PARE: A Simple and Strong Baseline for Monolingual and Multilingual Distantly Supervised Relation Extraction

Anonymous ACL submission

Abstract

Neural models for distantly supervised relation extraction (DS-RE) encode each sentence in an entity-pair bag separately. These are then aggregated for bag-level relation prediction. Since, at encoding time, these approaches do not allow information to flow from other sentences in the bag, we believe that they do not utilize the available bag data to the fullest. In response, we explore a simple baseline approach (*PARE*) in which all sentences of a bag are concatenated into a *passage* of sentences, and encoded jointly using BERT. The contextual embeddings of tokens are aggregated using attention with the candidate relation as query – this summary of whole passage predicts the candidate relation. We find that our simple baseline solution outperforms existing state-of-the-art DS-RE models in both monolingual and multilingual DS-RE datasets.

1 Introduction

Given some text (typically, a sentence) t mentioning an entity pair (e_1, e_2), the goal of relation extraction (RE) is to predict the relationships between e_1 and e_2 that can be inferred from t. Let $B(e_1, e_2)$ denote the set of all sentences (bag) in the corpus mentioning e_1 and e_2 and let $R(e_1, e_2)$ denote all relations from e_1 to e_2 in a KB. Distant supervision (DS) trains RE models given $B(e_1, e_2)$ and $R(e_1, e_2)$, without sentence level annotation (Mintz et al., 2009). Most DS-RE models use the "at-least one" assumption: $\forall r \in R(e_1, e_2)$, $\exists t^r \in B(e_1, e_2)$ such that t^r expresses (e_1, r, e_2).

Recent neural approaches to DS-RE encode each sentence $t \in B(e_1, e_2)$ and then aggregate sentence embeddings using an aggregation operator – the common operator being intra-bag attention (Lin et al., 2016). Various models differ in their approach to encoding (e.g., PCNNs, GCNs, BERT) and their loss functions (e.g., contrastive learning, MLM), but agree on the design choice of encoding each sentence independently of the others (Vashishth et al., 2018; Alt et al., 2019; Christou and Tsoumakas, 2021; Chen et al., 2021). We posit that this choice leads to a suboptimal usage of the available data – information from other sentences might help in better encoding a given sentence.

We explore this hypothesis by developing a simple baseline solution. We first construct a *passage* $P(e_1, e_2)$ by concatenating all sentences in $B(e_1, e_2)$. We then encode the whole passage through BERT (Devlin et al., 2019) (or mBERT for multilingual setting). This produces a contextualized embedding of every token in the bag. To make these embeddings aware of the candidate relation, we take a (trained) relation query vector, $\mathbf{r}$, to generate a relation-aware summary of the whole passage using attention. This is then used to predict whether (e_1, r, e_2) is a valid prediction.

Despite its simplicity, our baseline has some conceptual advantages. First, each token is able to exchange information with other tokens from other sentences in the bag – so the embeddings are likely more informed. Second, in principle, the model may be able to relax a part of the at-least-one assumption. For example, if no sentence individually expresses a relation, but if multiple facts in different sentences collectively predict the relation, our model may be able to learn to extract that.

We name our baseline model Passage-Attended Relation Extraction, *PARE* (*mPARE* for multilingual DS-RE). We experiment on four DS-RE datasets – three in English, NYT-10d (Riedel et al., 2010), NYT-10m, and Wiki-20m (Gao et al., 2021), and one multilingual, DiS-ReX (Bhartiya et al., 2021). We find that in all four datasets, our proposed baseline significantly outperforms existing state of the art, yielding up to 5 point AUC gain. Further attention analysis and ablations provide additional insight into model performance. We release our code for reproducibility. We believe that our work represents a simple but strong baseline that can form the basis for further DS-RE research.

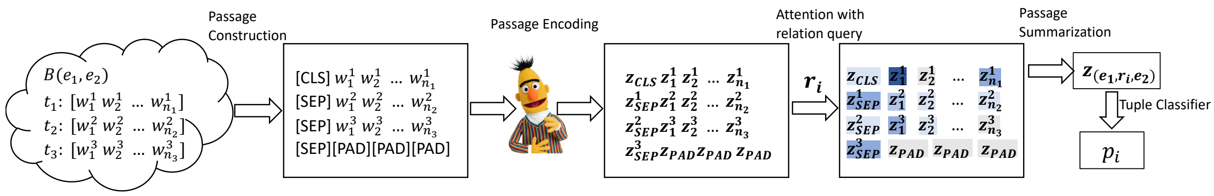


Figure 1: Model architecture for *PARE*. Entity markers not shown for brevity.

2 Related Work

Monolingual DS-RE: Early works in DS-RE build probabilistic graphical models for the task (e.g., (Hoffmann et al., 2011; Ritter et al., 2013). Most later works follow the multi-instance multi-label learning framework (Surdeanu et al., 2012) in which there are multiple labels associated with a bag, and the model is trained with at-least-one assumption. Most neural models for the task encode each sentence separately, e.g., using Piecewise CNN (Zeng et al., 2015), Graph Convolution Net (e.g., *RESIDE* (Vashishth et al., 2018)), GPT (*DISTRE* (Alt et al., 2019)) and BERT (*REDSandT* (Christou and Tsoumakas, 2021), *CIL* (Chen et al., 2021)). They all aggregate embeddings using intra-bag attention (Lin et al., 2016). Beyond Binary Cross Entropy, additional loss terms include masked language model pre-training (*DISTRE*, *CIL*), RL loss (Qin et al., 2018), and auxiliary contrastive learning (*CIL*). We show that *PARE* is competitive with *DISTRE*, *RESIDE*, *CIL*, and other natural baselines, without using additional pre-training, side information or auxiliary losses during training, unlike some comparison models.

