Rethinking Generalization of Neural Models: A Named Entity Recognition Case Study

Jinlan Fu†*, Pengfei Liu♯*, Qi Zhang†, Xuanjing Huang†
†School of Computer Science, Fudan University,
♯ Language Technologies Institute, Carnegie Mellon University,
{fujl16,qz,xjhuang}@fudan.edu.cn, pliu3@andrew.cmu.edu

Abstract

While neural network-based models have achieved impressive performance on a large body of NLP tasks, the generalization behavior of different models remains poorly understood: Does this excellent performance imply a perfect generalization model, or are there still some limitations? In this paper, we take the NER task as a testbed to analyze the generalization behavior of existing models from different perspectives and characterize the differences of their generalization abilities through the lens of our proposed measures, which guides us to better design models and training methods. Experiments with in-depth analyses diagnose the bottleneck of existing neural NER models in terms of breakdown performance analysis, annotation errors, dataset bias, and category relationships, which suggest directions for improvement. **We have released the datasets: (*ReCoNLL*, *PLONER*) for the future research at our project page:** http://pfliu.com/InterpretNER/.[1].

1 Introduction

Neural network-based models have achieved great success on a wide range of NLP tasks (Devlin et al. 2018; Bahdanau, Cho, and Bengio 2014). However, the generalization behaviors of neural networks remain largely unexplained. Recently, some researchers are beginning to realize this problem and attempt to understand the generalization behavior of neural networks in terms of network architectures or optimization procedure (Zhang et al. 2016; Baluja and Fischer 2017; Schmidt et al. 2018). However, it is incomplete to ignore the characteristics of tasks and datasets for generalization analysis since it not only depends on the model's architectures but on the data itself (Arpit et al. 2017).

In NLP, there is a massive gap between the growing task performance and the understanding of model generalization behavior. Many tasks have reached a plateau in the performance on a particular dataset (Rajpurkar, Jia, and Liang 2018; Devlin et al. 2018), which calls for a data-dependent understanding of models' generalization behavior.

In this paper, we take a step further towards diagnosing and characterizing generalization in the context of a specific task. Concretely, we take named entity recognition (NER) task as a study case and investigate three crucial yet rarely raised questions through *entity*- and *class*-centric generalization analyses.

Q1: Does our model really have generalization ability, or it just pretends to understand and make some shallow template matches as observed in (Jia and Liang 2017)? We devise a measure, which can break down the test set into different interpretable groups, helping us diagnosing inadequacies in the generalization of NER models (Sec. 4.1). Furthermore, this measure makes it easier to find human annotation errors, which cover the actual generalization ability of the existing models (Sec. 4.1). **Q2**: What factor of a dataset can distinguish neural networks that generalize well from those that don't? We introduce two metrics to quantify the dataset bias in a cross-dataset experimental setting, enabling us to better understand how the dataset bias influences the models' generalization ability (Sec. 4.2). **Q3**: How does the relationship between entity categories influence the difficulty of model learning? Our class-centric analysis shows that if two categories, e.g. $\mathscr{C}_1$ and $\mathscr{C}_2$, have overlaps (i.e. sharing a subset of entities), then most of the errors on $\mathscr{C}_1$ made by the model are due to mistakenly predicting $\mathscr{C}_1$ as $\mathscr{C}_2$ (Sec. 4.3). Our experiment results show the prospects for further gains for these problems from novel architecture design and knowledge pre-training seem quite limited (Sec. 4.3). Tab. 1 shows the framework of our experimental designs.

Main Contributions This paper understands the generalization behavior from multiple novel angles, which contributes from the following two perspectives: 1) For the task itself, we identify the bottleneck of existing methods on the NER task in terms of breakdown performance analysis, annotation errors, dataset bias, and category relationships, which suggest directions for improvement and can drive the progress of this area. 2) Other tasks can benefit from the research evidence found in this study. For example, this paper not only shows that utilizing less but more relevant data can achieve better performance, but also provides an effective and princi-

*These two authors contributed equally

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

[1]As a by-product of this paper, we have open-sourced a project that involves a comprehensive summary of recent NER papers and classifies them into different research topics: https://github.com/pfliu-nlp/Named-Entity-Recognition-NER-Papers.

pled way to select more relevant training samples.

Observations Our findings are summarized as follows: (1) The fine-grained evaluation based on our proposed measure reveals that the performance of existing models (including the state-of-the-art model) heavily influenced by the degree to which test entities have been seen in training set *with the same label* (Exp-I in Sec. 4.1). (2) The proposed measure enables us to detect human annotation errors, which cover the actual generalization ability of the existing model. We observe that once these errors are fixed, previous models can achieve new state-of-the-art results, 93.78 F1-score on CoNLL2003, which refers to Exp-II in Sec. 4.1. (3) We introduce two measures to characterize the data bias and the cross-dataset generalization experiment shows that the performance of NER systems is influenced not only by whether the test entity has been seen in the training set but also by whether the context of the test entity has been observed (Exp-III in Sec. 4.2). (4) Providing more training samples is not a guarantee of better results. A targeted increase in training samples will make it more profitable, which refers to Exp-IV in Sec. 4.2. (5) The relationship between entity categories influences the difficulty of model learning, which leads to some hard test samples that are difficult to solve using common learning methods, which refer to Exp-V and Exp-VI in Sec. 4.3.

