A Rigorous Study on Named Entity Recognition: Can Fine-tuning Pretrained Model Lead to the Promised Land?

Hongyu Lin[1], **Yaojie Lu**[1,3], **Jialong Tang**[1,3], **Xianpei Han**[1,2,*], **Le Sun**[1,2,*]
Zhicheng Wei[4], **Nicholas Jing Yuan**[4]

[1]Chinese Information Processing Laboratory [2]State Key Laboratory of Computer Science
Institute of Software, Chinese Academy of Sciences, Beijing, China
[3]University of Chinese Academy of Sciences, Beijing, China
[4]Huawei Cloud&AI
{hongyu,yaojie2017,jialong2019,xianpei,sunle}@iscas.ac.cn
{weizhicheng1,nicholas.yuan}@huawei.com

Abstract

Fine-tuning pretrained model has achieved promising performance on standard NER benchmarks. Generally, these benchmarks are blessed with strong name regularity, high mention coverage and sufficient context diversity. Unfortunately, when scaling NER to open situations, these advantages may no longer exist. And therefore it raises a critical question of whether previous creditable approaches can still work well when facing these challenges. As there is no currently available dataset to investigate this problem, this paper proposes to conduct randomization test on standard benchmarks. Specifically, we erase name regularity, mention coverage and context diversity respectively from the benchmarks, in order to explore their impact on the generalization ability of models. To further verify our conclusions, we also construct a new open NER dataset that focuses on entity types with weaker name regularity and lower mention coverage to verify our conclusion. From both randomization test and empirical experiments, we draw the conclusions that 1) name regularity is critical for the models to generalize to unseen mentions; 2) high mention coverage may undermine the model generalization ability and 3) context patterns may not require enormous data to capture when using pretrained encoders.

1 Introduction

Named entity recognition (NER), or more generally name tagging, aims to identify text spans pertaining to specific entity types. NER is a fundamental task of information extraction which enables many downstream NLP applications, such as relation extraction (GuoDong et al., 2005; Mintz et al., 2009), event extraction (Ji and Grishman, 2008; Li et al., 2013) and machine reading comprehension (Rajpurkar et al., 2016; Wang et al.,

*: Corresponding authors.

	Regular NER	Open NER
Typical Categories	Person, Location, Organization, etc.	Movie, Song, Book, TV Series, etc.
Name Regularity	Entity types with strong regularity	Entity types with weak or no regularity
Mention Coverage	Training set with high mention coverage	Many new and unseen mentions
Context Pattern	With decent training instances to capture	Fully-annotated training data is rare
Examples	**Location** Train: *starting from [**Cherry Street**] ... at [**8ᵗʰ Avenue**] ...* ⇨ Test: *... at [**Cherry Street**]... ... go to [**9ᵗʰ Avenue**] ...*	**Movie** Train: *I watched [**avatar**]last night ...[**the matrix**] is the best...* ⇨ Test: *Wow...[**Joker**] was great! Love [**inception**] so much.*

Figure 1: Comparison between regular NER benchmarks and open NER tasks in reality.

2016). Recently, neural network-based supervised models dominate the NER task. By supervised fine-tuning upon large-scale language model pretrained architectures (e.g., ELMo (Peters et al., 2018), BERT (Devlin et al., 2018), XLNet (Yang et al., 2019), etc.), we have witnessed superior performances on almost all widely-used NER benchmarks, including CoNLL03, ACE2005 and TAC-KBP datasets (Li et al., 2019b; Akbik et al., 2019; Zhai et al., 2019; Li et al., 2019a).

Despite the success of recent models, there are specific advantages in current NER benchmarks which significantly facilitate supervised neural networks. First, these benchmarks focus on limited entity types, and most mentions of these types have strong *name regularity*. For example, nearly all person names follow the "FirstName LastName" or "LastName FirstName" patterns, while location names mostly end with indicator words such as "street" or "road". Second, the training and test data in these benchmarks are sampled from the same corpus, and therefore the training data usually have high *mention coverage*

7291

on the test data, i.e., a large proportion of mentions in the test set have been observed in the training set. However, it is obvious that this high coverage is inconsistent with the primary goal of NER models, which is expected to identify unseen mentions from new data by capturing the generalization knowledge about names and contexts. For observed mentions, other techniques, such as entity linking (Lin et al., 2012), would be more appropriate and effective. Third, these benchmarks generally provide decent training data, and therefore the ***context diversity*** of all entity types can be sufficiently learned. In this paper, we refer to the NER tasks with strong name regularity, high mention coverage and with sufficient training instances as *regular NER*. And it proves that the state-of-the-art neural networks can easily exploit such name regularity, mention coverage and context diversity knowledge, and therefore achieve state-of-the-art performance in these benchmarks.