To evaluate DS-RE, at test time, the model makes a prediction for an unseen bag. Unfortunately, most popular DS-RE dataset (NYT-10d) has a noisy test set, as it is automatically annotated (Riedel et al., 2010). Recently Gao et al. (2021) has released NYT-10m and Wiki-20m, which have manually annotated test sets. We use all three datasets in our work.

Multilingual DS-RE: A bilingual DS-RE model named MNRE (tested on English and Mandarin) introduced cross-lingual attention in language-specific CNN encoders (Lin et al., 2017). Recently, Bhartiya et al. (2021) has released a dataset, DiS-ReX, for four languages – English, Spanish, German and French. We compare *mPARE* against the state of the art on DiS-ReX, which combines MNRE architecture with mBERT encoder. See Appendix E for details on all DS-RE models.

3 Passage Attended Relation Extraction

PARE explores the value of cross-sentence attention during encoding time. It uses a sequence of three key steps: passage construction, encoding and summarization, followed by prediction. Figure 1 illustrates these for a three-sentence bag.

Passage Construction constructs a *passage* $P(e_1, e_2)$ from sentences $t \in B(e_1, e_2)$. The construction process uses a sequential sampling of sentences in the bag without replacement. It terminates if (a) adding any new sentence would exceed the maximum number of tokens allowed by the encoder (512 tokens for BERT), or (b) all sentences from the bag have been sampled.

Passage Encoding takes the constructed passage and sends it to an encoder (BERT or mBERT) to generate contextualized embeddings $\mathbf{z}_j$ of every token w_j in the passage. For this, it first creates an encoder input. The input starts with the [CLS] token, followed by each passage sentence separated by [SEP], and pads all remaining tokens with [PAD]. Moreover, following best-practices in RE (Han et al., 2019), each mention of e_1 and e_2 in the passage are surrounded by special entity marker tokens <e1>,</e1>, and <e2>,</e2>, respectively.

Passage Summarization maintains a (randomly-initialized) query vector $\mathbf{r}_i$ for every relation r_i. It then computes α_j^i, the normalized attention of r_i on each token w_j, using dot-product attention. Finally, it computes a relation-attended summary of the whole passage $\mathbf{z}_{(e_1,r_i,e_2)} = \sum_{j=1}^{j=L} \alpha_j^i \mathbf{z}_j$, where L is the input length. We note that this summation also aggregates embeddings of [CLS], [SEP], [PAD], as well as entity marker tokens.

Tuple Classifier passes $\mathbf{z}_{(e_1,r_i,e_2)}$ through an MLP followed by Sigmoid activation to return the probability p_i of the triple (e_1, r_i, e_2). This MLP is shared across all relation classes. At inference, a positive prediction is made if $p_i >$ threshold (0.5).

Loss Function is simply Binary Cross Entropy between gold and predicted label set for each bag. No additional loss terms are used.

Model	AUC	P@M
PCNN-Att	34.1	69.4
RESIDE	41.5	77.2
DISTRE	42.2	66.8
REDSandT	42.4	75.3
CIL	50.8	86.0
PARE	51.8	89.0

Model	NYT-10m		Wiki-20m	
	AUC	M-F1	AUC	M-F1
B+Att	51.2	25.8	70.9	64.3
B+Avg	56.7	35.7	89.9	82.0
B+One	58.1	33.9	88.9	81.1
CIL	59.4	36.3	89.7	82.6
PARE	61.2	37.3	90.9	83.4

Model	AUC	μF1	M-F1
PCNN+Att	67.8	63.4	43.7
mB+Att	80.6	74.1	69.9
mB+One	80.9	74.0	68.9
mB+Avg	82.4	75.3	71.0
mB+MNRE	82.1	76.1	72.7
mPARE	86.9	79.4	76.3

Table 1: Results on (a) NYT-10d, (b) NYT-10m & Wiki-20m, and (c) DiS-ReX. B=BERT and mB=mBERT. *PARE* and *mPARE* outperforms all models by statistically significant margins (McNemar's test): all p values $< 10^{-5}$.

4 Experiments and Analysis

We compare *PARE* and *mPARE* against the state of the art models on the respective datasets. We also perform ablations and analyses to understand model behavior and reasons for its performance.

Datasets and Evaluation Metrics: We evaluate *PARE* on three English datasets: NYT-10d, NYT-10m, Wiki-20m. *mPARE* is compared using the DiS-ReX benchmark. Data statistics are in Table 2, with more details in Appendix C. We use the evaluation metrics prevalent in literature for each dataset. These include AUC: area under the precision-recall curve, M-F1: macro-F1, μ-F1: micro-F1, and $P@M$: average of P@100, P@200 and P@300, where P@k denotes precision calculated over a model's k most confidently predicted triples.

Comparison Models and Hyperparameters: Since there is substantial body of work on NYT-10d, we compare against several recent models: *RESIDE*, *DISTRE*, *REDSandT* and the latest state of the art, *CIL*. For NYT-10m and Wiki-20m, we report comparisons against models in the original paper (Gao et al., 2021), and also additionally run CIL for a stronger comparison. For DiS-ReX, we compare against mBERT based models. See Appendix E for details. For *PARE* and *mPARE*, we use base-uncased checkpoints for BERT and mBERT, respectively. Hyperparameters are set based on a simple grid search over devsets. (see Appendix A).

Dataset	#Rels	#Total	#Test	Test set
NYT-10d	58	694k	172k	Distant Sup.
NYT-10m	25	474k	9.74k	Manual
Wiki-20m	81	901k	140k	Manual
DiS-ReX	37	1.84M	334k	Distant Sup.

Table 2: Dataset statistics.

4.1 Comparisons against State of the Art

The results are presented in Table 1, in which, the best numbers are highlighted and second best numbers are underlined. On NYT-10d (Table 1(a)), *PARE* has 1 pt AUC improvement and 3 pts P@M gains over *CIL*, the current state of the art. This is also reflected in the P-R curve (Figure 2), where *PARE* is the only model which is able to achieve near 100% precision for very high threshold values. Our model beats *REDSandT* by 9 AUC pts, even though both use BERT, and latter uses extra side-information (e.g., entity-type, sub-tree parse).

