Composited-Nested-Learning with Data Augmentation for Nested Named Entity Recognition

Handi Guo
102304132
Nanjing University of Posts and Telecommunications
Jiangsu,Nanjing China

***Abstract*—Nested Named Entity Recognition (NNER) focuses on addressing overlapped entity recognition. Compared to Flat Named Entity Recognition (FNER), annotated resources are scarce in the corpus for NNER. Data augmentation is an effective approach to address the insufficient annotated corpus. However, there is a significant lack of exploration in data augmentation methods for NNER. Due to the presence of nested entities in NNER, existing data augmentation methods cannot be directly applied to NNER tasks. Therefore, in this work, we focus on data augmentation for NNER and resort to more expressive structures, Composited-Nested-Label Classification (CNLC) in which constituents are combined by nested-word and nested-label, to model nested entities. The dataset is augmented using the Composited-Nested-Learning (CNL). In addition, we propose the Confidence Filtering Mechanism (CFM) for a more efficient selection of generated data. Experimental results demonstrate that this approach results in improvements in ACE2004 and ACE2005 and alleviates the impact of sample imbalance.**

***Index Terms*—Named Entity Recognition, Nested Named Entity Recognition, Data Augmentation**

I. Introduction

Named entity recognition (NER) is a fundamental task in information extraction that aims to identify text spans corresponding to specific types, such as person, organization, and locations. It plays an essential role in various downstream tasks and applications. Traditional approaches for NER rely on sequence labeling. In this type of method, each entity is treated as a separate and non-overlapping unit, and the goal is to assign entity labels to individual words or spans in the text [1]. Nested Named Entity Recognition (NNER) offers greater flexibility compared to Flat Named Entity Recognition (FNER), enabling more fine-grained semantic representations and broader applications. However, a single word can have multiple labels, using these methods cannot effectively address NNER issues. To fill this gap, Tan et al. [2] pointed out the importance of boundaries and addressed NNER using a boundary detection task. Fu et al. [3] employed a span-based constituency parser to address NNER. They treated the annotated entity spans as a partially observed constituency tree and incorporated latent spans during training.

However, the limited availability of annotated resources for nested entities constrains the performance of NNER. While some research efforts have focused on enhancing NER models through data augmentation. Ding et al. [4] introduced a novel data augmentation technique for sequence labeling tasks which can be an effective solution for low-resource NER. Using weakly labeled data augmentation method that can extract named entities from social media texts more effectively [5]. To alleviate token-label misalignment issues, Zhou et al. [6] proposed MELM as a new data augmentation framework for low-resource NER. But so far as I know, the use of data augmentation in NNER has been very limited, and existing data augmentation techniques cannot be directly applied to NNER tasks.

TABLE I: Label correlation analysis

Outside Inside	PER	ORG	GPE	FAC	WEA	LOC	VEH
PER	3018	1047	927	170	13	106	42
ORG	843	696	394	25	0	35	11
GPE	527	230	939	69	12	158	14
FAC	25	35	153	108	0	49	0
WEA	2	3	0	16	59	2	0
LOC	22	33	131	0	4	157	0
VEH	60	53	51	0	9	8	48

Outside means containing nested labels, *Inside* means nested labels

In FNER, labels are assigned to tokens within a span, but it overlooks the possibility that one token may correspond to multiple labels, as shown in Fig. 1 (left). Labels containing nested labels are initially defined as *Outside*, with the nested labels being regarded as *Inside*. For example, consider the input "*The Chinese embassy in France*". In this context, "*The Chinese embassy*" is identified as the type of "FAC", while "*Chinese*" is also assigned with the "GPE" label. Consequently, in this case, the label "FAC" is designated as *Outside*, and the label "GPE" is categorized as *Inside*. As shown in Table I, it can be observed that the "PER" label is highly related to the "PER" label and "ORG" label. Simultaneously, there are cases where some labels have little or no correlation, so we can deduce that there is a certain level of correlation between labels. To fill this gap, we introduce a Composite-Nested-Label Classifier (CNLC) mechanism, which can simultaneously consider different labels for one token. Subsequently, the processed data is used for data augmentation through the Composited-Nested-Learning (CNL) module. As the augmented data exhibits the self-reinforcement effect [7] not all augmented data positively

