Knowledge-Driven Cross-Document Relation Extraction

Monika Jain and **Raghava Mutharaju**

Knowledgeable Computing and Reasoning Lab, IIIT-Delhi, India

{monikaja, raghava.mutharaju}@iiitd.ac.in

Kuldeep Singh

Cerence GmbH and Zerotha Research, Germany

kuldeep.singh1@cerence.com

Ramakanth Kavuluru

University of Kentucky, Lexington, Kentucky, United States

ramakanth.kavuluru@uky.edu

Abstract

Relation extraction (RE) is a well-known NLP application often treated as a sentence or document-level task. However, a handful of recent efforts explore it across documents or in the cross-document setting (CrossDocRE). This is distinct from the single document case because different documents often focus on disparate themes, while text within a document tends to have a single goal. Current CrossDocRE efforts do not consider domain knowledge, which are often assumed to be known to the reader when documents are authored. Here, we propose a novel approach, KXDocRE, that embed domain knowledge of entities with input text for cross-document RE. Our proposed framework has three main benefits over baselines: 1) it incorporates domain knowledge of entities along with documents' text; 2) it offers interpretability by producing explanatory text for predicted relations between entities 3) it improves performance over the prior methods. Code and models are available at https://github.com/kracr/cross-doc-relation-extraction.

1 Introduction

Identifying relations between entity pairs within unstructured text is known as relation extraction (RE). Earlier studies concentrated on uncovering relations between entities within a sentence (Zhang et al., 2017; Hsu et al., 2022; Zeng et al., 2014; dos Santos et al., 2015; Cai et al., 2016; Jain et al., 2023). However, an entity pair need not necessarily occur in a sentence. It can also be part of a document. There is substantial literature on extracting relations between entities based on the document itself (Liu et al., 2023; Xu et al., 2020; Christopoulou et al., 2019; Nan et al., 2020; Zeng et al., 2020; Li et al., 2021; Jain et al., 2024). Beyond a single sentence or a document, entity pairs can also be present in different documents (Yao et al., 2021).
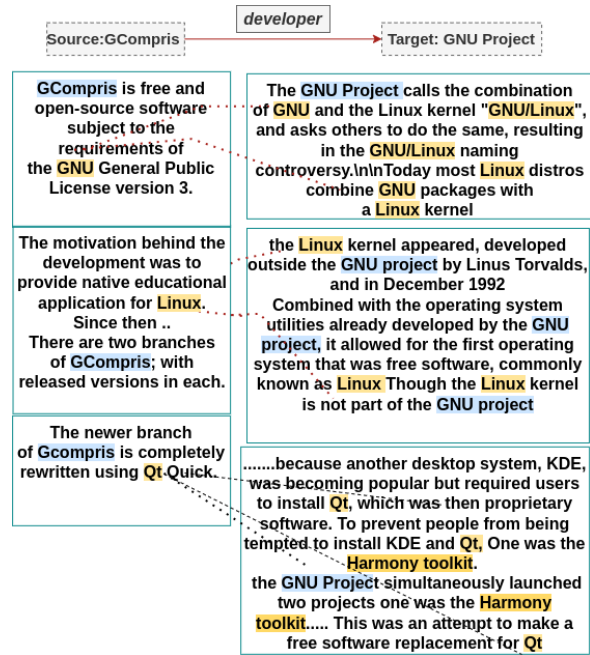

Figure 1: Three textual paths indicate the relationship path between the source entity (GCompris) and the target entity (GNU Project). These connections are established through pairs of documents, where one document features the source entity, and the other contains the target entity. In each path, the connection between the source and target entities is led by a mentioned entity in both documents (e.g., Linux).

An analysis on Wikipedia shows that more than 57.6 relational facts do not co-occur in a single document (Vrandečić and Krötzsch, 2014). This indicates that many facts occur across the documents. Considering that, little work has been performed in cross-document relation extraction (CrossDocRE). CodRED (cross-document relation extraction dataset) is the first dataset published in this line of work (Yao et al., 2021), which serves as a starting point to solve CrossDocRE. Documents containing the source entity are identified and retrieved from multiple documents; the same is done for the target entity. Various text paths be-

tween entity pairs (source and target entity) across documents are recognized using the mentioned entity (other than the source and target entity). Text paths refer to the paths that connect the source and target entities via the mentioned entities. These paths are retrieved from both the source and target documents.

Figure 1 shows an example of CrossDocRE. Between entity pairs **GCompris** and **GNU Project**, multiple text paths via the mentioned entities, such as *Linux, Qt* and *GNU*, can be used to get the correct relation label as **developer**. CodRED uses these text paths in a bag and performs reasoning (Yao et al., 2021). Although this approach seems reasonable, along with relevant text, these text paths also contain noisy data. To overcome these issues, ECRIM (entity-centered cross-document relation extraction) proposed filtering text paths using a mentioned entity (Wang et al., 2022). However, ECRIM only works in one of the settings of CrossDocRE where text paths are provided for reasoning. To address this issues, MR.COD has been proposed, which is a multi-hop reasoning framework based on path mining and ranking (Lu et al., 2023). These models rely on the knowledge between entities in a text and do not consider the domain knowledge associated with entities. Past work along these lines uses entity types and entity aliases to predict the relation (Fernàndez-Cañellas et al., 2020). RECON (Bastos et al., 2021) encoded attribute and relation triples in the Knowledge Graph. KB-Both (Wu and He, 2019) uses entity details from hyperlinked text documents of Wikipedia and Knowledge Graph (KG) from Wikidata to enhance performance. Given input text as a sentence or a document, these models use basic details of entities to improve the performance. In this paper, we explore whether incorporating domain knowledge can enhance the performance of CrossDocRE tasks. The main contributions of this work are as follows.