2 Related Work

Our work can be uniquely positioned in the context of the following two aspects.

Neural Network-based Models for NER Some researchers design different architectures which vary in word encoder (Chiu and Nichols 2016; Ma and Hovy 2016), sentence encoder (Huang, Xu, and Yu 2015; Ma and Hovy 2016; Chiu and Nichols 2016) and decoder (CRF) (Huang, Xu, and Yu 2015). Some works explore how to transfer learned parameters from the source domain to a new domain (Chen and Moschitti 2019; Lin and Lu 2018; Cao et al. 2018). Recently, (Yang, Liang, and Zhang 2018; Reimers and Gurevych 2017) systematically analyze neural NER models to provide useful guidelines for NLP practitioners. Different from the above works, instead of exploring the possibility for a new state-of-the-art in this paper, we aim to bridge the gap between the growing task performance and the understanding of model generalization behavior.

Analyzing Generalization Ability of Neural Networks Most existing works have analyzed the generalization power of DNNs by quantifying how the number of parameters, noise label, regularization, influence training process on a set of simple classification tasks. (Fort, Nowak, and Narayanan 2019) investigate neural network training and generalization by introducing a measure and study how it varies with training iteration and learning rate. (Zhang et al. 2016; Arpit et al. 2017) explore the generalization of the neural network by showing how the impact of representational capacity changes with varying noise levels and regularization. The goal of this paper is to study in the light of a specific NLP task, discussing how neural networks achieve linguistic generalization abilities. More recently, (Zhong et al. 2019) try to analyze the generalization of neural networks in the text summarization task, which is mainly performed from a dataset perspective.

3 Task, Methods, and Datasets

3.1 Task Description

Named entity recognition (NER) is usually formulated as a sequence labeling problem (Borthwick et al. 1998). Formally, let $X = \{x_1, x_2, \ldots, x_T\}$ be an input sequence and $Y = \{y_1, y_2, \ldots, y_T\}$ be the output tags. The goal of this task is to estimate the conditional probability: $P(Y|X) = P(y_t|X, y_1, \cdots, y_{t-1})$

Why do We Choose the NER Task? The goal of this paper is to study how neural networks achieve linguistic-level generalization abilities via the lens of a well-chosen NLP task. Compared with other general classification tasks, the NER task is particularly suitable here because 1) it contains more category labels; 2) different categories contain a number of training samples, which provides an ideal testbed for us to observe the generalization behavior of neural networks. Although our focus is on NER tasks, our solution can be ported to the other tagging problems.

3.2 Neural Network-based Methods for NER

To evaluate the importance of different components of the NER systems, we varied our models mainly in terms of three aspects: different choices of character-, word-, and sentence-level encoders and decoders. Tab.2 illustrates the models we have studied in this paper. Specifically, for Exp-I, we mainly focus on how different choices of pre-trained models (Mikolov et al. 2013; Peters et al. 2018; Devlin et al. 2018) influence systems' generalization abilities. All models adopt LSTM as sentence encoder and CRF as the decoder. For Exp-III and Exp-IV, we use *CnoneWrandlstmCrf* model to achieve cross-dataset generalization evaluation.

For Exp-V, we adopt *CcnnWglovelstmMLP* architecture since *MLP* decoder is easy to compute the measure *Consistency*. For Exp-VI, detailed choices of evaluated models are listed in Tab.7.

3.3 NER Datasets for Evaluation

We conduct experiments on three benchmark datasets: the CoNLL2003 NER dataset, the WNUT16 dataset, an d OntoNotes 5.0 dataset. The CoNLL2003 NER dataset (Sang and De Meulder 2003) is based on Reuters data (Collobert et al. 2011). WNUT16 dataset is provided by the second shared task at WNUT-2016. The OntoNotes 5.0 dataset (Weischedel et al. 2013) is collected from telephone conversations (TC), newswire (NW), newsgroups, broadcast news (BN), broadcast conversation (BC) and weblogs (WB), pivot text (PT) and magazine genre (MZ). Due to the lack of NER labels of PT and the insufficient amount of data of TC, we only evaluate the other five domains.

Views	Q.	Measures	Applications
Entity	Q1 (Sec. 4.1)	Entity Coverage Ratio	(Exp I) Breaking down the test set (Exp-II) Annotation errors detecting and fixing
	Q2 (Sec. 4.2)	Expectation of Coverage Ratio Contextual Coverage Ratio	(Exp-III) Cross-dataset generalization (Exp-IV) Order matters for data augmentation
Category	Q3 (Sec. 4.3)	Consistency	(Exp V) Probing inter-category relationships (Exp-VI) Exploring the errors of hard cases

Table 1: Outline of our experiment designs. Q1: Does our model really have generalization? Q2: What factor of a dataset can distinguish neural networks that generalize well from those that don't? Q3: How does the relationship between entity categories influence the difficulty of model learning?

Models	Character					Word			Sentence		Decoder	
	none	cnn	elmo	flair	bert	none	rand	glove	lstm	cnn	crf	mlp
CcnnWglovecnnMlp		√						√		√		√
CcnnWglovelstmMlp		√						√	√			√
CcnnWglovecnnCrf		√						√		√	√	
CnoneWglovelstmCrf	√							√	√		√	
CcnnWglovelstmCrf		√						√	√		√	
CelmWnonelstmCrf			√			√			√		√	
CelmWglovelstmCrf			√					√	√		√	
CnoneWrandlstmCrf	√						√		√		√	
CflairWglovelstmCrf				√				√	√		√	
CbertWnonelstmCrf					√			√	√		√	

Table 2: Neural NER systems with different architectures and pre-trained knowledge, which we studied in this paper.