On manually annotated testsets (Table 1(b)), *PARE* achieves up to 1.8 pt AUC and 1 pt F1 gains against *CIL*. We note that Gao et al. (2021) only published numbers on simpler baselines (BERT followed by attention, average and max aggregators), which are substantially outperformed by *PARE*. *CIL*'s better performance is likely attributed to its contrastive learning objective – it will be interesting to study this in the context of *PARE*.

For multilingual DS-RE (Table 1(c)), *mPARE*

Figure 2: PR Curve for Models on NYT-10d

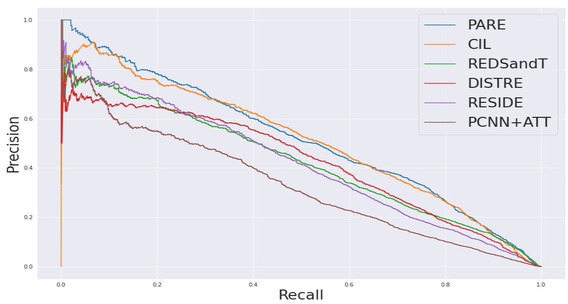
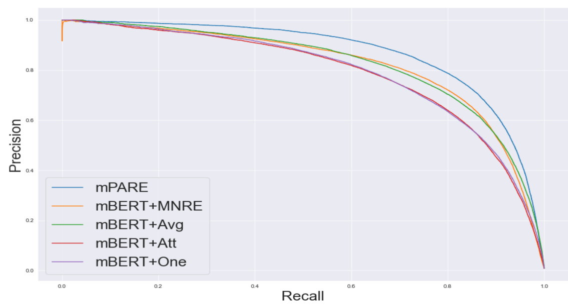


Figure 3: PR Curve for Models on DiS-ReX



obtains a 4.8 pt AUC gain against mBERT+MNRE. P-R curve in Figure 3 shows that it convincingly outperforms others across the entire domain of recall values. We provide language-wise and relation-wise metrics in Appendix L – the gains are consistent on all languages and nearly all relations.

4.2 Analysis and Ablations

Generalizing to Unseen KB: Recently, Ribeiro et al. (2020) has proposed a robustness study in which entity names in a bag are replaced by other names (from the same type) to test whether the extractor is indeed reading the text, or is simply overfitting on the regularities of the given KB. We also implement a similar robustness study (details in Appendix K), where entity replacement results in an entity-pair bag that does not exist in the original KB. We find that on this modified NYT-10m, all models suffer a drop in performance, suggesting that models are not as robust as we intend them to be. We, however, note that *CIL* suffers a 28.1% drop in AUC performance, but *PARE* remains more robust with only a 16.8% drop. We hypothesize that this may be because of *PARE*'s design choice of attending on all words for a given relation, which could reduce its focus on entity names themselves.

Scaling with Size of Entity-Pair Bags: Due to truncation when the number of tokens in a bag exceed 512 (limit for BERT), one would assume that the *PARE* may not be suited for cases where the number of tokens in a bag is large. To study this, we divide the test set of NYT-10m into 7 different groups based on the number of tokens present in the untruncated passage (Appendix J). We find that *PARE* shows consistent 2 to 5 pt AUC gains against *CIL* for all groups except the smallest group. This is not surprising, since for smallest group, there is likely only one sentence in a bag, and *PARE* would not gain from inter-sentence attention. For large bags, relevant information is likely already present in truncated passage, due to redundancy.

Attention Patterns: In *PARE*, each relation class has a trainable query vector, which attends on every token. The attention scores could give us some insight about the words the model is focusing on. We observe that for a candidate relation that is not a gold label for a particular bag, surprisingly, the highest attention scores are obtained by [PAD] tokens. In fact, for such bags, on an average, roughly 90% of the attention weight goes to [PAD] tokens, whereas this number is only 0.1% when the relation is in the gold set (see Appendices H and I). We find this to be an example of model ingenuity – *PARE* seems to have creatively learned that whenever the most appropriate words for a relation are not present, it could simply attend on [PAD] embeddings, which may lead to similar attended summaries, which may be easily decoded to a low probability of tuple validity. In fact, as a further test, we perform an ablation where we disallow relation query vectors to attend on [PAD] tokens – this results in an over 3 pt drop in AUC on NYT-10d, indicating the importance of padding for prediction.

Ablations: We perform further ablations of the model by removing [SEP] tokens, entity markers and removing the relation-attention step that computes a summary (instead using [CLS] token for predicting each relation). *PARE* loses significantly in performance in each ablation obtaining 49.4, 14.9 and 46.1 AUC, respectively (as against 51.8 for full model) on NYT-10d. The critical importance of entity markers is not surprising, since without them the model does not know what is the entity-pair it is predicting for. We also notice a very significant gain due to relation attention, suggesting that this is an important step for the model – it allows focus on specific words relevant for predicting a relation. More details on this experiment in Appendix G.

Effect of Sentence Order: We build 20 random passages per bag (by varying sentence order and also which sentences get selected if passage needs truncation). On all four datasets (Appendix M), we find that the standard deviation to be negligible.

5 Conclusion

We introduce *PARE*, a simple baseline for the task of distantly supervised relation extraction. It converts a bag of sentences containing an entity-pair into a passage. Contextual embeddings from encoding the passage can potentially benefit from attention across words from different sentences. It then creates an relation-attended summary of all contextual embeddings, which is decoded for tuple validity. Our experiments demonstrate that this simple baseline produces very strong results for the task, and outperforms existing top models by varying margins across four datasets in monolingual and multilingual settings. Several experiments for studying model behavior show its consistent performance across settings. We posit that our framework would serve as a strong backbone for further research in the field of DS-RE.

References

Christoph Alt, Marc Hübner, and Leonhard Hennig. 2019. Fine-tuning pre-trained transformer language models to distantly supervised relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1388–1398, Florence, Italy. Association for Computational Linguistics.

