STOP JUST RECALLING MEMORIZED RELATIONS: EXTRACTING UNSEEN RELATIONAL TRIPLES FROM THE CONTEXT

Anonymous authors
Paper under double-blind review

ABSTRACT

The ability to extract entities and their relations from unstructured text is essential for automated maintenance of large-scale knowledge graphs. To keep a knowledge graph up-to-date, it is required of an extractor to possess not only the ability to recall the triples encountered during training, but also the triples it has never seen before. In this paper, we show that although existing extraction models are able to 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 combat memorization and promote generalization, we present a simple yet effective noising framework that can be combined with existing models. By carefully noising the entities and their surrounding context, we refrain the model from simply memorizing the entities and their context, and promote generalization. To properly evaluate the generalization performance, we propose test set augmentation and train set sifting to emphasize unseen data. Experiments show that our model not only outperforms the current state-of-the-art in terms of generalization on the newly augmented unseen test data, but is also able to retain its memorization capabilities - achieving competitive results on the standard test data.

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 construction and maintenance of knowledge graphs such as Dbpedia (Auer et al., 2007), Freebase (Bollacker et al., 2008), and Wikidata (Vrandečić & Krötzsch, 2014) from documents containing a large number of new and emerging information.

The simplest approach to this task would be dictionary lookup: construct a dictionary of triples from the training data, then search for all (subject, object) entity pair occurrences for the dictionary entries and assign relations accordingly for a given test sample. However, such an approach can only work for triples already seen in the training data, and cannot generalize to *unseen* triples. Thus, an ideal RTE model must not only be able to *recall* and extract triples it has seen in training time (The [United States] President [Trump]), but also should be able to *generalize* beyond simple recalling and extract unseen triples by utilizing the context information (The [United States] President [Biden]).

To tackle RTE, early models have taken a pipeline approach (Zelenko et al., 2002), where first an entity recognition module captures the entities inside a given sentence, and a relation classification module identifies the relations between the entities. However, with recent advances in deep learning, trends have shifted to data-driven approaches, employing deep neural networks and jointly training the modules (Zeng et al., 2018). The arrival and integration of pretrained language models (Devlin et al., 2019; Radford et al., 2019) further elevated the performance of deep learning models (Wei et al., 2020; Wang et al., 2020). Zheng et al. (2021) adapted a BERT (Devlin et al., 2019) based entity extractor with a novel relation classifier architecture, achieving state-of-the-art performance. However, whether the performance of these methods attribute to their capabilities of recalling already seen data or their ability to generalize and extract relations from unseen data is yet to be scrutinized.

In this work, we uncover for the first time the fallacy of common RTE Benchmarks (Riedel et al., 2010; Gardent et al., 2017). Even though the exact sentences included in the training set and the test set are different, there is a significant overlap of relational triples between the training and the test set. Thus, the performance results achieved using these datasets are heavily biased towards recalling seen data. Re-evaluating the current state-of-the-art method using our *rectified* datasets reveal that the performance of previous models are overestimated, and their ability to generalize to unseen triples is impaired.

Previously introduced approaches are exposed to potential overfitting as their output is directly conditioned on the word embeddings of entities. While this approach is seemingly innocent, direct conditioning on input words is likely to guide the model to memorize the identities and meanings of entity words, being reduced to a lookup-based model. In other words, it is unclear whether if the model is truly extracting relations based on the context and structure of the sentence, or jumping to hasty conclusions based on the occurrence of certain entity pairs.

To resolve this generalization problem, we propose a simple yet effective training technique called **Entity Noising**. By replacing the entities in a training sentence with randomly sampled words and subwords, we prevent the model from memorizing the subject and object entity pair itself and its relations (i.e., memorizing triples), but instead utilize the context (i.e., the non-entity words) of the sentence to detect the entities and determine their relations.

We further propose **Context Noising** which supports Entity Noising. Although **Entity Noising** is surprisingly good at making the model to understand the relation between entities from the context of the sentences, it has a potential risk to bias on a certain structure or form of sentences since it produces a numerous samples with exactly same sentence with different entities. To alleviate this problem, we also perform some perturbation on the context of the sentence. On unseen data, models trained with Entity Noising along with Context Noising achieves superior performance to the previous methods.

Our contributions are:

- We show for the first time that the current benchmark datasets for relation triple extraction exhibit significant entity pair overlap between training and test data. Moreover, we confirm the current state-of-the-art models trained on such datasets cannot generalize well to unseen triples.
- We propose entity noising, a novel technique that efficiently promotes generalization in RTE models. By substituting the original entities with completely random words, the model learns to focus on the context of the sentence rather than the meaning of the words. This method is surprisingly good at extracting triples from unseen data.
- We further propose context noising which performs some perturbation on the context of the sentence to alleviate the problem that the model focuses on extracting noised entities from fixed non-noised context. With help of this simple perturbation, the proposed model performs well not only unseen data but also seen data.
- We validate the effectiveness of our framework through experiments on both original and rectified benchmark datasets, which shows that our model is both capable of recalling seen data and generalizing to unseen data.

The rest of the paper is organized as follows. In Section 2, we will introduce existing works for the relational triple extracting task. In Section 3, we will address the problem of existing works that just recalling the memorized relation between seen entity pairs by carefully analyzing the results on two standard benchmark datasets. Furthermore, we add new test cases or remove some of training cases to make the benchmark datasets to be more appropriate to the practical relational extracting task. In Section 4, we will propose simple but effective nosing methods for the practical relational extracting task. Then we will show extensive experimental results with a real case study in Section 5. Finally, we will conclude our paper in Section 6.

2 Related Work

Relational triple extraction Early attempts on tackling the relational triple extraction task opted for a divide-and-conquer strategy, where pipelined approaches were employed (Zelenko et al., 2002;

Table 1: Statistics of *entirely seen*, *partially seen* and *unseen* triples in validation, test, and *augmented test set*s of NYT and WebNLG datasets.

