An End-to-end Model for Entity-level Relation Extraction using Multi-instance Learning

Markus Eberts **Adrian Ulges**
RheinMain University of Applied Sciences
Wiesbaden, Germany
{markus.eberts, adrian.ulges}@hs-rm.de

Abstract

We present a joint model for entity-level relation extraction from documents. In contrast to other approaches – which focus on local intra-sentence mention pairs and thus require annotations on mention level – our model operates on entity level. To do so, a multi-task approach is followed that builds upon coreference resolution and gathers relevant signals via multi-instance learning with multi-level representations combining global entity and local mention information. We achieve state-of-the-art relation extraction results on the DocRED dataset and report the first entity-level end-to-end relation extraction results for future reference. Finally, our experimental results suggest that a joint approach is on par with task-specific learning, though more efficient due to shared parameters and training steps.

1 Introduction

Information extraction addresses the inference of formal knowledge (typically, entities and relations) from text. The field has recently experienced a significant boost due to the development of neural approaches (Zeng et al., 2014; Zhang and Wang, 2015; Kumar, 2017). This has led to two shifts in research: First, while earlier work has focused on sentence level relation extraction (Hendrickx et al., 2010; Han et al., 2018; Zhang et al., 2017), more recent models extract facts from longer text passages (document-level). This enables the detection of inter-sentence relations that may only be implicitly expressed and require reasoning across sentence boundaries. Current models in this area do not rely on mention-level annotations and aggregate signals from multiple mentions of the same entity.

The second shift has been towards multi-task learning: While earlier approaches tackle entity mention detection and relation extraction with separate models, recent joint models address these tasks at once (Bekoulis et al., 2018; Nguyen and Verspoor, 2019; Wadden et al., 2019). This does not only improve simplicity and efficiency, but is also commonly motivated by the fact that tasks can benefit from each other: For example, knowledge of two entities' types (such as *person*+*organization*) can boost certain relations between them (such as *ceo_of*).

We follow this line of research, and present JEREX[1] ("**J**oint **E**ntity-Level **R**elation **Ex**tractor"),

The **Portland Golf Club** is a private golf club in the northwest **United States**, in suburban Portland, Oregon. The **PGC** is located in the unincorporated **Raleigh Hills** area of eastern Washington County, southwest of downtown Portland and east of Beaverton. **PGC** was established in the winter of **1914**, when a group of nine businessmen assembled to form a new club after leaving their respective clubs. The **golf club** hosted the Ryder Cup matches of 1947, the first renewal in a decade, due to World War II. The **U.S.** team defeated Great Britain 11 to 1 in wet conditions in early November.

Figure 1: Our goal is to perform end-to-end entity-level relation extraction on whole documents. We extract entity mentions ("PGC"), entity clusters ({Portland Golf Club, PGC, golf club}), their types (ORG) and relations to other entities in the document, such as ($\{\text{Portland Golf Club, PGC, golf club}\}_{ORG}$, *inception*, $\{1914\}_{TIME}$), with a single, joint model. Note that document-level relation extraction requires the aggregation of relevant information from multiple sentences, such as in ($\{\text{Raleigh Hills}\}_{LOC}$, *country*, $\{\text{United States, U.S.}\}_{LOC}$). Other entities in the example document are omitted for clarity.

[1]The code for reproducing our results is available at https://github.com/lavis-nlp/jerex.

a novel approach for joint information extraction. JEREX is to our knowledge the first approach that combines a multi-task model with entity-level relation extraction: In contrast to previous work, our model jointly learns relations and entities without annotations on mention level, but extracts document-level entity clusters and predicts relations between those clusters using a *multi-instance learning* (MIL) (Dietterich et al., 1997; Riedel et al., 2010; Surdeanu et al., 2012) approach. The model is trained jointly on mention detection, coreference resolution, entity classification and relation extraction (Figure 1).

While we follow best practices for the first three tasks, we propose a novel representation for relation extraction, which combines global entity-level representations with localized mention-level ones. We present experiments on the DocRED (Yao et al., 2019) dataset for entity-level relation extraction. Though it is arguably simpler compared to recent graph propagation models (Nan et al., 2020) or special pre-training (Ye et al., 2020), our approach achieves state-of-the-art results.

We also report the first results for end-to-end relation extraction on DocRED as a reference for future work. In ablation studies we show that (1) combining a global and local representations is beneficial, and (2) that joint training appears to be on par with separate per-task models.

2 Related Work

Relation extraction is one of the most studied natural language processing (NLP) problems to date. Most approaches focus on classifying the relation between a given entity mention pair. Here various neural network based models, such as RNNs (Zhang and Wang, 2015), CNNs (Zeng et al., 2014), recursive neural networks (Socher et al., 2012) or Transformer-type architectures (Wu and He, 2019) have been investigated. However, these approaches are usually limited to local, intra-sentence, relations and are not suited for document-level, inter-sentence, classification. Since complex relations require the aggregation of information distributed over multiple sentences, document-level relation extraction has recently drawn attention (e.g. Quirk and Poon 2017; Verga et al. 2018; Gupta et al. 2019; Yao et al. 2019). Still, these models rely on specific entity mentions to be given. While progress in the joint detection of entity mentions and intra-sentence relations has been made (Gupta et al., 2016; Bekoulis et al., 2018; Luan et al., 2018), the combination of coreference resolution with relation extraction for entity-level reasoning in a single, jointly-trained, model is widely unexplored.

