DOREMI: Optimizing Long Tail Predictions in Document-Level Relation Extraction⋆

Laura Menotti[a,*], Stefano Marchesin[a], Gianmaria Silvello[a]

[a]*Department of Information Engineering, University of Padova, Padova, Italy*

Abstract

Document-Level Relation Extraction (DocRE) presents significant challenges due to its reliance on cross-sentence context and the long-tail distribution of relation types, where many relations have scarce training examples. In this work, we introduce **DO**cument-level **R**elation **E**xtraction opti**M**izing the long ta**I**l (DOREMI), an iterative framework that enhances underrepresented relations through minimal yet targeted manual annotations. Unlike previous approaches that rely on large-scale noisy data or heuristic denoising, DOREMI actively selects the most informative examples to improve training efficiency and robustness. DOREMI can be applied to any existing DocRE model and is effective at mitigating long-tail biases, offering a scalable solution to improve generalization on rare relations.

Keywords: Document-Level Relation Extraction, Active Learning, Long-Tail Relations, Natural Language Processing

1. Introduction

Document-Level Relation Extraction (DocRE) is an NLP task that identifies all relations between a given entity pair within a document. DocRE is a more realistic setting than the more commonly studied *sentence-level* Relation Extraction (RE), as many relational facts are typically expressed across multiple sentences. Sentence-level RE considers only a single entity pair connected by a maximum of one relation; whereas, DocRE involves documents that contain multiple entity pairs, each of which can appear multiple times and be associated with different relations. Figure 1 shows the typical case where an entity pair has multiple relations spanning multiple sentences; indeed, the pair (*The Hitch-Hiker*, *Ida Lupino*) has two relations: "*director*" derived from the first sentence and "*screenwriter*" inferred from combining the first and the second sentence. This case is common, for instance, around 40% of relational facts in Wikipedia documents can only be extracted from multiple sentences [1].

The reference test collection for the task is DocRED, a large-scale dataset constructed from Wikipedia and Wikidata [1]. It comprises two training datasets: one includes 3,053 manually annotated documents, while the other 101,873 documents are annotated using Distant Supervision (DS). Although DocRED is commonly used for evaluating DocRE models, recent works showed that the collection struggles with false negatives [2, 3]. Tan et al. [3] found out that almost 65% of ground truth relations were not annotated in DocRED; they corrected the manual training and development set and released Re-DocRED, meant as an improved version of DocRED.

[1] The Hitch-Hiker is a 1953 film noir directed by Ida Lupino, about two fishing buddies who pick up a mysterioys hitchhiker during a trip to Mexico. [2] Inspired by the crime spree of the psychopathid murderer Billy Cook (1928-1952), the screenplay was written by Robert L. Joseph, Lupino, ..(.. 2 sentences..) [5] The film is in the public domain.

Subject: The Hitch-Hiker **Object:** Ida Lupino

Relation: director; screenwriter

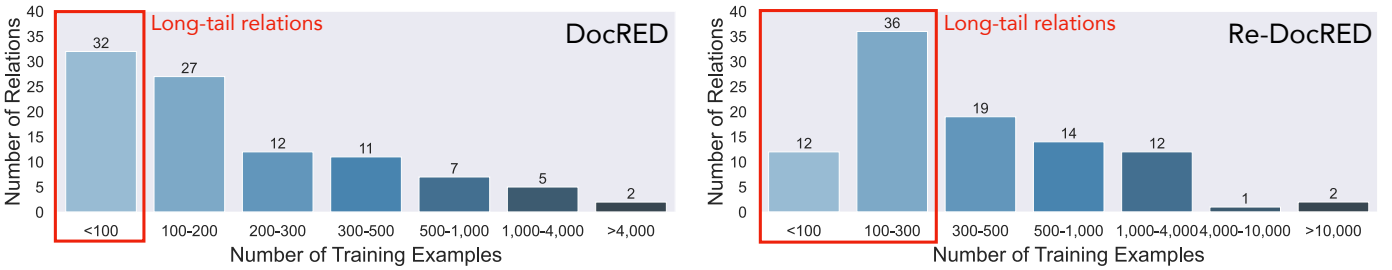
Figure 1: An example of multi-entity and multi-label problems from the DocRED dataset. Subject entity *"The Hitch-Hiker"* (in pink) and object entity *"Ida Lupino"* (in blue) express two relations in the document (*director* and *screenwriter*). Other entities are in grey.