Unfortunately, when it comes to a more general scenario, there are significant discrepancies between regular benchmarks and NER in open settings. Table 1 overviews their discrepancies on name regularity, mention coverage and context pattern acquisition. Specifically, mentions of many entity types do not follow regular compositional structures. For example, a movie name can be any n-gram utterance and even is not a regular noun phrase(e.g., "Gone with the Wind"). Furthermore, fully-annotated training data will be rare due to the expensive cost. As a consequence, training set can only cover a minor part of test mentions and diverse context patterns must be learned from minimal instances. It is obvious that these discrepancies will lead to the biased estimation of the open NER performance using regular NER benchmarks.

In this paper, we want to shed some light on the impact of the discrepancies between regular and open NER, and provides some valuable insights into the construction of general NER models in a more effective and efficient way. Specifically, we want to answer the following question:

Can pretrained supervised neural networks still generalize well on NER when either weaker name regularity, lower mention coverage or inadequate context diversity exists?

It is non-trivial to answer this question because currently no well-established benchmark concentrates on these issues. To this end, this paper exploits the efficacy of the above three kinds of information by conducting a series of experiments based on *randomization test* (Edgington and Onghena, 2007; Zhang et al., 2016). Specifically, we design several mention replacing mechanisms, which can erase specific kinds of information on-demand from current NER benchmarks. By applying the same supervised models on both vanilla and information-erased data, we can investigate how much the models will rely on particular erased information to identify entity mentions. Generally, we propose to erase name regularity, mention coverage and context diversity respectively using the following kinds of randomization test, whose examples are shown in Table 1:

- **Name Permutation (NP)** is used to investigate the necessity of name regularity for NER, which replaces the same entity mention with an identical, random n-gram string. In this way, the structural correlation between mentions of the same type is removed. For the example in Table 1, all mention "Putin" is replaced by the same utterance "the united".

- **Mention Permutation (MP)** is used to investigate the impact of mention coverage. Different from NP, MP replaces each mention with a unique n-gram string, and even two mentions with the same utterance will be replaced by different strings. For the example in Table 1, two mentions of "Putin" are replaced by "the united" and "which girl" respectively. In this way, the mention coverage is erased and the model should merely rely on context knowledge for NER prediction.

- **Context Reduction (CR)** and **Mention Reduction (MR)** are used to investigate the influence of less training data. CR decreases the diversity of sentences but preserves all mentions in vanilla data, while MR keeps all sentences but only preserves a small part of the original mentions. We compare these two settings, to figure out how much original training data are needed to learn context patterns and name regularity.

To verify the above findings, we further conduct a verification experiment by constructing a new dataset derived from Wikipedia, which focuses on entity types with weak name regularity. To the best of our knowledge, this is the first work that tries to investigate such critical differences between regular and open NER. From both the randomization

Settings	Name	Mention	Context	Examples
Vanilla Baseline	√	√	√	Train { *[Putin] concluded his two days of talks.* / *[Blair] spoke to [Bush] on April 5.* } Test *[Putin] will face re-election in March 2004.*
Name Permutation (NP)	×	√	√	Train { *[the united] concluded his two days of talks.* / *[Hillsborough] spoke to [analysts] on April 5.* } Test *[the united] will face re-election in March 2004.*
Mention Permutation (MP)	×	×	√	Train { *[the united] concluded his two days of talks.* / *[Hillsborough] spoke to [analysts] on April 5.* } Test *[which girl] will face re-election in March 2004.*
Context Reduction (CR)	√	√	↓	Train { *[Putin] concluded his two days of talks.* / *[Blair] concluded his two days of talks.* / *[Bush] concluded his two days of talks.* } Test *[Putin] will face re-election in March 2004.*
Mention Reduction (MR)	↓	↓	√	Train { *[Blair] concluded his two days of talks.* / *[Blair] spoke to [Blair] on April 5.* } Test *[Putin] will face re-election in March 2004.*

Table 1: Illustration of our four kinds of randomization test. The utterances in square brackets are entity mentions. Name: name regularity knowledge; Mention: high mention coverage; Context: sufficient training instances for context diversity √: the knowledge is preserved in this setting; ×: the knowledge is erased from the data in the setting; ↓: the knowledge decreases.

test and the verification experiment, we reach the following main conclusions:

- ***Decent name regularity is vital to the generalization over unseen entity mentions.*** When name regularity is erased, the performance on unseen mentions will be significantly undermined. This finding indicates that it will be challenging to build models for open entity types with weak name regularity.

