Does it Really Generalize Well on Unseen Data? Systematic Evaluation of Relational Triple Extraction Methods

Juhyuk Lee[1*†], **Min-Joong Lee**[2†], **June Yong Yang**[3†], **Eunho Yang**[3]
Samsung Research, Samsung Electronics, South Korea[1],
Samsung Advanced Institute of Technology, Samsung Electronics, South Korea[2],
Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea[3]
{juhyuk.lee, minjoong.lee}@samsung.com
{laoconeth, eunhoy}@kaist.ac.kr

Abstract

The ability to extract entities and their relations from unstructured text is essential for the automated maintenance of large-scale knowledge graphs. To keep a knowledge graph up-to-date, an extractor needs not only the ability to recall the triples it encountered during training, but also the ability to extract the new triples from the context that it has never seen before. In this paper, we show that although existing extraction models are able to easily memorize and recall already seen triples, they cannot generalize effectively for unseen triples. This alarming observation was previously unknown due to the composition of the test sets of the go-to benchmark datasets, which turns out to contain only 2% unseen data, rendering them incapable to measure the generalization performance. To separately measure the generalization performance from the memorization performance, we emphasize unseen data by rearranging datasets, sifting out training instances, or augmenting test sets. In addition to that, we present a simple yet effective augmentation technique to promote generalization of existing extraction models, and experimentally confirm that the proposed method can significantly increase the generalization performance of existing models.

1 Introduction

Relational Triple Extraction (RTE), a more generalized version of Relation Extraction, is the task of extracting all relational triples in the form of *(subject, relation, object)* from a given sentence. The ability to extract such triples is much required in the construction and maintenance of knowledge graphs such as Dbpedia (Auer et al., 2007), Freebase (Bollacker et al., 2008), and Wikidata (Vrandečić and Krötzsch, 2014) from documents containing a large number of new and emerging information.

With language model pretraining (Devlin et al., 2019; Radford et al., 2019), RTE methods achieved a new state-of-the-art (Wei et al., 2020; Wang et al., 2020; Zheng et al., 2021). However, whether the performance of these methods attributes to their capabilities of recalling already seen data or their ability to generalize and extract relations from unseen data is yet to be scrutinized.

To separately evaluate memorization and generalization, we categorize the triples in the test set into three types: *entirely seen* (completely overlaps with triples in their respective training sets), *partially seen* (overlaps partially), and *unseen* (completely new). We analyze common RTE benchmark datasets NYT (Riedel et al., 2010) and WebNLG (Gardent et al., 2017) using these categories, and find that 89.61% and 91.10% of triples in NYT and WebNLG *test sets* are of the *entirely seen* type. This suggests that benchmark results on these datasets are heavily biased towards recalling seen data. Thus, more reliable systematic evaluation methods are in need to test generalization performance.

In this paper, we propose three natural strategies for evaluating generalization performance from a limited number of given *partially seen* and *unseen* triples. For the first two strategies, we directly increase the proportion of *partially seen* and *unseen* triples in test sets by 1) rearranging their respective datasets or 2) sifting out instances in their respective training sets that overlap with the test set, rendering them unobserved. For the last strategy, we 3) augment test sets by replacing entities in each test instance with similar (and probably not pre-observed) words in order to increase diversity as well as the proportion of *partially seen* and *unseen* triples. In addition to evaluating recent RTE methods with the above evaluation strategies, we propose a simple yet effective augmentation technique called *Entity Noising* to help RTE methods to generalize beyond training data.

[*] This work was done when Juhyuk Lee was with KAIST as a student.

[†] Equal contribution.

Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 3849 - 3858

Triple type	NYT						WebNLG					
	Ori.	Rearr.	Sift-1	Sift-2	Sift-3	Aug.	Ori.	Rearr.	Sift-1	Sift-2	Sift-3	Aug.
Entirely seen (%)	**89.61**	14.20	63.24	55.45	49.27	5.76	**91.10**	45.47	78.03	56.50	39.20	17.21
Partially seen (%)	8.64	66.72	31.56	38.09	43.19	46.33	7.47	34.20	17.05	30.86	37.40	36.17
Unseen (%)	1.75	19.08	5.20	6.46	7.54	47.91	1.43	20.33	4.92	12.63	23.40	46.62

Table 1: Triple type statistics of original test sets, *rearranged*, *overlap sifted* datasets, and *augmented test sets*.