Category	NYT			WebNLG		
	Valid	Test	Augmented Test	Valid	Test	Augmented Test
Entirely seen (%)	**89.99**	**89.61**	5.76	**91.33**	**91.10**	17.21
Partially seen (%)	8.28	8.64	46.33	6.38	7.47	36.17
Unseen (%)	1.72	1.75	**47.91**	2.29	1.43	**46.62**

Chan & Roth, 2011; Mintz et al., 2009). The entities in the given sentence were extracted first, then the relationships between the entities were determined. However, such models have failed to consider the information present in the correlation between entity extraction and relation classification. Moreover, the division of tasks introduced error propagation into the framework.

Subsequently, works toward joint learning of entity and relation extraction were proposed. Yu & Lam (2010) constructed a conditional random field (CRF) model to jointly model entity tagging and relation extraction. Li & Ji (2014) proposed a segment-based decoder with local and global features to jointly extract entities and their relation in an incremental fashion. Albeit constructing joint models to utilize more information from data, they were heavily reliant on feature engineering and human knowledge.

With the successful introduction of deep learning into the field of natural language processing, data-driven deep learning methods were proposed to circumvent feature engineering issues. Katiyar & Cardie (2016); Miwa & Bansal (2016) used Long Short-Term Memory (Hochreiter & Schmidhuber, 1997) networks to process input sentences to extract the entities and predict their relations. Parameter sharing was employed to facilitate information sharing between the entity extraction task and the relationship classification task. However, the model was not jointly trained. Zheng et al. (2017) proposed to jointly train a deep neural network to extract entities and their relations. However, they could not extract overlapping relation due to architectural restrictions.

With the popularization of language model pretraining (Devlin et al., 2019; Radford et al., 2018; 2019; Brown et al., 2020), Wei et al. (2020) proposed a cascaded model built upon a BERT (Devlin et al., 2019) encoder, while Wang et al. (2020) proposed a joint prediction model. The current state-of-the-art is achieved by Zheng et al. (2021), where instead of considering all relations, potential relations are predicted first and a global correspondence component is employed to align the entities.

Data augmentation Unlike Computer Vision where data augmentation has become almost a free lunch, augmentation in the Natural Language Processing (NLP) domain is nontrivial (Feng et al., 2021) due to the discrete nature of languages. However, a number of works are proposed to effectively port augmentation methods into the NLP field, or to propose augmentations tailored to the field. Wei & Zou (2019) proposed Easy Data Augmentation, which is a rule-based augmentation policy comprised of synonym replacement, swap, insert and delete for randomly chosen words. Instead of relying on fixed rules, another line of work proposed to utilize pretrained models to augment samples by paraphrasing existing samples (Sennrich et al., 2016) or generating totally new samples (Anaby-Tavor et al., 2020). However, such methods are unfit for the RTE task as their policies are entity-agnostic.

3 Generalization Capabilities of the Current State-of-the-arts

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

Toward this, we first disclose that the current de facto benchmark datasets NYT (Riedel et al., 2010) and WebNLG (Gardent et al., 2017) are inadequate for testing generalization as 89.61% and 91.10% of triples in NYT and WebNLG *test sets* completely overlap with triples in their respective training sets (denoted as *entirely seen* in Table 1), while *partially seen* (only overlaps partially) and *unseen* (completely new) samples that require generalization to predict are but a small portion (in Table 1). This implies that these test sets are severely biased towards assessing the capabilities of recalling the triples already present in the training set. The detailed descriptions of these three categories can be found in Appendix A.1.

Table 2: Precision, recall and f1 score of recent RTE models on standard and augmented NYT and WebNLG test sets. Numbers in ( ) are from their papers, other numbers are from our reproduction.

	Method	NYT			WebNLG		
		Prec.	Rec.	F1	Prec.	Rec.	F1
Standard	CasRel (Wei et al., 2020)	90.2	90.0	90.1 (89.0)	90.1	86.6	88.3 (86.4)
	TPLinker (Wang et al., 2020)	92.7	92.2	92.4 (92.0)	90.3	88.3	89.3 (86.7)
	PRGC (Zheng et al., 2021)	90.3	89.4	89.9 (92.7)	89.5	86.0	87.7 (88.5)
Augmented	CasRel (Wei et al., 2020)	39.6	22.4	28.6	66.9	32.1	43.4
	TPLinker (Wang et al., 2020)	45.9	22.6	30.3	69.4	39.1	50.0
	PRGC (Zheng et al., 2021)	37.9	21.0	27.1	61.6	33.2	43.2

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 1: Selected examples from WebNLG *augmented test set*.

To resolve this issue and further reveal the true generalization performance on unseen cases, we develop two simple ways to modify the standard benchmark datasets. Although it is possible to focus only on *partially seen* or *unseen* triples from the current test sets for testing generalization, their numbers are too small (around 10%; see Table 1), rendering the evaluations unreliable. Therefore, we increase the proportion of *partially seen* and *unseen* triples in the test sets by augmenting them (Section 3.1) or sifting out training instances that overlap with the test set (Section 3.2), rendering them unobserved. In Section 5, we evaluate the generalization performance of existing methods and ours using the two revised datasets above. The revised datasets will be publicly available soon. In addition to the tests using the revised datasets, we apply the RTE models trained on the standard datasets to sentences from Wikipedia to further analyze how they behave on unseen cases in more realistic situations.

3.1 Augmented Test Set for Testing Generalization

To fairly evaluate the generalization performance, we first create an *augmented test set* $T_{\text{Augmented}}$ by increasing the proportion of *partially seen* and *unseen* triples in the standard test set. The key idea of constructing $T_{\text{Augmented}}$ 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). 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 are sequentially substituted using the language model. The detailed construction of $T_{\text{Augmented}}$ are found in Appendix A.2.

