (2021) is similar to NNShot. The only difference is that StructShot employs the Viterbi algorithm. As we all know, the Viterbi algorithm plays an essential role in the decoder stage of the named entity task. By introducing the Viterbi algorithm, the optimal path prediction results are selected. ProtoShot: is similar to ProtoBert, on which the Viterbi decoder is introduced.

4.3. Results and analysis

This part shows the comparison results between our proposed method and the typical approaches under the same hyper-parameters.

The experiment carried out three rounds of data calculation, and the results will be illustrated with the average value. Tables 2 and 3 show that FSECI has achieved good results in 5-way and 10-way. FSECI (ours) consistently outperforms state-of-the-art models, achieves 61.14% for Few-NERD_INTER and 39.2% for Few-NERD_INTRA, which obtain 4.83% and 3.16% improvements when compared with the existing highest model (Struct) for 5-way 1–2-shot task. For the 5-way 5–10-shot task, FSECI outperforms Protoshot by 10.02% and 8.04% in different datasets separately. Also, our model gets 56.13%, 31.71% and 63.96%, 43.32% for 10-way 1–2-shot and 10-way 5–10-shot in Few-NERD_INTER and Few-NERD_INTRA. In comparing different models, ProtoBert achieved better performance in calculating the value of R in Few-NERD_INTRA. On the one hand, because Few-NERD_INTRA is divided according to the fine-grained entity types, the degree of correlation between the training set and the test set is low, and the training is more complicated. On the other hand, the regularized nature of ProtoBert also plays a role. On the whole, the performance of the model in Few-NERD_INTER is generally better than that in Few-NERD_INTRA.

To further measure the performance of the model, we conduct horizontal and vertical error analysis in coarser-grained types to explore

Table 5
Comparison of intervention F1 results at different levels for FSECI.

Dataset		5-way 1∼2-shot	5-way 5∼10-shot	10-way 1∼2-shot	10-way 5∼10-shot
Few-NERD _INTER	FSECI_1_layer	**60.4**	67.18	**54.83**	60.63
	FSECI_2_layer	59.56	**68.05**	53.91	**62.22**
Few-NERD _INTRA	FSECI_1_layer	37.94	**52.02**	**31.64**	**43.02**
	FSECI_2_layer	**39.18**	51.93	29.63	42.66

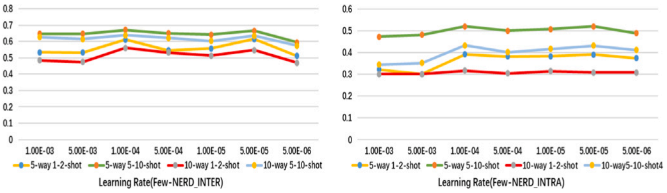

Fig. 6. Accuracy changes with different learning rate on FSECI model.

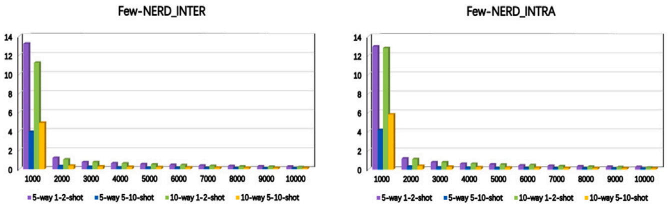

Fig. 7. The curve of the loss with the number of steps changed for FSECI under N-way K-shot on two datasets.

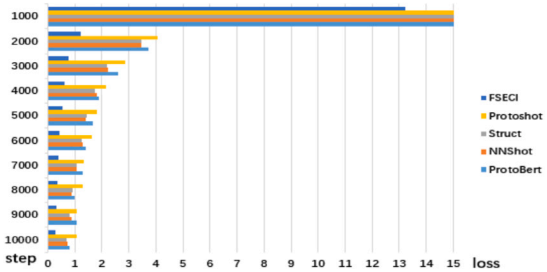

Fig. 8. The curve of the loss with the number of steps changed on 5-way 1–2-shot.

whether the entity span can be learned accurately. Table 4 shows the outcome. FP indicates that the "Other" token is identified as an entity, FN indicates that the entity token is identified as "Other". So, FP and FN are Span Error, meaning that boundary correctly identified, type incorrectly judged. Within indicates that the entity is misjudged to be of another type under the same span(coarse-grained). Outer represents that the entity is misjudged to be of another type under a different span. Whether entities can be accurately detected in span greatly influences model performance. For 5-way 1–2-shot, both FSECI and baselines show good results. Struct model has good performance in FP, but has the largest error in FN. NNShot performs best on Outer, but does not show an advantage on Span Error. Combined with Span Error and Type Error, FSECI is more stable. In terms of FSECI itself, as shown in Fig. 5, the error analysis on 5-way 5–10-shot and 10-way 5–10-shot will be lower. This means that FSECI has a better recognition effect for multiple samples.