	Method	F1	Entire	Partial	Unseen
NYT	CasRel	90.1 (89.0⋆)	93.8	64.6	45.4
	TPLinker	92.4 (92.0†)	96.0	65.9	50.3
	PRGC	89.1 (92.7†)	92.9	65.4	44.5
WebNLG	CasRel	88.3 (86.4⋆)	92.0	54.3	45.5
	TPLinker	89.0 (86.7†)	92.6	62.6	56.0
	PRGC	88.0 (88.5†)	92.1	56.2	34.5

Table 2: F1 and type F1 of recent RTE methods. Results with † marks are from their papers. Results with ⋆ marks are reported by Ren et al. (2021). Other results are our reproductions using official implementations.

Our contributions are:

- We show for the first time that the current benchmark datasets for relational triple extraction exhibit significant entity pair overlap between training and test data.
- We confirm that the current state-of-the-art models trained on such datasets cannot generalize well to unseen triples.
- We propose three evaluation strategies to evaluate RTE methods systematically, and show that the proposed simple augmentation technique called *Entity Noising* can assist RTE methods in generalizing to unseen data.

2 Fine-grained Re-evaluation of the Current State-of-the-arts

In this section, we mainly scrutinize the generalization capabilities of current Relational Triple Extraction (RTE) methods and show for the first time that they indeed struggle in extracting relational triples from the context for unseen cases.

2.1 Datasets and Evaluation Metrics

We use two well-known benchmark datasets NYT (Riedel et al., 2010) and WebNLG (Gardent et al., 2017) for evaluation, following Wang et al. (2020) and Zheng et al. (2021). Also, predicted triples are considered correct only if their whole entity spans of both subject and object and their relation are exactly matched with ground truth. We report the standard micro F1 for the overall performance.

To assess the memorization and generalization performances separately, we also compute type F1 with three triple types: *entirely seen*, *partially seen*, and *unseen* (Section 2.2). Type F1 is nothing but F1 evaluated using instances which only consist of a single triple type.

2.2 Triple Types

We describe three triple types - *entirely seen*, *partially seen*, and *unseen* - in detail. For a set of triples in the training set $S = \{(s_i, r_i, o_i)\}_{i=1}^{n}$, the type of each triple (s, r, o) in the test set are defined as follows. A triple (s, r, o) belongs to the *entirely seen* type if $(s, r, o) \in S$. For *partially seen* type, triples (s, r, o) which satisfy conditions $[(s, r, \cdot) \in S$ or $(\cdot, r, o) \in S]$ and $(s, r, o) \notin S$ belong to it. Other triples belong to *unseen* type.

2.3 Detailed Evaluation with Triple Types

Using type F1, we show that the current state-of-the-arts CasRel (Wei et al., 2020), TPLinker (Wang et al., 2020), and PRGC (Zheng et al., 2021) are only able to memorize and recall already seen triples, and are unable to generalize effectively for unseen triples (See Table 2). This observation was previously unknown due to the overlaps between training and test data of benchmark datasets NYT and WebNLG.

Indeed, as shown in Table 1, 89.61% and 91.10% of triples in NYT and WebNLG *test sets* completely overlap with triples in their respective training sets (such triples are defined as *entirely seen* type), while *partially seen* and *unseen* samples that require generalization to predict are but a small portion.