- We propose a novel model that integrates domain knowledge with a cross-document relation extraction model.

- Our validation demonstrates the effectiveness of our model, with nontrivial performance gains in cross-document RE.

- We enhance the predicted relationships through textual explanations, offering insights into how the relations were predicted.

2 Related work

The relation between entities can extend across multiple documents, and researchers have investigated the extraction of entities, events, and relationships in a cross-document context with unlabeled data (Yao et al., 2010). CodRED (Yao et al., 2021) is the first open source human-annotated cross-document dataset. In this work, the authors address the problem using two approaches. The first method involves a pipeline approach in which they construct a relational graph for each document and reason over these graphs to extract the desired relation. The second method, referred as the joint approach, combines various text path representations through a selective attention mechanism to predict relations. Although this method is effective, it does not consider the mentioned entity-based sentences. ECRIM uses an entity-based document context filter to retain useful information in the given documents by using the mentioned entities in the text paths. Secondly, they solve CrossDocRE using cross-path entity relation attention, allowing entity relations across text paths to interact with each other (Wang et al., 2022). Nevertheless, this work focuses on a closed setting where evidential context has been given instead of all documents. A multi-hop evidence retrieval method based on evidence path mining and ranking has been proposed in MR.COD (Lu et al., 2023). In evidence path mining, a multi-document passage graph is constructed, where passages are linked by edges corresponding to shared entities. A graph traversal algorithm mines the passage paths from source-to-target entities. In evidence path ranking, paths are ranked based on relevance and top-K evidence paths are selected as input for downstream relation extraction models. Alternatively, a causality-guided global reasoning algorithm is also used to filter confusing information and achieve global reasoning to solve cross-document relation extraction (Wu et al., 2023). Proposed models for CrossDocRE until now does not consider background knowledge.

3 Methodology

3.1 Problem statement

For a given entity pair, $\langle e_s, e_o \rangle$ our task is to predict the target relation $r^c \in \mathcal{R}$ that holds between e_s and e_o within a given corpus of documents $\mathcal{C}_{i=1}^n$, where $\mathcal{R}$ is the relation set and n is total number of documents. If no relation is inferred, it returns the *NA* label. Besides e_s and e_o, we introduce the
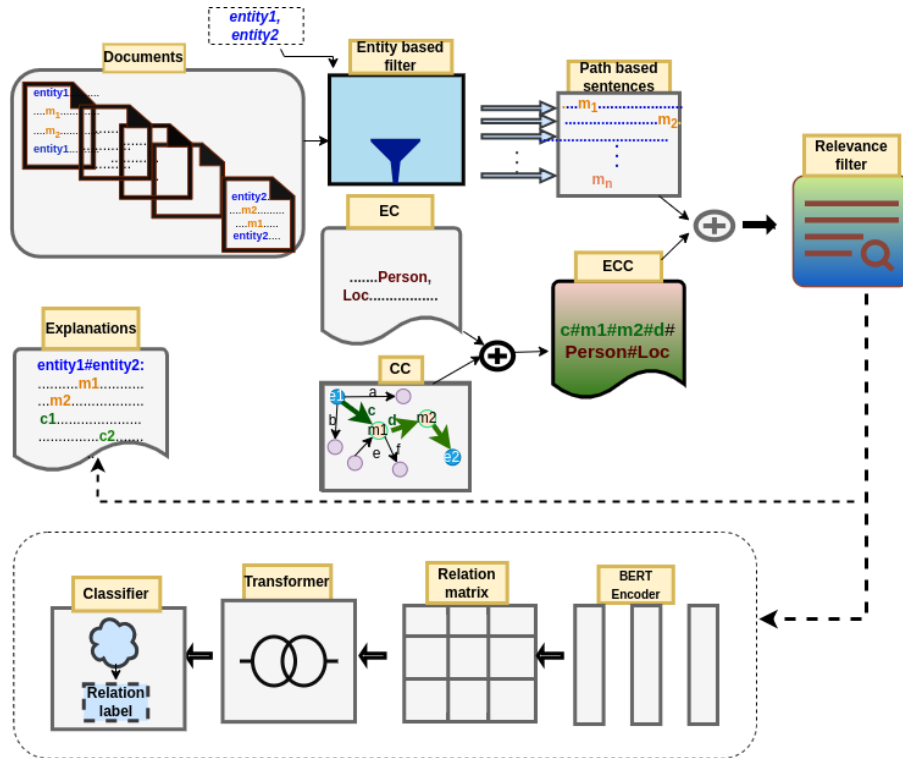
Figure 2: Architecture diagram of KXDocRE for cross-document relation extraction. Here EC represents entity context, CC represents connecting context and ECC represents both entity and connecting path context.

notion of potential bridge entities, which are other mentioned entities in a document that could act as intermediate concepts that help link e_s and e_o via multi-hop connections. The documents provided in CrossDocRE are also annotated with these mentioned entities. We denote $E_i = \{e_i^1, \ldots, e_i^m\}$ to be the set of m mentioned entities in the i-th document, $i = 1, \ldots, M$. CrossDocRE works in two settings, closed and open. In the closed setting, only the related documents are provided to the models, and the relations are inferred from the provided documents. In the more challenging and realistic open setting, the whole corpus of documents is provided, and the model needs to efficiently and precisely retrieve related evidence from the corpus. We now discuss the architecture of KXDocRE (Figure 2) in the subsequent sections.