Document-level Relation Extraction Recent work on document-level relation extraction directly learns relations between entities (i.e. clusters of mentions referring to the same entity) within a document, requiring no relation annotations on mention level. To gather relevant information across sentence boundaries, multi-instance learning has successfully been applied to this task. In multi-instance learning, the goal is to assign labels to bags (here, entity pairs), each containing multiple instances (here, specific mention pairs). Verga et al. (2018) apply multi-instance learning to detect domain-specific relations in biological text. They compute relation scores for each mention pair of two entity clusters and aggregate these scores using a smooth max-pooling operation. Christopoulou et al. (2019) and Sahu et al. (2019) improve upon Verga et al. (2018) by constructing document-level graphs to model global interactions. While the aforementioned models tackle very specific domains with few relation types, the recently released DocRED dataset (Yao et al., 2019) enables general-domain research on a rich relation type set (96 types). Yao et al. (2019) provide several baseline architectures, such as CNN-, LSTM- or Transformer-based models, that operate on global, mention averaged, entity representations. Wang et al. (2019) use a two-step process by identifying related entities in a first step and classifying them in a second step. Tang et al. (2020) employ a hierarchical inference network, combining entity representations with attention over individual sentences to form the final decision. Nan et al. (2020) apply a graph neural network (Kipf and Welling, 2017) to construct a document-level graph of mention, entity and meta-dependency nodes. The current state-of-the-art constitutes the CorefRoBERTa model proposed by Ye et al. (2020), a RoBERTa (Liu et al., 2019) variant that is pre-trained on detecting co-referring phrases. They show that replacing RoBERTa with CorefRoBERTa improves performance on DocRED.

All these models have in common that entities and their mentions are both assumed to be given. In contrast, our approach extracts mentions, clusters them to entities, and classifies relations jointly.

Joint Entity Mention and Relation Extraction Prior joint models focus on the extraction of mention-level relations in sentences. Here, most approaches detect mentions by BIO (or BILOU) tagging and pair detected mentions for relation classification, e.g. (Gupta et al., 2016; Zhou et al., 2017; Zheng et al., 2017; Bekoulis et al., 2018; Nguyen and Verspoor, 2019; Miwa and Bansal, 2016). However, these models are not able to detect relations between overlapping entity mentions. Recently, so-called span-based approaches (Lee et al., 2017) were successfully applied to this task (Luan et al., 2018; Eberts and Ulges, 2020): By enumerating each token span of a sentence, these models handle overlapping mentions by design. Sanh et al. (2019) train a multi-task model on named entity recognition, coreference resolution and relation extraction. By adding coreference resolution as an auxilary task, Luan et al. (2019) propagate information through coreference chains. Still, these models rely on mention-level annotations and only detect intra-sentence relations between mentions, whereas our model explicitly constructs clusters of co-referring mentions and uses these clusters to detect complex entity-level relations in long documents using multi-instance reasoning.

3 Approach

JEREX processes documents containing multiple sentences and extracts entity mentions, clusters them to entities, and outputs types and relations on entity level. JEREX consists of four task-specific components, which are based on the same encoder and mention representations, and are trained in a joint manner. An input document is first tokenized, yielding a sequence of n byte-pair encoded (BPE) (Sennrich et al., 2016) tokens. We then use the pre-trained Transformer-type network BERT (Devlin et al., 2019) to obtain a contextualized embedding sequence $(\mathbf{e}_1, \mathbf{e}_2, ...\mathbf{e}_n)$ of the document. Since our goal is to perform end-to-end relation extraction, neither entities nor their corresponding mentions in the document are known in inference.

3.1 Model Architecture

We suggest a multi-level model: First, we localize all entity mentions in the document (a) by a *span-based* approach (Lee et al., 2017). After this, detected mentions are clustered into entities by *coreference resolution* (b). We then classify the type (such as *person* or *company*) of each entity cluster by a fusion over local mention representations (*entity classification*) (c). Finally, relations between entities are extracted by a reasoning over mention pairs (d). The full model architecture is illustrated in Figure 2.