3 Evaluating Generalization Performance

As shown in Table 1, the proportion of *partially seen* and *unseen* triples in the original benchmark test sets are so small that they are not diverse

	Method	Original						Rearranged					
		Prec.	Rec.	F1	Entire	Partial	Unseen	Prec.	Rec.	F1	Entire	Partial	Unseen
NYT	CasRel	90.2	**90.0**	**90.1**	**93.8**	64.6	**45.4**	**65.9**	**60.1**	**62.9**	**85.8**	**65.0**	42.3
	CasRel+EN	**91.6**	88.8	**90.1**	93.7	**65.0**	44.8	65.2	59.3	62.1	81.1	64.9	**44.0**
	TPLinker	**92.3**	**92.5**	**92.4**	**96.0**	65.9	50.3	69.0	**60.8**	**64.7**	83.3	**66.7**	46.8
	TPLinker+EN	92.2	91.8	92.0	95.5	**66.0**	**54.4**	**69.2**	60.3	64.5	**84.2**	66.3	**47.2**
	PRGC	88.4	**89.9**	**89.1**	**92.9**	**65.4**	44.5	63.5	**61.6**	**62.6**	**81.6**	**64.2**	45.1
	PRGC+EN	**89.1**	88.7	88.9	92.3	**65.4**	**51.2**	**63.9**	60.6	62.2	79.8	**64.2**	**46.2**
WebNLG	CasRel	**90.1**	86.6	**88.3**	**92.0**	**54.3**	45.5	**73.6**	**64.2**	**68.6**	**89.6**	52.3	41.5
	CasRel+EN	88.8	**86.8**	87.8	91.3	48.9	**53.8**	72.5	63.2	67.5	85.7	**54.0**	**45.8**
	TPLinker	**90.2**	**87.7**	**89.0**	**92.6**	**62.6**	56.0	**75.1**	63.9	69.1	88.5	52.7	42.9
	TPLinker+EN	89.3	87.4	88.3	91.8	60.0	**71.4**	73.5	**66.2**	**69.7**	**88.7**	**53.6**	**49.3**
	PRGC	**89.7**	**86.4**	**88.0**	**92.1**	56.2	34.5	61.6	62.0	61.8	79.2	47.2	28.3
	PRGC+EN	87.6	85.4	86.5	90.2	**57.5**	**40.0**	**68.0**	**62.5**	**65.2**	**82.8**	**52.8**	**34.4**

Table 3: Results of recent RTE methods with and without *Entity Noising* on original and *rearranged* datasets. Every result are our reproduction.

enough, rendering the evaluations of generalization capabilities unreliable. Equipped with this observation, we propose three strategies to increase the proportion of *partially seen* and *unseen* triples and add diversity to them for reliable evaluation of Relational Triple Extraction (RTE) methods.[1]

3.1 Rearranged Dataset

The basic approach to increasing the proportion of *partially seen* and *unseen* triples is to rearrange the given dataset splits. However, it is not possible to emphasize unseen data just by randomly rearranging the dataset, since it inadvertently incurs overlaps between training and test data[2].

To emphasize the unseen data, we repeatedly select a triple and distribute every instance which contains that triple to the test set, rendering them unobserved in the training set. In order to minimize redundancy in the test set, we select a triple one by one which occurs less. The detailed statistics are shown in Table 1 and Appendix B.

3.2 Overlap Sifted Dataset

We propose another simple strategy to emphasize unseen test samples. To render a triple in the test set unobserved, we remove the instances containing that triple from the training set. Specifically, we randomly choose $k\%$ of the unique triples from the test set, then remove all the instances containing the selected triples from the *training set* to construct an *overlap sifted dataset*. For demonstration, we construct three such datasets by choosing $k = 5, 10, 15\%$, respectively. The detailed statistics are presented in Table 1 and Appendix B.

3.3 Augmented Test Set

To add more diversity to *partially seen* and *unseen* samples as well as increasing their proportion, we create an *augmented test set*. The key idea is to substitute every entity defined in every triple with probable alternative words by utilizing the knowledge of Masked Language Models (Radford et al., 2019; Devlin et al., 2019) and GloVe word embeddings (Pennington et al., 2014), similar to the data augmentation technique used in Jiao et al. (2020). With the *augmented test set*, it is able to assess whether the ability of an RTE method is influenced by the authenticity of the given text[3]. The details are in Appendix C and statistics are present in Table 1 and Appendix B.