3.2 Domain Knowledge

We consider three forms of domain knowledge knowledge for cross-document relation extraction (Nadgeri et al., 2021): 1) The type information of the source and target entity; 2) The connecting path between the source and target entities, extracted using Wikidata; and 3) A combination of 1) and 2). We use the term *context* in the rest of the paper to capture these three forms of knowledge.

Entity type context (EC): Predicting the relationship between two entities relies heavily on their respective entity types. For instance, if both entities belong to the *Person* category, certain relationships such as *has organization* and *has location* are not viable. However, relationships such as *child* and *spouse* become possible. Therefore, incorporating entity type information can assist the model in excluding obvious relationships, thereby improving its performance. Consider the example given in Figure 3. The type of the source entity *Jim Lynagh* is *Person* and the type of the target entity *Irish Republic*, is *Geo Political Entity*, which is used to identify locations, countries, cities, or geopolitical regions. The entity type context for this example is, EC{Q6196505, Q1140152}={Person, GeoPoliticalEntity}

Connecting path context (CC): The contextual path pertains to an entity pair $\langle e_s, e_o \rangle$. We consider context paths of length up to N_h, the hop distance (a tunable parameter) between the entity pair. In Figure 3, the entity *Jim Lynagh* is five hops away from the entity *Irish Republic*. The four nodes between the entity pair on the path are intermediary entities, and the five edges are the intermediary properties. The contextual path (CP_{e_i,e_j}) is formed using the intermediary enti-
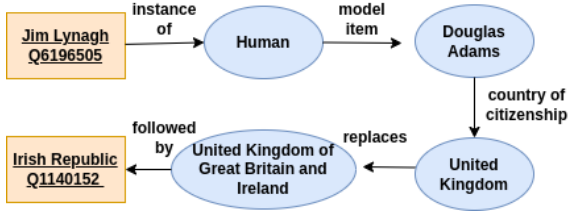
Figure 3: Context path constructed from Wikidata between Jim Lynagh (source) and Irish Republic (target).

ties and properties. So, the connecting path context, $CC_{\{Q6196505,Q1140152\}}$={instance of, Human, model item, Douglas Adams, country of citizenship, United Kingdom, replaces, United Kingdom of Great Britain and Ireland, followed by}.

Entity type with connecting path context (ECC): ECC combines the entity type and the connecting path context. For Figure 3, $ECC_{\{Q6196505,Q1140152\}} = EC_{\{Q6196505,Q1140152\}} \cup CC_{\{Q6196505,Q1140152\}}$.

The steps to generate EC, CC, and ECC for a given entity pair are outlined in Algorithm 1. The ContextGeneration function (lines 3-12) computes the EC, CC, and ECC for the given entity pair. CC is constructed in steps of one hop. In lines 14-17, the adjacent edge and node information is retrieved and added to the path. This continues for N_h hops (lines 18-20). The EC of the source and the target entity is obtained using the named entity recognition technique (lines 27-35).

3.3 Entity-based filter

In this step, we select sentences that contain meaningful connecting information about the source and target entity and remove irrelevant sentences. The sentences are filtered using the mentioned entity, e^m. Similar to the baseline (Wang et al., 2022), our underlying premise is that a sentence holds significance if it contains either a source or a target entity. Additionally, a sentence is relatively significant if it mentions another entity in conjunction with the entity pair found in both documents. We calculate each entity's score based on three conditions. 1) The source and target entities co-occur in some sentence with e^m (Θ_1), 2) An entity, e^o, co-occurring with e^m, also co-occurs with the source entity in a sentence and the target entity in a different sentence (Θ_2), 3) e^m is part of a text path connecting source and target entities (Θ_3). Figure 4 depicts these conditions using the example from Figure 1. The red color represents direct occurrence with the source or target entity in the same sentence, the

---

Algorithm 1: ContextGeneration

Input : Entity pair (v_s,v_o), Number of hops (N_h)
Output : Context

1 **Initialization:** i ← 1, source ← v_s, $path_{i-1}$, finalcontext, finalentitytype, adjacent_node, hop_path_i ← **None**
2 Context ← ContextGeneration(v_s, v_o, N_h)
3 **Function** ContextGeneration(v_s, v_o, N_h):
4   contextpath, entitytype ← { }
5   **foreach** *entity pair v_s, v_o ∈ KnowledgeBase* **do**
6   contextpath.append(ExplorePath(v_s, v_o, N_h))
7   **end**
8   **foreach** *entity pair v_s, v_o* **do**
9   entitytype.append(ExploreEntityType(v_s, v_o)
10   **end**
11   finalcontext ← contextpath ∪ entitytype
12   **return** {finalcontext}
13 **Function** ExplorePath(v_s, v_o, N_h):
14     $edge_i$, adjacent $node_i$ = GetOneHopFromSource(v_s)
15     $path_i$ = {$edge_i$, adjacent $node_i$}
16   i ← 1
17   **while** v_o ≠ *adjacent node and* i ≤ N_h **do**
18        $path_i$ = $path_i$ ∪ GetOneHopFromSource(adjacent node)
19   i ← i + 1
20   **end**
21   **if** *adjacent node == v_o* **then**
22   **return** {$path_i$}
23   **end**
24   **else**
25   **return** {}
26   **end**
27 **Function** ExploreEntityType(v_s, v_o):
28     $entity\_type_s$, $entity\_type_o$ ← { }
29   **if** *EntityTypeExist(v_s)* **then**
30        $entity\_type_s$ ← GetEntityType(v_s)
31   **end**
32   **if** *EntityTypeExist(v_o)* **then**
33        $entity\_type_o$ ← GetEntityType(v_o)
34   **end**
35   finalentitytype ← $entity\_type_s$ ∪ $entity\_type_o$
    return {finalentitytype}

