A Additional Implementation Details

We used an Nvidia V100 server with 16BG VRAM for our experiments. They can be run with a single Nvidia GTX 1080 with 8GB VRAM with the same hyperparameters as experimented during prototyping. We report the average number of epochs and time for every configuration in Table 1. We report the number of parameters in our models in Table 2.

	CoNLL04		ACE05	
Model	Ep.	Time	Ep.	Time
BERT + Span	52	166	25	160
BERT + BILOU	16	20	22	50
BiLSTM + Span	20	52	17	100
BiLSTM + BILOU	14	7	14	18

Table 1: Average number of epochs before early stopping and corresponding runtime in minutes for a training with early stopping on the dev RE Strict μ F1 score.

Module	CoNLL04	ACE05
BERT Embedder	108 M	108 M
GloVe Embedder	2.6 M	5.6 M
charBiLSTM	34 k	35 k
BiLSTM Encoder	2.3 M	2.3 M
Span NER	4 k	7 k
BILOU NER	13 k	22 k
RE Decoder	12 k	14 k
BERT + Span	108 M	108 M
BERT + BILOU	108 M	108 M
BiLSTM + Span	5 M	8 M
BiLSTM + BILOU	5 M	8 M

Table 2: Number of parameters in the different modules of our models.

B Additional Datasets Statistics

We provide more detailed statistics on the two datasets we used for our experimental study in Tables 3 and 4. We believe that reporting the number of sentences, entity mentions and relation mentions per training partition is a minimum to enable sanity checks ensuring data integrity.

	Reference	Train	Dev	Test	Total
Sentences	(R&Y, 04)	-	-	-	1437
	(G, 16)	922	231	288	1441
	Ours	922	231	288	1441
Tokens	(A&S, 17)	23,711	6,119	7,384	37,274
	Ours	26,525	6,993	8,336	41,854
Entities	(R&Y, 04)	-	-	-	5,336
	(A&S, 17)	3,373	858	1,071	5,302
	Ours	3,377	893	1,079	5,349
Relations	(R&Y, 04)	-	-	-	2,040
	(A&S, 17)	1,270	351	422	2,043
	Ours	1,283	343	422	2,048

Table 3: Detailed statistics of our CoNLL04 dataset, as preprocessed by Eberts and Ulges (2020) [1]. We compare to previously reported statistics (Roth and Yih, 2004; Gupta et al., 2016; Adel and Schütze, 2017). The test sets from (Gupta et al., 2016), (Adel and Schütze, 2017) and (Eberts and Ulges, 2020) are supposedly the same but we observe differences. Only (Eberts and Ulges, 2020) released their complete training partition.

	Reference	Train	Dev	Test	Total
Documents	(L&J, 14)	351	80	80	511
	Ours	351	80	80	511
Sentences	(L&J, 14)	7,273	1,765	1,535	10,573
	Ours	10,051	2,420	2,050	14,521
Tokens	Ours	144,783	35,548	30,595	210,926
Entities	(L&J, 14)	26,470	6,421	5,476	38,367
	Ours	26,473	6,421	5,476	38,370
Relations	(L&J, 14)	4,779	1,179	1,147	7,105
	Ours	4,785	1,181	1,151	7,117

Table 4: Detailed statistics of our ACE05 dataset, following Miwa and Bansal (2016)'s preprocessing scripts[2]. We compare to previously reported statistics by (Li and Ji, 2014). The large difference in the number of sentences is likely due to a different sentence tokenizer.

[1] https://github.com/markus-eberts/spert
[2] https://github.com/tticoin/LSTM-ER

C Additional Comparison of ACE05 and CoNLL04

ACE05 and CoNLL04 have key differences we propose to visualize with global statistics. First, in CoNLL04 every sentence contains at least two entity mentions and one relation while the majority of ACE05 contains no entities nor relations as depicted in Fig. 1.We can also notice that among sentences containing relations, a higher proportion of ACE05 contains several of them. Second, the variety of combinations between relation types and argument types makes RE on ACE05 much more difficult than on CoNLL04 (Fig. 2 and 3).