Abhyuday Bhartiya, Kartikeya Badola, and Mausam. 2021. Dis-rex: A multilingual dataset for distantly supervised relation extraction.

Tao Chen, Haizhou Shi, Siliang Tang, Zhigang Chen, Fei Wu, and Yueting Zhuang. 2021. CIL: Contrastive instance learning framework for distantly supervised relation 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 6191–6200, Online. Association for Computational Linguistics.

Despina Christou and Grigorios Tsoumakas. 2021. Improving distantly-supervised relation extraction through bert-based label and instance embeddings. *IEEE Access*, 9:62574–62582.

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.

Tianyu Gao, Xu Han, Keyue Qiu, Yuzhuo Bai, Zhiyu Xie, Yankai Lin, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2021. Manual evaluation matters: Reviewing test protocols of distantly supervised relation extraction. *arXiv preprint arXiv:2105.09543*.

Xu Han, Tianyu Gao, Yankai Lin, Hao Peng, Yaoliang Yang, Chaojun Xiao, Zhiyuan Liu, Peng Li, Jie Zhou, and Maosong Sun. 2020. More data, more relations, more context and more openness: A review and outlook for relation extraction. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 745–758, Suzhou, China. Association for Computational Linguistics.

Xu Han, Tianyu Gao, Yuan Yao, Deming Ye, Zhiyuan Liu, and Maosong Sun. 2019. OpenNRE: An open and extensible toolkit for neural relation extraction. In *Proceedings of EMNLP-IJCNLP: System Demonstrations*, pages 169–174.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 541–550, Portland, Oregon, USA. Association for Computational Linguistics.

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

Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2017. Neural relation extraction with multi-lingual attention. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 34–43, Vancouver, Canada. Association for Computational Linguistics.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133, Berlin, Germany. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

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

Jian Ni, Taesun Moon, Parul Awasthy, and Radu Florian. 2020. Cross-lingual relation extraction with transformers. *arXiv preprint arXiv:2010.08652*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Robust distant supervision relation extraction via deep reinforcement learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne,* *Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2137–2147. Association for Computational Linguistics.

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

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912.

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.

Alan Ritter, Luke Zettlemoyer, Mausam, and Oren Etzioni. 2013. Modeling missing data in distant supervision for information extraction. *Trans. Assoc. Comput. Linguistics*, 1:367–378.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465, Jeju Island, Korea. Association for Computational Linguistics.

Shikhar Vashishth, Rishabh Joshi, Sai Suman Prayaga, Chiranjib Bhattacharyya, and Partha Talukdar. 2018. RESIDE: Improving distantly-supervised neural relation extraction using side information. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1257–1266, Brussels, Belgium. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1753–1762.

A Experimental Settings

We train and test our model on two NVIDIA GeForce GTX 1080 Ti cards. We use a linear LR scheduler having weight decay of 1e-5 with AdamW (Loshchilov and Hutter, 2017; Kingma and Ba, 2014) as the optimizer. Our implementation uses PyTorch (Paszke et al., 2019), the Transformers library (Wolf et al., 2019) and OpenNRE [1] (Han et al., 2019). We use bert-base-uncased checkpoint for BERT initialization in the mono-lingual setting. For multi-lingual setting, we use bert-base-multilingual-uncased.

For hyperparameter tuning, we perform grid search over {1e-5, 2e-5} for learning rate and {16, 32, 64} for batch size and select the best performing configuration for each dataset.

PARE takes 2 epochs to converge on NYT-10d (152 mins/epoch), 3 epochs for NYT-10m (138 mins/epoch), 2 epochs for Wiki-20m (166 mins/epoch) and 4 epochs for DiS-ReX (220 mins/epoch).

The numbers we report for the baselines come from their respective papers. We obtained the code base of CIL, BERT+Att, BERT+Avg, BERT+One from their respective authors, so that we could run them on additional datasets. We were able to replicate same numbers as reported in their papers. We trained those models on other datasets as well by carefully tuning the bag size hyperparameter.

B Sizes of different models

We calculate no. of trainable parameters for each model. For fair comparison, we exclude the parameters of BERT encoder while reporting these numbers for every model. We stress that the number of parameters in BERT is same for all models including us i.e. 109482240 because all use the same bert-base-uncased checkpoint.

We note that the key reason why other models have significantly higher parameters is because they use *entity pooling* for constructing instance representations. Here, the encoded representations of tokens corresponding to the span of head and tail entity mentions are pooled together, followed by concatenation and passing through a linear layer of size $2D \times 2D$ (where D is the dimension of the encoded token). For BERT, $D = 768$ due to which the size of the linear layer is = 1536*1536 = 2359296. This causes the huge difference in number of parameters between our model and their models. However, it should be noted that the entity pooling generally performs better than using [CLS] embedding for encoding instances as has been shown in recent works (Ni et al., 2020).

Model	#Parameters (excluding BERT)
Att	2400793
One	2399257
Avg	2399257
CIL	2453052
PARE	45313

Table 3: Comparison of trainable parameters between our model and other state-of-the-art models

C Dataset Details

We evaluate our proposed model on four different datasets: NYT-10d (Riedel et al., 2010), NYT-10m (Gao et al., 2021), Wiki-20m (Gao et al., 2021) and DiS-ReX (Bhartiya et al., 2021). The statistics for each of the datasets is present in table 2.

NYT-10d

NYT-10d is the most popular dataset for monolingual DS-RE, constructed by aligning Freebase entities to the New York Times Corpus. The train and test splits are both distantly supervised.

NYT-10m

[1] https://github.com/thunlp/OpenNRE

NYT-10m is a recently released dataset to train and evaluate models for monolingual DS-RE. The dataset is built from the same New York Times Corpus and the Freebase KB but with a new relation ontology and a manually annotated test set. It aims to tackle the existing problems with the NYT-10d dataset by 1) establishing a public validation set 2) establishing consistency among the relation classes present in the train and test set 3) providing a high quality, manually labeled test set.