---

black line represents indirect co-occurrence, and the green line represents potential co-occurrence.

For a mentioned entity e^m in each text path p_i, the score of each mentioned entity e^m is calculated by the following equation,

$$score(e^m) = \lambda S_1(e^m) + \eta S_2(e^m) + \kappa S_3(e^m)$$

$$S_1(e^m) = \begin{cases} 1, & \text{if } \Theta_1(e^m) \\ 0, & \text{otherwise} \end{cases}$$

$$S_2(e^m) = \begin{cases} |\{e^o \mid (\Theta_1(e^o) \wedge I(e^o)) = 1\}|, & \text{if } \Theta_2(e^m) \\ 0, & \text{otherwise} \end{cases}$$

$$S_3(e^m) = \begin{cases} |\{p_i \mid e^m \in E_i^m\}|, & \text{if } \Theta_3(e^m) \\ 0, & \text{otherwise} \end{cases}$$

where λ, η, κ are hyper parameters and $I(e^o) = 1$ when e^o and e^m are co-occuring in the same
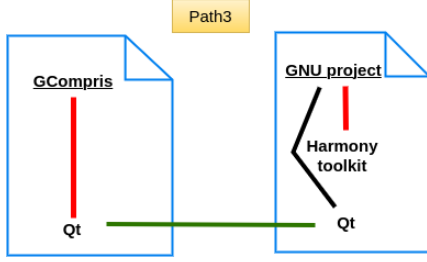
Figure 4: An example of a co-occurring graph for path 3 in Figure 1.

sentence, where $e^o \in E_i^m \setminus \{e^m\}$. S_2 sums the number of occurrences of e^o and S_3 sums the occurrence of mentioned entity in path p_i. Next, we calculate the importance score Imp^s of each sentence by aggregating the scores of each mentioned entity, $Imp^s = \sum_{e^m \in E_s^m} \text{score}(e^m)$, where E_s^m is mentioned entity (bridge) in sentence s. The sentences are then ranked based on their importance score, and the top K sentences form the candidate set, $S = \{s_1, s_2, ..., s_K\}$, where K is set to 16 based on the experiment results. We reused the entity based filter from ECRIM model (Wang et al., 2022).

3.4 Relevance-based filter

We aggregate the candidate set sentences from the previous step with the context selected using entity type with the connecting context. Let $S_{total} = S \cup \{S_{ECC}\}$, where S_{ECC} is the string based domain knowledge obtained from ECC. We then apply a relevance-based filter that considers the semantic relevance of the sentence. Here, we assume that if a sentence is semantically similar to another sentence, including the target entity, this sentence should be more informative than other sentences. Our objective is to get the most informative context I^* from the candidate set S_{total} for reasoning about the relation between entities.

3.5 Encoder

We mark the start and end of every entity and context using special tokens in sentences. Following the baseline (Yao et al., 2021), we have used the BERT model (Devlin et al., 2019) to encode the tokens selected from the previous step. Here, $start$ and end are the start and end positions of the j-th mention, e_j is the j-th mentioned entity, and n is the total number of sentences.

$$e_j = BERT(\{(S_{total}{}_{i=1}^n\})_{start}^{end}$$

3.6 Relation matrix

Besides the mentioned entity, common relations also exist across various text paths. To capture these details from the text path, we used the cross-path entity relation attention module based on Transformer (Vaswani et al., 2017). We collect all entity mentioned representation in a bag and then generate relation representations for entity pairs (e_u, e_v). Here, $\boldsymbol{E}_r$, $\boldsymbol{E}_u$, $\boldsymbol{E}_v$ are learnable parameters and e_u, e_v are combinations of all entities present, including the mentioned entity.

$$\boldsymbol{r}_{u,v} = \text{ReLU}\left(\boldsymbol{E}_r\left(\text{ReLU}\left(\boldsymbol{E}_u\boldsymbol{e}_u + \boldsymbol{E}_v\boldsymbol{e}_v\right)\right)\right)$$

To model the relation interaction across paths, we build a relation matrix $M \in \mathcal{R}^{|E| \times |E| \times d}$, where $E = \bigcup_{i=1}^N E_i$ denotes all the entities in the entity set E_i of text path p_i and $E_i = \{e_s, e_o\} \cup E_i^m$ and N is total number of entities.