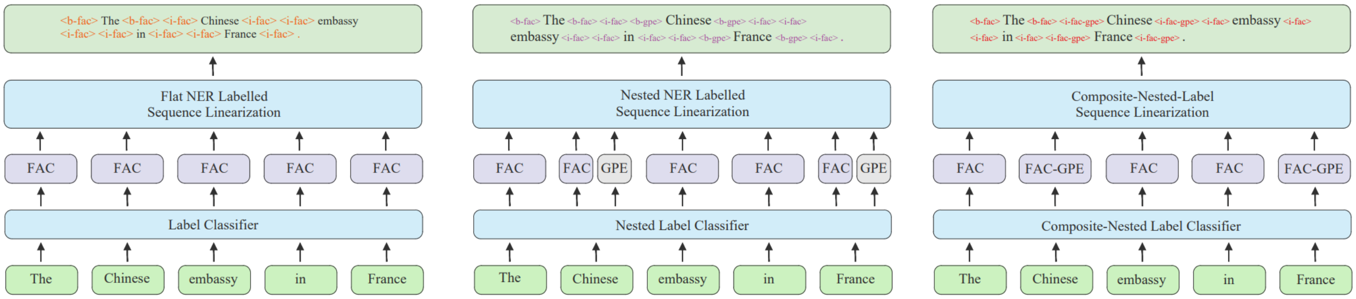

<b-fac> The <b-fac> <i-fac> Chinese <i-fac> <i-fac> embassy <i-fac> <i-fac> in <i-fac> <i-fac> France <i-fac> .
Flat NER Labelled Sequence Linearization
FAC
FAC
FAC
FAC
FAC
Label Classifier
The
Chinese
embassy
in
France
<b-fac> The <b-fac> <i-fac> <b-gpe> Chinese <b-gpe> <i-fac> <i-fac> embassy <i-fac> <i-fac> in <i-fac> <i-fac> <b-gpe> France <b-gpe> <i-fac> .
Nested NER Labelled Sequence Linearization
FAC
FAC
GPE
FAC
FAC
FAC
GPE
Nested Label Classifier
The
Chinese
embassy
in
France
<b-fac> The <b-fac> <i-fac-gpe> Chinese <i-fac-gpe> <i-fac> embassy <i-fac> <i-fac> in <i-fac> <i-fac-gpe> France <i-fac-gpe> .
Composite-Nested-Label Sequence Linearization
FAC
FAC-GPE
FAC
FAC
FAC-GPE
Composite-Nested Label Classifier
The
Chinese
embassy
in
France

Fig. 1: Three different patterns for recognizing label sequence templates, with each distinguished by different colors for their respective label formats. The left part represents the Flat-NER classification template, the middle part represents the current NNER classification template, and the right part represents the CNLC classification template we propose.

impact the model. Therefore, we design a Confidence Filtering Mechanism (CFM) to select augmented data, and experiments indicate that this approach can provide more valuable data for NNER. To summarize, our main contributions are as follows:

- We composite nested tokens and nested labels in NNER for solving NNER problems through data augmentation methods.
- For the data selection, our proposed CFM can select samples with higher confidence to improve the quality of the augmented data.
- Through the framework, we can enhance existing models' performance, and it can alleviate the problem of sample imbalance.
- We open-source our augmented dataset, which could provide a sliver dataset for other researchers.

II. Related Works

A. Data augmentation

In the field of Natural Language Processing (NLP), complex NER remains a relatively underexplored task. Data augmentation involves expanding the training dataset by applying transformations without changing the labels of the training instances. Using data augmentation methods effectively improves model generalization. Current research on data augmentation primarily addresses data scarcity in low-resource NLP, focusing on word-level modifications that prove beneficial for classification tasks, including simple synonym replacement strategies [8], Masked Language Modeling (MLM) using PLMs [9] or auto-regressive PLMs [9]. One of the initial studies that delved into effective data augmentation for NER involved replacing named entities (NEs) with NEs of the same type or substituting tokens within the sentence with synonyms retrieved from WordNet [10]. Subsequently, numerous neural learning systems were introduced, either altering the training objective of MLM using PLMs [6] or mBART [11], to generate entirely new sentences from scratch.