Table 1 shows that the *augmented test sets* exhibit large proportions of *partially seen* and *unseen* triples. In the NYT *augmented test set*, 46.33% of the triples are *partially seen* and 47.91% are *unseen*, meanwhile in the WebNLG *augmented test set*, 36.17% are *partially seen* and 46.62% are *unseen*. A number of selected examples are displayed in Figure 1.

It is also 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. However, the true meaning of the entity words is fundamentally irrelevant to the relation between them given the context. Thus, the ability of an RTE model to extract relational triples should not be influenced by the authenticity of the given text. For example, the ideal RTE model should be able

Table 3: Statistics of *entirely seen*, *partially seen*, and *unseen* triples in each *overlap sifted* NYT and WebNLG test sets including standard test sets.

Category	NYT				WebNLG			
	Original	D^1_{Sift}	D^2_{Sift}	D^3_{Sift}	Original	D^1_{Sift}	D^2_{Sift}	D^3_{Sift}
Sifted sent. (%)	0	9.7	15.8	21.4	0	4.8	21.3	36.4
Entirely seen (%)	89.61	63.24	55.45	49.27	91.10	78.03	56.50	39.20
Partially seen (%)	8.64	31.56	38.09	43.19	7.47	17.05	30.86	37.40
Unseen (%)	1.75	5.20	6.46	7.54	1.43	4.92	12.63	23.40

Table 4: F1 scores of recent RTE models on original and *overlap sifted datasets*. To make the triples are completely *unseen* even for the underlying BERT, we used randomly initialized BERT.

	Method	NYT				WebNLG			
		Entire	Partial	Unseen	Total	Entire	Partial	Unseen	Total
Original Train	CasRel$_{Random}$	86.7	47.4	20.7	82.1	78.2	25.0	9.5	73.9
	TPLinker$_{Random}$	93.2	48.8	28.6	88.4	81.1	36.6	26.1	76.5
D^1_{Sift}	CasRel$_{Random}$	93.0	48.8	24.3	76.5	84.2	47.3	40.8	73.4
	TPLinker$_{Random}$	97.5	51.7	22.9	80.5	86.3	53.7	15.0	74.3
D^2_{Sift}	CasRel$_{Random}$	93.4	49.1	22.6	73.2	86.9	40.0	14.1	63.6
	TPLinker$_{Random}$	97.8	52.1	23.2	77.1	90.6	29.9	12.8	62.9
D^3_{Sift}	CasRel$_{Random}$	94.3	49.3	22.5	70.8	90.0	40.0	10.4	52.7
	TPLinker$_{Random}$	98.0	53.3	20.8	74.5	94.2	29.3	2.8	51.7

to extract the relational triple (The [United States] President [Christopher]) if such fictitious content happens to exists in the given text.

Using the newly constructed test set $T_{\text{Augmented}}$, we test existing RTE models to verify whether they can correctly function on augmented test samples comprised of *partially seen* or *unseen* triples. We evaluate the recently proposed CasRel (Wei et al., 2020) and TPLinker (Wang et al., 2020), and the current state-of-the-art PRGC (Zheng et al., 2021) on the newly constructed *augmented test sets*. Table 2 shows that these models perform poorly on the *augmented test sets*, albeit achieving high F1 scores on the standard test sets.

3.2 Overlap Sifted Dataset for Testing Generalization

Albeit its effectiveness in testing generalization capabilities of existing RTE models, one might think of the augmented test dataset as *"unnatural"*, in the sense that they are not curated purely from existing texts. Thus, there exists a demand for an auxiliary dataset being comprised of only natural texts that is still capable for testing generalization (although sub-par compared to *augmented test set*).

With the above in mind, we propose another simple strategy to augment the dataset with unseen test samples while staying natural: removing train-test overlapped triples from the training data. To render a triple unobserved, we remove the sentences containing overlapped triples from the training set. Specifically, we randomly choose $k\%$ of the unique triples from the test set, then remove all the sentences containing the selected triples from the *training set* to construct an *overlap sifted dataset*. For demonstration, we construct three such datasets: D^1_{Sift}, D^2_{Sift}, and D^3_{Sift} by choosing $k = 5, 10, 15\%$, respectively. As a result, for the NYT dataset, approximately 10%, 16%, and 21% of sentences are removed and approximately 5%, 21%, and 36% of sentences are removed for the WebNLG dataset. The detailed statistics of the *overlap sifted datasets* are presented in Table 3.

Evaluating RTE models using *overlap sifted datasets* requires separate learning dependent on each experiment due to the difference in the training data. We train CasRel and TPLinker on D^1_{Sift}, D^2_{Sift}, and D^3_{Sift}, since they show better performances on standard than PRGC with the reproducibility issue in its official code[1] as shown in Table 2. The purpose of creating an *overlap sifted dataset* is not only