(a) Entity Mention Localization Here our model performs a search over all document token subsequences (or *spans*). In contrast to BIO/BILOU-based approaches for entity mention localization, span-based approaches are able to detect overlapping mentions. Let $s := (\mathbf{e}_i, \mathbf{e}_{i+1}, ..., \mathbf{e}_{i+k})$ denote an arbitrary candidate span. Following Eberts and Ulges (2020), we first obtain a span representation by max-pooling the span's token embeddings:

$$\mathbf{e}(s) := \text{max-pool}(\mathbf{e}_i, \mathbf{e}_{i+1}, ..., \mathbf{e}_{i+k}) \quad (1)$$

Our *mention classifier* takes the span representation $\mathbf{e}(s)$ as well as a span size embedding $\mathbf{w}^s_{k+1}$ (Lee et al., 2017) as meta information. We perform binary classification and use a sigmoid activation to obtain a probability for s to constitute an entity mention:

$$\hat{y}^s = \sigma\Big(\text{FFNN}^s(\mathbf{e}(s) \circ \mathbf{w}^s_{k+1})\Big) \quad (2)$$

where $\circ$ denotes concatenation and FFNN^s is a two-layer feedforward network with an inner ReLu activation. Span classification is carried out on all token spans up to a fixed length L. We apply a filter threshold α^s on the confidence scores, retaining all spans with $\hat{y}^s \geq \alpha_s$ and leaving a set $\mathcal{S}$ of spans supposedly constituting entity mentions.

(b) Coreference Resolution Entity mentions referring to the same entity (e.g. "Elizabeth II." and "the Queen") can be scattered throughout the input document. To later extract relations on entity level, local mentions need to be grouped to document-level entity clusters by coreference resolution. We use a simple mention-pair (Soon et al., 2001) model: Our component classifies pairs $(s_1, s_2) \in \mathcal{S} \times \mathcal{S}$ of detected entity mentions as coreferent or not, by combining the span representations $\mathbf{e}(s_1)$ and $\mathbf{e}(s_2)$ with an edit distance embedding $\mathbf{w}^c_d$: We compute the Levenshtein distance (Levenshtein, 1966) between spans $d := D(s_1, s_2)$ and use a learned embedding $\mathbf{w}^c_d$. A mention pair representation $\mathbf{x}^c$ is constructed by concatenation:

$$\mathbf{x}^c := \mathbf{e}(s_1) \circ \mathbf{e}(s_2) \circ \mathbf{w}^c_d \quad (3)$$

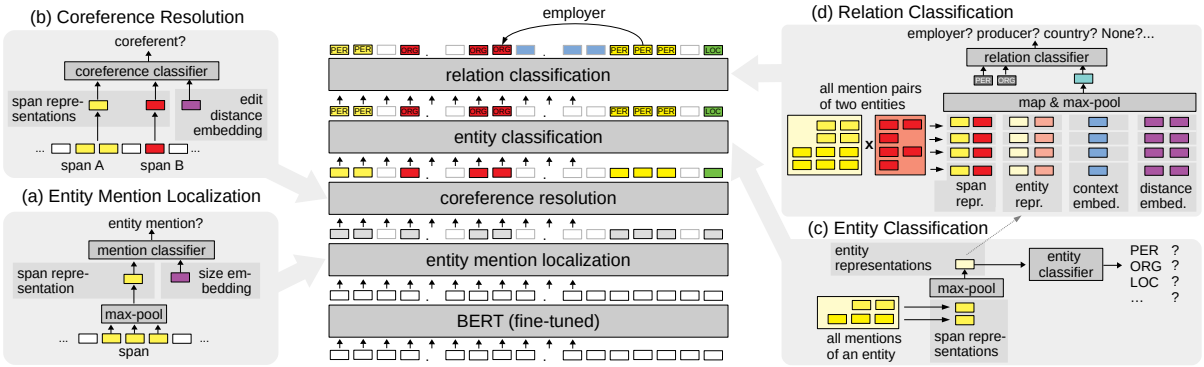

Figure 2: Our approach combines entity mention localization (a), coreference resolution (b), entity classification (c) and relation classification (d) within a joint multi-task model, which is trained jointly on entity-level relation extraction. The sub-components share a single BERT encoder for document encoding. Each input document is only encoded once (*single-pass*) to speed-up training/inference, with sub-components operating on the contextualized embeddings. Both entity classification and relation classification use multi-instance learning to synthesize relevant signals scattered throughout the input document.

Similar to span classification, we conduct binary classification using a sigmoid activation, obtaining a similarity score between the two mentions:

$$\hat{y}^c := \sigma\Big(\text{FFNN}^c(\mathbf{x}^c)\Big) \tag{4}$$

where FFNN^c follows the same architecture as FFNN^s. We construct a similarity matrix $C \in \mathbb{R}^{m \times m}$ (with m referring to the document's overall number of mentions) containing the similarity scores between every mention pair. By applying a filter threshold α^c, we cluster mentions using complete linkage (Müllner, 2011), yielding a set $\mathcal{E}$ containing clusters of entity mentions. We refer to these clusters as *entities* or *entity clusters* in the following.

(c) Entity Classification Next, we map each entity to a type such as $location$ or $person$: We first fuse the mention representations of an entity cluster $\{s_1, s_2, ..., s_t\} \in \mathcal{E}$ by max-pooling:

$$\mathbf{x}^e := \text{max-pool}(\mathbf{e}(s_1), \mathbf{e}(s_2), ..., \mathbf{e}(s_t)) \tag{5}$$

Entity classification is then carried out on the entity representation $\mathbf{x}^e$, allowing the model to draw information from mentions spread across different parts of the document. $\mathbf{x}^e$ is fed into a softmax classifier, yielding a probability distribution over the entity types:

$$\hat{y}^e := \text{softmax}\Big(\text{FFNN}^e(\mathbf{x}^e)\Big) \tag{6}$$

We assign the highest scored type to the entity.