Nevertheless, DocRED and Re-DocRED show a significantly imbalanced distribution of training examples across the 96 annotated relations [4]. Figure 2 shows the distribution of labels in the annotated training datasets of both reference datasets. They exhibit a typical power-law pattern, where a small number of relations – i.e., *frequent relations* – have many training examples, while the majority have only a few – i.e., *long-tail relations*. The DocRED annotated training dataset consists of 38,180 examples, half representing the four most frequent relations, while 31 relations (out of 96) have fewer than 100 training examples each. This class imbalance is even more pronounced in Re-DocRED, where the training dataset contains 85,932 examples (three times more than DocRED), with 20,402 (24% of all instances) about a single relation. In addition, half of the relations (48 out of 96) have fewer than 300 examples each, of which 12 (almost 13% of all relations) have fewer than 100. This class imbalance raises concerns about the actual effectiveness of the models, whose performance is biased

⋆**Paper accepted for publication in Knowledge-Based Systems.**

*Corresponding author.

Email addresses: laura.menotti@unipd.it (Laura Menotti), stefano.marchesin@unipd.it (Stefano Marchesin), gianmaria.silvello@unipd.it (Gianmaria Silvello)



(a). Label distribution in DocRED annotated training dataset. (b). Label distribution in Re-DocRED annotated training dataset.

Figure 2: Label distribution in DocRED (a) and Re-DocRED (b) annotated training dataset.

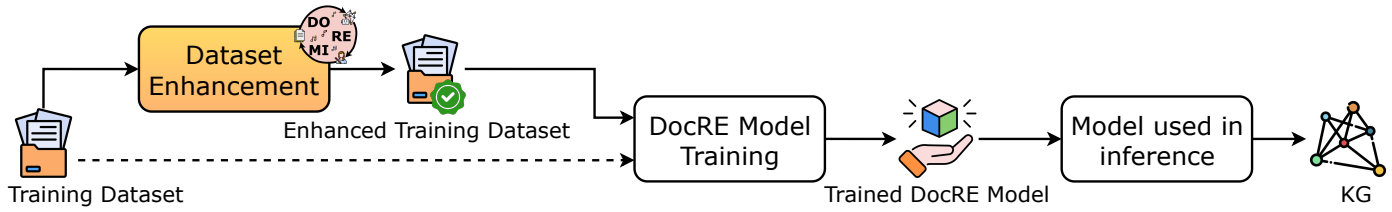


Figure 3: The general pipeline of a DocRE model. DOREMI enhances the training dataset, optimizing long-tail relations. Once the enhanced dataset is obtained, it can be used for training any DocRE model.

toward frequent relations, at the expense of accurately identifying long-tail relations [4, 5].

One potential approach to improve models' performance on the long-tail is to use DS data to augment the number of training examples for underepresented relations. However, the DocRED DS dataset is noisy and biased toward popular relations. This dataset was created by aligning Wikipedia documents with Wikidata triples, where the inclusion of entities and properties tends to favor frequently mentioned concepts [2]. The most effective label denoising strategy in DocRE is UGDRE [6], leveraging uncertainty accuracy estimation based on Monte Carlo dropout. However, it does not consider long-tail relations, requiring strategies that can selectively enhance underrepresented relations. Although promising, Large Language Models (LLMs) still underperform compared to current state-of-the-art approaches for DocRE, making them ineffective for the task [7]. The current best performing approaches for DocRE features ATLOP [8], a sequence-based model introducing an adaptive thresholding and localized context pooling, and DREEAM [9], enhancing predictions with an evidence-based attention network in a teacher-student paradigm.