[1] https://github.com/hy-struggle/PRGC

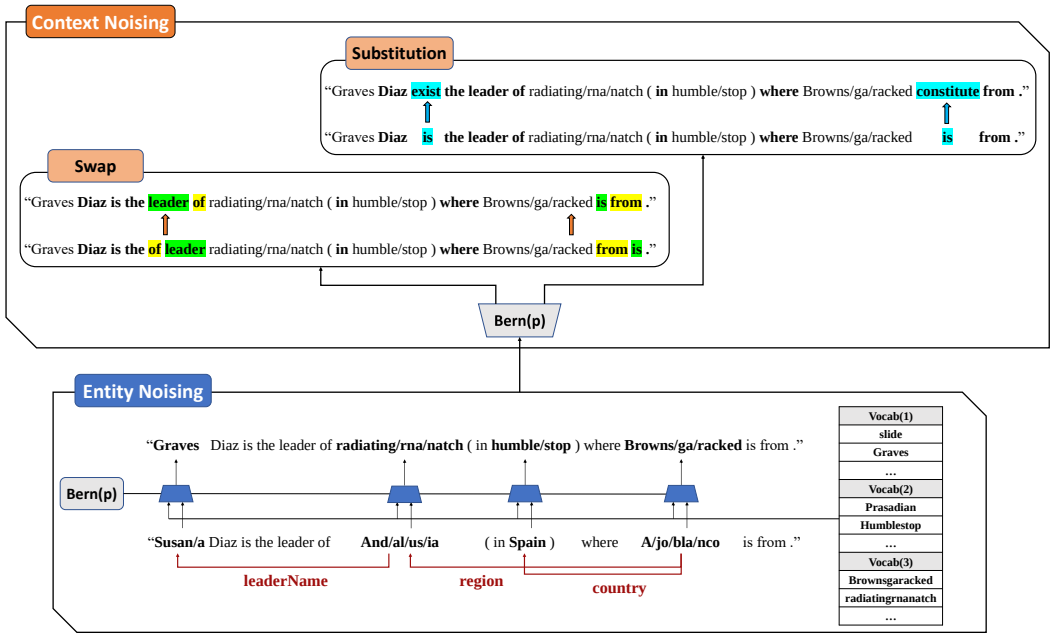


Figure 2: Overview of our framework.

to increase the portion of unseen test cases but to expose the RTE models to an increased amount of sentences containing completely unseen new facts. To this end, the utilization of a pretrained BERT backbone is problematic as the backbone contains vast amounts of subconscious factual knowledge. Therefore, for training RTE models for benchmark on *overlap sifted datasets*, we do not use a pretrained BERT but randomly initialize the backbone to completely deprive them of the BERT knowledge base and soak them thoroughly on the unobserved factual triples. We believe this also has the effect of clearly distinguishing entire/partial/unseen categories.

The results of existing RTE models on *overlap sifted datasets* are depicted in Table 4. In case of TPLinker$_{Random}$ on NYT dataset, F1 score is as high as 88.44 even without a pre-train BERT. However, the model degrades as the portion of unseen cases increases. Note that performances on *entirely seen* cases get better as the portion of unseen cases increases. We believe this is because, the pruning reduces the number of seen triples to be memorized and this is an evidence for how much RTE models rely on memorizing triples.

4 Noising Frameworks for Generalization

In this section, we present Entity Noising, a training technique to enhance the generalization performance of existing RTE methods. Entity noising allows the model to utilize entity-agnostic information, so that the model is able to extract triples from sentences by focusing on the context information rather than the information in the entities themselves. Therefore, with Entity Noising, the model is kept away from memorizing the entity pair along with its relation.

To assist the effect of Entity Noising, we further present Context Noising, an auxiliary noising scheme. Context Noising prevents the model from memorizing surrounding words and sentence structures, which can be unintentionally exploited as a proxy to memorizing the triple itself. The overall procedures of Entity noising along with Context Noising is described in Figure 2.

4.1 Entity Noising

We now describe Entity Noising in detail. The key idea of Entity Noising is to replace the entities in the given training input sentence with completely random noisy words. This is different from applying existing data augmentation methods such as Easy Data Augmentation (EDA) (Wei & Zou, 2019) to replace entities with words similar to them, since the entity information still persist with such replacement. In contrast, Entity Noising replace entities with completely random noisy words, so that RTE models can utilize entity-agnostic information to extract triples.

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, 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 $w' \sim P(w' \mid w)$, the 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:

$$w'_k = \begin{cases} w'_k \sim P(w'_k \mid w_k), & \text{if } w_k \text{ is an entity} \\ w_k, & \text{otherwise} \end{cases} \tag{1}$$

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 $P(\mathbf{x} = \mathbf{x}_{\text{original}}) = 1 - p_{en}$. This guides a model not to memorize entity specific information, but to utilize context information to extract triples.

With this training technique, the model takes multiple sentences which share context words and structure while having diverse noisy entities. Therefore, the entity specific information can no longer be exploited by the model to extract triples. Consequently, the model is trained to ignore entity specific information and only utilizes context information to extract triples.

4.2 Context Noising

Since Entity Noising repeatedly replaces entity words to noisy words while leaving the context (i.e., non-entity words) intact, the trained model witnesses a variety of similar sentences with different noised entities during the training. This poses a potential risk as the model can develop a bias towards exploitative solutions, as in catching certain structures cues and concentrating on distinguishing entity boundaries. To alleviate this problem, we introduce two sentence perturbation techniques on the context (non-entity parts) of the sentence: *Swap* and *Substitution*.

Swap *Swap* simply switches two strictly neighboring non-entity words with a small probability p_{cn}^{swap}. We do not swap two non-entity words if an entity word is placed in the middle of the two non-entity words in order not to damage the context information that our RTE models should focus on.

Substitution *Substitution* replaces a non-entity word to another similar word with a small probability p_{cn}^{sub}. To obtain a similar word, we can retrieve: either a synonym defined on Wordnet (Miller, 1995), or a predicted word from masking a target non-entity word using BERT (Devlin et al., 2019). However, we found that the BERT variation occasionally substitutes a word with a word of completely different meaning, since the prediction is conditioned on the words around the masked word and not the masked word itself. Therefore, we use Wordnet for the substitution process. To encourage generating diverse perturbed contexts from a sentence, we randomly select a word among Wordnet synonyms for substitution.

Note that we also set the maximum number of swaps or substitutions per sentence to 5 to maintain the contextual integrity of the original sentence in a broad sense. With Context Noising, it is now possible to show diverse contexts to the model so that the model may not memorize surrounding words or sentence structures that can be exploited to memorize triple itself.

5 Experiments

We conduct a series of experiments to test the generalization capabilities of a variety of RTE baselines, and verify that our method entity noising along with context noising can be applied to them to improve their generalization power. The results on *augmented test set* and *overlap sifted dataset* are in Section 5.2 and 5.3. Apart from evaluation on those two revised datasets, we also qualitatively study a baseline and our method with completely unseen sentences from Wikipedia in Section 5.4.

Table 5: Results of baselines RTE models and models with our nosing methods on standard and *augmented test set*.