4.4. Ablation studies

Our model focuses on the two-layer intervention mechanism. In order to verify the effect of feature fusion and class intervention, we evaluate the single model effect, respectively. As shown in Table 5: for the 5-way 1–2 shot task, a two-layer intervention achieves 39.18% higher than a one-layer intervention, In the Few-NERD-INTRA dataset. In other tasks, a one-layer intervention shows better performance in the Few-NERD-INTRA dataset. For the 5-way 5–10 shot and 10-way 5–10 shot task, two-layer intervention achieves 68.05% and 62.22% in the Few-NERD-INTER dataset, which obtain 0.87% and 1.59% improvements when compared with the one-layer intervention. The first-level intervention has greater advantages in the case of fewer samples, while the second-level intervention has greater advantages in the case of multiple samples under the coarse-grained dataset. And the first-level intervention has significant benefits in multiple samples and multiple classifications under the fine-grained dataset. FSECI combines the characteristics of both, which makes the model have good performance under a different classification and sample number.

In addition, the learning rate is one of the key metrics to improve the accuracy of the model. Fig. 6 shows that on Few-NERD_INTER and Few-NERD_INTRA, the FSECI model acquires the highest accuracy when the learning rate is 1e−4. Overall consideration, we set the learning rate to 1e−4 in our experiments. In order to verify the convergence rate of the model, we choose several points to present the changes in the loss of decline. For FSECI itself, it can be seen from Fig. 7 that the model is convergence on all datasets at 10 000 steps. Also, we verify the convergence rate of different models on the Few-NERD_INTER dataset as the step increases. In order to facilitate observation, only one segment of value was intercepted for the model with excessive loss value in 1000 steps. We can see from Fig. 8, the convergence rate of loss tends to be stable for all models with the increase of step length.

5. Conclusion

In this paper, we propose a novel framework: Few-shot Named Entity Recognition via Encoder and Class Intervention(FSECI). We exploited the causal inference to address the problem of difficulty in identifying similar semantic but not similar samples in Few-shot entity recognition. Intervention mitigated bias misdirection and captured more complex causal effects with real labels. Experiments show that our model is superior to the four baselines. It is worth noting that FSECI not only improves the accuracy of entity recognition, but also provides causal theoretical analysis and opens up a new research direction for Few-shot entity recognition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The State Key Program of National Natural Science of China, Grant/Award Number: 61533018; National Natural Science Foundation of China, Grant/Award Number: 61402220; The Philosophy and Social Science Foundation of Hunan Province, China, Grant/Award Number: 16YBA323; Natural Science Foundation of Hunan Province, China, Grant/Award Number: 2020JJ4525, 2022JJ30495; Scientific Research Fund of Hunan Provincial Education Department, Grant/Award Number: 18B279, 19A439, 22A0316.

References

Bellet, A., Habrard, A., Sebban, M., 2013. A survey on metric learning for feature vectors and structured data. arXiv preprint arXiv:1306.6709.

Bengio, Y., Deleu, T., Rahaman, N., Ke, R., Lachapelle, S., Bilaniuk, O., Goyal, A., Pal, C., 2019. A meta-transfer objective for learning to disentangle causal mechanisms. arXiv preprint arXiv:1901.10912.

Cotterell, R., Duh, K., 2017. Low-resource named entity recognition with cross-lingual, character-level neural conditional random fields. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing. pp. 91–96.

Devlin, J., Chang, M.W., Lee, K., 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., Liu, Z., 2021. Few-NERD: A few-shot named entity recognition dataset. In: Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing.

Fries, J., Wu, S., Ratner, A., Ré, C., 2017. Swellshark: A generative model for biomedical named entity recognition without labeled data. arXiv preprint arXiv:1704.06360.