3.7 Transformer

For capturing the intra and inter path dependencies, we apply a multi-layer Transformer to perform self attention on the flattened relation matrix $\hat{\boldsymbol{M}} \in \mathbb{R}^{|E|^2 \times d}$

$$\hat{\boldsymbol{M}}^{(t+1|1)} = \text{Transformer}(\hat{\boldsymbol{M}}^{(t)})$$

We obtain the target relation representation r_{h_i,t_i} for each path p_i from the last layer of the Transformer.

3.8 Classifier

After getting the relation representation r_{h_i,t_i} for each text path p_i for each pair of entities, r_{h_i,t_i} is used as a classification feature. We feed these features to the MLP classifier to get the score for each relation. The relation that gets the maximum score is the predicted relation.

$$\hat{y}_i = \boldsymbol{MLP}(r_{h_i,t_i})$$

A max pooling operation is applied to each relation label to obtain the final score for each relation type r.

$$\hat{y}^{(r)} = \text{Max}\left\{\hat{y}_i^{(r)}\right\}_{i=1}^N$$

After obtaining the scores for all relations, a global threshold θ is applied to filter out the categories lower than the threshold. A additional threshold is introduced by baseline paper (Wang et al., 2022) to control which class should be output. The scores of target classes should be greater than threshold and scores of non target class are less than threshold. Formally, for each Bag B, loss is defined as:

$$\mathcal{L} = \log \left(e^{\theta} + \sum_{r \in \Omega_{\text{neg}}^{B}} e^{\hat{\boldsymbol{y}}^{(r)}} \right)$$

$$+ \log \left(e^{-\theta} + \sum_{r \in \Omega_{\text{pos}}^{B}} e^{\hat{\boldsymbol{y}}^{(r)}} \right)$$

Here, $\hat{y}_r$ is score for relation r , θ represent threshold and is set to zero, Ω_{pos}^{B} and Ω_{neg}^{B} are positive and negative classes between target entity pair.

3.9 Explanation

To improve the interpretability of our model, we incorporated an explainable module that can explain the predicted relationship by providing the filtered sentences given to the model. This represents a notable advancement, as contemporary state-of-the-art models cannot often furnish explanations alongside their predictions. In CrossDocRE, the most challenging issue lies in the length of the documents because it affects the amount of noise the model has to handle. So having an explanation module also helps in getting to know how well the model is able to handle noise. Along with that, it also helps in facilitating error analysis. We retrieve the tokens that are fed to the model (I^*), converting them into strings. This way, we get the exact token data that was used to make the relation prediction. This process enables us to obtain the precise token data, which drives predictions.

4 Experimental setup

Metrics. We used F1 and area under curve (AUC) scores following the baseline (Yao et al., 2021) for a fair comparison. For the open setting, we retrieve the top k document paths from the Wikipedia corpus[1] and use the models trained in the closed setting to predict a relation. These paths are scored using three conditions. 1) Entity count: the number of occurrences of the source entity in the document containing the source and likewise with the target entity. 2) Shared entity: the number of shared entities that appear in source and target documents 3) TF-IDF: The TF-IDF similarity between the two documents. Similar to baseline, we selected the top 16 paths with the highest scores. Hyper-parameters used in our model are shown in Table 1.

Baseline model for comparison. We used all the baseline models available in CrossDocRE for comparison. 1) CodRED (Yao et al., 2021) is

---

[1] We follow the baseline paper and set k to 16 (Yao et al., 2021)

Hyper parameters	Value
Learning rate	3e-5
Embedding dimesion	768
Encoder layers	3
γ, η, κ	0.1,0.01, 0.001
Optimizer	AdamW

Table 1: Hyper-parameters setting

used, which extracts text snippets surrounding the two entities in the document as input and feeds it into a BERT-based model 2) ECRIM (Wang et al., 2022); an entity-based document filter is constructed and then fed into a BERT-based model. 3) MR.COD (Lu et al., 2022) is a multi-hop evidence retrieval method based on evidence path mining and ranking. 4) LGCR (Wu et al., 2023) discusses local to global reasoning method which enables efficient distinguishing, filtering and global reasoning on complex information from a causal perspective.

We conducted our evaluation in response to the following research questions. **RQ1**: What is the effectiveness of KXDocRE in combining context knowledge with reasoning in solving CrossDocRE? **RQ2**: Does the explanation generated by our approach provide sufficient grounds to support the inferred relation?

Dataset. We used the CodRED dataset for evaluation. CodRED contains 11,971 entities and 276 relation types in this dataset. The details of Co-DRED dataset are given in Table 2. Bags represent relational facts of dataset, 2733 are positive labeled facts and 16,668 are labeled as NA in training set.

		Train	Dev	Test
Bags	Pos	2733	1010	1012
	NA	16,668	4558	4523
Text paths	-	129,548	40,740	40,524

Table 2: Statistics of CoDRED dataset

5 Results

Our model works in the open and closed settings of CrossDocRE. We evaluated our model with three variations: using entity context (EC), connecting context (CC), and both (ECC). We used an NVIDIA A100-SXM4 tensor core GPU with 40GB of memory on Linux 5.4.0-125 with Python version 3.8.5. The results for closed and open settings are available in Table 3 and Table 4. The results on the test

set were obtained from CodaLab[2]. In contrast to a closed setting, performance declines in an open setting due to the retrieval of paths, not all of which significantly contribute to reasoning. Compared to the baseline model, KXDocRE improves by ≈ 3% in the F1 score (closed setting) and ≈ 4% in the F1 score (open setting). In all settings of KX-DocRE, it was able to reason over text better than baseline. Our findings demonstrated the significant role domain knowledge can play in the process of reasoning. Hence, our to answer **RQ1** is affirmative. In addition to BERT, we did attempt using RoBERTa[3] and GPT2[4] in our evaluations. However, there was no improvement in the F1 score. Hence, we decided to report BERT scores.