Method	NYT-Standard			NYT-Augmented			WebNLG-Standard			WebNLG-Augmented		
	Prec.	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*
Novel Tagging†	32.8	30.6	31.7	-	-	-	52.5	19.3	28.3	-	-	-
MultiHead†	60.7	58.6	59.6	-	-	-	57.5	54.1	55.7	-	-	-
ETL-Span†	85.5	71.7	78.0	-	-	-	84.3	82.0	83.1	-	-	-
CasRel†	89.8	88.2	89.0	-	-	-	88.3	84.6	86.4	-	-	-
TPLinker†	91.4	92.6	92.0	-	-	-	88.9	84.5	86.7	-	-	-
PRGC†	93.5	91.9	92.7	-	-	-	89.9	87.2	88.5	-	-	-
CasRel§	90.2	90.0	90.1	39.6	22.4	28.6	90.1	86.6	88.3	66.9	32.1	43.4
CasRel+EN+CN	**90.5**	**90.2**	**90.3**	**50.4**	**33.0**	**39.9**	**90.8**	**87.7**	**89.2**	**69.6**	**46.0**	**55.4**
TPLinker§	92.7	92.2	92.4	45.9	22.6	30.3	90.3	88.3	89.3	69.4	39.1	50.0
TPLinker+EN+CN	**92.9**	**92.6**	**92.7**	**56.9**	**33.2**	**41.9**	**91.2**	**88.4**	**89.8**	**71.7**	**46.2**	**56.2**
PRGC§	90.3	**89.4**	89.9	37.9	21.0	27.1	89.5	86.0	87.7	61.6	33.2	43.2
PRGC+EN+CN	**91.6**	88.3	**90.0**	**52.3**	**32.9**	**40.4**	**89.7**	**87.2**	**88.4**	**66.6**	**50.9**	**57.7**

†: Paper reported score §: Our reproduced score

5.1 Dataset and Training Details

We evaluate our method on two well known benchmark datasets NYT (Riedel et al., 2010) and WebNLG (Gardent et al., 2017) following Wang et al. (2020) and Zheng et al. (2021). Both NYT and WebNLG have reduced versions, named NYT* and WebNLG*, where only the last word of entities are annotated. Considering that the ultimate goal of enhancing generalization capabailties of RTE models is to append new information to knowledge graphs, extracting triples which have only last word of subjects and objects are not useful. Therefore, we only evaluate RTE models on full original version of NYT and WebNLG. Since it is difficult to measure the generalization capabilities of RTE models with standard NYT and WebNLG datasets, we construct two revised datasets - *augmented test set*, *overlap sifted dataset* - and use them for testing generalization (Section 3).

We compare our method against the state-of-the-art RTE baselines: CasRel, TPLinker, and PRGC. For comparison purpose, we also depicted performances of previous strong baselines: NovelTagging (Zheng et al., 2017), MultiHead (Bekoulis et al., 2018), and ETL-Span (Yu et al., 2020) on standard setting. To reproduce current state-of-the-arts, we follow the officially published implementations of them unless specified. We defer training details to Appendix A.3.

5.2 Generalization Performance: Augmented Test Set

We evaluate the generalization capabilities of RTE models with *augmented test set*, which is constructed by augmenting the test sets of NYT and WebNLG so that the proportion of *partially seen* and *unseen* triples are increased 5.2. Table 5 shows the generalization performance of our method against current state-of-the-arts on the *augmented test set* as well as the performance on the standard test set. Equipped with Entity Noising and Context Noising (EN+CN), the generalization performance of every current state-of-the-arts on the *augmented test set* increased significantly. Furthermore, the performance on the standard test sets also increased after applying EN+CN to the current state-of-the-arts, demonstrating that the noising framework does not harm the capabilities of recalling triples already seen in the training set. More detailed analysis of the results on the *augmented test sets* can be found in Appendix A.4.

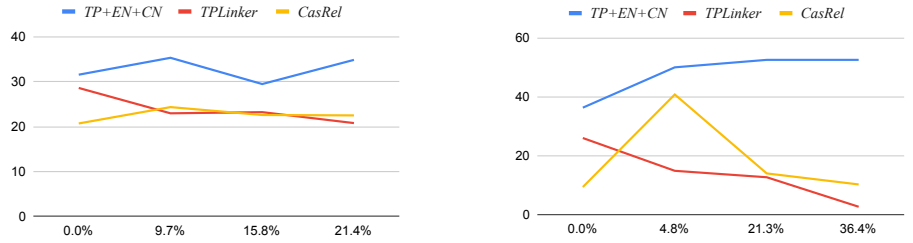
5.3 Generalization Performance: Overlap Sifted Dataset

To evaluate the effectiveness of our method on a purely natural dataset, we train and test our method on *overlap sifted datasets* created in Section 3.2. As in Section 3.2, a randomly initialized BERT backbone was employed to deprive the model of the BERT knowledge base. As shown in Table 6, RTE models with Entity Noising along with Context Noising (EN+CN) consistently outperform the baseline RTE models. For *unseen* cases of WebNLG-D^3_{Sift} dataset which has the highest sifted ratio, the models with EN+CN show substantial improvements (2.8→52.5 for TPLinker, and 10.4→47.5 for CasRel). We report the F1 scores of unseen cases on RTE models trained on *overlap sifted datasets* in Figure 3.

Table 6: Result on *overlap sifted dataset*. Randomly initialized BERT used.