However, all these methods were designed for low-resource FNER, but for emerging and nested label complex entities, data augmentation in these domains remains underexplored.

B. Nested NER

Novel methods for NNER have recently been proposed and can be categorized into three main types: Sequence-to-sequence (Seq2Seq) methods, setting prediction methods [12], and Span-based methods, respectively. Yan et al. [13] formulated NER as an entity span sequence generation problem, using a BART-based Seq2Seq model [14] with a pointer network to solve most NER problems in a unified framework. Tan et al. [15] proposed a non-autoregressive decoder for predicting entity sets, formulating NER as an entity set prediction task. Yuan et al. [16] focused on span-based methods and introduced three imitating mechanisms to handle heterogeneous factors in span-based methods.

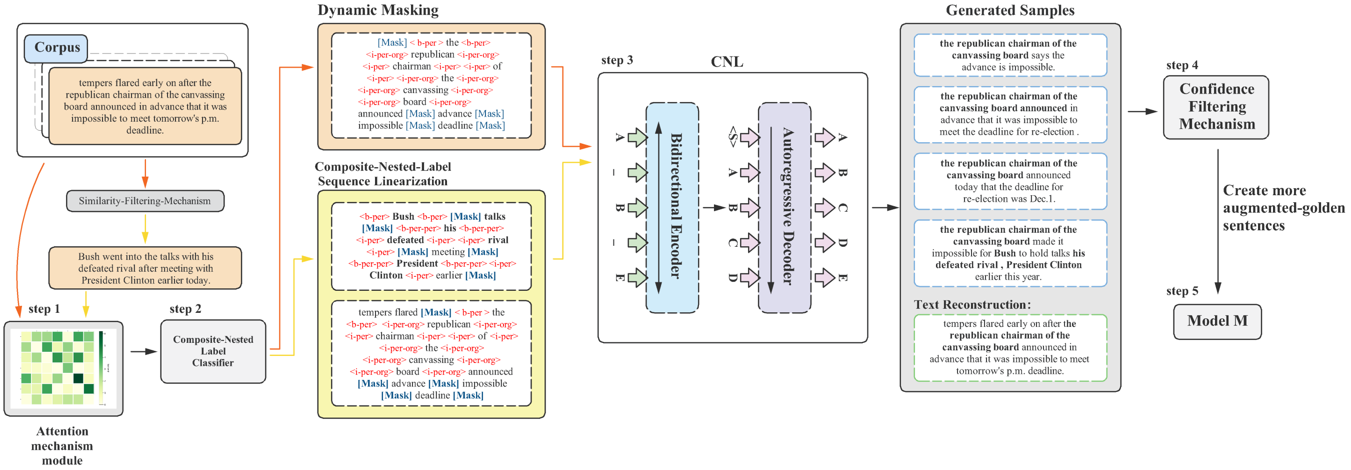
Previous research on NNER has mainly studied the relationships between tokens, labels, and boundaries. The current data augmentation techniques are not sufficient to address the challenges of NNER. Therefore, we focus on labels of nested words within each span, combining different labels present within one word to address the limitation of data augmentation in NNER. Additionally, a data filtering mechanism is introduced to ensure that the generated samples are more readable and fluent.

III. Method

In this section, we first introduce the label classifier about the CNLC in III-A and then describe our method for data augmentation in III-B. The overview of the CNL augmentation is shown in Fig. 2. The augmentation data requires selecting the CFM, and the selecting mechanism will be introduced in III-C.

A. Composite-Nested-Label Classification Template

Previous NNER studies have primarily focused on spans and tokens as they are distinctive markers with rich semantics, with the primary goal of identifying whether a span constitutes an entity. Many methods have been explored how accurately models traverse or enumerate all spans. As shown in Fig. 1 (middle), given an input sequence "*The Chinese embassy in France.*", enumerates each span to populate <label> of the template. In the data augmentation field, the approach can


Corpus
Similarity-Filtering-Mechanism
step 1
Attention mechanism module
step 2
Composite-Nested Label Classifier
Dynamic Masking
Composite-Nested-Label Sequence Linearization
step 3
CNL
Bidirectional Encoder
Autoregressive Decoder
A
_
B
_
E
<S>
A
B
C
D
A
B
C
D
E
Generated Samples
Text Reconstruction:
step 4
Confidence Filtering Mechanism
Create more augmented-golden sentences
step 5
Model M

tempers flared early on after the republican chairman of the canvassing board announced in advance that it was impossible to meet tomorrow's p.m. deadline.