- ***High mention coverage weakens the model ability to capture informative context knowledge.*** In other words, high mention coverage will mislead models to memorize popular mentions, rather than to learn generalization knowledge. This also reveals that current performance on regular NER benchmarks is highly biased, i.e., the performance on open NER will be significantly lower than that on regular benchmarks.

- ***Sufficient context diversity may not require enormous training data to capture.*** We show that with simple data augmentation techniques to preserving name regularity, required training data can be significantly reduced. This observation also raises the possibility of designing more effective NER models with less annotated data.

2 Experiment Settings

2.1 Dataset Summary

We use ACE2005 (LDC2006T06) as our primary experiment dataset for randomization test. Other openly-available datasets, such as CoNLL03 and Ontonotes, are not suitable for our randomization test. This is because they only annotate named mentions but ignore nominal and pronominal mentions. However, the context of named and nominal/pronominal mentions is generally identical, and therefore the models will be unable to distinguish between them once name regularity is removed.

For better illustration and reproduction, we will report experiment results using the same dataset splits corresponding to Wang et al. (2018); Wang and Lu (2018); Lin et al. (2019a); Xia et al. (2019). For all experiments, we only consider the outmost mentions similar to the majority of the previous work. Finally, there are 18739/2531/2314 mentions in the train/dev/test set respectively. We found that 58.4% mentions in the test set have appeared in the training data, which confirms our high mention coverage concern. We also have conducted multiple experiments using the 8:1:1 train/dev/test data split. And we found all the above experiments lead to the same conclusions which we will illustrate in the next section.

7293

Data Setting	PER	ORG	GPE	FAC	LOC	WEA	VEH	ALL
Baseline	86.31	76.49	80.89	69.23	40.58	74.70	61.97	81.76
Name Permutation	73.41	44.34	49.71	37.96	28.24	33.33	23.93	62.28
- Drop Compared with Baseline	15%	42%	39%	45%	44%	55%	61%	24%
Mention Permutation	61.78	39.40	33.27	32.16	18.60	9.38	21.92	51.58
- Drop Compared with Baseline	28%	48%	59%	54%	54%	87%	65%	34%

Table 2: Micro-F1 scores of BERT-CRF tagger on original data, name permutation setting and mention permutation setting respectively. We can see that erasing name regularity and mention coverage will significantly undermine the model performance.

2.2 Baseline

We use the BERT-CRF tagger as our baseline. Specifically, a Transformer (Vaswani et al., 2017) is used as the encoder and then two dense layers are used to map the hidden representation into the label space. Finally, a linear-chain CRF is applied. The transformer is initialized using *bert_uncased_L-24_H-1024_A-16*, which achieves the best performance on our auxiliary experiments. All model parameters are fine-tuned later. We used Adam (Kingma and Ba, 2014) as the optimizer and set the learning rate to 10^{-5}. Finally, this model achieves 81.76 micro F1 score on ACE2005.

3 Randomization Test on NER

To investigate the discrepancies between regular and open NER, this paper controllably erases target information from the vanilla data via a variant of randomization test (Edgington and Onghena, 2007; Zhang et al., 2016) in non-parametric statistics. Concretely, to probe the effect of a specific kind of information, we erase it in vanilla data by randomly replacing entity mentions with particular irregular utterances. After that, we learn and compare NER models on both the vanilla benchmark and the information-erased version, in order to evaluate the models' robustness and generalization ability when target information is absent. The results of our randomization test can serve as a frame of reference for open NER, where the erased information is often truly absent.

Specifically, three kinds of information are particularly considered, and four kinds of strategies are used in our randomization test. Table 1 shows all our randomization test with examples. In the following, we will illustrate the empirical findings through the test, with one subsection for one kind of information. For each kind of information, we first present the critical conclusion and then demonstrate how we reach the conclusion.

3.1 Effect of Name regularity

Conclusion 1 *Name regularity is critical for supervised NER model to generalize over unseen entity mentions.*

One critical difference between regular and open NER is whether names of the same entity type share inner compositional structure. In regular NER, entities (e.g., PER, ORG and LOC) have long been observed with strong name regularity. In open NER, however, most entities (e.g., movie, song and book) do not have such strong name regularity, and some of mentions can even be random utterances. Therefore, it is critical to evaluate the impact of name regularity on generalization.