		NYT / overlap sifted ratio %				WebNLG / overlap sifted ratio %			
Category	Method	Original 0%	D^1_{Sift} 3.0%	D^2_{Sift} 15.8%	D^3_{Sift} 21.4%	Original 0%	D^1_{Sift} 4.5%	D^1_{Sift} 21.3%	D^1_{Sift} 36.4%
Entirely seen	CasRel+EN+CN	86.8(+0.1)	93.6(+0.5)	93.8(+0.4)	95.3(+1.0)	84.3(+6.1)	89.0(+4.9)	91.7(+4.8)	92.9(+2.9)
	TPLinker+EN+CN	94.3(+1.0)	97.73(+0.2)	98.28(+0.5)	98.37(+0.4)	88.8(+7.7)	91.5(+5.2)	92.9(+2.2)	95.0(+0.8)
Partially seen	CasRel+EN+CN	48.2(+0.7)	54.7(+5.9)	51.9(+2.9)	51.7(+2.4)	46.3(+21.3)	51.4(+4.1)	46.1(+6.1)	52.5(+12.5)
	TPLinker+EN+CN	55.5(+6.7)	56.6(+4.9)	55.1(+2.9)	55.2(+1.9)	50.6(+14.0)	57.3(+3.6)	49.3(19.4)	52.9(+23.7)
Unseen	CasRel+EN+CN	22.4(+1.7)	33.1(+8.8)	34.4(+11.8)	31.1(+8.6)	46.7(+37.1)	51.0(+10.2)	55.0(+40.9)	47.5(+37.1)
	TPLinker+EN+CN	31.5(+3.0)	35.3(+12.3)	29.4(+6.2)	34.8(+14.0)	36.4(+10.3)	50.0(+35.0)	52.5(+39.7)	52.5(+49.7)
Total	CasRel+EN+CN	82.3(+0.2)	78.4(+1.9)	75.1(+1.8)	72.6(+1.8)	80.3(+6.4)	80.3(+6.9)	70.9(+7.3)	62.7(+10.0)
	TPLinker+EN+CN	89.8(+1.4)	82.2(+1.7)	78.9(+1.8)	76.1(+1.5)	85.0(+8.6)	82.5(+8.3)	70.8(+8.0)	62.2(+10.5)

( ) shows increments over the baseline CasRel/TPLinker model



(a) Unseen F1 on *overlap sifted* NYT.

(b) Unseen F1 on *overlap sifted* WebNLG.

Figure 3: Unseen F1 on *overlap sifted dataset*. Randomly initialized BERT used. Proposed entity nosing with context nosing method consistently outperform baselines by a huge margin.

5.4 Case Study on Real True Unseen Samples

Although we extensively verified the effectiveness of our noising methods so far, we further investigate the ability of the proposed methods on "actual true" *unseen* triples from completely unseen sentences. To this end, we select 16 triples that "was true in the past" but "no longer true" from the NYT training dataset. The details regarding selecting the 16 triples are elaborated in Appendix A.6. For each triple, to compose a real true unseen sentence, we manually select sentences which imply a new true fact(s) from the subject entity's Wikipedia page. The complete list of 16 sentences and extracted triples from baseline TPlinker and proposed TPlinker+EN+CN models are shown Tables 10 and 11 in Appendix A.6. As shown in Table 7, TPlinker+EN+CN outperforms baseline TPLinker by a large margin.

Table 7: Result on real true unseen samples.

	Precision	Recall	F1
TPLinker	29.4	25.3	27.8
TPLinker+EN+CN	**61.9**	**68.4**	**65.0**

6 Conclusion

To the best of our knowledge, we firstly disclosed two well-known benchmark datasets NYT and WebNLG for the relational triple extraction task are not inadequate for testing generalization since about 90% of triples in test sets completely overlap with triples in training sets. This leads to poor generalization performances of existing RTE models which mainly benchmarked on two datasets. To reveal a true generalization capability, we developed two strategies called *augmented test set* and *overlap sifted dataset* that can be applied to both datasets. Furthermore, we proposed a simple yet effective noising method to improve the generalization performance. Our method advances generalization capabilities of existing RTE models in a huge margin, while also able to retain its memorization capabilities.

REFERENCES

Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper, and Naama Zwerdling. Do not have enough data? deep learning to the rescue! In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7383–7390, 2020.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. ISWC'07/ASWC'07, pp. 722–735, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 3540762973.

Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. Joint entity recognition and relation extraction as a multi-head selection problem. *Expert Systems with Applications*, 114: 34–45, 2018.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 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, pp. 1247–1250, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605581026. doi: 10.1145/1376616.1376746. URL https://doi.org/10.1145/1376616.1376746.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. 2020.

Yee Seng Chan and Dan Roth. Exploiting syntactico-semantic structures for relation extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 551–560, 2011.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. A survey of data augmentation approaches for nlp. *arXiv preprint arXiv:2105.03075*, 2021.

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

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4163–4174, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.372. URL https://aclanthology.org/2020.findings-emnlp.372.

Arzoo Katiyar and Claire Cardie. Investigating lstms for joint extraction of opinion entities and relations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 919–929, 2016.

Qi Li and Heng Ji. Incremental joint extraction of entity mentions and relations. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 402–412, 2014.

George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11): 39–41, 1995.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 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*, pp. 1003–1011, 2009.

Makoto Miwa and Mohit Bansal. End-to-end relation extraction using LSTMs on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1105–1116, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1105. URL https://aclanthology.org/P16-1105.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014. URL http://www.aclweb.org/anthology/D14-1162.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

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

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

Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 86–96, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1009. URL https://aclanthology.org/P16-1009.

Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, September 2014. ISSN 0001-0782. doi: 10.1145/2629489. URL https://doi.org/10.1145/2629489.

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

Jason Wei and Kai Zou. 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)*, pp. 6382–6388, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1670. URL https://aclanthology.org/D19-1670.

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

Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Tingwen Liu, Yubin Wang, Bin Wang, and Sujian Li. Joint extraction of entities and relations based on a novel decomposition strategy. In *ECAI 2020*, pp. 2282–2289. IOS Press, 2020.

Xiaofeng Yu and Wai Lam. Jointly identifying entities and extracting relations in encyclopedia text via a graphical model approach. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pp. 1399–1407, 2010.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. Kernel methods for relation extraction. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pp. 71–78. Association for Computational Linguistics, July 2002.

Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 506–514, Melbourne, Australia, July 2018. Association for Computational Linguistics.