4 Experiment and Analysis

4.1 Diagnosing Generalization with *Entity Coverage Ratio*

The generalization ability of neural models is often evaluated based on a holistic metric over the whole test set. For example, the performance of the NER system is commonly measured by the F1 score. Despite its effectiveness, this holistic metric fails to provide fine-grained analysis and as a result, we are not clear about what the strengths and weaknesses of a specific NER system are.

Driven by **Q1**, we propose to shift the focus of evaluation from a holistic way to fine-grained way, navigating directly to the parts which influence the generalization ability of neural NER models. We approach the above end by introducing the notion of *entity coverage ratio* (ECR) for each test entity, by which the test set will be divided into different sub-sets, and the overall performance could be broken down into interpretable categories.

Entity Coverage Ratio (ECR) The measure entity coverage ratio is used to describe the degree to which entities in the test set have been seen in the training set with the same category. Specifically, we refer to e_i as a test entity, whose coverage ratio is defined as:

$$\rho(e_i) = \begin{cases} 0 & C = 0 \\ (\sum_{k=1}^{K} \frac{\#(e_i^{tr,k})}{C^{tr}} \dot{\#}(e_i^{te,k}))/C^{te} & \text{otherwise} \end{cases} \quad (1)$$

$\rho(e)$	Interpretation
$\rho = 1$	Entity e appears in train set with only label k
$\rho \in (0, 1)$	Entity e appears in train set with diverse labels
$\rho = 0 \wedge C \neq 0$	Entity e appears in train set but without label k
$\rho = 0 \wedge C = 0$	Entity e doesn't appear in train set

Table 3: Interpretation of ρ with different values.

where $e_i^{tr,k}$ is the entity e_i in the training set with ground truth label k, $e_i^{te,k}$ is the entity e_i in the test set with ground truth label k, $C^{tr} = \sum_{k=1}^{K} \#(e_i^{tr,k})$, $C^{te} = \sum_{k=1}^{K} \#(e_i^{te,k})$, and $\#$ denotes the counting operation.

For example, in the training set, "`chelsea`" is labeled as the category `Person` 6 times, and `Organization` 4 times, while in the test set, labeled as `Person` 3 times and `Organization` 2 times, so ρ ("`chelsea`") $= (0.6 \times 3 + 0.4 \times 2)/3 = 0.52$. According to Eq.1, we can investigate the relationship between the coverage ratio of the entity e_i and model's generalization ability on this entity. The possible values of $\rho(e_i)$ and their corresponding interpretation can be found in Tab. 3.

Exp-I: Breaking Down the Test Set

Instead of utilizing a holistic metric on the whole dataset, we break down the test set into interpretable regions by the measure ρ and then observe how the generalization ability of the NER models varies with it.

Results Based on Tab. 4 and driven by **Q1**, our observations are: 1) In general, the part of test entities with high performance are usually the ones that appear in the training set. By contrast, if the test entity is unseen, it will achieve a lower performance. 2) No matter what level (character or word) pre-trained embeddings are introduced, the performances of unseen entities are largely improved. 3) Comparing two different levels of pre-trained methods, `ELMo` and `FLAIR` achieve better performances on unseen entities but have not shown significant gain on seen entities. 4) Compared to `Rand`, `CNN` shows its superior performance on the prediction of unseen entities. 5) For different parts of the test set, we find $C \neq 0$ is the most challenging part (even for

Datasets	Embed-layer		Entity Coverage Rate					
	Char	**Word**	**Overall**	1	$(0.5, 1)$	$(0, 0.5]$	$C \neq 0$	$C = 0$
CoNLL	CNN	-	76.42	79.94	86.99	78.84	69.74	77.61
	FLAIR	-	89.98	95.30	95.58	82.39	72.16	90.39
	ELMo	-	91.79	97.61	95.98	85.15	71.43	92.22
	BERT	-	91.34	97.72	95.17	86.66	**77.83**	92.37
	-	Rand	78.43	95.05	94.75	73.54	37.97	66.40
	-	GloVe	89.10	98.44	96.31	81.34	57.80	87.23
	CNN	Rand	82.88	94.13	94.48	74.25	47.78	78.91
	CNN	GloVe	90.33	98.32	95.94	80.33	59.67	89.74
	ELMo	GloVe	92.46	98.08	**96.46**	86.14	69.79	93.08
	FLAIR	GloVe	**93.03**	**98.56**	96.38	**87.07**	73.58	**93.42**
WNUT	CNN	-	20.88	45.99	67.01	40.25	19.14	19.74
	FLAIR	-	41.49	81.15	88.14	54.36	39.56	43.44
	ELMo	-	43.70	88.72	90.83	55.56	**44.19**	43.32
	BERT	-	44.08	77.75	81.61	49.74	34.65	41.92
	-	Rand	14.97	60.62	83.84	50.00	3.90	4.77
	-	GloVe	37.28	89.29	92.62	45.65	35.34	35.15
	CNN	Rand	22.29	48.88	71.43	39.08	16.75	18.83
	CNN	GloVe	40.72	86.12	**92.24**	49.74	26.67	40.06
	ELMo	GloVe	45.33	90.38	89.92	56.57	37.8	46.58
	FLAIR	GloVe	**45.96**	**90.52**	89.92	**61.69**	42.07	**48.38**

Table 4: The breakdown performance on CoNLL and WNUT datasets with different pre-training strategies, which is based on the LSTM as sentence encoder and CRF as the decoder. "`Rand`" represents the word representations are randomly initialized. "`Overall`" denotes the F1 score on the whole test set and the names of the last five columns correspond to ρ definition in Tab.3.

the state-of-the-art model), followed by $C = 0$ and $(0, 0.5]$. Interestingly, on the CoNLL dataset, we find that if the test entity is labeled as a different category in the training set, it will be more difficult to learn compared with entities which have not been seen in the training set. 6) *We find that the character- and the word-level pre-trained embeddings are complementary to each other*. Combining these two types of *pre-trained* knowledge will further improve the performance by a considerable margin.

Exp-II: Annotation Errors Detecting and Fixing

For each test entity with tag k, the measure *ECR* quantifies its *label ambiguity*: the proportion that this entity is labeled as k in the training set. Its intriguing property could help us find the annotation errors of the dataset.

Detecting Errors Specifically, since ρ measures the degree to which entities in the test set have been seen in the training set with the same label, the value of ρ within some ranges suggests that corresponding entities are more prone to annotation errors, such as $\rho = 0, C \neq 0$ (entity e^k appeared in train set but without label k) and $\rho \in (0, 0.5]$ (entity e^k appeared in train set with diverse labels).

Fixing Errors While researchers have been aware of annotation errors, such as on the tasks of Part-of-Speech (Manning 2011) and Chinese word segmentation (Ma, Ganchev, and Weiss 2018), yet few attempts have been made to fix them. The significance of correcting annotation errors for tagging tasks has been originally mentioned by (Manning 2011). In this paper, we argue that fixing annotation errors can not only boost the NER performance, but can reflect the true generalization ability of the existing models, making it possible to identify the real weaknesses of current systems.

Evaluation on Revised CoNLL (*ReCoNLL*) Many errors and inconsistencies in NER datasets are quite non-systematic and are hard to fix by deterministic rules. Therefore, we manually fixed errors with the instruction of the measure ECR (entity coverage ratio). Finally, we corrected 65 sentences in the test set, and 14 sentences in training set. When the revised dataset is ready, we re-train several typical NER models and make a comparison to the original ones.

The results are shown in Tab. 5. We find that once these errors are fixed, the performance of all these models has been improved, which indicates that human annotation errors cover the actual generalization ability of the existing model. Notably, the NER model *FLAIR* has driven the state-of-the-art result to a new level.

4.2 Measuring Dataset Bias

To answer the question **Q2**: "what factor of a dataset can distinguish neural networks that generalize well from those that don't", in this section, we introduce two measures, which can quantify the relationship of entities between training and test sets from dataset-level and help us understand the generalization behavior.

Expectation of Entity Coverage Ratio (EECR) Here, we define the expectation of the coverage ratio over all entities

Model	CoNLL	ReCoNLL
(Devlin et al. 2018)	92.80	-
(Peters et al. 2018)	92.22	-
(Akbik, Blythe, and Vollgraf 2018)	93.09	-
(Akbik, Bergmann, and Vollgraf)	**93.18**	-
Our Implementation		
GloVe	89.10	89.85
ELMo	91.79	92.65
BERT	91.34	92.16
FLAIR	93.03	**93.78**

Table 5: The test performance (F1 score) on CoNLL 2003 and its revised version.

in test data as $E_\rho(e)$ as follows:

$$E_\rho(e) = \sum_{i \in N_e} \rho(e_i) * \text{freq}(e_i) \quad (2)$$

in which N_e denotes the number of unique test entities and $\text{freq}(e_i)$ represents the frequency of the test entity e_i.

This index measures the degree to which the test entities have been seen in the training set. A higher value is suggestive of a larger proportion of entities with high coverage ratio.

Contextual Coverage Ratio (CCR) We introduce a notion of η to measure the contextual similarity of entities belonging to the same category but from the training and the test sets, respectively.

$$\eta^k(D_{tr}, D_{te}) = \sum_{f_i \in \phi^k_{te}} \sum_{f_j \in \phi^k_{tr}} p_{f_i} p_{f_j} \text{Sim}(v_{f_i}, v_{f_j}) \quad (3)$$

where k denotes the category of an entity. D_{tr} and D_{te} represents the training and test sets. ϕ^k_{tr} denotes a set of the high-frequency contextual patterns in which entities in training set reside in. We set the window size to 3, and choose 30 bigrams and 20 trigrams, then we obtain their vector representation v_{f_*} of each word span using BERT followed by a mean operation. $\text{Sim}(\cdot)$ is a cosine-similarity function. p_{f_i} is the probability of the contextual pattern f_i, which is using the frequency of the contextual pattern divided by the total contextual patterns' frequency.

Exp-III: Cross-dataset Generalization

The Expectation of Entity Coverage Ratio and Contextual Coverage Ratio can measure the similarity between training and test set from a different perspective. Next, we show how these two measures correlate with the model's performance by a cross-dataset generalization experiment.

Data Construction: *PLONER* We re-purpose a dataset for cross-domain generalization evaluation, in which three types of entities (PERSON, LOCATION, ORGANIZATION) from different domains are involved, therefore named "**PLO**NER" dataset. Specifically, we pick a set of representative NER datasets including: `WNUT16`, `CoNLL03`, `OntoNotes-bn`, `OntoNotes-wb`, `OntoNotes-mz`, `OntoNotes-nw`, and `OntoNotes-bc`. These datasets use disparate entity classification schemes, which makes it hard to conduct zero-shot transfer. We collapse types into standard categories used in the MUC (Grishman and Sundheim 1996) competitions (PERSON, LOCATION, ORGANIZATION) and the other categories are dropped. [2] To be fair, we limited the number of samples in each dataset to the same 2,500.

Matrix	Train	WN.	Co.	BN	WB	MZ	NW	BC	P-row
M_{F1}	WN.	**46.6**	16.7	12.0	13.8	11.4	6.50	11.7	0.98
	Co.	22.2	**70.2**	19.4	17.5	12.7	19.8	17.3	0.93
	BN	24.4	35.2	**65.7**	35.4	28.6	36.2	42.3	0.93
	WB	19.6	25.6	28.0	**55.7**	17.3	28.3	25.8	0.83
	MZ	15.3	25.0	29.7	21.8	**67.8**	32.2	26.6	0.89
	NW	22.2	24.2	32.6	29.3	24.6	**68.4**	26.9	0.87
	BC	28.6	29.1	41.1	43.6	25.5	33.2	**70.3**	0.90
	P-col	0.93	0.97	0.97	0.83	0.96	0.94	0.96	0.88
M_ρ	WN.	**1.00**	0.26	0.29	0.21	0.07	0.24	0.31	0.96
	Co.	0.325	**1.00**	0.35	0.37	0.29	0.36	0.37	0.94
	BN	0.35	0.33	**1.00**	0.49	0.32	0.45	0.65	0.95
	WB	0.27	0.22	0.43	**1.00**	0.29	0.34	0.52	0.91
	MZ	0.15	0.15	0.24	0.15	**1.00**	0.31	0.23	0.99
	NW	0.20	0.27	0.38	0.34	0.34	**1.00**	0.41	0.92
	BC	0.28	0.22	0.52	0.49	0.35	0.36	**1.00**	0.87
	P-col	0.95	0.95	0.91	0.89	0.97	0.92	0.92	0.78
M_ϕ	WN.	**1.00**	0.06	0.08	0.11	0.07	0.03	0.12	0.99
	Co.	0.176	**1.00**	0.20	0.35	0.20	0.13	0.27	0.91
	BN	0.19	0.22	**1.00**	0.56	0.34	0.18	0.65	0.88
	WB	0.29	0.26	0.41	**0.88**	0.42	0.23	0.70	0.70
	MZ	0.32	0.28	0.43	1.00	**1.00**	0.33	0.68	0.57
	NW	0.36	0.22	0.42	0.90	0.53	**1.00**	0.69	0.70
	BC	0.26	0.22	0.40	0.52	0.32	0.19	**1.00**	0.90
	P-col	0.85	0.97	0.96	0.45	0.93	0.93	0.83	0.89

Table 6: Illustration of F1 score, EECR, and CCR on cross-dataset setting. `P-row` and `P-col` represent row- and column-wise Pearson correlation coefficient. `Green`, `Pink` and `Yellow` regions denote the correlation between M_{F1} and M_ρ+M_ϕ, M_ρ, M_ϕ respectively. The `Blue` is the overall correlation coefficient.

Results Tab. 6 shows the cross-dataset expectation of coverage ratio (M_ρ), contextual coverage ratio (M_ϕ), and F1 score (M_{F1}). Each column corresponds to the performance when testing on one dataset and training on each of other datasets. We detail our findings as follows:

1) The diagonal elements of the M_{F1} achieve the highest values, which suggests that models generalize poorly on the samples from different distributions (domains).

2) The highest values are also achieved on the diagonal in M_ρ and M_ϕ. Additionally, from the values of Pearson coefficient, we could find the two measures: expectation of

[2] We have released the dataset.

coverage ratio (M_ρ), contextual coverage ratio (M_ϕ) correlate closely with F1-score M_ρ.

3) Column-wisely, given a test dataset, ρ, ϕ, and $F1$ can usually achieve the highest values on the same training set, which suggests we can select the most useful training sets through the measures ρ and ϕ when the distribution to be tested is given and we have some samples from it as the validation set.

4) Given a test set, the training set with higher EECR (Expectation of Entity Coverage Ratio) value could also obtain a lower F1 score, since **entity coverage ratio is not the only factor that effects generalization and the contextual coverage ratio also matters**.

A significant case can be found in Tab. 6 (numbers in boxes), taking the `WB` as a test set, we observe that `WNUT` and `CoNLL` have higher ECR(ρ) value than `MZ` while obtaining lower $F1$ score. We can speculate the reason from the ϕ-M, that the contextual coverage ratio between `WB` and `MZ` is much higher than utilizing `WNUT` and `CoNLL` as training sets. The above results show that *the generalization ability of NER models is influenced not only by whether the test entity has been seen in the training set but also by whether the context of the test entity has been seen.*

Exp-IV: Order Matters for Data Augmentation

The measure ECCR can be used to quantify the importance of different source domains, therefore allowing us to select suitable ones for data augmentation. Next we will show how to utilize the ECCR metric to make better choices of source domains from the seven candidates: `WNUT16`, `CoNLL03`, `OntoNotes-bn`, `OntoNotes-wb`, `OntoNotes-mz`, `OntoNotes-nw`, and `OntoNotes-bc`. We take `WNUT` as the tested object and continuously increase the training samples of above seven datasets in three ways: 1) random order of EECR scores; 2) descending order of EECR scores; 3) ascending order of EECR scores;

Results Fig. 1 shows the results and we can find that *it is not that the more training data we have, the better performance we will obtain*. When we introduce multiple training sets for data augmentation, the order of the distance between training sets and validation sets can help us select the most useful training sets.

4.3 Diagnosing Generalization with *Consistency*

Above entity-centric analyses encourage us to find interpretable factors that affect the model's generalization ability. We can also understand the models' generalization behavior from the perspective of the category of each entity. To answer the question **Q3**, we propose to use a proxy measure *consistency* via the angle of gradients to investigate how the relationship between entity categories influence the difficulty of model learning.

The core idea behind the measure *consistency* is to quantify the effect of different test samples on the trained parameters in NER models. Formally, given a training sample x and its ground truth label y, we refer to $f(x, \theta)$ as the parameterized neural network. Generally, the loss function $\mathcal{L}(f(x, \theta), y)$ shows the difference between model's output and ground truth label. And the gradients of the loss with respect to θ can be formulated as: $\mathbf{g} = \nabla_\theta \mathcal{L}(f(x, \theta), y)$ Here, we propose to characterize the generalization ability of neural networks by observing the gradients' behaviors of test samples. Specifically, given any two samples, the measure consistency Cs can be defined as the cosine angle of their gradients: $\mathrm{Cs}(_{1,2}) = \frac{1 \cdot 2}{|1||2|}$, where $|\mathbf{v}|$ denotes the $L2$-norm of vector $\mathbf{v}$. $_1$ and $_2$ represent two gradient vectors derived from two samples. The idea of utilizing the angle of gradients induced by two test examples has been originally explored on image classification (Fort, Nowak, and Narayanan 2019). Here, we extend this idea to NLP tasks.

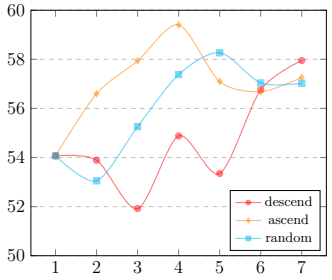


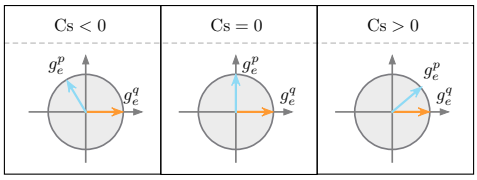
Figure 1: Changes of F1-score as more source domains are introduced in three different orders: descending order (red), ascending order (orange) and random order (blue) of EECR scores.

Algorithm 1 Consistency calculation and evaluation for Named Entity Recognition

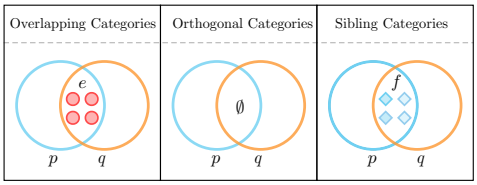
Require: Training dataset $\mathcal{D}^{tr}$ and multiple subsets of validation data $\mathcal{D}_1^{val}, \cdots, \mathcal{D}_K^{val} \subset \mathcal{D}^{val}$. ▷ K is the number of categories
Require: Parameters of the model $\theta \leftarrow \theta^0$
1: Train the model using $\mathcal{D}^{tr}$: $\theta \leftarrow \hat{\theta}$
2: **for** $p \in \{1 \cdots K\}$ **do**
3: **for** $q \in \{1 \cdots K\}$ **do**
4: Compute class-level consistency: $\delta(p, q) = \frac{1}{|\mathcal{D}_p^{val}||\mathcal{D}_q^{val}|} \sum_i^{|\mathcal{D}_p^{val}|} \sum_j^{|\mathcal{D}_q^{val}|} \mathrm{Cs}(_{e_i, e_j}^{p, q})$ ▷ Eq.4
5: **end for**
6: **end for**
7: Obtain a consistency matrix $M \in \mathcal{R}^{K \times K}$ =0

Consistency Evaluation for NER Formally, given an entity e^k and its label y^k, we refer to $\mathbf{g}_e^k = \nabla_\theta \mathcal{L}(f(x, \theta), y^k)$ as its generated gradient vector, where x is the input sample containing entity e^k. Then, for any two samples that contain two entities (e_i and e_j) with different categories (p and q), we introduce the measure $\delta(p, q)$ to quantify the difference between two directions along which the parameters are updated.

$$\delta(p, q) = \frac{1}{C_p C_q} \sum_i^{C_p} \sum_j^{C_q} \mathrm{Cs}(_{e_i, e_j}^{p, q}) \tag{4}$$



(a) The concept of *consistency* Cs



(b) Entity categories relationship

Figure 2: Illustration of the concept of *consistency* Cs and entity categories relationship. g_e^p and g_e^q represents the gradient of entity belongs to category p and q, respectively. e and f are the collection of entities and features overlapped by categories p and q, respectively.

where p and q denote different entity categories. C_p represents the number of test entities with category p. Alg. 1 illustrates the process for consistency calculation and evaluation.

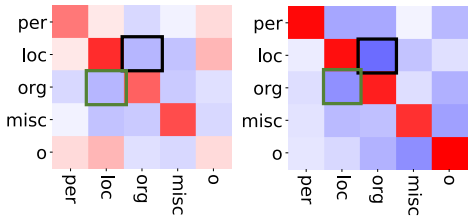
Exp-V: Probing Inter-category Relationships via *Consistency*

Given an NER model, we can understand its generalization ability by calculating the consistency matrix based on Alg. 1. As shown in Fig. 3, the sub-figure (a) illustrates the *consistency* matrice of two NER models trained on `CoNLL` As expected, the on-diagonal elements of $M_{p,q}$ $(p = q)$ usually stay high, since it is easier for the model to find shared features between different entities within the same category. Algorithmically speaking, a gradient step taken with respect to one test entity can reduce the loss on another test entity.

Additionally, a larger value of off-diagonal elements indicates that the two categories share more common properties. As a result, a correct judgment of one category is useful for another. For example, `Percent` category and `Ordinal` category shared a common property of "digit". We name this relationship between them as *Sibling Categories*, shown in Fig. 2(b).

However, if the off-diagonal elements are negative, it suggests that a gradient step taken with respect to one test entity would increase loss on another test entity with different categories, which we define as *Overlapping Categories*, shown in Fig. 2(b). This phenomenon usually occurs when two categories have some overlapped entities. For instance, "`New York University`" is usually a `Location` name, but when "`New York University`" represents as the New York University football team, "`New York University`" is an `Organization` name.

Particularly, if the off-diagonal elements are close to zero, it means the features of two categories tend to be orthogonal: they share few entities or common properties. We name the relationship of these categories as *Orthogonal Categories*, shown in Fig. 2(b).



(a) Consistency: CoNLL (b) Error: CoNLL

P-row	P-col
0.85	0.91
0.94	0.90
0.98	0.99
0.98	0.95
0.45	0.40
Overall: 0.83	

(c) Peason coefficient

Figure 3: The alignment between *consistency* Cs and the error ratio. Sub-figure (a) is the category-membership dependence of *consistency* Cs. Sub-figure (b) is the cross-category error ratios. Sub-figure (c) denotes the Pearson coefficients between consistency values and error values. The change in color from blue to red represents the change in value from negative to positive.

	CoNLL				WNUT				ON-BN			
Models	**F1**	**org-loc**	**loc-org**	**misc-o**	**F1**	**fac.-loc**	**spo.-loc**	**mus.-per**	**F1**	**ord.-date**	**per-org**	**gpe-loc**
CcnnWglovecnnMlp	87.6	43.2	57.3	37.5	31.0	23.3	30.4	41.1	83.5	44.5	4.80	40.5
CcnnWglovelstmMlp	88.4	43.3	58.9	32.9	37.8	31.1	34.6	43.8	84.1	37.5	9.09	29.7
CcnnWglovecnnCrf	89.1	50.6	64.2	29.2	34.2	34.0	34.4	37.8	85.2	27.3	30.0	51.2
CnoneWglovelstmCrf	89.1	33.6	37.5	59.1	37.3	32.2	26.4	37.3	86.2	44.4	11.1	40.5
CcnnWglovelstmCrf	90.3	39.9	55.1	33.7	39.2	53.7	26.9	40.8	88.6	25.0	47.2	26.5
CelmWnonelstmCrf	91.8	41.0	56.4	40.5	43.7	60.0	51.6	43.5	89.3	50.0	50.0	40.7
CelmWglovelstmCrf	92.3	48.8	54.4	35.1	44.0	51.0	43.4	46.8	90.0	28.6	33.3	33.3

Table 7: Error ratios for hard cases (typical error types) with various NER systems. The detail model architecture is shown in Table 2.

Exp-VI: Exploring the Errors of Hard Cases

As shown in Fig. 3, the two sub-figure (a-b) illustrate the consistency and error matrices of the NER model trained on `CoNLL`. In the error matrix, the off-diagonal elements of $Er_{p,q}$ $(p \neq q)$ is computed as the number of entity belonging to category p predicted as category q, divided by the total number of prediction errors of the category p. The on-diagonal elements of $Er_{p,q}$ $(p = q)$ is the accuracy of the category p. Notably, we find that the *consistency* values correlate closely with error ratios based on the Pearson coefficient in Fig. 3-(c). Taking the marked positions in sub-figures (a-b) for example, We find that **if two categories have low *consistency*, the model tends to have difficulty distinguishing them, and it is easy to mis-predict each other**. This observation demonstrates that relationships between entity categories influence model's generalization ability. We

additionally find the prospects for further gains from architecture design and knowledge pre-training seem quite limited based Tab.7. To address these issues, more contextual knowledge or prior linguistic knowledge is needed.

5 Acknowledgments

Thanks Jie Fu for useful comments, and thank the anonymous reviewers for their helpful comments. This work was partially funded by China National Key RD Program (No. 2018YFB1005104, 2018YFC0831105), National Natural Science Foundation of China (No. 61976056, 61532011, 61751201), Science and Technology Commission of Shanghai Municipality Grant (No.18DZ1201000, 16JC1420401, 17JC1420200), Shanghai Municipal Science and Technology Major Project (No.2018SHZDZX01), and ZJ Lab.

References

[Akbik, Bergmann, and Vollgraf] Akbik, A.; Bergmann, T.; and Vollgraf, R. Pooled contextualized embeddings for named entity recognition.

[Akbik, Blythe, and Vollgraf 2018] Akbik, A.; Blythe, D.; and Vollgraf, R. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th COLING*, 1638–1649.

[Arpit et al. 2017] Arpit, D.; Jastrzebski, S.; Ballas, N.; Krueger, D.; Bengio, E.; Kanwal, M. S.; Maharaj, T.; Fischer, A.; Courville, A.; Bengio, Y.; et al. 2017. A closer look at memorization in deep networks. In *Proceedings of the 34th ICML-Volume 70*, 233–242. JMLR. org.

[Bahdanau, Cho, and Bengio 2014] Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. *ArXiv e-prints*.

[Baluja and Fischer 2017] Baluja, S., and Fischer, I. 2017. Adversarial transformation networks: Learning to generate adversarial examples. *arXiv preprint arXiv:1703.09387*.

[Borthwick et al. 1998] Borthwick, A.; Sterling, J.; Agichtein, E.; and Grishman, R. 1998. Exploiting diverse knowledge sources via maximum entropy in named entity recognition. In *Sixth Workshop on Very Large Corpora*.

[Cao et al. 2018] Cao, P.; Chen, Y.; Liu, K.; Zhao, J.; and Liu, S. 2018. Adversarial transfer learning for chinese named entity recognition with self-attention mechanism. In *Proceedings of the 2018 Conference on EMNLP*, 182–192.

[Chen and Moschitti 2019] Chen, L., and Moschitti, A. 2019. Transfer learning for sequence labeling using source model and target data.

[Chiu and Nichols 2016] Chiu, J. P., and Nichols, E. 2016. Named entity recognition with bidirectional lstm-cnns. *TACL* 4:357–370.

[Collobert et al. 2011] Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; and Kuksa, P. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research* 12(Aug):2493–2537.

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

[Fort, Nowak, and Narayanan 2019] Fort, S.; Nowak, P. K.; and Narayanan, S. 2019. Stiffness: A new perspective on generalization in neural networks. *arXiv preprint arXiv:1901.09491*.

[Grishman and Sundheim 1996] Grishman, R., and Sundheim, B. 1996. Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, volume 1.

[Huang, Xu, and Yu 2015] Huang, Z.; Xu, W.; and Yu, K. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

[Jia and Liang 2017] Jia, R., and Liang, P. 2017. Adversarial examples for evaluating reading comprehension systems. *arXiv preprint arXiv:1707.07328*.

[Lin and Lu 2018] Lin, B. Y., and Lu, W. 2018. Neural adaptation layers for cross-domain named entity recognition. *arXiv preprint arXiv:1810.06368*.

[Ma and Hovy 2016] Ma, X., and Hovy, E. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of ACL*, volume 1, 1064–1074.

[Ma, Ganchev, and Weiss 2018] Ma, J.; Ganchev, K.; and Weiss, D. 2018. State-of-the-art chinese word segmentation with bi-lstms. In *Proceedings of the 2018 Conference on EMNLP*, 4902–4908.

[Manning 2011] Manning, C. D. 2011. Part-of-speech tagging from 97% to 100%: is it time for some linguistics? In *International conference on intelligent text processing and computational linguistics*, 171–189. Springer.

[Mikolov et al. 2013] Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[Peters et al. 2018] Peters, M.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of NAACL*, volume 1, 2227–2237.

[Rajpurkar, Jia, and Liang 2018] Rajpurkar, P.; Jia, R.; and Liang, P. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

[Reimers and Gurevych 2017] Reimers, N., and Gurevych, I. 2017. Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. *arXiv preprint arXiv:1707.06799*.

[Sang and De Meulder 2003] Sang, E. F., and De Meulder, F. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

[Schmidt et al. 2018] Schmidt, L.; Santurkar, S.; Tsipras, D.; Talwar, K.; and Madry, A. 2018. Adversarially robust generalization requires more data. In *Advances in NIPS*, 5014–5026.

[Weischedel et al. 2013] Weischedel, R.; Palmer, M.; Marcus, M.; Hovy, E.; Pradhan, S.; Ramshaw, L.; Xue, N.; Taylor, A.; Kaufman, J.; Franchini, M.; et al. 2013. Ontonotes release 5.0 ldc2013t19. *LDC, Philadelphia, PA*.

[Yang, Liang, and Zhang 2018] Yang, J.; Liang, S.; and Zhang, Y. 2018. Design challenges and misconceptions in neural sequence labeling. *arXiv preprint arXiv:1806.04470*.

[Zhang et al. 2016] Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; and Vinyals, O. 2016. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*.

[Zhong et al. 2019] Zhong, M.; Wang, D.; Liu, P.; Qiu, X.; and Huang, X. 2019. A closer look at data bias in neural extractive summarization models. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*. Hong Kong, China: Association for Computational Linguistics.