This work proposes **DO**cument-level **R**elation **E**xtraction opti**M**izing the long ta**I**l (DOREMI), an iterative system that enhances the training data through targeted manual annotation of highly informative examples selected leveraging a disagreement-based criteria inspired by [10]. DOREMI differs from prior denoising strategies for its iterative refinement architecture and the disagreement-based instance selection for manual annotation. To our knowledge, this marks the first effort to tailor an active learning approach specifically for DocRE, featuring a human-in-the-loop strategy for denoising. As shown in Figure 3, DOREMI operates upstream in the general DocRE pipeline by augmenting the dataset in a model-agnostic fashion, enabling any downstream DocRE model to benefit from improved long-tail coverage. To ensure high-reliability extraction, DOREMI is precision-oriented in its design. In real-world applications of automatic Knowledge Base Construction (KBC), prioritizing precision over recall is crucial to ensure the reliability of extracted relations. Including fewer but more accurate triples minimizes the risk of introducing incorrect information, which can significantly degrade downstream tasks [11]. Experimental results on the DocRED development dataset indicate that annotating merely *0.001%* of the DS dataset in DocRED (400 triples in total) significantly enhances the performance of leading models like ATLOP and DREEAM. Compared to the leading label denoising technique, that is, UGDRE, DOREMI registers an increase in overall PRECISION and RECALL of up to +8.2% and +3.3%, respectively (F1 improves by up to +0.7%). For long-tail triples (< 100 examples in training), PRECISION increases by +76.0% and F1 by +5.0%. In long-tail triples involving entity pairs not seen in training, DOREMI boosts IGNF1 by up to 28.7% and IGNPRECISION by 137.6% with respect to UGDRE. The improvements are confirmed on the Re-DocRED test dataset, with a gain of PRECISION and IGNPRECISION up to +16.2% and +19.2% on long-tail triples (< 300 examples in training) from *annotating only 0.003%* of the distant dataset (1,200 triples). Moreover, if we consider *extreme long-tail triples* in Re-DocRED (< 100 examples in training, DOREMI registers a substantial improvement in PRECISION (+83.2%), IGNPRECISION (+207.9%), F1 (1.3%), and IGNF1 (+33.5%). The hybrid setting, i.e., considering UGDRE predicitions for frequent relations and DOREMI for the long-tail, registers an improvement in overall PRECISION and IGNPRECISION of up to +3.7% and +4.5%, respectively. These results indicates DOREMI effectively complements exisiting denoising approaches.

The core **contributions** of this work are listed as follows:

(1) We propose DOREMI, a novel iterative system tailored for long-tail relations to enhance the distantly supervised dataset through disagreement-driven annotations (Section 3); (2) We demonstrate for the first time that measuring the disagreement between multiple models is a good proxy to identify Hard-To-Classify examples specifically for DocRE and yields substantial performance improvements with negligible human effort (Section 6.1); (3) We release two new Denoised Distantly Supervised Datasets (DDSs), one based on DocRED and one on Re-DocRED, which can be used to train any DocRE model [12]. Unlike existing resources, these datasets are explicitly optimized for long-tail relations and can be directly used to train arbitrary DocRE models, leading to consistent and significant improvements as validated by extensive experiments.(Section 5). The implementation of DOREMI is available in GitHub at: `https://github.com/mntlra/DOREMI`.

The rest of this work is organized as follows. Section 2 presents previous efforts in DocRE, with special attention to label denoising and sequence-based models. Section 3 describes DOREMI. Section 5 reports the performance of DOREMI compared to other denoising strategies evaluated on the DocRED development dataset (Section 5.1) and Re-DocRED test dataset (Section 5.2). Section 6 reports an ablation study about disagreement-based sampling and some statistics on the manual annotation process. To conclude, section 7 draws some final remarks.

2. Related Work

DocRE models are categorized into graph-based and sequence-based approaches.