Wiki-20m

Wiki-20m is also a recently released dataset for training DS-RE models and evaluating them on manually annotated a test set. The test set in this case corresponds to the Wiki80 dataset (Han et al., 2019). The relation ontology of Wiki80 is used to re-structure the Wiki20 DS-RE dataset (Han et al., 2020), from which the training and validation splits are created. It is made sure that their is no overlap between the instances present in the testing and the training and validation sets.

DiS-ReX

DiS-ReX is a recently released benchmarking dataset for training and evaluating DS-RE models on instances spanning multiple languages. The entities present in this dataset are linked across the different languages which means that a bag can contain sentences from more than one languages. The training, validation and testing sets present in the dataset are constructed in a way that there is no head and tail entity pair overlap between the bags present in any two different sets.

We obtain the first three datasets from OpenNRE and DiS-ReX from their official repository.

D Description of Intra-Bag attention

Let $t_1, t_2, ..., t_n$ denote n instances sampled from $B(e_1, e_2)$. In all models using intra-bag attention for instance-aggregation, each t_i is independently encoded to form the instance representation, $E(t_i)$, following which the relation triple representation B_r for the triple (e_1, e_2, r) is given by $B_r = \sum_{i=0}^{i=n} \alpha_i^r E(t_i)$. Here r is any one of the relation classes present in the dataset and α_i^r is the normalized attention score allotted to instance representation $E(t_i)$ by relation query vector $\overrightarrow{r}$ for relation r. The model then predicts whether the relation triple is a valid one by sending each B_r through a feed-forward neural network. In some variants, $\overrightarrow{r}$ is replaced with a shared query vector for all relation-classes, $\overrightarrow{q}$, resulting in a bag-representation B corresponding to (e_1, e_2) as opposed to triple-representation.

E Baselines

The details for each baseline is provided below:

PCNN-Att

Lin et al. (2016) proposed the intra-bag attention aggregation scheme in 2016, obtaining the then state-of-the-art performance on NYT-10d using a piecewise convolutional neural network (PCNN (Zeng et al., 2015)).

RESIDE

Vashishth et al. (2018) proposed RESIDE which uses side-information (in the form of entity types and relational aliases) in addition to sentences present in the dataset. The model uses intra-bag attention with a shared query vector to combine the representations of each instance in the bag. The sentence representations are obtained using a Graph Convolutional Network (GCN) encoder.

DISTRE

Alt et al. (2019) propose the use of a pre-trained transformer based language model (OpenAI GPT Radford et al. (2018)) for the task of DS-RE. The model uses intra-bag attention for the instance aggregation step.

REDSandT

Christou and Tsoumakas (2021) propose the use of a BERT encoder for DS-RE by using sub-tree parse of the input sentence along with special entity type markers for the entity mentions in the text. The model uses intra-bag attention for the instance aggregation step.

CIL
Chen et al. (2021) propose the use of Masked Language Modeling (MLM) and Contrastive Learning (CL) losses as auxilliary losses to train a BERT encoder + Intra-bag attention aggregator for the task.

BERT+Att/mBERT+Att
The model uses intra-bag attention aggregator on top of a BERT/mBERT encoder.

BERT+Avg/mBERT+Avg
The model uses "Average" aggregator which weighs each instance representation uniformly, hence denoting bag-representation as the average of instance-representations.

BERT+One/mBERT+One
The model independently performs multi-label classification on each instance present in the bag and then aggregates the classification results by performing class-wise max-pooling (over sentence scores). In essence, the "One" aggregator ends up picking one instance for each class (the one which denotes the highest confidence for that particular class), hence the name.

mBERT+MNRE
The MNRE aggregator was originally introduced by Lin et al. (2017) and used with a shared mBERT encoder by Bhartiya et al. (2021) [2]. The model assigns a query vector for each $(relation, language)$ tuple. A bag is divided into sub-bags where each sub-bag contains the instances of the same language. In essence, a bag has L sub-bags and each relation class corresponds to L query vectors, where L denotes the number of languages present in the dataset. These are then used to construct L^2 triple representations (using intra-bag attention aggregation on each *(sub-bag,query vector)* pair for a candidate relation) which are then scored independently. The final confidence score for a triple is the average of L^2 triple scores.

F Statistical Significance

We compare the predictions of our model on the non-NA triples present in the test set with the predictions of the second-best model using the McNemar's test of statistical significance (McNemar, 1947). In all cases, we obtained the *p-value* to be many orders of magnitude smaller than 0.05, suggesting that the improvement in results is statistically significant in all cases.

G Ablation on NYT-10d

Modification	**AUC**
Ours	51.8
w/ max length = 256	48.1
w/ max length = 32	31.2
w/o passage summarization	46.1
w/o [PAD] attention	48.3
w/o [SEP] tokens	49.4
w/o entity markers	14.9

Table 4: Model ablation on NYT-10d

[2]Obtained from the original repository for DiS-ReX

We perform ablation studies on the NYT-10d dataset to understand which components are most beneficial for our proposed model. We provide the results in table 4.

We observe that the performance increases with increase in maximum allowed length of the passage. This result is expected since the model would be exposed to more information for a given entity pair, allowing it to make more confident predictions for the validity of a particular candidate relation.

Upon replacing our passage summarization step with multi-label classification using [CLS] token (present at the start of the passage), we observe a significant decrease in AUC, indicating that contextual embedding of [CLS] token might not contain enough information for multi-label prediction of bag.

It is interesting to note here that the AUC is still higher than that of REDSandT, a model which uses BERT+Att as the backbone. This means that one can simply obtain an improvement in performance by creating a passage from multiple instances in a bag.

Removing entity markers resulted in the most significant drop in performance. However, this is also expected since without them, our model would have no way to understand which entities to consider while performing relation extraction.