Closed setting					
Base model	PLM/GNN	Dev		Test	
		F1	AUC	F1	AUC
CoDRED	BERT	51.26	47.94	51.02	47.46
ECRIM	BERT	61.12	60.91	62.48	60.67
MR.COD	BERT	61.20	59.22	62.53	61.68
LGCR	BERT	61.67	63.17	61.08	60.75
LGCR	RoBERTa	63.18	64.76	63.79	63.03
KXDocRE$_{EC}$	**BERT**	**63.57**	**62.80**	**65.30**	**64.45**
KXDocRE$_{CC}$	**BERT**	**64.00**	**63.70**	**65.80**	**64.90**
KXDocRE$_{ECC}$	**BERT**	**64.97**	**64.30**	**66.30**	**65.55**

Table 3: Results on CodRED dataset for closed setting

Open setting					
Base model	PLM/GNN	Dev		Test	
		F1	AUC	F1	AUC
CoDRED	BERT	47.23	40.86	45.06	39.05
MR.COD	BERT	53.06	51.00	57.88	53.30
LGCR	BERT	52.96	51.48	53.45	50.15
LGCR	RoBERTa	55.15	52.36	55.37	49.05
KXDocRE$_{EC}$	**BERT**	**55.50**	**54.30**	**56.15**	**50.11**
KXDocRE$_{CC}$	**BERT**	**55.90**	**54.80**	**57.12**	**50.60**
KXDocRE$_{ECC}$	**BERT**	**56.70**	**55.20**	**57.93**	**57.12**

Table 4: Results on CodRED dataset for open setting

5.1 Ablation Study

Effectiveness of relevance and entity-based filter: We studied the impact of relevance and entity-based filters on the performance of KXDocRE. Figure 5 shows the F1 score obtained

using the Dev dataset. After removing the relevance filter, the performance dropped by 7.7% and if the entity-based filter is removed, the performance dropped by 4.9%. After removing both filters, the F1 score drops significantly by 12%. This indicates that the two filters play an important role in KXDocRE.
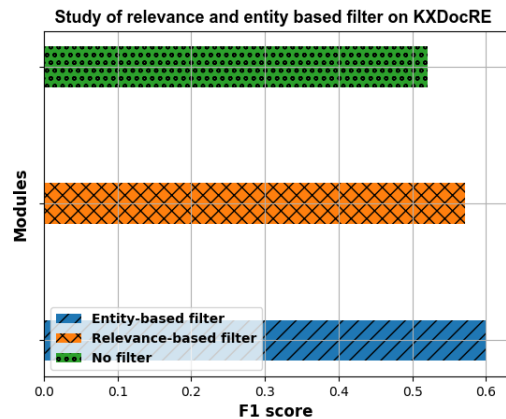

Figure 5: Study of relevance and entity based filter on KXDocRE

Effectiveness of explanations: In cross-document setting, predicting the relationship between entity pairs involves considering multiple text paths (averaging 3646) found across various documents. The explanation module aids in surfacing the most relevant paths that influence the prediction. This module is expected to help us to understand the prediction using a single paragraph containing the relevant paths, rather than scrutinizing each document individually. We discuss the explanation module using a case study (Figure 6). The labeled relation between Oichi (Q635214) and Ohatsu (Q1050395) is *Child (P40)*. The explanation text provides text paths between Oichu and Ohatsu via Azai, Toyotomi and Yodo-dono (Oichi is the spouse of Azai Nagamasa, who is the child of Yodo-dono, who is the sibling of Ohatsu); hence, Oichi is the child of Ohatshu. Therefore, the explanation text provides enough evidence along with the context to reason over the data to understand the predicted relationship. Hence, we answer **RQ2**.

Effectiveness of number of hops on KX-DocRE: We also studied the impact of the number of hops considered for extracting context in a CC setting on the F1 score (Figure 7). The F1 score increases with the number of hops until a point,
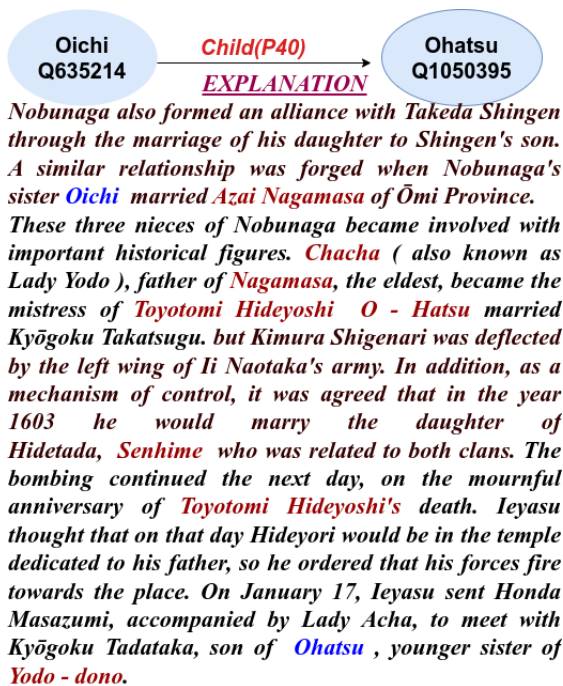
Figure 6: Explanation generated using KXDocRE. The source and target entities are blue in color. Intermediate entities are in red, and aggregated context is in green.