Graph-based methods construct a document graph where nodes represent words, mentions, entities, or sentences, and edges capture various dependencies. Early approaches used syntactic dependencies to build these graphs [13]. Christopoulou et al. [14] introduced the edge-oriented graph model, which employs different types of edges to capture intra- and inter-sentential relations. Subsequent models, like GAIN [15], constructed both mention-level and entity-level graphs to perform multi-hop reasoning for relation extraction. These methods leverage graph neural networks to propagate information across the graph and infer relations between entities. In this context, SSAN introduces self-attention to incorporate the coreference and co-occurrence structure of entities into training [16]. DocuNet instead considers DocRE as a semantic segmentation task and exploits an entity-level relation matrix to capture local and global information [17]. Xu et al. [18] proposes a reconstruction mechanism to allow the model to capture path dependencies between an entity pair and its ground-truth relationship. Wang et al. [19] leverages Hierarchical Dependency Tree and Bridge Path (HDT-BP) to represent fine-grained features aiding relation predictions. DocRE-CLiP frames DocRE as a link prediction problem combining link prediction with contextual knowledge and achieves state-of-the-art performance [20]. DocRE-CLiP is the leading graph-based model but cannot be reproduced due to the absence of specific files in the shared GitHub repository. The other graph-based models are outperformed by the leading sequence-based model; hence, we do not consider them in our evaluation. Moreover, graph-based methods cannot effectively scale to the size of distantly supervised datasets.

Sequence-based models treat documents as a sequence of tokens and learn contextual representations for all entity pairs. Early approaches using CNNs and LSTMs struggled with long-range dependencies [1]. Transformer-based models leveraging contextual embeddings, such as BERT, markedly improve the task and are now standard in DocRE [21].

A key transformer-based model is **ATLOP**, which introduces adaptive thresholding for entity-dependent multi-label classification, and localized context pooling for improved predictions [8]. KD-DocRE extends ATLOP by addressing class imbalance via adaptive focal loss and leveraging knowledge distillation in a teacher-student paradigm [22]. Wang et al. [19] frames DocRE as a metric learning problem to learn similar representations for entity embeddings and ground truth relations. To enhance robustness, Duan et al. [23] proposes COMM, a two-stage model employing Concentrated Margin Maximization to adaptively adjust decision boundaries. COMM is an enhancement strategy that can be applied to any off-the-shelf DocRE models. Thus, this work is orthogonal to DOREMI and can be integrated into the pipeline as future work. **DREEAM** follows the same architecture as KD-DocRE and introduces an evidence-aware distillation mechanism that better guides learning using noisy DS data, achieving state-of-the-art performance in the task [9]. Other transformer-based or evidence-oriented models, such as SAIS [24] and EIDER [25], exhibit inferior results and suffer from scalability issues due to reliance on manual datasets. As DREEAM has shown superior performance over all these systems, we do not consider them as baselines for the experimental evaluation.

The long-tail problem in DocRE has been addressed by three methods: ERA [26], PRISM [5], DocRE-Co-Occur [4]. ERA leverages a relation augmentation mechanism to enhance the entity pair representation by applying a random mask on pooled context representation. However, ERA shows inferior performance compared to DREEAM and long-tail results are inferior to DocRE-Co-Occur. PRiSM is a calibration-based approach that learns to adapt logits based on relation and entity pairs' embeddings similarity; it improves 0.5% on ATLOP and is inferior to DREEAM. DocRE-Co-Occur introduces the concept of relation correlations, leveraging the co-occurrence patterns of relations to transfer knowledge from data-rich to data-scarce relations. By modeling these correlations through relation embeddings and incorporating auxiliary co-occurrence coarse and fine-grained predictions, DocRE-Co-Occur enhances the model's ability to extract rare relations. However, the performance of DocRE-Co-Occur is inferior to that of DREEAM, even considering only long-tail relations. .