Hengyi Zheng, Rui Wen, Xi Chen, Yifan Yang, Yunyan Zhang, Ziheng Zhang, Ningyu Zhang, Bin Qin, Xu Ming, and Yefeng Zheng. 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)*, pp. 6225–6235. Association for Computational Linguistics, August 2021. doi: 10.18653/v1/2021.acl-long.486. URL https://aclanthology.org/2021.acl-long.486.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1227–1236, 2017.

A APPENDIX

A.1 DEFINITIONS OF TRIPLE CATEGORIES

Since triples can be overlapped partially or entirely, we classify triples in the validation and test sets into three categories - *entirely seen*, *partially seen* and *unseen*. For a set of triples in the training set $S = \{(s_i, r_i, o_i)\}_{i=1}^{n}$, the category of each triple (s, r, o) in the validation and test sets are defined as follows. First, a triple (s, r, o) belongs to the *entirely seen* category if $(s, r, o) \in S$. For *partially seen* category, 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* category.

A.2 CONSTRUCTION DETAILS OF AUGMENTED TEST SET

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 $t^i_{\text{Augmented}} \in T_{\text{Augmented}}$. The details of getting similar entities are described in Sections 3.1.

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 sampled from $E^{ij}_s \backslash E_{\text{Train}}$ 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^{ij}_s is sampled from $E^{ij}_s \backslash E_{\text{Train}}$ uniformly.

A.3 TRAINING DETAILS

In general, we train CasRel and TPLinker for 300, 500 epochs on NYT, WebNLG datasets and train PRGC for 200 epochs on both NYT, WebNLG datasets. We select the best model by only using F1 score of given validation set. For Entity Noising, we set p_{en} to 0.1 and 0.05 for NYT and WebNLG datasets and set p^{len}_{en} to 0.4. For Context Noising, we set context noising probability p_{cn} to 0.3 for both NYT and WebNLG datasets and set both swap probability p^{swap}_{cn} and substitution probability p^{sub}_{cn} for each word to 0.1.

A.4 DETAILED ANALYSIS ON AUGMENTED TEST SET

Since the *augmented test set* is the union of four components T_{ss}, T_{su}, T_{us}, T_{uu} (Appendix A.2), we also evaluate on each component as well. Table 8 shows that not only the generalization performance on the *augmented test set* increased significantly when Entity Noising along with Context Noising (EN+CN) applied to existing RTE models (See Table 5), but also EN+CN consistently perform well on every four components. This implies that RTE models equipped with EN+CN can extract both

partially seen and *unseen* triples, since T_{ss}, T_{su}, T_{us} have large proportion of *partially seen* triples and T_{uu} has large proportion of *unseen* triples.

Table 8: Results on four components T_{ss}, T_{su}, T_{us}, T_{uu} of *augmented test set*.

Method	NYT-Enlarged				WebNLG-Enlarged			
	$\mathbf{T_{ss}}$	$\mathbf{T_{su}}$	$\mathbf{T_{us}}$	$\mathbf{T_{uu}}$	$\mathbf{T_{ss}}$	$\mathbf{T_{su}}$	$\mathbf{T_{us}}$	$\mathbf{T_{uu}}$
CasRel	58.4	23.8	17.8	8.6	71.3	39.3	31.7	20.9
CasRel+EN+CN	**59.6**	**34.3**	**27.6**	**24.1**	**78.6**	**53.5**	**45.7**	**39.2**
TPLinker	9.9	26.4	18.7	60.9	27.9	45.4	40.1	77.3
TPLinker+EN+CN	**28.8**	**38.3**	**32.9**	**64.2**	**38.0**	**54.6**	**50.2**	**80.6**

A.5 Ablation Study of Entity Noising and Context Noising

We compare Entity Noising (EN) and Entity Noising along with Context Noising (EN+CN) on TPLinker model and two revised WebNLG dataset: *overlap sifted dataset* and *augmented test set*. Table 9 shows that EN enhance the generalization capabilities of existing RTE models, and CN can assist EN to further enhance the generalization capabilities.

Table 9: Entity noising and context noising ablation results.

Method	WebNLG-Sifted $\left(D^3_{\text{Sift}}\right)$				WebNLG-Augmented
	Entire	Partial	Unseen	Total	Total
TPLinker	**96.6**	63.5	60.8	70.9	46.6
TPLinker+EN	95.6	64.0	59.2	71.3	53.6
TPLinker+EN+CN	95.0	**64.3**	**64.8**	**72.3**	**55.5**

A.6 Detailed Real True Unseen Samples

We select some triples that "was true in the past" but "no longer true" from NYT training dataset. Since finding those triples among whole triples in training data is intractable. We narrow down candidates to the triples having "/business/person/company" (hereafter referred to as "company") relation since turnovers are common in the business world. We checked first 50 triples having "company" relation in the original NYT dataset[2] in order of their appearances in the training sentences. Note that we also restrict candidates to triples that appear more than 10 sentences and less than 50 sentences in the training data, to ensure the model sufficiently witnesses the triples and rule out abnormally duplicately labeled triples.

Among 50 candidate triples, we are able to find 16 triples that are no longer true by checking the Wikipedia webpage of the subject entity (i,e, Person's wikipedia page). To construct truly unseen sentences having real existential *unseen* triples, we carefully choose sentences imply a new fact(s) from Wikipedia. We tried to minimize modifying the original sentence from Wikipedia, but removing reference indices (i.e., remove "[ ]") and replacing a pronoun or a simplified name refers the subject to the subject entity word were inevitable (i.e., June 2018, Disney announced that [he →John Lasseter] would be leaving the company ...). We also simply concatenate two or more sentences from Wikipedia if needed for implying new facts. The complete list of 16 sentences and extracted triples from baseline TPlinker and proposed TPlinker+EN+CN models are shown Tables 10 and 11.

[2]from CopyRE (Zeng et al. (2018)) Github repository https://github.com/xiangrongzeng/copy_re

Table 10: Real true unseen sample sentences from Wikipedia.