H Attention on [PAD] tokens

In the passage summarization step (described in section 3), we allow the relation query vector $\overrightarrow{r}$ to also attend over the encodings of the [PAD] tokens present in the passage. We make this architectural choice in-order to provide some structure to the relation-specific summaries created by our model. If a particular relation class r is not a valid relation for entity pair (e_1, e_2), then ideally, we would want the attended-summary of the passage $P(e_1, e_2)$ created by the relation vector $\overrightarrow{r}$ to represent some sort of a null state (since information specific to that relation class is not present in the passage). Allowing [PAD] tokens to be a part of the attention would provide enough flexibility to the model to represent such a state. We test our hypothesis by considering 1000 non-NA bags correctly labelled by our trained model in the test set of NYT-10d. Let $R(e_1, e_2)$ denote the set of valid relation-classes for entity pair (e_1, e_2) and let R denote all of the relation-classes present in the dataset. We first calculate the percentage of attention given to [PAD] tokens for a given passage $P(e_1, e_2)$ for all relation-classes in R. The results are condensed into two scores, sum of scores for $R(e_1, e_2)$ and sum of scores for $R \setminus R(e_1, e_2)$. The results are aggregated for all 1000 bags, and then averaged out by dividing with the total number of positive triples and negative triples respectively. We obtain that on an average, only 0.07% of attention weight is given to [PAD] tokens by relation vectors corresponding to $R(e_1, e_2)$, compared to 88.35% attention weight given by relation vectors corresponding to $R \setminus R(e_1, e_2)$. We obtain similar statistics on other datasets as well. This suggests that for invalid triples, passage summaries generated by the model resemble the embeddings of the [PAD] token. Furthermore, since we don't allow [PAD] tokens to be a part of self-attention update inside BERT, the [PAD] embeddings at the output of the BERT encoder are not dependent on the passage, allowing for uniformity across all bags.

Finally, we train a model where we don't allow the relation query vectors to attend on the [PAD] token embeddings and notice a 3.5pt drop in AUC on NYT-10d (table 4). We also note that the performance is still significantly higher than models such as REDSandT and DISTRE, suggesting that our instance aggregation scheme still performs better than the baselines, even when not optimized fully.

I Examples of Attention Weighting during Passage Summarization

To understand how the query vector of a relation attends over passage tokens to correctly predict that relation, we randomly selected from correctly predicted non-NA triples and selected the token obtaining the highest attention score (by the query vector for the correct relation). For the selection, we ignore the stop words, special tokens and the entity mentions. The results are presented in table 5.

J Performance vs Length of test passages

Our instance aggregation scheme truncates the passage if the number of tokens exceed the maximum number of tokens allowed by the encoder. In such cases, one would assume that the our model is not suited for cases where the number of instances present in a bag is very large. To test this hypothesis,

Input Passage (tokenized by BERT)	correctly predicted label
[CLS] six months later , his widow met the multi ##mill ##ion ##aire [unused2] vincent astor [unused3] , a **descendant** of the fur trader turned manhattan real - estate magnate [unused0] john jacob astor [unused1] , and a man considered so unpleasant by his peers 1 ##rb and even by his own mother rr ##b - that he reportedly required a solitary seating for lunch at his club because nobody would share a meal with him . [SEP]	/people/person/children
[CLS] the [unused2] robin hood foundation [unused3] , **founded** by [unused0] paul tudor jones [unused1] ii and perhaps the best - known hedge fund charity , raised $ 48 million at its annual benefit dinner last year . [SEP]	/business/person/company
[CLS] she is now back in the fourth round , where she will face 11th - seeded je ##lena jan ##kovic of serbia , a 6 - 3 , 6 - 4 winner over [unused0] victoria az ##are ##nka [unused1] **of** [unused2] belarus [unused3] . [SEP]	/people/person/nationality
[CLS] [unused2] boston [unused3] what : a two - bedroom condo how much : $ 59 ##9 , 000 per square foot : $ 83 ##6 located in the [unused0] back bay [unused1] area of the city , this 71 ##6 - square - foot condo has views from the apartment and its private roof deck of the charles river , one block away . [SEP] seven years ago , when nad ##er tehran ##i and monica ponce de leon , partners at office da , an architecture firm in [unused2] boston [unused3] , were asked to reno ##vate a five - story town house in the [unused0] back bay [unused1] **neighborhood** , they faced a singular design challenge . [SEP] far more inviting is first church in [unused2] boston [unused3] , in [unused0] back bay [unused1] , which replaced a gothic building that burned in 1968 . [SEP]	/location/neighborhood/neighborhood_of
[CLS] [unused2] steve new ##comb [unused3] , a [unused0] powers ##et [unused1] **founder** and veteran of several successful start - ups , said his company could become the next google . [SEP]	/business/company/founders
[CLS] [unused0] michael sm ##uin [unused1] , a choreographer who worked for major ballet companies and led his own , marshal ##ing eclectic dance forms , robust athletic ##ism and striking theatrical ##ity to create works that appealed to broad audiences , **died** yesterday in [unused2] san francisco [unused3] . [SEP]	/people/deceasedperson/place_of_death

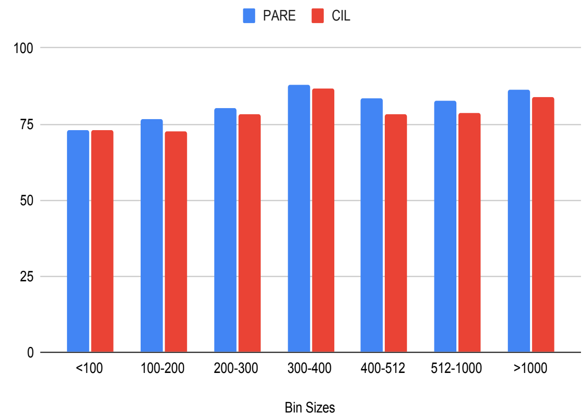
Table 5: Attention analysis on a few random correctly predicted non-NA triples on NYT-10m test set. The highest attention-scored token (excluding entity mentions and special markers and stop words) are present in bold. [unused0], [unused1] denote the start and end head entity markers. [unused2], [unused3] denote the start and end tail entity markers.