and after that, the F1 score starts decreasing and saturates. Increasing the hops do not add much new and relevant information after a certain point. Due to this reason, we considered the number of hops (N_h) up to 5.
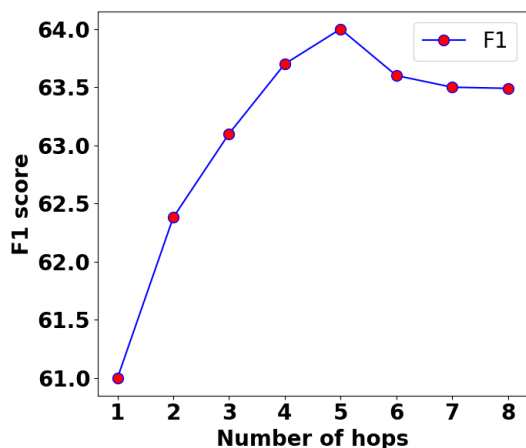

Figure 7: Effectiveness of hops in KXDocRE

Error analysis: We studied the successful and failed cases of KXDocRE based on the domain knowledge of the entity pair. From Table 5, we can say that the likelihood of a correct prediction is higher if context is available for the given entity pair.

	# Dev entity pair	# Correct prediction	# Incorrect prediction	Correct prediction%
KXDocRE$_{EC}$	952	894	58	93.9
KXDocRE$_{CC}$	921	890	31	96.6

Table 5: Impact of context in successful cases on Dev dataset

Complexity analysis: In the CodRED dataset, the text path consists of 129,548 instances in the training set, 40,740 instances in the development set, and 40,524 instances in the test set. The average time of a single epoch's execution in the baseline and our model is given in Table 6. The execution time for KXDocRE is longer (linear increment) due to the addition of context in the module. The time taken to create domain knowledge is available in Table 7.

Case study: We discuss two successful cases and one failed case of KXDocRE and compared them with the baseline model (Table 8). **Case1:** To identify the relation between *Dreamlover (Q909801)* and *If it's Over (Q1095958)* from Documents 1 and 2, we use domain knowledge of both entities. EC does not exist for the given entity pair, hence we use CC. CC for both entities is {*part of, Emotions, tracklist, If It's Over, followed by*}. Adding CC helps KXDocRE to predict the correct relation compared to our baseline model, ECRIM. **Case2:** In second case study, for entity pairs *Airbus A320neo family (Q6488)* and *Airbus (Q67)*, we add the ECC context as {*owned by, ORG, ORG*}, which helps KXDocRE help in predicting the relation. **Case3:** We studied a failed case of KXDocRE for entity pairs *Adium (Q58058)* and *x86_64 (Q272629)*. CC for this entity pair is {*instance of, free software, subclass of, software, model item, Mozilla Firefox model item*}. This context does not contribute significantly in predicting the relationship.

6 Conclusion

In this paper, we introduced a novel model KXDocRE with three settings (EC, CC, ECC) that incorporate domain knowledge in CrossDocRE. Our results suggest that model performance is improved by integrating diverse domain knowledge. Also, we provide an explanation text for each prediction that makes our model interpretable. As a future work, researchers can create a versatile model that can work on diverse document sets, accumulating domain knowledge.

Model	Total Time (in hours)	Total (#epochs)	Average time (in seconds) per epoch
ECRIM (baseline)	76	10	7.6
KXDocRE(ours)	96.8	10	9.6

Table 6: Complexity analysis of KXDocRE as compared to baseline.

Module	Average time (in sec)
Entity context	0.001
Context path (1-hop)	0.11
Context path (2-hop)	0.21
Context path (3-hop)	0.38

Table 7: Average time taken to create the context in KXDocRE.

Case1:

Document 1 : Dreamlover is a song by American singer Mariah Carey ....Dreamlover marked a more pronounced attempt on Carey's part to incorporate .... arey began to alter her songwriting style and genre choices, most notably in Dreamlover. Dreamlover an Vision of Love Carey's best, calling them the original hits...........................

Document 2 : If It's Over If It's Over is a song written by American singers and songwriters Mariah Carey and .....if it's over, let me go Several months after the release of Emotions Carey performed the song...If It's Over is a downtempo ballad, which incorporates several genres...........

Correct answer: followed by

CodRED: N/A

ECRIM: N/A

KXDocRE$_{ECC}$: followed by

Case2:

Document 1 : It is the core development area of of Bandai Namco group. ......main video game branch of Bandai Namco Holdings. In Feb 2005, primariliy set in fictional Japanese city of ......, in association with the Japanese government, suppressed information can be......................

Document 2 : Bandai Visual, Bandai Entertainment, Dentsu, Nippon...........and original Japanese (one late-night screening).......It was released on 4 March 2004 in Japan and 8 November 2004 in North America.....The second volume Ghost in the shell.............in Japan and on 26 September 2006......

Correct answer: parent organization

CodRED: N/A

ECRIM: N/A

KXDocRE$_{ECC}$: parent organization

Case3:

Document 1 : Adium is a free and open source instant messaging client for macOS that supports... including Windows Live Messenger...It is written using macOS...under the GNU....Adium makes use of a plugin architecture...