To address this issue, we propose *name permutation*, which replaces each mention utterance with a randomly sampled n-gram string, and the mentions with the same name will all be replaced by the same string. To ensure that no structural correlation between these mentions will be retained, the replacing strings are randomly sampled. For example in Table 1, all mentions "Putin" are replaced by "the united", and all "Bush" are replaced by "analysts'". In this way, the name regularity will be erased, but the mention coverage will still retain because the same mention in the training and test data will still be the same.

Table 2 shows the overall results. We can see that when we erase name regularity from the dataset, the performance significantly undermined. The overall drop on micro-F1 is 24%. Moreover, in the majority of entity types, the performance slips more than 40%. This verifies the importance of name regularity on model generalization ability. To investigate the reasons behind, we split mentions for evaluation by whether the predicted/golden mention is covered by the training data, which we refer to as the in-dictionary

portion (InDict) and the out-of-dictionary portion (OutDict) respectively. The results [1] of these two portions on the vanilla dataset and name permutation setting are shown in Table 3.

Vanilla Baseline						
	Precison			Recall		
	InDict	OutDict	**Diff**	InDict	OutDict	**Diff**
PER	88.03	75.40	14%	92.90	85.20	8%
ORG	73.51	72.77	1%	81.93	76.56	7%
GPE	79.55	78.21	2%	85.37	77.22	10%
FAC	65.91	65.67	0%	86.05	65.67	24%
ALL	83.37	72.97	**12%**	89.08	79.11	**11%**
Name Permutation						
	Precison			Recall		
	InDict	OutDict	**Diff**	InDict	OutDict	**Diff**
PER	88.58	46.91	47%	87.00	62.30	28%
ORG	70.40	37.01	47%	51.76	27.80	46%
GPE	70.20	18.60	74%	64.63	30.38	53%
FAC	63.64	27.40	57%	48.84	29.85	39%
ALL	82.47	38.29	**54%**	76.31	46.72	**39%**

Table 3: Comparasion between baseline and name permutation on in-dictionary and out-of-dictionary portions. We can see that the performance gap between In-Dict and OutDict is significantly enlarged when name regularity was erased.

From Table 3, we can find that erasing name regularity leads to more severe performance drop on mentions not covered by training set (Out-Dict) than those appearing in the training set (InDict). For the vanilla dataset, the performance gap between InDict and OutDict is not very large, which shows the good generalizing ability of pretrained supervised model over unseen mentions with strong name regularity. However, after erasing name regularity, this gap is significantly enlarged. The performance on the InDict portion does not drop too much, but the performance on the OutDict portion drops dramatically. This result shows that it is quite difficult to recognize unseen entity mentions when name regularity is missing. Besides, we can see that after erasing name regularity, the model can still perform quite well on the in-dictionary portion, whose precision is still quite high. This demonstrates the strong ability of neural networks to memorize and disambiguate observed mentions even their names are irregular.

Note that Name permutation may result in ambiguous entity instances with only sentence contexts. However, this phenomenon is very common in real world, open-ended NER tasks. For example, a twitter is often a simple sentence

like "La La Land is great, we like it", where "La La Land" is ambiguous in this context and can only be recognized as a Movie using world knowledge about movie. Our experiments also confirm that the less context provides, the more name knowledge is needed.

To summarize, name regularity is very critical for model generalization over unseen mentions. Without name regularity, current models can only work well on mentions covered by training data via memorizing and disambiguating names, but cannot generalize well to unseen mentions.

3.2 Effect of Mention Coverage

Conclusion 2 *High mention coverage weakens the models' ability of capturing informative generalization knowledge for NER.*

Another critical difference between regular and open NER is whether the training data can cover a majority of mentions in the test scenario. High mention coverage can provide misleading evidence during model learning because neural networks can achieve considerable performance by just memorizing and disambiguating observed entity names. This ability, obviously, is not what we desire because 1) in real world applications, most entity mentions are new and unseen, which means out-of-dictionary mentions will dominate the test process; 2) because the training instances are very limited in open situations, it is too expensive to achieve high mention coverage; 3) many long-tail mentions in the training set would be one-shot, i.e., the mention only appears once in the training data. Therefore, it is necessary to exploit whether NER models can still reach reasonable performance in low mention coverage situation.