Addressing noise in DS datasets is critical and could help balance long-tail relations via the injection of additional training data. Xiao et al. [27] mitigate noise using multi-task learning across mention-entity matching, relation detection, and fact alignment. Zhang et al. [28] focuses on relational reasoning and proposes a self-distillation framework with a multihead at-

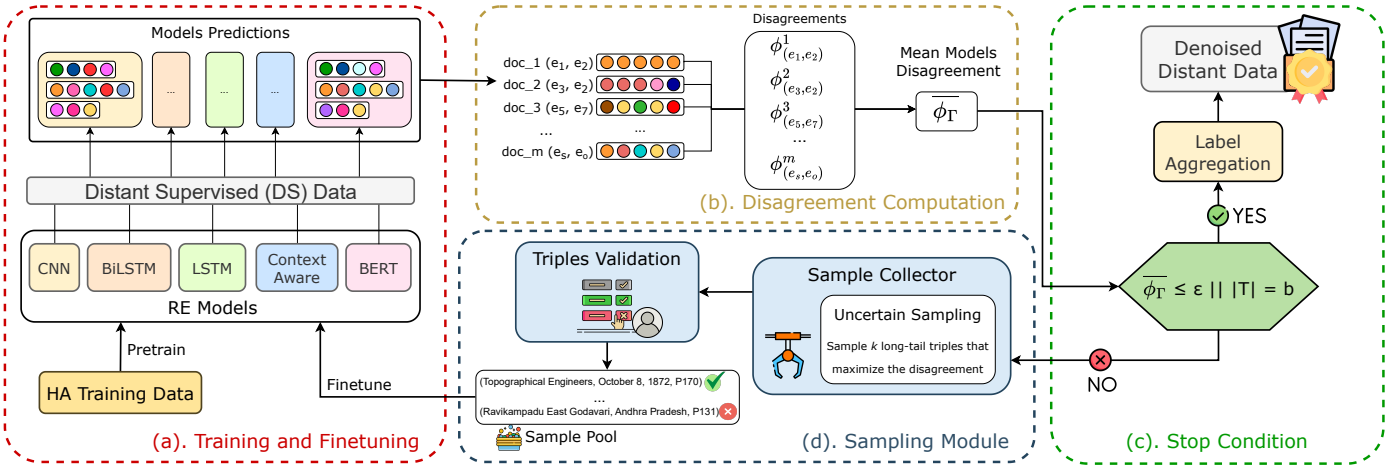


Figure 4: The DOREMI architecture including: (a) *Training and Finetuning* of five models on long-tail relations; (b) *Disagreement Computation* for each entity pair; (c) *Stop Condition* checks if the disagreement is below a threshold or the annotation budget is exhausted; (d) *Sampling Module* samples k high-disagreement triples to be annotated.

tention unit that models common reasoning patterns. Li et al. [29] proposes integrating a LLM and a Natural Language Inference (NLI) module to generate relation triples and perform data augmentation. The current state-of-the-art and best-perfoming approach in label denoising is **UGDRE**, which applies Monte Carlo dropout to estimate pseudo-label uncertainty, selectively filtering low-confidence labels to improve training data quality [6].

In summary, we focus on **ATLOP**, **DREEAM**, and **UGDRE** as our primary baselines due to their strong performance, scalability, and representativeness of the best practices in modeling and data denoising. Other approaches are either incorporated within these models or perform worse and are therefore omitted from further comparison.

3. DOREMI

DOREMI aims to enhance the prediction of long-tail relations by selectively enriching the training data with minimal human-annotated triples. Given the high cost of expert annotations, we annotate as few examples as possible while maximizing their impact on improving long-tail performance. Figure 4 shows the main computational blocks involved in DOREMI. Table 1 reports the main symbols and notation used throughout the paper.

DOREMI maintains a pool Γ of n diverse DocRE core models, each fine-tuned on the available Human-Annotated (HA) training data (Figure 4a). These models then predict relations over the noisy DS dataset and, for each entity pair (s, o), these predicts are used to compute the disagreement ($\phi_\Gamma(s, o)$) among models. The disagreement across all pairs is then averaged $\overline{\phi}_\Gamma$ and evaluated against a stopping criterion (Figure 4b). Under the hypothesis that higher inter-model disagreement indicates greater annotation utility, we stop annotating when $\overline{\phi}_\Gamma \leq \varepsilon$ or when the annotation budget b is over. At that point, we aggregate model outputs into the final DDS (Figure 4c). Otherwise, we sample the k pairs with maximum disagreement for manual labeling, augment the training set, and repeat the cycle of model training and uncertainty-driven sampling (Figure 4a,d).