4 Entity noising

We further propose *Entity Noising*, a simple augmentation technique to enhance the generalization performance of existing Relational Triple Extraction methods. The key idea of *Entity Noising* is to replace the entities in the given training input sentence with completely random noisy words. To apply Entity Noising, we sample a random noisy word w' for each entity w, i.e., $w' \sim P(w' \mid w)$. The sampling strategy is defined as follows. First,

[1]Three versions of datasets can be found in https://github.com/sehkmg/rte-eval.

[2]We are only able to emphasize unseen data to at most 2% with 10^6 random trials.

[3]An ideal RTE model should be able to extract the relational triple (The [United States] President [Christopher]) if such fictitious content happens to exist in the given text.

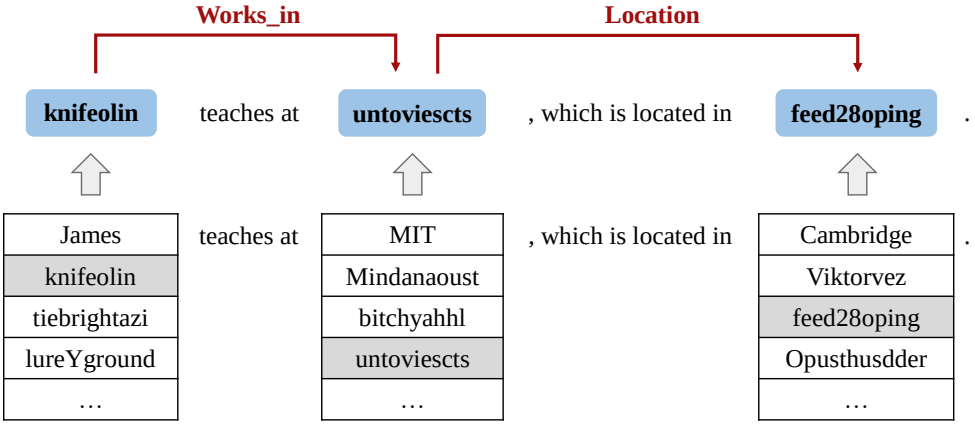

Figure 1: Overview of *Entity Noising*.

	Method	NYT				WebNLG			
		F1	Entire	Partial	Unseen	F1	Entire	Partial	Unseen
CasRel	Original	90.1 (+0.0)	93.7 (-0.1)	65.0 (+0.4)	44.8 (-0.6)	87.8 (-0.5)	91.3 (-0.7)	48.9 (-5.4)	53.8 (+8.3)
	Sift-1	84.8 (+0.0)	96.0 (+0.1)	69.2 (+0.2)	51.4 (+1.8)	85.4 (-1.0)	93.7 (-0.9)	58.9 (-6.0)	56.1 (-1.6)
	Sift-2	83.4 (+0.6)	96.3 (-0.2)	71.1 (+2.9)	48.8 (-1.3)	77.8 (-0.6)	94.8 (-0.4)	59.6 (+0.6)	68.4 (+11.3)
	Sift-3	81.7 (+0.2)	96.8 (+0.3)	70.8 (-0.2)	48.4 (+0.1)	71.3 (-1.2)	95.9 (-0.1)	66.9 (+2.2)	63.8 (+7.5)
TPLinker	Original	92.0 (-0.4)	95.5 (-0.5)	66.0 (+0.1)	54.4 (+4.1)	88.3 (-0.7)	91.8 (-0.8)	60.0 (-2.6)	71.4 (+15.4)
	Sift-1	87.1 (+0.1)	98.0 (-0.6)	71.1 (+1.0)	56.1 (+6.3)	86.5 (-0.5)	93.3 (-0.6)	75.3 (+2.7)	60.7 (+0.0)
	Sift-2	85.4 (-0.2)	98.3 (-0.1)	72.4 (-0.3)	56.5 (+1.5)	79.3 (+0.4)	93.6 (-1.9)	69.8 (+3.1)	67.3 (+0.6)
	Sift-3	83.6 (+0.1)	98.2 (-0.1)	72.7 (+0.1)	53.1 (+0.5)	72.1 (+1.0)	95.4 (-1.3)	68.7 (+3.4)	66.7 (+5.9)
PRGC	Original	88.9 (-0.2)	92.3 (-0.6)	65.4 (+0.0)	51.2 (+6.7)	86.5 (-1.5)	90.2 (-1.9)	57.5 (+1.3)	40.0 (+5.5)
	Sift-1	84.5 (-0.6)	96.3 (-0.3)	67.2 (-1.2)	54.0 (+0.7)	84.5 (+0.2)	92.5 (-0.6)	64.4 (+9.6)	51.5 (+4.6)
	Sift-2	83.2 (+0.0)	96.9 (+0.3)	68.4 (-1.2)	49.2 (-1.9)	75.9 (+1.5)	93.5 (-0.3)	60.1 (+8.8)	57.1 (+9.2)
	Sift-3	81.6 (+0.4)	96.5 (-0.1)	70.3 (+0.0)	51.3 (+0.4)	68.4 (+2.4)	93.5 (+0.2)	62.6 (+1.7)	56.8 (-1.3)