Bush went into the talks with his defeated rival after meeting with President Clinton earlier today.

[Mask] < b-per > the <b-per> <i-per-org> republican <i-per-org> <i-per> chairman <i-per> <i-per> of <i-per> <i-per-org> the <i-per-org> <i-per-org> canvassing <i-per-org> <i-per-org> board <i-per-org> announced [Mask] advance [Mask] impossible [Mask] deadline [Mask]

<b-per> Bush <b-per> [Mask] talks [Mask] <b-per-per> his <b-per-per> <i-per> defeated <i-per> <i-per> rival <i-per> [Mask] meeting [Mask] <b-per-per> President <b-per-per> <i-per> Clinton <i-per> earlier [Mask]

tempers flared [Mask] < b-per > the <b-per> <i-per-org> republican <i-per-org> <i-per> chairman <i-per> <i-per> of <i-per> <i-per-org> the <i-per-org> <i-per-org> canvassing <i-per-org> <i-per-org> board <i-per-org> announced [Mask] advance [Mask] impossible [Mask] deadline [Mask]

the republican chairman of the canvassing board says the advance is impossible.

the republican chairman of the canvassing board announced in advance that it was impossible to meet the deadline for re-election .

the republican chairman of the canvassing board announced today that the deadline for re-election was Dec.1.

the republican chairman of the canvassing board made it impossible for **Bush** to hold talks **his defeated rival , President Clinton** earlier this year.

tempers flared early on after **the republican chairman of the canvassing board** announced in advance that it was impossible to meet tomorrow's p.m. deadline.

Fig. 2: Model architecture of CNL: Model CNL is divided into five steps, serving as the input to the model during fine-tuning and generation. Step 1: Similar sentences are obtained from the corpus using a similarity filtering mechanism. Then, the important keywords related to NEs are extracted using attention maps obtained from the fine-tuned RoBERTa model. Step 2: After adding label tokens before and after each entity in the sentences using CNLC, the sentences are divided into two parts. The original sentence template undergoes further masking, with a small portion of keywords dynamically masked. The other part is obtained by merging the sentence with similar sentences using a FUSION mechanism to create a template. Step 3: The CNL model is used to generate augmented data. Step 4: The generated samples are further filtered through the CFM to obtain high-confidence sentences, which are then concatenated with the golden data. Step 5: The obtained final data is used as the input for model M.

not address the issue of a single token having multiple labels in NNER. However, we've discovered that for nested words, the label of the word itself is a crucial determinant. From our Table I, It's evident that there is a correlation among labels. Therefore, We introduce CNLC and use it to construct a new template. Combining tokens and labels, allowing the model to learn labels for span perception and considering different labels' contributions to tokens, as shown in Fig. 1 (right). With this template, we can obtain the nested labels with a span by traversing it only once and can effectively address the problem of NNER in data augmentation.

B. Composited-Nested-Learning Model

For every sentence in our training dataset, we initially identify a set of none-Named Entity (n-NE) tokens that receive the highest attention from the Named Entities (NEs) within that sentence. These selected tokens are termed as keywords. In our research, an n-NE token is defined as a keyword when it holds the most contextual significance for the NEs within the sentence. The contextual dependency is assessed by utilizing the attention scores extracted from attention maps derived from a RoBERTa model that we fine-tuned with the golden dataset. Our primary objective is to identify the top k% of n-NE tokens, referred to as keywords. To ensure robustness, we make certain that no more than 10% of their combined attention is directed towards a single token [17]. Additionally, stop words, punctuation, and other NEs are excluded from the top k% of n-NE tokens to derive the ultimate set of keywords. Once the top k% of n-NE tokens in the sentence are designated as keywords, we end up with the K n-NE keywords and E entity tokens. To construct the template, we replace each n-NE token that is not part of the K keywords with a mask token and then eliminate consecutive mask tokens.