Γ	Set of core models
γ_i	i-th core model in Γ
$\phi_\Gamma(s, o)$	Disagreement between models in Γ for the entity pair (s, o)
$\overline{\phi}_\Gamma$	Mean disagreement between models in Γ for all entity pairs in the dataset
ϵ	Disagreement threshold
k	Sample size
b	Annotation budget
τ	Label aggregation threshold

Table 1: Notation table. We report the symbols used throughout the paper and in the figures to define the key elements of DOREMI.

This section is organized as follows. Subsection 3.1 details how disagreement is computed in DOREMI to select entity pairs for annotation and to determine the stopping condition. Subsection 3.2 defines the label aggregation criteria employed by DOREMI to build the DDS in the final iteration. To conclude, Subsection 3.3 describes the iterative training procedure followed by DOREMI.

3.1. Disagreement Computation

In the following, we define how we compute the disagreement between models in DOREMI. The disagreement between models is employed to assess the stopping condition (Figure 4c) and select the entity pairs to annotate (Figure 4d). Le et al. [10] demonstrated that sampling by model disagreement outperforms single-model confidence sampling. Inspired by this finding, we posit that inter-model disagreement effectively identifies Hard-To-Classify examples in DocRE. Below, we formalize our disagreement measure using prediction probabilities produced by the models.

Let $\Gamma = \{\gamma_i\}_{i=1}^n$ be a set of n independently trained DocRE models. For a candidate triple (s, r, o), denote by $\mathbb{1}_{\gamma_i}(r|(s, o))$

the indicator function returning 1 if model γ_i predicts relation r between s and o, and 0 otherwise. Conversely, we write $\mathbb{1}_{\gamma_i'}(r|(s,o)) = 1 - \mathbb{1}_{\gamma_i}(r|(s,o))$ for the event "no relation r". When considering a single relation r, the models agree if they all predict r or all predict "no relation r". Therefore, the probability of disagreement on r can be computed as:

$$\phi_\Gamma(r|(s,o)) = 1 - \Big[P\Big(\bigcap_{i=1}^{n} \mathbb{1}_{\gamma_i}(r|(s,o))\Big) + P\Big(\bigcap_{i=1}^{n} \mathbb{1}_{\gamma_i'}(r|(s,o))\Big)\Big] \quad (1)$$

Given that model predictions are independent, the probability that all models predict the relation can be expressed as the product of the individual model probabilities.

We denote $P(\mathbb{1}_{\gamma_i}(r|(s,o)))$ as $p_{\gamma_i}(r|(s,o)) = P(\gamma_i \text{ predicts relation } r \mid s,o)$. Thus, the probability that a model does not predict the relation is $1 - p_{\gamma_i}(r|(s,o))$. Substituting into Equation 1, the disagreement is

$$\phi_\Gamma(r \mid (s,o)) = 1 - \Big[\prod_{i=1}^{n} p_{\gamma_i}(r|(s,o)) + \prod_{i=1}^{n} \big(1 - p_{\gamma_i}(r|(s,o))\big)\Big] \quad (2)$$

In DocRE, each of the R target relations may hold independently for a given entity pair (s,o), resulting in a multi-label classification problem. Assuming independence across relations, the overall disagreement on the relation set R for the entity pair (s,o) is computed as the product of the individual per-relation disagreements:

$$\phi_\Gamma(s,o) = \prod_{r \in R} \phi_\Gamma(r|(s,o)) \quad (3)$$

Since the disagreement $\phi_\Gamma(r|(s,o))$ lies in $[0,1]$, we apply a logarithmic transformation to amplify small differences and enhance interpretability. To ensure the logarithm is well-defined, we add a small, scale-invariant constant $\delta > 0$ to shift the values away from zero.[1] This transformation maps disagreement values into the range $(-\infty, \delta]$, with complete disagreement corresponding to δ:

$$\psi_\Gamma(s,o) = \sum_{r \in R} \log \phi_\Gamma(r|(s,o)) \quad (4)$$

Based on it, we define the sampling criterion for Hard-To-Classify instances as selecting the top-k entity pairs with the highest disagreement:

$$\arg\max_{(s,o)} \psi_\Gamma(s,o) \quad (5)$$

This strategy prioritizes entity pairs (s,o) for which the model ensemble Γ exhibits the highest uncertainty, thereby maximizing the expected benefit of manual annotation for each selected instance.