ID	Sentences
1	Nissan shareholders voted to remove Carlos Ghosn from the company board. Shareholders also voted to remove Carlos Ghosn's former right-hand man Greg Kelly, and to appoint Renault chairman Jean-Dominique Senard as a director.
2	Morgan Stanley announced that Zoe Cruz was resigning as co-president of the firm and that she would retire immediately. Following Morgan Stanley, Cruz was on the Board of Trustees for the Harlem Children 's Zone.
3	Jeff Zucker worked with fellow NBC News alum, former Today host Katie Couric, producing her daytime talk show for Disney-ABC Domestic Television, Katie. However, Jeff Zucker left the show to be the president of CNN Worldwide.
4	In June 2018, Disney announced that John Lasseter was leaving the company at the end of the year. On January 9, 2019, John Lasseter was hired to head Skydance Animation.
5	As of January 1, 2012, George Bodenheimer was the executive chairman of ESPN, with John Skipper replacing him as president.
6	Sony announced that Howard Stringer would step down as president and CEO, effective 1 April to be replaced by Kazuo Hirai.
7	Edward R. Murrow resigned from CBS to accept a position as head of the United States Information Agency, parent of the Voice of America, in January 1961.
8	The Green Bay Packers traded Brett Favre to the New York Jets on August 7, 2008, in exchange for a conditional fourth-round pick in the 2009 NFL Draft with performance escalation.
9	Robert S. Miller left Delphi in October 2009 . American International Group named Robert S. Miller as their chairman in July 2010 .
10	In addition, Eric Ripert partnered with The Ritz-Carlton Hotel Company to open Blue in Grand Cayman.
11	On September 6, 2010, Mark V. Hurd was named president of Oracle Corporation alongside Safra A. Catz, succeeding former president Charles Phillips. Mark V.
12	Dan Glickman left the Motion Picture Association of America in 2010 to serve as president of Refugees International.
13	In January 2018, Kenneth I. Chenault announced he would become chairman and managing director of General Catalyst Partners and joined the board of directors of Airbnb.
14	Robert B. Willumstad left Citigroup in July 2005, saying that he wanted to run a major company, after CEO Charles Prince decided to take back control of operations.
15	Since Peter Chernin departure from News Corporation. in 2009, Peter Chernin has been the chairman of his own company, The Chernin Group (TCG).
16	In the spring of 2001, Marc Jacobs introduced his secondary line, Marc by Marc Jacobs.

Table 11: Detailed results on real true unseen samples. Relation "company" are omitted for triples. We colored correctly predicted triples in blue.

ID	GOLD	Trained Triples (biased, no longer true)	TPlinker(baseline)	TPlinker+EN+CN (Proposed)
1	(Jean-Dominique Senard, Renault)	(Carlos Ghosn, Renault)	(Carlos Ghosn, Renault)	(Jean-Dominique Senard, Renault)
2	(Zoe Cruz, Harlem Children's Zone)	(Zoe Cruz, Morgan Stanley)	(Zoe Cruz, Morgan Stanley)	(Zoe Cruz, Morgan Stanley)
3	(Jeff Zucker, CNN)	(Jeff Zucker, NBC News)	(Katie Couric, NBC News) (Jeff Zucker, NBC News)	(Katie Couric, NBC News)
4	(John Lasseter, Skydance Animation)	(John Lasseter, Pixar)	{}	(John Lasseter, Skydance Animation)
5	(John Skipper, ESPN) (George Bodenheimer, ESPN)	(George Bodenheimer, ESPN)	(George Bodenheimer, ESPN)	(John Skipper, ESPN) (George Bodenheimer, ESPN)
6	(Kazuo Hirai, Sony)	(Howard Stringer, Sony)	(Howard Stringer, Sony)	(Kazuo Hirai, Sony) (Howard Stringer, Sony)
7	(Edward R. Murrow, United ... Agency)	(Edward R. Murrow, CBS)	(Edward R. Murrow, United ... Agency) (Edward R. Murrow, CBS)	(Edward R. Murrow, United ... Agency)
8	(Brett Favre, New York Jets)	(Brett Favre, Green Bay Packers)	(Brett Favre, Green Bay Packers)	(Brett Favre, Green Bay Packers)
9	(Robert S. Miller, American ...Group)	(Robert S. Miller, Delphi)	(Robert S. Miller, American ...Group) (Robert S. Miller, Delphi)	(Robert S. Miller, American ...Group) (Robert S. Miller, Delphi)
10	(Eric Ripert, Blue)	(Eric Ripert, Le Bernardin)	{}	(Eric Ripert, Blue)
11	(Mark V. Hurd, Oracle Corporation) (Safra A. Catz, Oracle Corporation)	(Mark V. Hurd, Hewlett-Packard)	(Mark V. Hurd, Oracle Corporation)	(Mark V. Hurd, Oracle Corporation) (Safra A. Catz, Oracle Corporation)
12	(Dan Glickman, Refugees International)	(Dan Glickman, Motion ... of America)	(Dan Glickman, Motion ... America)	(Dan Glickman, Motion ... America)
13	(Kenneth I. Chenault, General ... Partners) (Kenneth I. Chenault, Airbnb)	(Kenneth I. Chenault, American Express)	(Kenneth I. Chenault, General ...Partners)	(Kenneth I. Chenault, General ...Partners) (Kenneth I. Chenault, Airbnb)
14	(Charles Prince, Citigroup)	(Robert B. Willumstad, Citigroup)	(Robert B. Willumstad, Citigroup)	(Robert B. Willumstad, Citigroup)
15	(Peter Chernin, The Chernin Group)	(Peter Chernin, News Corporation)	(Peter Chernin, News Corporation)	(Peter Chernin, The Chemin Group) (Peter Chernin, News Corporation)
16	(Marc Jacobs, Marc by Marc Jacobs)	(Marc Jacobs, Louis Vuitton)	(Marc Jacobs, Children, Marc Jacobs)	{}