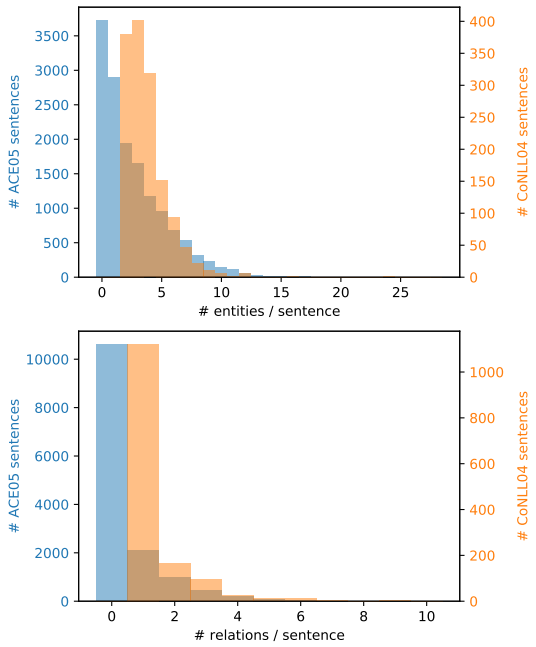

Figure 1: Distribution of the number of entity and relation mentions per sentence in ACE05 and CoNLL04.

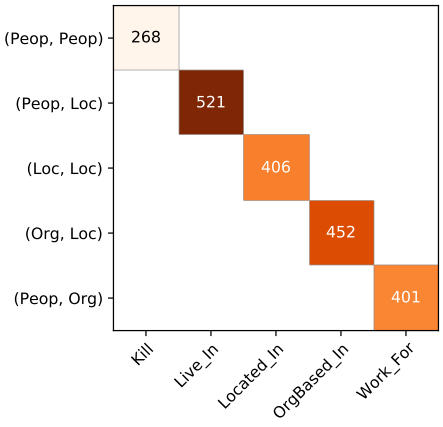

Figure 2: Occurrences of each relation / argument types combination in CoNLL04.

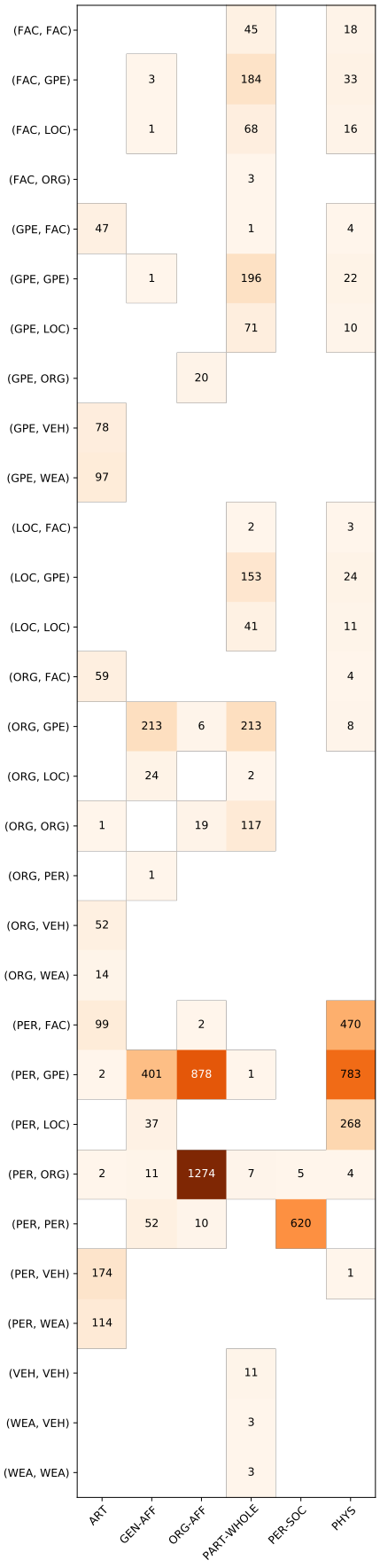

Figure 3: Occurrences of each relation / argument types combination in ACE05.

References

Heike Adel and Hinrich Schütze. 2017. Global normalization of convolutional neural networks for joint entity and relation classification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1723–1729, Copenhagen, Denmark. Association for Computational Linguistics.

Markus Eberts and Adrian Ulges. 2020. Span-based Joint Entity and Relation Extraction with Transformer Pre-training. In *Proceedings of the 12th European Conference on Artificial Intelligence (ECAI)*.

Pankaj Gupta, Hinrich Schütze, and Bernt Andrassy. 2016. Table filling multi-task recurrent neural network for joint entity and relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2537–2547, Osaka, Japan. The COLING 2016 Organizing Committee.

Qi Li and Heng Ji. 2014. 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)*, pages 402–412, Baltimore, Maryland. Association for Computational Linguistics.

Makoto Miwa and Mohit Bansal. 2016. 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)*, pages 1105–1116, Berlin, Germany. Association for Computational Linguistics.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 1–8, Boston, Massachusetts, USA. Association for Computational Linguistics.