(d) Relation Classification Our final component assigns relation types to pairs of entities. Note that the directionality, i.e. which entity constitutes the head/tail of the relation, needs to be inferred, and that the input document can express multiple relations between different mentions of the same entity pair. Let $\mathcal{R}$ denote a set of pre-defined relation types. The relation classifier processes each entity pair $(e_1, e_2) \in \mathcal{E} \times \mathcal{E}$, estimating which, if any, relations from $\mathcal{R}$ are expressed between these entities. To do so, we score every candidate triple (e_1, r_i, e_2), expressing that e_1 (as head) is in relation r_i with e_2 (as tail). We design two types of relation classifiers: A *global relation classifier*, serving as a baseline, which consumes the entity cluster representations $\mathbf{x}^e$, and a *multi-instance classifier*, which assumes that certain entity mention pairs support specific relations and synthesizes this information into an entity-pair level representation.

Global Relation Classifier (GRC) The global classifier builds upon the max-pooled entity cluster representations $\mathbf{x}^e_1$ and $\mathbf{x}^e_2$ of an entity pair (e_1, e_2). We further embed the corresponding entity types ($\mathbf{w}^e_1$ / $\mathbf{w}^e_2$), which was shown to be beneficial in prior work (Yao et al., 2019), and compute an entity-pair representation by concatenation:

$$\mathbf{x}^p := \Big(\mathbf{x}^e_1 \circ \mathbf{w}^e_1\Big) \circ \Big(\mathbf{x}^e_2 \circ \mathbf{w}^e_2\Big) \tag{7}$$

This representation is fed into a 2-layer FFNN (similar to FFNN^s), mapping it to the number of relation types $\#\mathcal{R}$. The final layer features sigmoid activations for multi-label classification and assigns

any relation type exceeding a threshold α^r:

$$\hat{y}^r := \sigma\Big(\text{FFNN}^p(\mathbf{x}^p)\Big) \quad (8)$$

Multi-instance Relation Classifier (MRC) In contrast to the global classifier (GRC), the multi-instance relation classifier operates on mention level: Since only entity-level labels are available, we treat entity mention pairs as latent variables and estimate relations by a fusion over these mention pairs. For any pair of entity clusters $e_1{=}\{s_1^1, s_2^1, ..., s_{t_1}^1\}$ and $e_2{=}\{s_1^2, s_2^2, ..., s_{t_2}^2\}$, we compute a mention-pair representation for any $(s_1, s_2){\in}e_1{\times}e_2$. This representation is obtained by concatenating the global entity embeddings (Equation (5)) with the mentions' local span representations (Equation (1))

$$\mathbf{u}(s_1, s_2) := \Big(\mathbf{e}(s_1) \circ \mathbf{x}_1^e\Big) \circ \Big(\mathbf{e}(s_2) \circ \mathbf{x}_2^e\Big) \quad (9)$$

Further, as we expect close-by mentions to be stronger indicators of relations, we add meta embeddings for the *distances* d_s,d_t between the two mentions, both in sentences (d_s) and in tokens (d_t). In addition, following Eberts and Ulges (2020), the max-pooled context between the two mentions ($\mathbf{c}(s_1, s_2)$) is added. This *localized context* provides a more focused view on the document and was found to be especially beneficial for long, and therefore noisy, inputs:

$$\mathbf{u}'(s_1,s_2){:=}\mathbf{u}(s_1,s_2) \circ \mathbf{c}(s_1,s_2) \circ \mathbf{w}_{d_s}^r \circ \mathbf{w}_{d_t}^{r'} \quad (10)$$

This mention-pair representation is mapped by a single feed-forward layer to the original token embedding size (768):

$$\mathbf{u}''(s_1, s_2) := \text{FFNN}^p(\mathbf{u}'(s_1, s_2)) \quad (11)$$

These focused representations are then combined by max-pooling:

$$\mathbf{x}^r{=}\text{max-pool}(\{\mathbf{u}''(s_1, s_2)|s_1{\in}e_1,s_2{\in}e_2\}) \quad (12)$$

Akin to GRC, we concatenate $\mathbf{x}^r$ with entity type embeddings $\mathbf{w}_1^e/\mathbf{w}_2^e$ and apply a two-layer FFNN (again, similar to FFNN^s). Note that for both classifiers (GRC/MRC), we need to score both (s_1, r_i, s_2) and (s_2, r_i, s_1) to infer the direction of asymmetric relations.

3.2 Training

We perform a supervised multi-task training, whereas each training document features ground truth for all four subtasks (mention localization, coreference resolution, as well as entity and relation classification). We optimize the joint loss of all four components:

$$\mathcal{L} := \beta_s \cdot \mathcal{L}^s + \beta_c \cdot \mathcal{L}^c + \beta_e \cdot \mathcal{L}^e + \beta_r \cdot \mathcal{L}^r \quad (13)$$

$\mathcal{L}^s$, $\mathcal{L}^c$ and $\mathcal{L}^r$ denote the binary cross entropy losses of the span, coreference and relation classifiers. We use a cross entropy loss ($\mathcal{L}^e$) for the entity classifier. A batch is formed by drawing positive and negative samples from a single document for all components. We found such a *single-pass approach* to offer significant speed-ups both in learning and inference:

- Entity mention localization: We utilize all ground truth entity mentions $\mathcal{S}^{gt}$ of a document as positive training samples, and sample a fixed number N_s of random non-mention spans up to a pre-defined length L_s as negative samples. Note that we only train and evaluate on the full tokens according to the dataset's tokenization, i.e. not on byte-pair encoded tokens, to limit computational complexity. Also, we only sample intra-sentence spans as negative samples. Since we found intra-mention spans to be especially challenging ("New York" versus "New York City"), we sample up to $\frac{N_s}{2}$ intra-mention spans as negative samples.

- Coreference resolution: The coreference classifier is trained on all span pairs drawn from ground truth entity clusters $\mathcal{E}^{gt}$ as positive samples. We further sample a fixed number N_c of pairs of random ground truth entity mentions that do not belong to the same cluster as negative samples.

- Entity classification: Since the entity classifier only receives clusters that supposedly constitute an entity during inference, it is trained on all ground truth entity clusters of a document.

- Relation classification: Here we use ground truth relations between entity clusters as positive samples and N_r negative samples drawn from $\mathcal{E}^{gt}{\times}\mathcal{E}^{gt}$ that are unrelated according to the ground truth.

Each component's loss is obtained by averaging over all samples. We learn the weights and biases of sub-component specific layers as well as the

Level	Task	Joint Model*			Pipeline		
		Precision	Recall	F1	Precision	Recall	F1
(a)	Mention Localization	93.29	92.70	92.99	92.87	92.46	92.66
(b)	Coreference Resolution	82.52	83.06	82.79	82.11	82.66	82.39
(c)	Entity Classification	79.84	80.36	80.10	79.00	79.52	79.26
(d)	Relation Classification	42.76	38.25	40.38	43.61	37.50	40.32
	Relation Classification (GRC)	38.69	37.32	37.98	39.07	36.44	37.70

Table 1: Test set evaluation results of our multi-level end-to-end system JEREX on DocRED (using the end-to-end split). We either train the model jointly on all four sub-components (left) or arrange separately trained models in a pipeline (right) (* joint results are for MRC except for the last row).

meta embeddings during training. BERT is fine-tuned in the process.

4 Experiments

We evaluate JEREX on the DocRED dataset (Yao et al., 2019). DocRED ist the most diverse relation extraction dataset to date (6 entity and 96 relation types). It includes over 5,000 documents, each consisting of multiple sentences. According to Yao et al. (2019), DocRED requires multiple types of reasoning, such as logical or common-sense reasoning, to infer relations.

Note that previous work only uses DocRED for relation extraction (which equals our relation classifier component) and assumes entities to be given (e.g. Wang et al. 2019; Nan et al. 2020). On the other hand, DocRED is exhaustively annotated with mentions, entities and entity-level relations, making it suitable for end-to-end systems. Therefore, we evaluate JEREX both as a relation classifier (to compare it with the state-of-the-art) and as a joint model (as reference for future work on joint entity-level relation extraction).

While prior joint models focus on mention-level relations (e.g. Gupta et al. 2016; Bekoulis et al. 2018; Chi et al. 2019), we extend the strict evaluation setting to entity level: A mention is counted as correct if its span matches a ground truth mention span. An entity cluster is considered correct if it matches the ground truth cluster exactly and the corresponding mention spans are correct. Likewise, an entity is considered correct if the cluster as well as the entity type matches a ground truth entity. Lastly, we count a relation as correct if its argument entities as well as the relation type are correct. We measure precision, recall and micro-F1 for each sub-task and report micro-averaged scores.

Split	#Doc.	#Men.	#Ent.	#Rel.
Train	3,008	78,677	58,708	37,486
Dev	300	7,702	5,805	3,678
Test	700	17,988	13,594	8,787
Total	4,008	104,367	78,107	49,951

Table 2: DocRED dataset split used for end-to-end relation extraction.

Dataset split The original DocRED dataset is split into a train (3,053 documents), dev (1,000) and test (1,000) set. However, test relation labels are hidden and evaluation requires the submission of results via Codalab. To evaluate end-to-end systems, we form a new split by merging train and dev. We randomly sample a train (3,008 documents), dev (300 documents) and test set (700 documents). Note that we removed 45 documents since they contained wrongly annotated entities with mentions of different types. Table 2 contains statistics of our end-to-end split[2]. We release the split as a reference for future work.

Hyperparameters We use $\text{BERT}_{\text{BASE}}$ (cased)[3] for document encoding, an attention-based language model pre-trained on English text (Devlin et al., 2019). Hyperparameters were tuned on the end-to-end dev set: We adopt several settings from (Devlin et al., 2019), including the usage of the Adam Optimizer with a linear warmup and linear decay learning rate schedule, a peak learning rate of 5e-5[4] and application of dropout with a rate of 0.1 throughout the model. We

[2]Note that DocRED contains some duplicate annotations. These are included in the statistics, but are filtered for evaluation in the end-to-end setting.

[3]We use the implementation from (Wolf et al., 2019).

[4]We performed a grid search over [5e-6, 1e-5, 5e-5, 1e-4, 5e-4].

Model	Ign F1	F1
CNN (Yao et al., 2019)	40.33	42.26
LSTM (Yao et al., 2019)	47.71	50.07
Ctx-Aware (Yao et al., 2019)*	48.40	50.70
BiLSTM (Yao et al., 2019)	48.78	51.06
Two-Step (Wang et al., 2019)*	-	53.92
HIN (Tang et al., 2020)*	53.70	55.60
JEREX (GRC)*	53.76	55.91
LSR (Nan et al., 2020)*	56.97	59.05
CorefRo (Ye et al., 2020)*	57.90	60.25
JEREX (MRC)*	**58.44**	**60.40**

Table 3: Comparison of our relation classification component (GRC/MRC) with the state-of-the-art on the DocRED relation extraction task. We report test set results on the original DocRED split. Ign F1 ignores relational facts also present in the train set. Models marked with $*$ use a Transformer-type model for document encoding.

	JM*	SM
Task	F1	F1
Mention Localization	92.99	92.66
Coreference Resolution	90.54	90.46
Entity Classification	95.66	95.29
Relation Classification	59.46	59.76
Relation Classification (GRC)	56.45	56.55

Table 4: Single-task performance of the joint model (left) and separate models (right) on the end-to-end split (* joint results are for MRC except for the last row).

set the size of meta embeddings ($\mathbf{w}^s$, $\mathbf{w}^c$, $\mathbf{w}^e$, $\mathbf{w}^r_{d_s}$, $\mathbf{w}^{r'}_{d_t}$) to 25 and the number of epochs to 20. Performance is measured once per epoch on the dev set, out of which the best performing model is used for the final evaluation on the test set. A grid search is performed for the mention, coreference and relation filter threshold (α^s=0.85, α^c=0.85, α^r(GRC)=0.55, α^r(MRC)=0.6) with a step size of 0.05. The number of negative samples (N_s=N_c=N_r=200) and sub-task loss weights (β_s=β_c=β_r=1, β_e=0.25) are manually tuned. Note that some documents in DocRED exceed the maximum context size of BERT (512 BPE tokens). In this case we train the remaining position embeddings from scratch.

4.1 End-to-End Relation Extraction

JEREX is trained and evaluated on the end-to-end dataset split (see Table 2). We perform 5 runs for each experiment and report the averaged results. To study the effects of joint training, we experiment with two approaches: (a) All four sub-components are trained jointly in a single model as described in Section 3.2 and (b) we construct a pipeline system by training each task separately and not sharing the document encoder.

Table 1 illustrates the results for the joint (left) and pipeline (right) approach. As described in Section 3, each sub-task builds on the results of the previous component during inference. We observe the biggest performance drop for the relation classification task, underlining the difficulty in detecting document-level relations. Furthermore, the multi-instance based relation classifier (MRC) outperforms the global relation classifier (GRC) by about 2.4% F1 score. We reason that the fusion of local evidences by multi-instance learning helps the model to focus on appropriate document sections and alleviates the impact of noise in long documents. Moreover, we found the multi-instance selection to offer good interpretability, usually selecting the most relevant instances (see Figure 3 for examples). Overall, we observe a comparable performance by joint training versus using the pipeline system.

This is also confirmed by the results reported in Table 4, where we evaluate the four components independently, i.e. each component receives ground truth samples from the previous step in the hierarchy (e.g. ground truth mentions for coreference resolution). Again, we observe the performance difference between the joint and pipeline model to be negligible. This shows that it is not necessary to build separate models for each task, which would result in training and inference overhead due to multiple expensive BERT passes. Instead, a single neural model is able to jointly learn all tasks necessary for document-level relation extraction, therefore easing training, inference and maintenance.

4.2 Relation Extraction

We also compare our model with the state-of-the-art on DocRED's relation extraction task. Here, entity clusters are assumed to be given. We train and test our relation classification component on the original DocRED dataset split. Since test set labels are hidden, we submit the best out of 5 runs on the development set via CodaLab to retrieve the test set results. Table 3 includes previously re-

Queequeg is a fictional character in the 1851 novel Moby-Dick by American author **Herman Melville**. The son of a South Sea chieftain who left home to explore the world, **Queequeg** is the first principal character encountered by the narrator, Ishmael. The quick friendship and relationship of equality between the tattooed cannibal and the white sailor shows **Melville**'s basic theme of shipboard democracy and racial diversity...

Shadowrun:Hong Kong is a turn-based tactical role-playing video game set in the Shadowrun universe. It was developed and published by **Harebrained Schemes**, who previously developed **Shadowrun Returns** and its standalone expansion. It includes a new single - player campaign and also shipped with a level editor that lets players create their own Shadowrun campaigns and share them with other players. In January 2015, **Harebrained Schemes** launched a Kickstarter campaign in order to fund additional features and content they wanted to add to the game, but determined would not have been possible with their current budget. The initial funding goal of US $ 100,000 was met in only a few hours. The campaign ended the following month, receiving over $ 1.2 million. The game was developed with an improved version of the engine used with **Shadowrun Returns** and Dragonfall. **Harebrained Schemes** decided to develop the game only for Microsoft Windows, OS X, and Linux, ...

Figure 3: Two example documents of the DocRED dataset. Highlighted are relations "creator" between "Queequeg" and "Herman Melville" (top) and "developer" between "Shadowrun Returns" and "Harebrained Schemes" (bottom). Bordered pairs are the top selections of the multi-instance relation classifier.

ported results from current state-of-the-art models. Note that our global classifier (GRC) is similar to the baseline by (Yao et al., 2019). However, we replace mention span averaging with max-pooling and also choose max-pooling to aggregate mentions into an entity representation, yielding considerable improvement over the baseline. Using the multi-instance classifier (MRC) instead further improves performance by about 4.5%. Here our model also outperforms complex methods based on graph attention networks (Nan et al., 2020) or specialized pre-training (Ye et al., 2020), achieving a new state-of-the-art result on DocRED's relation extraction task.

4.3 Ablation Studies

We perform several ablation studies to evaluate the contributions of our proposed multi-instance relation classifier enhancements: We remove either the global entity representations $\mathbf{x}_1^e, \mathbf{x}_2^e$ (Equation 5) (a) or the localized context representation $\mathbf{c}(s_1, s_2)$ (Equation 10) (b). The performance drops by about 0.66% F1 score when global entity representations are omitted, indicating that multi-instance reasoning benefits from the incorporation of entity-level context. When the localized context representation is omitted, performance is reduced by about 0.90%, confirming the importance of guiding the model to relevant input sections. Finally, we limit the model to fusing only intra-sentence mention pairs (c). In case no such instance exists for an entity pair, the closest (in token distance) mention pair is selected. Obviously, this modification reduces computational complexity and memory consumption, especially for large documents. Nevertheless, while we observe intra-sentence pairs to cover most relevant signals, exhaustively pairing all mentions of an entity pair yields an improvement of 0.67%.

Model	F1
Relation Classification (MRC)	59.76
- (a) Entity Representations	59.10
- (b) Localized Context	58.85
- (c) Exhaustive Pairing	59.09

Table 5: Ablation studies for the multi-level relation classifier (MRC) using the end-to-end split. We either remove global entity representations (a), the localized context (b) or only use intra-sentence mention pairs (c). The results are averaged over 5 runs.

5 Conclusions

We have introduced JEREX, a novel multi-task model for end-to-end relation extraction. In contrast to prior systems, JEREX combines entity mention localization with coreference resolution to extract entity types and relations on an entity level. We report first results for entity-level, end-to-end, relation extraction as a reference for future work. Furthermore, we achieve state-of-the-art results on the DocRED relation extraction task by enhancing multi-instance reasoning with global entity representations and a localized context, outperforming several more complex solutions. We showed that training a single model jointly on all subtasks instead of using a pipeline approach performs roughly on par, eliminating the need of training separate models and accelerating inference. One of the remaining shortcomings lies in the detection of false positive relations, which may be expressed according to the entities' types but are actually not

expressed in the document. Exploring options to reduce these false positive predictions seems to be an interesting challenge for future work.

Acknowledgments

This work was funded by German Federal Ministry of Education and Research (Program FHprofUnt, Project DeepCA (13FH011PX6)).

References

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

Renjun Chi, Bin Wu, Linmei Hu, and Yunlei Zhang. 2019. Enhancing Joint Entity and Relation Extraction with Language Modeling and Hierarchical Attention. In *Proc. APWeb-WAIM, LNCS 11641*, pages 314–328.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the Dots: Document-level Neural Relation Extraction with Edge-oriented Graphs. In *Proc. of EMNLP and IJCNLP 2019*, pages 4925–4936, Hong Kong, China. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of NAACL-HLT 2019*, pages 4171–4186, Minneapolis, Minnesota. ACL.

Thomas G. Dietterich, Richard H. Lathrop, and Toms Lozano-Prez. 1997. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1):31 – 71.

Markus Eberts and Adrian Ulges. 2020. Span-based Joint Entity and Relation Extraction with Transformer Pre-training. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI 2020)*, pages 2006 – 2013. IOS Press.

Pankaj Gupta, Subburam Rajaram, Hinrich Schütze, and Thomas A. Runkler. 2019. Neural Relation Extraction within and across Sentence Boundaries. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6513–6520. AAAI Press.

Pankaj Gupta, Hinrich Schütze, and Bernt Andrassy. 2016. Table Filling Multi-Task Recurrent Neural Network for Joint Entity and Relation Extraction. In *Proc. of COLING 2016*, pages 2537–2547, Osaka, Japan. The COLING 2016 Organizing Committee.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. FewRel: A Large-Scale Supervised Few-Shot Relation Classification Dataset with State-of-the-Art Evaluation. In *Proc. of EMNLP 2018*, pages 4803–4809, Brussels, Belgium. ACL.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó. Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. SemEval-2010 Task 8: Multi-way Classification of Semantic Relations Between Pairs of Nominals, booktitle = Proc. of the 5th International Workshop on Semantic Evaluation. SemEval '10, pages 33–38, Stroudsburg, PA, USA. ACL.

Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Shantanu Kumar. 2017. A Survey of Deep Learning Methods for Relation Extraction. *CoRR*, abs/1705.03645.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end Neural Coreference Resolution. In *Proc. of EMNLP 2017*, pages 188–197, Copenhagen, Denmark. ACL.

V. I. Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet physics. Doklady*, 10:707–710.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction. In *Proc. of EMNLP 2018*, pages 3219–3232, Brussels, Belgium. ACL.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A General Framework for Information Extraction using Dynamic Span Graphs. In *Proc. of NAACL-HLT 2019*, volume 1, pages 3036–3046, Minneapolis, Minnesota. ACL.

Makoto Miwa and Mohit Bansal. 2016. End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures. In *Proc. of ACL 2016*, pages 1105–1116, Berlin, Germany. ACL.

Daniel Müllner. 2011. Modern hierarchical, agglomerative clustering algorithms. *CoRR*, abs/1109.2378.

Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. 2020. Reasoning with Latent Structure Refinement for Document-Level Relation Extraction. In

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1546–1557, Online. Association for Computational Linguistics.

Dat Quoc Nguyen and Karin Verspoor. 2019. End-to-end neural relation extraction using deep biaffine attention. In *Advances in Information Retrieval*, pages 729–738, Cham. Springer International Publishing.

Chris Quirk and Hoifung Poon. 2017. Distant Supervision for Relation Extraction beyond the Sentence Boundary. In *Proceedings of EACL: Volume 1, Long Papers*, pages 1171–1182, Valencia, Spain. Association for Computational Linguistics.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling Relations and Their Mentions without Labeled Text. In *Machine Learning and Knowledge Discovery in Databases*, pages 148–163, Berlin, Heidelberg. Springer Berlin Heidelberg.

Sunil Kumar Sahu, Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Inter-sentence Relation Extraction with Document-level Graph Convolutional Neural Network. In *Proc. of the 57th ACL*, pages 4309–4316, Florence, Italy. Association for Computational Linguistics.

Victor Sanh, Thomas Wolf, and Sebastian Ruder. 2019. A Hierarchical Multi-Task Approach for Learning Embeddings from Semantic Tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:6949–6956.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic Compositionality Through Recursive Matrix-vector Spaces. In *Proc. of EMNLP-CoNLL 2012*, pages 1201–1211, Stroudsburg, PA, USA. ACL.

Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A Machine Learning Approach to Coreference Resolution of Noun Phrases. *Computational Linguistics*, 27(4):521–544.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance Multi-label Learning for Relation Extraction. In *Proceedings of EMNLP-CoNLL 2012*, pages 455–465, Jeju Island, Korea. Association for Computational Linguistics.

Hengzhu Tang, Yanan Cao, Zhenyu Zhang, Jiangxia Cao, Fang Fang, Shi Wang, and Pengfei Yin. 2020. HIN: Hierarchical Inference Network for Document-Level Relation Extraction. In *Advances in Knowledge Discovery and Data Mining*, pages 197–209, Cham. Springer International Publishing.

Patrick Verga, Emma Strubell, and Andrew McCallum. 2018. Simultaneously Self-Attending to All Mentions for Full-Abstract Biological Relation Extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 872–884, New Orleans, Louisiana. Association for Computational Linguistics.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, Relation, and Event Extraction with Contextualized Span Representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789, Hong Kong, China. Association for Computational Linguistics.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William W. J. Wang. 2019. Fine-tune Bert for DocRED with Two-step Process. *ArXiv*, abs/1909.11898.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rmi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv*, abs/1910.03771.

Shanchan Wu and Yifan He. 2019. Enriching Pre-trained Language Model with Entity Information for Relation Classification. *CoRR*, abs/1905.08284.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A Large-Scale Document-Level Relation Extraction Dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy. Association for Computational Linguistics.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential Reasoning Learning for Language Representation. In *Proc. of EMNLP*, pages 7170–7186, Online. Association for Computational Linguistics.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation Classification via Convolutional Deep Neural Network. In *Proc. of COLING 2014*, pages 2335–2344, Dublin, Ireland. Dublin City University and ACL.

Dongxu Zhang and Dong Wang. 2015. Relation Classification via Recurrent Neural Network. *CoRR*, abs/1508.01006.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware Attention and Supervised Data Improve Slot Filling. In *Proc. of the EMNLP 2017*, pages 35–45. ACL.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint Extraction of Entities and Relations Based on a Novel Tagging Scheme. In *Proc. of ACL 2017*, pages 1227–1236, Vancouver, Canada. ACL.

Peng Zhou, Suncong Zheng, Jiaming Xu, Zhenyu Qi, Hongyun Bao, and Bo Xu. 2017. Joint Extraction of Multiple Relations and Entities by Using a Hybrid Neural Network. In *Proc. of CCL 2017*, pages 135–146.