Inspired by Zhou et al. [6], we perform CNLC sequence linearization on the CNL template proposed in Section III-A, and subsequently incorporate label information into the fine-tuning and augmentation generation process. Similar to the approach [6], Label tokens are inserted both before and after each entity token, considering them as part of the regular context within the sentence. Furthermore, these label tokens positioned both before and after each NE offer boundary supervision, particularly for NEs that span multiple tokens.

After CNLC sequence linearization, This process is initiated by sampling a dynamic masking rate ϵ from a Gaussian distribution:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/(2\sigma^2)} \tag{1}$$

where the variance σ is set to $\frac{1}{K}$, and x is the keyword index list of the CNLC processed template.

Subsequently, we randomly select tokens from the set of K keywords in the sentence based on the masking rate ϵ, replace them with mask tokens, and then eliminate consecutive mask tokens. The aforementioned process serves two primary purposes: 1) It allows for the creation of different templates, leading to the generation of a more diverse set of sentences and enhancing sentence diversity. 2) The varied combinations of keywords and entities activate the model's generation capacity.

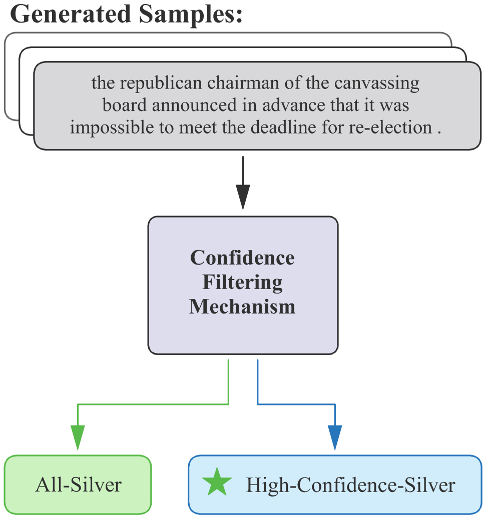

Generated Samples:
the republican chairman of the canvassing board announced in advance that it was impossible to meet the deadline for re-election .
Confidence Filtering Mechanism
All-Silver
High-Confidence-Silver

Fig. 3: After acquiring data-augmented samples, the samples are subsequently filtered using the CFM. Within the sample filtering process, sentences with low PLLs are excluded, and high-confidence sentences are retained as our final silver dataset.

As previously explained, the CNL model is fine-tuned on text with missing information, and it learns to recover the original text from the templates. This is employed as our fine-tuning objective, and templates are created that deviate from existing pre-training objectives through our strategy.

The sentences involved in the aforementioned process are referred to as sentence A. To retrieve sentence B similar to A, we traverse all sentences in the dataset and identify the top-n sentences with the highest similarity to A. To calculate the semantic similarity for each sentence in the training set, embeddings for each sentence are initially extracted using a multi-lingual Sentence-BERT [18]. Then compute the semantic similarity using the following method:

$$\text{sim}(S_a, S_b) = \frac{S_a \cdot S_b}{\|S_a\| \, \|S_b\|} \tag{2}$$

where $sim(\cdot)$ is the cosine similarity between two embeddings and S_a, S_b respectively represent the aforementioned Sentence A, B. In addition, a, $b \in N$ where $a \neq b$ and N is the size of the training set.

Upon obtaining the similar sentence B, we apply the aforementioned missing information template operation. After acquiring templates for both sentences A and B, we propose the use of FUSION, a novel template fusion algorithm that combines sentence A and sentence B, resulting in the creation of a new context-enriched sentence named C. Similarly, C is input into the CNL model, allowing the sentence to be regenerated by the model.

C. Confidence Filtering Mechanism

As a post-processing step, following the operation in Section III-B, an augmented dataset is obtained. Initially, we calculate the accuracy of the model's predicted-label compared to the golden-label of the original sentences. This results in two subsets: "silver" samples and "none-silver" samples. It is important to note that the "silver" category may contain sentences where only the labels match, but the sentences have low pseudo-log-likelihood scores (PLLs) [19]. Therefore, we propose CFM to effectively alleviate the issues mentioned above. By inputting the generated samples into a module and filtering them based on PPLs, we can construct a "high-confidence silver" dataset as shown in Fig. 3. When the "high-confidence silver" samples are concatenated with the original training set samples, we obtain an enriched NNER dataset. We can input the augmented data into Model M as part of its input.