Document 2 : In computer architecture, 64-bit integers, memory.....ALU architectures are those that are based on.........AMD released its first x86-64...Java program can run on a 32- or ...

Correct answer: N/A

CodRED: N/A

ECRIM: N/A

KXDocRE$_{ECC}$: N/A

Table 8: Case study

7 Limitations

Our work has limitations in terms of creating a connecting context — such a context will only be created if there is some path between two entities in Wikidata. We plan to overcome this limitation by incorporating other external knowledge bases for creating context. With the increase in the number of text paths and mentioned entities, the GPU memory consumption increases, and the speed decreases.

References

Anson Bastos, Abhishek Nadgeri, Kuldeep Singh, Isaiah Onando Mulang, Saeedeh Shekarpour, Johannes Hoffart, and Manohar Kaul. 2021. Recon: relation extraction using knowledge graph context in a graph neural network. In *Proceedings of the Web Conference 2021*, New York, NY, USA.

Rui Cai, Xiaodong Zhang, and Houfeng Wang. 2016. Bidirectional recurrent convolutional neural network for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 756–765, Berlin, Germany. Association for Computational Linguistics.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. 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 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 *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.

Cícero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 626–634, Beijing, China. Association for Computational Linguistics.

Dèlia Fernàndez-Cañellas, Joan Marco Rimmek, Joan Espadaler, Blai Garolera, Adrià Barja, Marc Codina, Marc Sastre, Xavier Giro-i Nieto, Juan Carlos Riveiro, and Elisenda Bou-Balust. 2020. Enhancing online knowledge graph population with semantic knowledge. In *The Semantic Web – ISWC 2020*.

I-Hung Hsu, Kuan-Hao Huang, Shuning Zhang, Wen-Huang Cheng, Premkumar Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. A simple and unified tagging model with priming for relational structure predictions. *ArXiv*, abs/2205.12585.

Monika Jain, Raghava Mutharaju, Ramakanth Kavuluru, and Kuldeep Singh. 2024. Revisiting document-level relation extraction with context-guided link prediction. In *Proceedings of The 38th AAAI Conference on Artificial Intelligence*, Vancouver, Canada. AAAI.

Monika Jain, Kuldeep Singh, and Raghava Mutharaju. 2023. Reonto: A neuro-symbolic approach for biomedical relation extraction. In *Machine Learning and Knowledge Discovery in Databases: Research Track: European Conference, ECML PKDD 2023, Turin, Italy, September 18–22, 2023, Proceedings, Part IV*, page 230–247, Berlin, Heidelberg. Springer-Verlag.

Jingye Li, Kang Xu, Fei Li, Hao Fei, Yafeng Ren, and Donghong Ji. 2021. MRN: A locally and globally mention-based reasoning network for document-level relation extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1359–1370, Online. Association for Computational Linguistics.

Yiming Liu, Hongtao Shan, Feng Nie, Gaoyu Zhang, and George Xianzhi Yuan. 2023. Document-level relation extraction with local relation and global inference. *Information*, 14(7).

K. Lu, I-Hung Hsu, Wenxuan Zhou, Mingyu Derek Ma, and Muhao Chen. 2022. Multi-hop evidence retrieval for cross-document relation extraction. *ArXiv*, abs/2212.10786.

Keming Lu, I-Hung Hsu, Wenxuan Zhou, Mingyu Derek Ma, and Muhao Chen. 2023. Multi-hop evidence retrieval for cross-document relation extraction. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 10336–10351. Association for Computational Linguistics.

Abhishek Nadgeri, Anson Bastos, Kuldeep Singh, Isaiah Onando Mulang, Johannes Hoffart, Saeedeh Shekarpour, and Vijay Saraswat. 2021. Kgpool: Dynamic knowledge graph context selection for relation extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 535–548.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

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

Fengqi Wang, Fei Li, Hao Fei, Jingye Li, Shengqiong Wu, Fangfang Su, Wenxuan Shi, Donghong Ji, and Bo Cai. 2022. Entity-centered cross-document relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9871–9881, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Haoran Wu, Xiuyi Chen, Zefa Hu, Jing Shi, Shuang Xu, and Bo Xu. 2023. Local-to-global causal reasoning for cross-document relation extraction. *IEEE/CAA Journal of Automatica Sinica*, 10(7):1608–1621.

Shanchan Wu and Yifan He. 2019. Enriching pretrained language model with entity information for relation classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2361–2364, New York, NY, USA. ACM.

Wang Xu, Kehai Chen, and Tiejun Zhao. 2020. Document-level relation extraction with reconstruction. In *AAAI Conference on Artificial Intelligence*.

Limin Yao, Sebastian Riedel, and Andrew McCallum. 2010. Collective cross-document relation extraction without labelled data. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, page 1013–1023, USA. Association for Computational Linguistics.

Yuan Yao, Jiaju Du, Yankai Lin, Peng Li, Zhiyuan Liu, Jie Zhou, and Maosong Sun. 2021. CodRED: A cross-document relation extraction dataset for acquiring knowledge in the wild. In *Proceedings of EMNLP 2021*, pages 4452–4472.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2335–2344, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. 2020. Double graph based reasoning for document-level relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1630–1640, Online. Association for Computational Linguistics.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark. Association for Computational Linguistics.