Table 4: Results of recent RTE methods applied with *Entity Noising* on original and *overlap sifted* datasets. Numbers in ( ) show performance gaps between baseline and *Entity Noising*.

we sample token length $l' \in \{l-1, l, l+1\}$ of w' with probability $P(l' = l) = p_{en}^{len}$ and $P(l' = l-1) = P(l' = l+1) = \left(1 - p_{en}^{len}\right)/2$, where l is a token length of w. This sampling process introduces a small(± 1) perturbation to the token length l to prevent the model from memorizing the number of tokens. After sampling l', we sample w' from the uniform distribution $w' \sim \text{Uniform}(V_{l'})$, where $V_{l'}$ is a subset of the vocabulary V which consists of all words of token length l'.

With sampling strategy $w' \sim P(w' \mid w)$, Entity Noising is applied to a given training sentence $\mathbf{x}_{\text{original}} = (w_1, w_2, \cdots, w_K)$ to produce a noised sentence $\mathbf{x}_{\text{noised}} = (w'_1, w'_2, \cdots, w'_K)$ according to the following rule:

$$w'_k = \begin{cases} w'_k \sim P(w'_k \mid w_k), & \text{if } w_k \text{ is an entity} \\ w_k, & \text{otherwise} \end{cases}$$

Finally, we determine which input $\mathbf{x}$ is fed to the extractor model with probability $P(\mathbf{x} = \mathbf{x}_{\text{noised}}) = p_{en}$ and $P(\mathbf{x} = \mathbf{x}_{\text{original}}) = 1 - p_{en}$. An overview illustration of *Entity Noising* is shown in Figure 1.

Entity Noising is different from a commonly used data augmentation method such as Wei and Zou (2019) which replaces entities with words similar to them. *Entity Noising* replaces entities with completely random noisy words. This feature allows the model to utilize entity-agnostic information, so that the model can learn to extract triples from sentences by focusing on the context information rather than the entities themselves. Therefore, with *Entity Noising*, the model is kept away from memorizing the entity pair along with its relation.

5 Experiments

We conduct a series of experiments with recent Relational Triple Extraction (RTE) methods on newly constructed datasets (Section 3).

Rearranged Dataset (Section 3.1) Table 3 shows the lack of generalization capabilities of recent RTE methods in *rearranged datasets* as well as original datasets. On *rearranged datasets*, *Entity Noising* consistently improves the ability of generalization on *unseen* triples, and for *partially seen* triples, it at least does not hurt the generalization capabilities. For original datasets, the evaluation can be biased on some specific *partially seen* and *unseen* samples since their proportion in test sets is small, rendering inconsistent results.