To this end, we conducted experiments via *mention permutation*, which replaces each mention with a random n-gram similar to the name permutation. However, to erase mention coverage information, the replacing string for each mention is independently sampled, and therefore even mentions with the same utterance in vanilla data will be replaced by different strings. For example, two "Putin" in Table 1 are replaced by different utterances. In this way, (almost) no mention in the test set is covered by the training data, and no name information remains in the data. Consequently, the models should only rely on context knowledge for NER prediction.

The mention permutation results are shown in

---

[1]We only present the performance on four entity types with sufficient training and testing instances.

	Precision		Recall	
	NP	MP	NP	MP
PER	46.91	**58.01**	62.30	**66.21**
ORG	37.01	**40.76**	27.80	**38.94**
GPE	18.60	**32.92**	30.38	**34.05**
FAC	27.40	**27.74**	29.85	**36.54**
ALL	38.29	**49.40**	46.72	**54.01**

Table 4: Experiment results on *OutDict* portion. We can see that mention permutation significantly performs better than name permutation, which indicates that high mention coverage may undermine the generalization ability of models.

Table 2. We can see that the performance of MP further drops compared with NP, which demonstrates high mention coverage can make mention detection much simpler. To further investigate whether high mention coverage will influence the models' generalization ability, we also compared MP with NP on the *out-of-dictionary portion*. The results are shown in Table 4.

Surprisingly, the model performs significantly better in the MP setting than in the NP setting in all entity types. In other words, high mention coverage undermines the models' ability to generalize to unseen mentions. We believe this is because, as some previous studies in other tasks (Zhang et al., 2016; Lu et al., 2019) have pointed out, neural networks have strong ability and tendency to memorize training instances. Consequently, the high mention coverage will mislead the models to mainly memorize and disambiguate frequent entity names even though they are irregular, but ignore informative context patterns which are useful for generalization over unseen mentions. These results reveal that NER models should focus more on context knowledge for generalization, rather than memorizing popular mentions. This is even more important for entity types without or with weak name regularity because context patterns are more critical in this circumstance.

3.3 Effect of Context Diversity

Conclusion 3 *Sufficient context patterns may not require enormous training data to capture when learning upon pretrained neural networks.*

Current NER benchmarks commonly provide decent training data for learning context patterns of entities. However, due to the expensive cost, it is usually impractical for open NER to assume enough fully-annotated training data. If we can figure out how many training instances are sufficient for context pattern and name regularity respectively, it will provide valuable insights for constructing open NER datasets and models more effectively and efficiently.

To this end, we conduct *context reduction (CR)* and *mention reduction (MR)* on the vanilla training set using simple data augmentation strategies. The purpose of context reduction is to reduce context diversity in training data but still keeps all name regularity in the vanilla setting. Specifically, CR only keeps a subset of sentences in the vanilla training data, and then duplicates preserved sentences and randomly replace mentions in them with mentions of the same type in the vanilla training data. In this way, all mentions will share identical frequency in the vanilla and CR dataset. On the contrast, MR aims to reduce mention diversity for name regularity, but retains context diversity by keeping all of the original contexts. For this, MR only keeps a part of mentions in the original training set as seeds, and replace other mentions in the training data with a mention randomly sampled from the seeds of the same type. In this way, only part of name knowledge will retain, but all contexts will be preserved. Furthermore, we also compare CR and MR with a naive reduction strategy which simply subsamples sentences in the training data, and we refer it as *sentence reduction*.

We varied the ratio of preserved information in each setting ranging from 5% to 100% respectively. The overall results are shown in Figure 2. We can see that in the sentence reduction setting, the performance steadily improved as the training data grows. This phenomenon is also observed in the mention permutation setting, which indicates that increasing training data will introduce more name regularity knowledge, and thus results in better performance. However, for context permutation setting, there is no significant performance improvement on PER, ORG and GPE when the preserved sentences are more than 30% of the vanilla data. But for FAC, increasing the preserving data will still improve the performance. This may because the number of FAC instances are significantly smaller than PER, ORG and GPE in the vanilla dataset. From the above experiments, it seems that once it reaches a certain amount, the instances in training data are enough to capture sufficient context patterns. And increasing training instances can mainly provide more name