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

Gao, T., Han, X., Liu, Z., Sun, M., 2019. Hybrid attention-based prototypical networks for noisy few-shot relation classification. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 6407–641.

Kim, Y.B., Stratos, K., Sarikaya, R., Jeong, M., 2015. New transfer learning techniques for disparate label sets. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. pp. 473–482.

Lee, S., Song, Y., Choi, M., Kim, H., 2016. Bagging-based active learning model for named entity recognition with distant supervision. In: Proceedings of the 2016 International Conference on Big Data and Smart Computing. pp. 321–324.

Li, Y., Li, H., Chen, H., Chen, H., 2021. Hierarchical representation based query-specific prototypical network for few-shot image classification. arXiv preprint arXiv:2103.11384.

Li, X., Sun, X., Meng, Y., Liang, J., Wu, F., Li, J., 2019a. Dice loss for data-imbalanced nlp tasks. arXiv preprint arXiv:1911.02855.

Li, W., Wang, L., Xu, J., Huo, J., Gao, Y., Luo, J., 2019b. Revisiting local descriptor based image-to-class measure for few-shot learning. In: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7260–7268.

Lin, Qiang, Liu, Yongbin, Wen, Wen, Tao, Zhihua, Ouyang, Chunping, Wan, Yaping, 2022. Ensemble making few-shot learning stronger. Data Intell. 4, 529–551.

Lukas, S., Rauber, J., Matthias, B., Brendel, W., 2018. Towards the first adversarially robust neural network model on MNIST. arXiv preprint arXiv:1805.09190.

Ma, Y., Cambria, E., Gao, S., 2016. Label embedding for zero-shot fine-grained named entity typing. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. pp. 171–180.

Pan, S.J., Tsang, I.W., Kwok, J.T., Yang, Q., 2010. Domain adaptation via transfer component analysis. IEEE Trans. Neural Netw. 22 (2), 199–210.

Qi, H., Brown, M., Lowe, D.G., 2018. Low-shot learning with imprinted weights. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5822–5830.

Snell, J., Swersky, K., Zemel, R., 2017. Prototypical networks for few-shot learning. Adv. Neural Inf. Process. Syst. 30, 4077–4087.

Sun, S., Sun, Q., Zhou, K., Lv, T., 2019. Hierarchical attention prototypical networks for few-shot text classification. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. pp. 476–485.

Tsai, Y., Salakhutdinov, R., 2017. Improving one-shot learning through fusing side information. arXiv preprint arXiv:1710.08347.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 5998–6008.

Vinyals, O., Blundell, C., Lillicrap, T., kavukcuoglu, K., Wierstra, D., 2016. Matching networks for one shot learning. Adv. Neural Inf. Process. Syst. 29, 3630–3638.

Wang, X., Girshick, R., Gupta, A., He, K., 2018a. Non-local neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7794–7803.

Wang, Z., Qu, Y., Chen, L., Shen, J., Zhang, W., Zhang, S., Gao, Y., Gu, G., Chen, K., Yu, Y., 2018b. Label-aware double transfer learning for cross-specialty medical named entity recognition. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Vol. 1, pp. 1–15.

Yang, Z., He, X., Gao, J., Deng, L., Smola, A., 2016. Stacked attention networks for image question answering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 21–29.

Yang, Y., Katiyar, A., 2020. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. pp. 6365–6375.

Yang, Zhen, Liu, Yongbin, Ouyang, Chunping, 2023. Causal intervention-based few-shot named entity recognition. In: Findings of the Association for Computational Linguistics: EMNLP 2023. Association for Computational Linguistics, Singapore, pp. 15635–15646.

Yang, X., Zhang, H., Qi, G., Cai, J., 2021. Causal attention for vision-language tasks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Yang, K., Zheng, Z., Dai, X., He, L., 2020. Enhance prototypical network with text descriptions for few-shot relation classification. In: CIKM '20: The 29th ACM International Conference on Information and Knowledge Management.

Yu, Z., Yu, J., Cui, Y., Tao, D., Tian, Q., 2019. Deep modular co-attention networks for visual question answering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6281–6290.

Yue, Z., Wang, T., Zhang, H., Sun, Q., Hua, X., 2021. Counterfactual zero-shot and open-set visual recognition. arXiv preprint arXiv:2103.00887.

Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., Xu, B., 2016. Attention-based bidirectional long short-term memory networks for relation classification. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics.