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

Jinlan Fu,†‡∗ Pengfei Liu,†‡∗ Qi Zhang†‡∗
†Shanghai Key Laboratory of Intelligent Information Processing, Fudan University
‡Research Institute of Intelligent and Complex Systems, Fudan University
∗School of Computer Science, Fudan University
825 Zhangheng Road, Shanghai, China
{fujl16, pfliu14, qz}@fudan.edu.cn

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: (ReCoNL, PLONER) for the future research at our project page: http://pfliu.com/InterpretNER/.

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. \( C_1 \) and \( C_2 \), have overlaps (i.e. sharing a subset of entities), then most of the errors on \( C_1 \) made by the model are due to mistakenly predicting \( C_1 \) as \( C_2 \) (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 principled 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.

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: $P(Y|X) = P(y_t|X, y_1, \ldots, y_{t-1})$

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. Fig.1 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.
We conduct experiments on three benchmark datasets: the CoNLL2003 NER dataset, the WNUT16 dataset, and OntoNotes 5.0 dataset. The CoNLL2003 NER dataset, the WNUT16 dataset, and 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.

### 3.3 NER Datasets for Evaluation

We conduct experiments on three benchmark datasets: the CoNLL2003 NER dataset, the WNUT16 dataset, and OntoNotes 5.0 dataset. The CoNLL2003 NER dataset (Sang and De Meulder 2003) is based on the 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.

### 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 entity coverage ratio is defined as:

$$
\rho(e_i) = \frac{\sum_{k=1}^{K} \frac{\#(e_{i}^{tr,k})}{\#(e_{i})}}{C_{te}}
$$

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 $\rho$ (“chelsea”) = $0.6 \times 3 + 0.4 \times 2)/5 = 0.52$. 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. 2.

#### 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 $\rho$ and then observe how the generalization ability of the NER models varies with it.

#### Results

Based on Tab. 3 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 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
Datasets | Embed-layer | Entity Coverage Rate
|---|---|---|---|---|---|
| | Char | Word | Overall | $1$ | $(0, 0.5)$ | $C \neq 0$ | $C = 0$

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 |

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 |

Table 3: 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 $\rho$ definition in Tab.2.

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 $\rho$ measures the degree to which entities in the test set have been seen in the training set with the same label, the value of $\rho$ 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. 4. 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
Table 4: The test performance (F1 score) on CoNLL 2003 and its revised version.

<table>
<thead>
<tr>
<th>Model</th>
<th>CoNLL</th>
<th>ReCoNLL</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Devlin et al. 2018)</td>
<td>92.80</td>
<td>-</td>
</tr>
<tr>
<td>(Peters et al. 2018)</td>
<td>92.22</td>
<td>-</td>
</tr>
<tr>
<td>(Akbik, Blythe, and Vollgraf 2018)</td>
<td>93.09</td>
<td>-</td>
</tr>
<tr>
<td>(Akbik, Bergmann, and Vollgraf)</td>
<td><strong>93.18</strong></td>
<td>-</td>
</tr>
</tbody>
</table>

Our Implementation

GloVe 89.10 89.85
ELMo 91.79 92.65
BERT 91.34 92.16
FLAIR 93.03 93.78

Table 4: The test performance (F1 score) on CoNLL 2003 and its revised version.

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

$$E_{\rho}(e) = \sum_{i \in N_e} \rho(e_i) \times \text{freq}(e_i)$$ (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 $\eta$ to measure the contextual similarity of entities belonging to the same category but from the training and the test sets, respectively.

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

where $k$ denotes the category of an entity. $D_{tr}$ and $D_{te}$ represents the training and test sets. $\phi_{tr}^k$ 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 representative $v_{f_i}$ 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 “PLONER” 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. To be fair, we limited the number of samples in each dataset to the same 2,500.

**Results** Fig. 2 shows the cross-dataset expectation of coverage ratio ($M_\rho$), contextual coverage ratio ($M_\phi$), 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 $M_\rho$ and $M_\phi$. Additionally, from the values of Pearson

```
Table 4: The test performance (F1 score) on CoNLL 2003 and its revised version.

<table>
<thead>
<tr>
<th>Model</th>
<th>CoNLL</th>
<th>ReCoNLL</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Devlin et al. 2018)</td>
<td>92.80</td>
<td>-</td>
</tr>
<tr>
<td>(Peters et al. 2018)</td>
<td>92.22</td>
<td>-</td>
</tr>
<tr>
<td>(Akbik, Blythe, and Vollgraf 2018)</td>
<td>93.09</td>
<td>-</td>
</tr>
<tr>
<td>(Akbik, Bergmann, and Vollgraf)</td>
<td><strong>93.18</strong></td>
<td>-</td>
</tr>
</tbody>
</table>

Our Implementation

GloVe 89.10 89.85
ELMo 91.79 92.65
BERT 91.34 92.16
FLAIR 93.03 93.78

Table 4: The test performance (F1 score) on CoNLL 2003 and its revised version.

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

$$E_{\rho}(e) = \sum_{i \in N_e} \rho(e_i) \times \text{freq}(e_i)$$ (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 $\eta$ to measure the contextual similarity of entities belonging to the same category but from the training and the test sets, respectively.

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

where $k$ denotes the category of an entity. $D_{tr}$ and $D_{te}$ represents the training and test sets. $\phi_{tr}^k$ 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 representative $v_{f_i}$ 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 “PLONER” 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. To be fair, we limited the number of samples in each dataset to the same 2,500.

**Results** Fig. 2 shows the cross-dataset expectation of coverage ratio ($M_\rho$), contextual coverage ratio ($M_\phi$), 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 $M_\rho$ and $M_\phi$. Additionally, from the values of Pearson

```
```
which suggests we can select the most useful training sets. The value of the ratio could also be validated through the measures usually achieve the highest values on the same training set, which suggests we can select the most useful training sets through the measures and the distribution to be tested is given and we have some samples from it as the validation set.

3) Column-wisely, given a test dataset, , and can usually achieve the highest values on the same training set, which suggests we can select the most useful training sets through the measures and 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 value could also obtain a lower F1 score, since entity coverage ratio is not the only factor that affects generalization and the contextual coverage ratio also matters. A significant case can be found in Fig. 2 (numbers in boxes), taking the as a test set, we observe that WNUT and CoNLL have higher EECR value than while obtaining lower F1 score. We can speculate the reason from the that the contextual coverage ratio between and 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 EECR 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 EECR 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. 3 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 and its ground truth label, we refer to as the parameterized neural network. Generally, the loss function shows the difference between model’s output and ground truth label. And the gradients of the loss with respect to can be formulated as: 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 can be defined as the cosine angle of their gradients:

\[
\cos(\mathbf{g}_1, \mathbf{g}_2) = \frac{\mathbf{g}_1 \cdot \mathbf{g}_2}{\|\mathbf{g}_1\| \|\mathbf{g}_2\|}, \quad |\mathbf{v}| \text{ denotes the } L2\text{-norm of vector } \mathbf{v}.
\]

\[
\mathbf{g}_1, \mathbf{g}_2 \text{ 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.}

**Consistency Evaluation for NER** Formally, given an entity and its label , we refer to as its generated gradient vector, where is the input sample containing entity . Then, for any two samples that contain two entities ( and ) with different categories ( and ), we introduce the measure to quantify the difference between two directions along which the parameters are updated.

---

**Algorithm 1 Consistency calculation and evaluation for Named Entity Recognition**

**Require:** Training dataset and multiple subsets of validation data .

**Require:** Parameters of the model with \( \theta \) and \( \theta^0 \)

1. Train the model using \( \mathcal{D}^{tr} \): \( \theta \leftarrow \theta^0 \)
2. for \( p \in \{1 \cdots K\} \) do
3. for \( q \in \{1 \cdots K\} \) do
4. Compute class-level consistency:
5. end for
6. end for

return A consistency matrix \( M \in \mathbb{R}^{K \times K} \)
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. 5, the sub-figure (a) illustrates the consistency matrix 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. 4b.

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. 4b. 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. 4b.

Exp-VI: Exploring the Errors of Hard Cases

As shown in Fig. 5, 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 \( E_{r_{p,q}} \) \( (p \neq q) \) 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 \( E_{r_{p,q}} \) \( (p=q) \) 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. 5-(c). Taking the marked positions in subfigures (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 Fig.6. To address these issues, more contextual knowledge or prior linguistic knowledge is needed.

5 Acknowledgments

The authors wish to thank the anonymous reviewers for their helpful comments. This work was partially funded by China National Key R&D Program (No. 2018YFB1005104, 2018YFC0831105), National Natural Science Foundation of China (No. 61976056, 61532011, 61751201), Science and Technology Commission of Shanghai Municipality
Figure 6: Error ratios for hard cases (typical error types) with various NER systems. The detailed model architecture is shown in Fig. 1.

Grant (No.18DZ1201000, 16JC1420401, 17JC1420200), Shanghai Municipal Science and Technology Major Project (No.2018SHZDZX01), and ZJ Lab.

References


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.