TABLE II: Statistics and silver for ACE2004 and ACE2005

	ACE2004			ACE2005		
	Train	Dev	Test	Train	Dev	Test
#S	6198	742	809	7285	968	1058
#NS	2718	294	388	2797	352	339
#E	22204	2514	3035	24827	3234	3041
#NE	10159	1092	1417	10039	1200	1186
*S	5910	-	-	5404	-	-
*NS	1062	-	-	983	-	-
*E	14693	-	-	13778	-	-
*NE	1676	-	-	1416	-	-

#represents Statistics, and * represents Silver. We report the number of sentences(#S,*S), the number of sentences containing nested entities(#NS,*NS), the number of entities(#E,*E), and the number of nested entities(#NE,*NE) on the two datasets.

IV. Experiments and Results

A. Datasets

We conducted experiments on two datasets: ACE2004 and ACE2005. During the data augmentation phase, label filtering was initially performed on the ACE2004 and ACE2005 datasets. Because overly complex situations were not considered, sentences in which a single token has more than three nested labels were excluded. After data augmentation, the augmented corpus was concatenated with the original training data, resulting in an aug-golden dataset. Following previous work, we measured the results using span-level precision, recall, and F1 scores. Models were selected based on the performance of development sets. We followed Yu et al. [20] to train the model on the concatenation of the train and dev sets. Table II provides relevant information about the dataset.

B. Experimental Setup

CNL. XLM-RoBERTa-large with an additional linear head was employed in our attention selection CNL model. We approach the task as token-level classification, implementing the BIO tagging scheme. Our model optimization is carried out using the Adam optimizer, with a fixed learning rate of 1e^{-2}, and training is performed with a batch size of 8. The attention selection CNL model is trained for 100 epochs, with

TABLE III: Results in the standard nested NER setting.

Model	ACE2004			ACE2005		
	Pr.	Rec.	F1	Pr.	Rec.	F1
Biaffine [20]	87.30	86.00	86.70	85.20	85.60	85.40
BARTNER [13]	87.27	86.41	86.84	83.16	86.38	84.74
UIE [21]	-	-	86.89	-	-	85.78
BuParser [22]	86.60	87.28	86.94	84.61	86.43	85.53
ERPG [23]	-	-	86.99	-	-	86.38
PromptNER* [24]	86.91	87.67	87.29	85.22	88.03	86.60
CNL	**87.12 (+0.21)**	**88.27 (+0.60)**	**87.69 (+0.40)**	**85.93 (+0.71)**	**88.60 (+0.57)**	**87.25 (+0.65)**

Bold and underline indicate the best and the second-best scores

*Baseline result given by our implementation. Other results are derived from related papers

an attention mask rate of 0.3. The model that exhibits the best performance on the development set is selected for testing.

We use mBart-50-large equipped with a condition generation head to enhance the performance of CNL. The fine-tuning process for CNL spans 10 epochs, facilitated by the Adam optimizer with a learning rate of 1e^{-5}. A batch size of 16 is used in dataset ACE2004, and a batch size of 8 is used in dataset ACE2005. In the FUSION module, the attention mask rate is set to 0.3.

PromptNER. PromptNER has been selected as our model M, and this model serves as our baseline. Our results are compared based on our reimplementation of PromptNER. As a result of the addition of more samples, the decision has been made to extend the training period for better refining the model. The original epoch value has been adjusted from 50 to 60 while keeping all other parameters unchanged. Similarly, we have chosen recent competitive models as our baseline, including parsing-based [20], generation-based [13] [21], span-based [22] [16], Prompt Guidance [23] [24]. These approaches utilize different pre-trained language models as the encoder. Therefore, in our experimental results, performance is presented using BERT-large as the benchmark.

Result. Table III illustrates the performance of data augmentation through CNL, the performance of the current state-of-the-art (SOTA) models in the NNER field can be further improved. The silver dataset can be extended to other models.

C. Parametric Search

After enhancing the data, the presence of sentences in the new silver dataset containing repeated occurrences of simple words or phrases was observed. Although the predicted labels matched the golden labels, these sentences lacked coherence and had low reference values. To address these low-confidence samples, the CFM was introduced to assign confidence scores to the samples. The higher the confidence score, the more complete the sentence. To select the best samples, we varied the proportion of silver samples (50%, 60%, 70%, 75%). As shown in Fig. 4, it can be noted that selecting 70% of the silver samples in ACE2004 yielded the best results. Using the same approach, we achieved the optimal outcome by choosing 35% of the silver samples in ACE2005.