we divide the non-NA bags, (e_1, e_2), present in the NYT-10m data into 7 bins based on the number of tokens present in $P(e_1, e_2)$ (after tokenized using BERT). We then compare the performance with CIL on examples present in each bin. The results in figure 4 indicate that a) our model beats CIL in each bin-size b) the variation among different bins is the same for both models. This trend is continued even for passages where the number of tokens present exceed the maximum number of tokens allowed for BERT (i.e. 512). This results indicate that 512 tokens provide sufficient information for correct classification of a triple. Moreover, models using intra-bag attention aggregation scheme fix the number of instances sampled from the bag in practice. For CIL, the best performing configuration uses a bag-size of 3. This analysis therefore indicates that our model doesn't particularly suffer a drop in performance on large bags when compared with other state-of-the-art models.

K Entity Permutation Test

To understand how robust our trained model would be to changes in the KB, we design the entity permutation test (inspired by Ribeiro et al. (2020)). An ideal DS-RE model should be able to correctly predict the relationship between an entity pair by understanding the semantics of the text mentioning them. Since DS-RE models under the MI-ML setting are evaluated on bag-level, it might be the case that such models are simply memorizing the KB on which they are being trained on.

Figure 4: AUC on test set with different bin sizes



To test this hypothesis, we construct a new test set using NYT-10m by augmenting its KB. Let $B(e_1, e_2)$ denote a non-NA bag already existing in the test set of the dataset. We augment this bag to correspond to a new entity-pair (which is not present in the combined KB of all three splits of this dataset). The augmentation can be of two different types: replacing e_1 with e_1' or replacing e_2 with e_2'. We restrict such augmentations to the same type (i.e the type of e_i and e_i' is same for $i = 1, 2$). For each non-NA entity pair in the test set of the dataset, we select one such augmentation and appropriately modify each instance in $B(e_1, e_2)$ to have the new entity mentions. We note that since each instance in NYT-10m is manually annotated and since our augmentation ensures that the type signature is preserved, the transformation is label preserving. For the NA bags, we use the ones already present in the original split. This entire transformation leaves us with an augmented test set, having same number of NA and non-NA bags as the original split. The non-NA entity pairs are not present in the KB on which the model is trained on.

L More Analysis on DiS-ReX

L.1 Relation-wise F1 scores

To show how our model performs on each relation label compared to other competitive baselines, we present relation-wise F1 scores on DiS-ReX in table 6.

L.2 Language-wise AUC scores

We compare the performance of our model compared to other baselines on every language in DiS-ReX. For this, we partition the test data into language-wise test sets i.e. containing instances of only a particular language. The results are presented in table 7. We observe that the order of performance across languages is consistent for all models including ours i.e. German < English < Spanish < French. Further we observe that our model beats the second best model by an AUC ranging from 3 upto 4 points on all languages.

L.3 Do multilingual bags improve performance?

To understand whether the currently available aggregation schemes (including ours) are able to benefit from multilingual bags or not, we conduct an experiment where we only perform inference on test-set bags that contain instances from all four languages. In the multilingual case, the *passage* constructed during the *Passage Summarization* step will contain multiple sentences of different languages. To understand whether such an input allows improves (or hampers) the performance, we devise an experiment where we perform inference by removing sentences from any one, two or three languages from the set of bags containing instances of all four languages. There are roughly 1500 bags of such kind. Note that removing any k languages ($k <= 3$) would result in $\binom{4}{k}$ different sets and we take average of AUC while reporting the numbers. The results are presented in figure 5.

We observe that in all aggregation schemes, AUC increases with increase in number of languages of a multilingual bag. *mPARE* consistently beats the other models in each scenario, indicating that the

Relation	*mPARE*	mBERT-MNRE	mBERT-Avg
http://dbpedia.org/ontology/birthPlace	**78.6**	75.3	74.9
http://dbpedia.org/ontology/associatedBand	**77.7**	70.9	74.7
http://dbpedia.org/ontology/director	**87.5**	83.2	85.5
http://dbpedia.org/ontology/country	**87.8**	86	85.2
http://dbpedia.org/ontology/deathPlace	**71.3**	67.3	65.5
http://dbpedia.org/ontology/nationality	**71.4**	67.7	68.7
http://dbpedia.org/ontology/location	**74.5**	70.5	67.5
http://dbpedia.org/ontology/related	**79.4**	75.5	73.2
http://dbpedia.org/ontology/isPartOf	**74.9**	68.6	64.7
http://dbpedia.org/ontology/influencedBy	57.0	**58.4**	57.4
http://dbpedia.org/ontology/starring	**87.6**	86.1	83.9
http://dbpedia.org/ontology/headquarter	**72.9**	70.7	66.7
http://dbpedia.org/ontology/successor	**76.1**	71.8	71.3
http://dbpedia.org/ontology/bandMember	**76.1**	**74.6**	74.3
http://dbpedia.org/ontology/producer	**58.5**	53.6	48.5
http://dbpedia.org/ontology/recordLabel	**90.5**	86.9	86.1
http://dbpedia.org/ontology/city	**85.2**	78.8	77.6
http://dbpedia.org/ontology/influenced	59.5	**61.9**	51.5
http://dbpedia.org/ontology/author	80.1	78.2	**80.5**
http://dbpedia.org/ontology/team	**84.6**	82.5	78.6
http://dbpedia.org/ontology/formerBandMember	57.2	**57.4**	56.5
http://dbpedia.org/ontology/state	**87.2**	83.9	82.4
http://dbpedia.org/ontology/region	**84.1**	80.4	78.8
http://dbpedia.org/ontology/subsequentWork	**73.4**	72.4	69.6
http://dbpedia.org/ontology/department	**96.3**	95.4	95.5
http://dbpedia.org/ontology/locatedInArea	**77.4**	72.5	72.3
http://dbpedia.org/ontology/artist	**80.6**	77.2	78.6
http://dbpedia.org/ontology/hometown	**77.7**	73.6	73.7
http://dbpedia.org/ontology/province	**81.3**	79.2	78.2
http://dbpedia.org/ontology/riverMouth	**76.9**	72.4	71.9
http://dbpedia.org/ontology/locationCountry	**68.9**	62.5	64.2
http://dbpedia.org/ontology/predecessor	**68.7**	68.1	62
http://dbpedia.org/ontology/previousWork	68.8	**69.6**	65.5
http://dbpedia.org/ontology/capital	**71.7**	55.1	58
http://dbpedia.org/ontology/leaderName	**80.1**	70.4	63.3
http://dbpedia.org/ontology/largestCity	**68.7**	59.1	48.6