Overlap Sifted Dataset (Section 3.2) With *overlap sifted datasets* and original datasets, we evaluate recent RTE methods with and without *Entity Noising* to get more insight into what extent they generalize on unseen data. Table 4 shows that recent RTE methods struggle in extracting triples from unseen data, while *Entity Noising* promotes their generalization capabilities in most cases.

Augmented Test Set (Section 3.3) To assess whether the ability of an RTE method is influenced by the authenticity of the given text, we evaluate recent RTE methods with and without *Entity Noising* on *augmented test set*. We find that current RTE methods are substantially influenced by the authenticity of the given text, while *Entity Noising* relieves that influence by a huge margin (See Table 5).

6 Related Work

Open Information Extraction (Open IE) Open IE is the task of extracting relations from the given text without predefined relation type (Stanovsky et al., 2018; Zhan and Zhao, 2020; Cui et al., 2018; Kolluru et al., 2020). Although Open IE is a more general task than Relational Triple Extraction, it is necessary to extract information using fixed relation type to get high quality relational triples from specific domains such as science and business.

Data Leakage in NLP The overlapping problem between training and test data makes the evaluation biased towards assessing memorization capabilities of models. Several works point out the overlapping problem and quantify data leakage in basic NLP tasks (Elangovan et al., 2021) and Open-Domain Question Answering (Lewis et al., 2021), but Relational Triple Extraction was not considered yet.

	Method	Prec.	Rec.	F1
NYT	CasRel	39.6	22.4	28.6
	CasRel+EN	**54.3**	**34.5**	**42.2**
	TPLinker	44.5	22.6	30.0
	TPLinker+EN	**56.2**	**34.7**	**42.9**
	PRGC	37.2	25.4	30.2
	PRGC+EN	**51.8**	**28.1**	**36.4**
WebNLG	CasRel	66.9	32.1	43.4
	CasRel+EN	**70.4**	**53.6**	**60.9**
	TPLinker	69.6	39.1	50.1
	TPLinker+EN	**73.4**	**55.2**	**63.0**
	PRGC	67.5	42.0	51.8
	PRGC+EN	**69.0**	**56.3**	**62.0**

Table 5: Results of recent RTE methods with and without *Entity Noising* on *augmented test sets*.

7 Conclusion

In this paper, we disclosed for the first time that recent Relational Triple Extraction (RTE) methods struggle to extract triples from unseen data, which was previously unknown due to the test-train overlap problem in popular benchmark datasets. To properly assess the generalization capabilities of RTE methods, we developed three strategies to construct *rearranged dataset*, *overlap sifted dataset*, and *augmented test set* from original datasets. Furthermore, we proposed a simple yet effective noising method to promote generalization and experimentally confirm that it effectively improves the generalization capabilities of existing RTE methods.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2019-0-00075 Artificial Intelligence Graduate School Program(KAIST), No.2019-0-01371 Development of Brain-inspired AI with Human-like Intelligence) and National Research Foundation of Korea (NRF) grant (2018R1A5A1059921).

References

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. ISWC'07/ASWC'07, page 722–735, Berlin, Heidelberg. Springer-Verlag.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, page 1247–1250, New York, NY, USA. Association for Computing Machinery.

Lei Cui, Furu Wei, and Ming Zhou. 2018. Neural open information extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 407–413, Melbourne, Australia. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Aparna Elangovan, Jiayuan He, and Karin Verspoor. 2021. Memorization vs. generalization : Quantifying data leakage in NLP performance evaluation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1325–1335, Online. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 179–188, Vancouver, Canada. Association for Computational Linguistics.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online. Association for Computational Linguistics.

Keshav Kolluru, Samarth Aggarwal, Vipul Rathore, Mausam, and Soumen Chakrabarti. 2020. IMoJIE: Iterative memory-based joint open information extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5871–5886, Online. Association for Computational Linguistics.

Patrick Lewis, Pontus Stenetorp, and Sebastian Riedel. 2021. Question and answer test-train overlap in open-domain question answering datasets. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1000–1008, Online. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Feiliang Ren, Longhui Zhang, Shujuan Yin, Xiaofeng Zhao, Shilei Liu, and Bochao Li. 2021. *A Conditional Cascade Model for Relational Triple Extraction*, page 3393–3397. Association for Computing Machinery, New York, NY, USA.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 885–895, New Orleans, Louisiana. Association for Computational Linguistics.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020. Tplinker: Single-stage joint extraction of entities and relations through token pair linking. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1572–1582.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A novel cascade binary tagging framework for relational triple extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1476–1488, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Junlang Zhan and Hai Zhao. 2020. Span model for open information extraction on accurate corpus. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9523–9530. AAAI Press.

Hengyi Zheng, Rui Wen, Xi Chen, Yifan Yang, Yunyan Zhang, Ziheng Zhang, Ningyu Zhang, Bin Qin, Xu Ming, and Yefeng Zheng. 2021. PRGC: Potential relation and global correspondence based joint relational triple extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6225–6235. Association for Computational Linguistics.

A Training Details

In general, we train CasRel, TPLinker, and PRGC for 300, 500 epochs on NYT, WebNLG datasets. It takes 5 GPU days for training models on NYT and 1 GPU day for training models on WebNLG. We select the best model by only using the F1 score of the given validation set except *overlap sifted dataset*. For *overlap sifted dataset*, the training instances are sifted out according to the test instances, rendering the triple type statistics of valid and test sets are different. Therefore, we select the best model by using the F1 score of *overlap sifted* test sets. For Entity Noising, we set p_{en} to 0.1 and 0.05 for NYT and WebNLG datasets and set p_{en}^{len} to 0.4. Every model is based on pre-trained BERT model *BERT-base-cased* from Huggingface Transformers (Wolf et al., 2020), which contains 110M parameters.

B Dataset Statistics

The statistics of dataset split are shown in Table 6. To compute type F1 defined in Section 2.1, stratification is necessary by extracting test instances which only consist of single triple type among *entirely seen*, *partially seen* and *unseen*. The stratification statistics are shown in Table 7.

C Augmented test sets

Discussions on augmented test set It is worthy to note that the samples in the *augmented test set* may not be "true" statements in the real world but rather invented, as by construction their entities are replaced with other similar words (See examples in Figure 2). However, the true meaning of the entity words is fundamentally irrelevant to the relation between them given the context. Also, it is unknown whether the relation in the sentence is a fact. Thus, the ability of an RTE model to extract relational triples should not be influenced by the authenticity of the given text. Note that an ideal RTE model should be able to extract the relational triple (The [United States] President [Christopher]) if such fictitious content happens to exist in the given text.

Although the ideal RTE model should not be influenced by the authenticity of the given text, there exists potential risk. It is that the deployed RTE model might extract the invalid triple from wrong text. Therefore, the validation process which checks the triple is needed before adding it to the knowledge graph.

Construction details of augmented test set We now describe the construction details of the augmented test set. First, we preemptively run the language tokenizer to flag the wordpieces in the entity words. we substitute all entity words in the triples with masks (one mask *per word*, not *per wordpiece*). For single-word-single-wordpiece entities, we use the language model to fill in their masks independently. For single-word-multi-piece entities, we do not use the language model but search and substitute for the k-nearest words of the original entity word in the GloVe embedding space. For multi-word entities, each word constituting an entity is sequentially substituted using the language model.

Now we describe the detailed construction of $T_{\text{Augmented}}$. To measure the generalization performance properly, it is required that the *augmented test set* $T_{\text{Augmented}}$ consists of *partially seen* triples as well as *unseen* triples since the ideal RTE model is required to effectively extract both *partially seen* and *unseen* triples. Therefore, we first construct four augmented components of the test set T_{ss}, T_{su}, T_{us}, T_{uu} and take a union of them to create the final *augmented test set* $T_{\text{Augmented}} = T_{\text{ss}} \cup T_{\text{su}} \cup T_{\text{us}} \cup T_{\text{uu}}$. Among the four components, T_{ss} consists of triples with seen subject and object; T_{su} consists of triples with seen subject and unseen object; T_{us} is symmetrical with T_{su}; T_{uu} consists of triples with unseen subject and object.

We now describe the construction details of four components: T_{ss}, T_{su}, T_{us} and T_{uu}. First, for each sample in the test set $t^i_{\text{Standard}} \in T_{\text{Standard}}$, we get a set of top-$k$ similar entities E^{ij}_s for each entity e^{ij} in t^i_{Standard} independently, so that there is no correlation between each E^{ij}_s. Then, we uniformly sample e^{ij}_s from E^{ij}_s and replace e^{ij} with e^{ij}_s to get $t^i_{\text{Augmented}} \in T_{\text{Augmented}}$.

Construction of $\mathbf{T_{ss}}$ T_{ss} mainly consists of triples in which both subject and object entities are already seen in the training set. Therefore, every subject and object entity e^{ij}_s is sampled from $E^{ij}_s \cap E_{\text{Train}}$ uniformly, where E_{Train} is a set of entities appeared in the training set. If we encounter to sample from an empty set, we assign $e^{ij}_s = e^{ij}$.

Construction of $\mathbf{T_{su}}, \mathbf{T_{us}}$ T_{su} mainly consists of triples in which subject entities are seen and object entities are unseen in the training set. Therefore, subject and subject/object entities e^{ij}_s are sampled from $E^{ij}_s \cap E_{\text{Train}}$, and object entities e^{ij}_s are

Split	NYT						WebNLG					
	Ori.	Rearr.	Sift-1	Sift-2	Sift-3	Aug.	Ori.	Rearr.	Sift-1	Sift-2	Sift-3	Aug.
Train	56196	56196	50599	47152	44003	-	5019	5019	4776	3951	3193	-
Valid	5000	5000	5000	5000	5000	-	500	500	703	703	703	-
Test	5000	5000	5000	5000	5000	20000	703	703	703	703	703	2812

Table 6: Dataset statistics of original, *rearranged*, *overlap sifted* datasets, and *augmented test sets*.

Type	NYT					WebNLG				
	Ori.	Rearr.	Sift-1	Sift-2	Sift-3	Ori.	Rearr.	Sift-1	Sift-2	Sift-3
Entirely seen	4292	348	2733	2349	2064	580	155	435	249	160
Partially seen	473	3307	1703	2027	2265	42	178	82	133	172
Unseen	88	886	238	262	274	17	174	34	63	99
Others	147	459	326	362	397	64	196	152	258	272
Total	5000	5000	5000	5000	5000	703	703	703	703	703

Table 7: Stratified test set statistics of original, *rearranged*, and *overlap sifted* datasets. Each number indicates the number of instances which only consist of respective triple type. Note that an instance can have multiple triples associated with multiple triple types, which are defined with *Others* type.

sampled from $E_s^{ij} \backslash E_{\text{Train}}$ uniformly. T_{us} is constructed symmetrically.

Construction of $\mathbf{T_{uu}}$ T_{uu} mainly consists of triples in which both subject and object entities are unseen in the training set. Therefore, every subject and object entity e_s^{ij} is sampled from $E_s^{ij} \backslash E_{\text{Train}}$ uniformly.

Original Test Samples		Augmented Test Samples	
Above the Veil, from Australia, is the third book in a series after **Aenir** and **Castle**.	(**Above the Veil**, *precededBy*, **Aenir**) (**Aenir**, *precededBy*, **Castle**)	**Dark Wars Rising**, from Australia, is the third book in a series after **Sword** and **Avalon**.	(**Dark Wars Rising**, *precededBy*, **Sword**) (**Sword**, *precededBy*, **Avalon**)
Populous was the architect of **3Arena** in **Dublin** which was completed in December 2008.	(**3Arena**, *location*, **Dublin**) (**3Arena**, *architect*, **Populous**)	**Monolith** was the architect of **Trinity** in **Miami** which was completed in December 2008.	(**Trinity**, *location*, **Miami**) (**Trinity**, *architect*, **Monolith**)

Figure 2: Selected examples from WebNLG *augmented test set*.