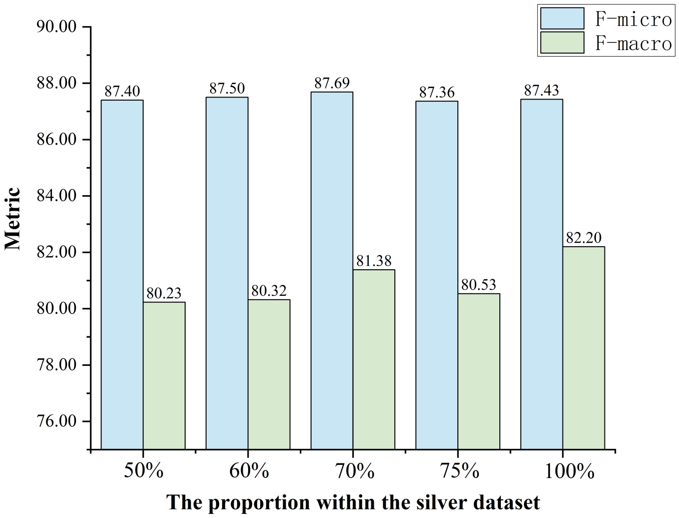


Fig. 4: A parameter search was conducted for the silver dataset generated for ACE2004. The Rate represents the proportion of silver selected, and we tested it using model M, obtaining F-micro and F-macro scores at different proportions.

D. Sample Imbalance Inquiry

Following data augmentation with the CNL model, it is observable from Table I that the distribution of the dataset tends to exhibit bias towards the first few classes, leading to an excessive leaning of the model towards the majority class. Through the data augmentation methods presented in Table IV, We appropriately expanded some underrepresented labels, which alleviated the problem of sample imbalance to a certain extent and enhanced the balance of the model among different categories. Similarly, alleviating sample imbalance can help the model generalize better to diverse data from various categories, thereby improving the model's practicality and robustness.

V. Conclusion

This work introduces a CNL model for NNER and its application in data augmentation. We demonstrate the existence of correlations between labels, and based on these label correlations, we propose CNLC templates to address the

TABLE IV: Results in sample imbalance inquiry.

	ACE2004	ACE2004*	ACE2005	ACE2005*
WEA	75.29	71.60	78.85	79.21
PER	91.04	91.50	90.01	90.29
FAC	71.53	67.67	77.66	77.70
GPE	89.57	89.64	85.96	88.50
ORG	81.85	82.58	82.66	82.50
VEH	91.67	100.00	74.49	75.90
LOC	64.47	66.67	70.09	74.77
Fmacro	80.77	**81.38**	79.96	**81.27**

ACE2004 and ACE2005 for model PromptNER* results given by our implementation. ACE2004* and ACE2005* for CNL.

challenge of applying NNER in data augmentation. For the enhancement of generated silver samples, we introduce the CFM, effectively helping us filter sentences with high confidence. Our approach focuses on how to augment NNER samples, making them applicable to other models and providing more valuable samples. Through these samples, it can improve the model's performance and alleviate sample imbalance. In future work, we will endeavor to apply our method to few-shot NNER and explore methods for addressing more complex nested entity labels.

VI. Acknowledgment

Our work is supported by multiple funds in China, including Guangzhou Science and Technology Planning Project (2023B01J0001), the Key Program of NSFC-Guangdong Joint Funds (U2001201, U1801263), Industrial core and key technology plan of Zhuhai City (ZH22044702190034-HJL). Our work is also supported by Guangdong Provincial Key Laboratory of Cyber-Physical System (2020B1212060-069).

References

[1] F. Luo, P. Fang, Q. Qiu, and H. Xiao, "Features induction for product named entity recognition with crfs," in *Proceedings of the 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. IEEE, 2012, pp. 491–496.

[2] C. Tan, W. Qiu, M. Chen, R. Wang, and F. Huang, "Boundary enhanced neural span classification for nested named entity recognition," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 05, 2020, pp. 9016–9023.

[3] Y. Fu, C. Tan, M. Chen, S. Huang, and F. Huang, "Nested named entity recognition with partially-observed treecrfs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 14, 2021, pp. 12 839–12 847.

[4] B. Ding, L. Liu, L. Bing, C. Kruengkrai, T. H. Nguyen, S. Joty, L. Si, and C. Miao, "Daga: Data augmentation with a generation approach for low-resource tagging tasks," *arXiv preprint arXiv:2011.01549*, 2020.

[5] J. Kim, Y. Kim, and S. Kang, "Weakly labeled data augmentation for social media named entity recognition," *Expert Systems with Applications*, vol. 209, p. 118217, 2022.

[6] R. Zhou, X. Li, R. He, L. Bing, E. Cambria, L. Si, and C. Miao, "Melm: Data augmentation with masked entity language modeling for low-resource ner," *arXiv preprint arXiv:2108.13655*, 2021.

[7] J. Yan, J. Xu, C. Song, C. Wu, Y. Li, and Y. Zhang, "Understanding in-context learning from repetitions," *arXiv preprint arXiv:2310.00297*, 2023.

[8] J. Wei and K. Zou, "Eda: Easy data augmentation techniques for boosting performance on text classification tasks," *arXiv preprint arXiv:1901.11196*, 2019.

[9] V. Kumar, A. Choudhary, and E. Cho, "Data augmentation using pre-trained transformer models," *arXiv preprint arXiv:2003.02245*, 2020.

[10] X. Dai and H. Adel, "An analysis of simple data augmentation for named entity recognition," *arXiv preprint arXiv:2010.11683*, 2020.

[11] S. Ghosh, U. Tyagi, M. Suri, S. Kumar, S. Ramaneswaran, and D. Manocha, "Aclm: A selective-denoising based generative data augmentation approach for low-resource complex ner," *arXiv preprint arXiv:2306.00928*, 2023.

[12] H. Li, J. Song, W. Zheng, and Y. Xiao, "Scoring prediction model focusing on local feature interaction of text," in *2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. IEEE, 2023, pp. 320–325.

[13] H. Yan, T. Gui, J. Dai, Q. Guo, Z. Zhang, and X. Qiu, "A unified generative framework for various ner subtasks," *arXiv preprint arXiv:2106.01223*, 2021.

[14] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[15] Z. Tan, Y. Shen, S. Zhang, W. Lu, and Y. Zhuang, "A sequence-to-set network for nested named entity recognition," *arXiv preprint arXiv:2105.08901*, 2021.

[16] Z. Yuan, C. Tan, S. Huang, and F. Huang, "Fusing heterogeneous factors with triaffine mechanism for nested named entity recognition," *arXiv preprint arXiv:2110.07480*, 2021.

[17] K. Clark, U. Khandelwal, O. Levy, and C. D. Manning, "What does bert look at? an analysis of bert's attention," *arXiv preprint arXiv:1906.04341*, 2019.

[18] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[19] J. Salazar, D. Liang, T. Q. Nguyen, and K. Kirchhoff, "Masked language model scoring," *arXiv preprint arXiv:1910.14659*, 2019.

[20] J. Yu, B. Bohnet, and M. Poesio, "Named entity recognition as dependency parsing," *arXiv preprint arXiv:2005.07150*, 2020.

[21] Y. Lu, Q. Liu, D. Dai, X. Xiao, H. Lin, X. Han, L. Sun, and H. Wu, "Unified structure generation for universal information extraction," *arXiv preprint arXiv:2203.12277*, 2022.

[22] S. Yang and K. Tu, "Bottom-up constituency parsing and nested named entity recognition with pointer networks," *arXiv preprint arXiv:2110.05419*, 2021.

[23] X. Zhu, F. Dai, X. Gu, H. Fan, B. Li, and W. Wang, "Erpg: Enhancing entity representations with prompt guidance for complex named entity recognition," in *2023 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2023, pp. 2813–2818.

[24] Y. Shen, Z. Tan, S. Wu, W. Zhang, R. Zhang, Y. Xi, W. Lu, and Y. Zhuang, "Promptner: Prompt locating and typing for named entity recognition," *arXiv preprint arXiv:2305.17104*, 2023.