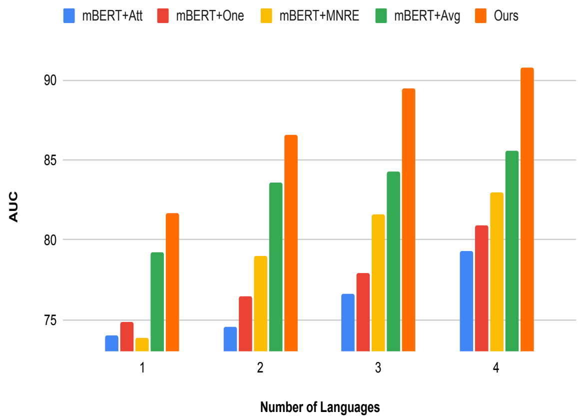
Table 6: Relation-wise F1 scores on DiS-Rex. Bold and underline represent best and second best models respectively on a class. Our model consistently beats the other 2 models in 31 out of 36 relation classes, thus showing how strong our approach is for the multilingual setting.

Model	English	French	German	Spanish
mPARE	**83.2**	**86.8**	**81.7**	**85.3**
mBERT-Avg	79.9	83.1	77.7	82.1
mBERT-MNRE	79.6	82.2	75.5	81.6

Table 7: Language-wise AUC comparison of our model v/s baseline models.

encoding of a multilingual passage and attention-based summarization over multilingual tokens doesn't hamper the performance of a DS-RE model with increasing no. of languages.

Figure 5: AUC vs number of languages in a bag in DiS-ReX test set



M Negligible effect of random ordering

Since we order the sentences randomly into a passage to be encoded by BERT, this may potentially cause some randomness in the results. However, we hypothesize that the BERT encoder must also be getting fine-tuned to treat the bag as a set (and not a sequence) of sentences when being trained with random ordering technique. And as a result, it's performance must be agnostic to the order of sentences it sees in a passage during inference. To validate this, we perform 20 inference runs of our trained model with different seeds i.e. the ordering of sentences is entirely random in each run. We measure mean and standard deviation for each dataset as listed in table 8. We observe negligible standard deviation in all metrics. A minute variation in Macro-F1 or P@M metrics may be attributed to the fact that these are macro-aggregated metrics and a variation in performance over some data points may also affect these to some extent.

	NYT-10m		NYT-10d		Wiki-20m		DiS-ReX	
	AUC	**M-F1**	**AUC**	**P@M**	**AUC**	**M-F1**	**AUC**	**M-F1**
	61.18	37.47	51.98	89.33	90.86	83.33	86.91	76.32
	61.23	37.58	52.01	89.67	90.86	83.34	86.94	76.49
	61.29	37.47	51.81	88.67	90.86	83.31	86.88	76.35
	61.25	37.11	51.78	89.0	90.87	83.34	86.86	76.24
	61.19	37.36	51.89	88.67	90.88	83.5	86.91	76.34
	61.21	37.75	51.83	88.67	90.87	83.28	86.92	76.38
	61.29	37.23	51.9	89.33	90.87	83.27	86.89	76.31
	61.26	37.49	51.79	88.67	90.86	83.37	86.87	76.29
	61.18	37.27	51.87	88.67	90.87	83.29	86.94	76.51
	61.3	37.41	51.68	88.33	90.86	83.27	86.9	76.37
	61.28	37.09	51.92	89.0	90.86	83.31	86.91	76.3
	61.26	37.29	51.78	88.67	90.86	83.31	86.94	76.45
	61.19	37.37	52.05	90.0	90.87	83.33	86.9	76.27
	61.25	37.25	51.78	89.0	90.88	83.37	86.89	76.28
	61.21	37.61	51.68	89.33	90.86	83.34	86.92	76.41
	61.22	37.55	51.96	89.33	90.86	83.37	86.91	76.35
	61.21	37.48	51.77	89.33	90.88	83.42	86.9	76.36
	61.23	37.22	51.76	88.67	90.86	83.27	86.92	76.44
	61.24	37.36	51.95	89.33	90.86	83.31	86.91	76.27
	61.19	37.24	51.87	88.67	90.87	83.33	86.89	76.32
Average	61.22	37.36	51.85	89.02	90.87	83.33	86.91	76.32
Std-Dev	0.05	0.16	0.08	0.42	0.01	0.06	0.01	0.07
Std-Dev(%)	0.08	0.4	0.15	0.48	0.01	0.07	0.01	0.1

Table 8: We perform 20 inference runs with random seeds of our trained model on each dataset and report the mean and standard deviation. All numbers have been rounded upto second decimal place. We observe negligible stdandard deviation in all metrics on all datasets thus validating our hypothesis that the model learns to treat a bag of sentences as a set (and not a sequence) of sentences treating any random order almost alike. Note that the results presented in main paper are for inference done with same seed value with which the model has been trained. However, in current analysis we select random seed values at inference (irrespective of the one with which it was trained).