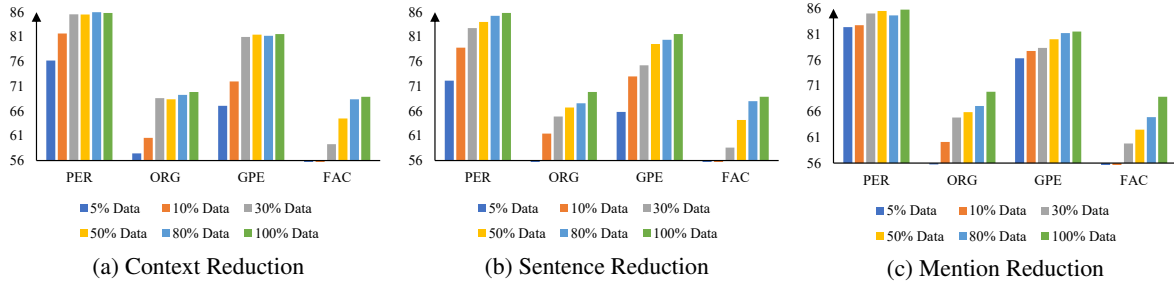(a) Context Reduction (b) Sentence Reduction (c) Mention Reduction

Figure 2: Experiments on context reduction, mention reduction and sentence reduction when the kept information ratio varies. We can see that when preserving all name regularity information, sufficient context patterns can be captured once the training sentences reaches a certain amount, and introducing more training data does not significantly improve the performance.

regularity knowledge rather than more context diversity.

The above results provide a valuable insight that the name regularity and the context patterns for NER can be learned separately, rather than jointly. For example, we can learn context patterns using a moderate number of training instances and attempt to incorporate more name regularity knowledge using other resources, e.g., gazetteers.

4 Experiments on Open NER

4.1 Data Preparation

To further verify the conclusions from our randomization test, we propose to conduct experiments on a real-world open NER dataset, which focuses on real world entity types with weaker name regularity than previously-used benchmarks. Because currently no suitable dataset is available for verifying our conclusions, this paper constructs a new dataset from Wikipedia. Specifically, we consider four entity types in our experiments, including *movie*, *song*, *book* and *tv series*. We extract all sentences in Wikipedia which contain mentions linking to entities of these types as our experiment dataset. From them, we randomly sample 10,000 sentences as the test set and 2,000 sentences as the development set, and part of the remaining data will be used as the training data according to the following different settings. Finally, there are 2875, 2791, 598 and 580 mentions for movie, tv series, song and book in the test set respectively. Note that different from real scenarios, this dataset only keeps sentences containing at least one mention, Due to the partial labeling nature of Wikipedia, this dataset only keeps sentences containing at least one mention, and the performance on this dataset may over-estimate the

precision than in real applications. Although this may different from real scenarios, we believe it can still lead to reasonable conclusions.

4.2 Generalizing over Unseen Mentions

The first group of experiments was conducted to verify the influence of name regularity on in-dictionary and out-of-dictionary mentions. To this end, we randomly sampled 5,000 sentences from the dataset as the training set, which is close to the training data size of ACE2005. We use *bert_cased_L-24_H-1024_A-16* rather than the uncased version of the pretrained model because we find that capitalization may have a significant impact on this dataset. Furthermore, different from the ACE2005 whose training data covers nearly 58% test set mentions, the training set of our Wikipedia dataset can only cover 27% mentions in the test set. This confirms our concern that the mention coverage is much lower in open NER than in regular NER.

	Baseline					
	Precison			Recall		
	InDict	OutDict	Diff	InDict	OutDict	Diff
Movie	80.68	71.48	11%	88.43	71.43	19%
Tv Series	91.48	65.77	28%	88.03	74.73	15%
Song	77.94	56.42	28%	68.83	62.50	9%
Book	83.72	53.99	36%	68.57	55.03	20%
ALL	87.44	66.59	**24%**	86.30	71.13	**18%**

Table 5: Comparison between in-dictionary portion and out-of-dictionary portion on Wikipedia dataset. We can see that there is a significant gap between these two portions.

Table 5 reports the experiment results on in-dictionary portion and out-of-dictionary portion respectively. We can see that the performance gap between in-dictionary portion and out-of-dictionary portion significant due to the weak

name regularity. This confirms our Conclusion 1 that name regularity is vital for NER system to generalize over unseen mentions. Compared with the name permutation setting shown in Table 3, the InDict-outDict performance gap is not so large. We believe this is because: 1) there still exist some kinds of name regularity for these entity types, e.g., the capitalization of the first letter; 2) Wikipedia documents are much formal than ACE2005 documents, which makes the context patterns much easier to capture. For example, a movie mention in Wikipedia will frequently share the same context of "in the film Xxx Xxx", where "Xxx" is mention word with the first letter being capitalized. Despite this, the performance gap between InDict and OutDict portions is still significant – more than 24% and 18% on precision and recall respectively, which verifies the necessity of name regularity for NER to achieve good generalization.

4.3 Influence of Training Data Size

To investigate the impact of training data size to the model performance, we varied the size of training set from 500 to 10,000, and investigate the performance improvement over the test set. Because the entity types we considered are with weak name regularity, the increment of training instances will mainly increase the context diversity. Therefore, this group of experiments can be used to verify the Conclusion 3 we proposed before.

Figure 3 shows the results. We can see that the performance improvements on all entity types are less significant after training data size exceeding 3000. This phenomenon is to the Figure 2 (a) results on the context reduction setting on ACE2005. This further verifies our Conclusion 3 and confirms that when sample size reaches a certain level, introducing more training data will not improve the learning of context knowledge.

5 Related Work

Named entity recognition has long been studied and has attracted much attention. Conventional methods (Zhou and Su, 2002; Chieu and Ng, 2002; Bender et al., 2003; Settles, 2004) commonly rely on handcraft features to build NER models, which are hard to transfer among different languages, domains and entity types. Recently, deep learning methods, which automatically extract high-level features and perform sequence tagging with neural
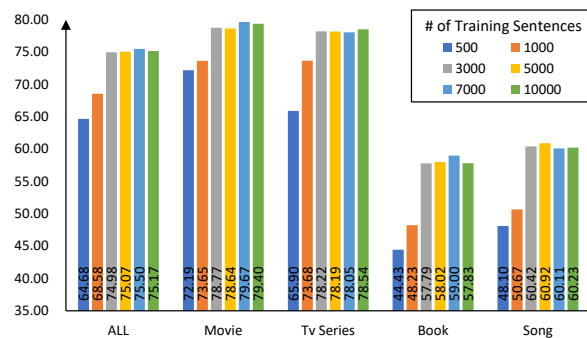

Figure 3: F1 scores on Wikipedia dataset when training data size varies. We can see that there are little improvement when training data size exceeds 3000.

networks (Santos and Guimaraes, 2015; Chiu and Nichols, 2016; Lample et al., 2016; Yadav and Bethard, 2019), have achieved significant progress especially under strong pretraining and fine-tuning paradigm (Li et al., 2019b; Akbik et al., 2019; Zhai et al., 2019; Li et al., 2019a; Xia et al., 2019; Lin et al., 2019a). These methods have achieved promising results in almost all popular NER benchmarks considering regular entity types.

Several researches have shift attention to name tagging in open scenarios, where entity types may have weaker name regularity and training data are often insufficient. These papers mainly focus on how to exploit weakly-supervised data (Täckström et al., 2013; Ni et al., 2017; Cao et al., 2019; Xue et al., 2019), or devoted to incorporate external resources (Yang et al., 2017; Peng and Dredze, 2016; Pan et al., 2017; Lin et al., 2018; Xie et al., 2018; Lin et al., 2019b).

By contrast, to the best of our knowledge, this is the first work which investigates the essential difference between regular and open NER. By conducting both randomization test (Edgington and Onghena, 2007) and verification experiments, we analyze the impact of name regularity, mention coverage and context pattern sufficiency and shed light on future open NER studies.

6 Conclusion and Future work

This paper investigates whether current state-of-the-art models on regular NER can still work well on open NER. From the perspective of name regularity, mention coverage and context diversity, we conducted both randomization test and verification experiments to evaluate the generalization ability of models. Our investigation leads to three valuable conclusions, which shows the necessity of de-

cent name regularity to identify unseen mentions, the hazard of high mention coverage to model generalization, and the redundancy of enormous data to capture context patterns.

The above findings shed light on the promising directions for open NER, including 1) exploiting name regularity more efficiently with easily-obtainable resources such as gazetteers; 2) preventing the overfit on popular in-dictionary mentions with constraints or regularizers; and 3) reducing the need of training data by decoupling the acquisition of context knowledge and name knowledge.

Acknowledgments

References

Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 2019. Pooled contextualized embeddings for named entity recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 724–728.

Oliver Bender, Franz Josef Och, and Hermann Ney. 2003. Maximum entropy models for named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 148–151. Association for Computational Linguistics.

Yixin Cao, Zikun Hu, Tat-Seng Chua, Zhiyuan Liu, and Heng Ji. 2019. Low-resource name tagging learned with weakly labeled data. *arXiv preprint arXiv:1908.09659*.

Hai Leong Chieu and Hwee Tou Ng. 2002. Named entity recognition: a maximum entropy approach using global information. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Eugene Edgington and Patrick Onghena. 2007. *Randomization tests*. Chapman and Hall/CRC.

Zhou GuoDong, Su Jian, Zhang Jie, and Zhang Min. 2005. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 427–434. Association for Computational Linguistics.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. *Proceedings of ACL-08: HLT*, pages 254–262.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

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

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 73–82.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2019a. A unified mrc framework for named entity recognition. *arXiv preprint arXiv:1910.11476*.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019b. Entity-relation extraction as multi-turn question answering. *arXiv preprint arXiv:1905.05529*.

Hongyu Lin, Yaojie Lu, Xianpei Han, and Le Sun. 2019a. Sequence-to-nuggets: Nested entity mention detection via anchor-region networks. *arXiv preprint arXiv:1906.03783*.

Hongyu Lin, Yaojie Lu, Xianpei Han, Le Sun, Bin Dong, and Shanshan Jiang. 2019b. Gazetteer-enhanced attentive neural networks for named entity recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6233–6238.

Thomas Lin, Oren Etzioni, et al. 2012. Entity linking at web scale. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, pages 84–88. Association for Computational Linguistics.

Ying Lin, Shengqi Yang, Veselin Stoyanov, and Heng Ji. 2018. A multi-lingual multi-task architecture for low-resource sequence labeling. In *ACL*.

Yaojie Lu, Hongyu Lin, Xianpei Han, and Le Sun. 2019. Distilling discrimination and generalization knowledge for event detection via delta-representation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4366–4376.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics.

Jian Ni, Georgiana Dinu, and Radu Florian. 2017. Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. In *ACL*.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *ACL*.

Nanyun Peng and Mark Dredze. 2016. Improving named entity recognition for chinese social media with word segmentation representation learning. In *ACL*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Cicero Nogueira dos Santos and Victor Guimaraes. 2015. Boosting named entity recognition with neural character embeddings. *arXiv preprint arXiv:1505.05008*.

Burr Settles. 2004. Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*, pages 104–107. Association for Computational Linguistics.

Oscar Täckström, Dipanjan Das, Slav Petrov, Ryan McDonald, and Joakim Nivre. 2013. Token and type constraints for cross-lingual part-of-speech tagging. *TACL*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Bailin Wang and Wei Lu. 2018. Neural segmental hypergraphs for overlapping mention recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 204–214. Association for Computational Linguistics.

Bailin Wang, Wei Lu, Yu Wang, and Hongxia Jin. 2018. A neural transition-based model for nested mention recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1011–1017. Association for Computational Linguistics.

Zhiguo Wang, Haitao Mi, Wael Hamza, and Radu Florian. 2016. Multi-perspective context matching for machine comprehension. *arXiv preprint arXiv:1612.04211*.

Congying Xia, Chenwei Zhang, Tao Yang, Yaliang Li, Nan Du, Xian Wu, Wei Fan, Fenglong Ma, and Philip Yu. 2019. Multi-grained named entity recognition. *arXiv preprint arXiv:1906.08449*.

Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A Smith, and Jaime Carbonell. 2018. Neural cross-lingual named entity recognition with minimal resources. In *EMNLP*.

Mengge Xue, W. Cai, Jinsong Su, Linfeng Song, Y. Ge, Y. Liu, and B. Wang. 2019. Neural collective entity linking based on recurrent random walk network learning. In *IJCAI*.

Vikas Yadav and Steven Bethard. 2019. A survey on recent advances in named entity recognition from deep learning models. *arXiv preprint arXiv:1910.11470*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Zhilin Yang, Ruslan Salakhutdinov, and William W Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. In *ICLR*.

Zenan Zhai, Dat Quoc Nguyen, Saber A Akhondi, Camilo Thorne, Christian Druckenbrodt, Trevor Cohn, Michelle Gregory, and Karin Verspoor. 2019. Improving chemical named entity recognition in patents with contextualized word embeddings. *arXiv preprint arXiv:1907.02679*.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2016. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*.

GuoDong Zhou and Jian Su. 2002. Named entity recognition using an hmm-based chunk tagger. In *proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 473–480. Association for Computational Linguistics.