[1] δ can be arbitrarily small; its value does not affect the approach. Hence, we omit δ from the equations.

Since DOREMI is tailored for target long-tail relations, the disagreement computation is limited to *candidate long-tail triples* – that is, (s,o) pairs for which at least one core model predicts a long-tail relation. This focus ensures that annotation efforts are directed toward instances with the greatest potential to improve long-tail predictions.

3.2. Denoised Distant Dataset Construction

Once the stopping condition is satisfied, we aggregate the core models predictions to construct the denoised distant dataset (figure 4c). This subsection describes the label aggregation criteria exploited by DOREMI. The overall mean disagreement $\overline{\phi}_\Gamma$ is computed by averaging $\phi_\Gamma(s,o)$ over all candidate pairs in the distantly supervised dataset. Once $\overline{\phi}_\Gamma \leq \varepsilon$ or the annotation budget b is exhausted, the active learning loop ends. We then aggregate predictions from the best iteration of each model – based on the highest long-tail F1 on the development set – to construct the DDS.

To maximize coverage while maintaining label quality in the DDS, we adopt a selective retention strategy that filters out highly uncertain relations. Specifically, for each entity pair (s,o) and relation r, we include r in the final label set only if at least one model predicts it with high confidence:

$$\max_{i=1,\ldots,n} p_{\gamma_i}(r|(s,o)) > \tau,$$

where τ is a confidence threshold. This approach acts as a precision-preserving filter, ensuring that only relations supported by strong evidence from at least one model are retained. As a result, we balance broad relational coverage with robustness, reducing noise without overly sacrificing recall. Thanks to this approach, we managed to retain around 60% of the DDS instances, while discarding the remaining.

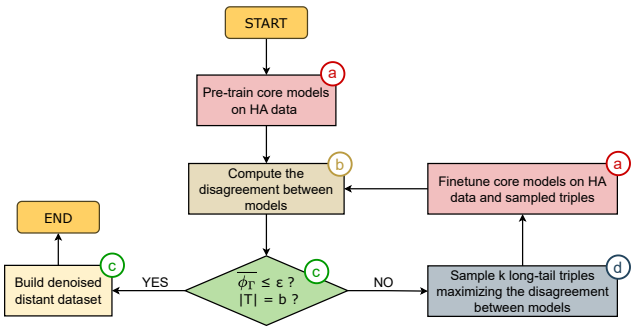


Figure 5: DOREMI flowchart. Each process or decision is linked to the corresponding computational blocks defined in Figure 4.

3.3. Iterative Training Procedure

This subsection details the iterative training procedure followed by DOREMI. The pseudocode is reported in Algorithm 1. To enhance clarity, Figure 5 reports the flowchart of DOREMI. In input, DOREMI takes the human-annotated (HA) and distantly supervised (DS) training data, a disagreement threshold (ε) for the stopping criterion, a set of DocRE

Algorithm 1 Iterative training for long-tail DocRE

Input: Human-annotated training data HA, distant training data DS, disagreement threshold ε, DocRE models $\Gamma = \gamma_1, .., \gamma_n$, sample size k, budget b

Output: Denoised distantly supervised data DDS

1: **for each** γ_i **in** Γ **do** ▷ Pretrain the models (see Figure 4a)
2: $\gamma_i^{pretrain} \leftarrow \texttt{train}(\gamma_i, HA)$
3: $\gamma_i^{prev} \leftarrow \gamma_i^{pretrain}$
4: $DDS^i \leftarrow \texttt{predict}(\gamma_i^{pretrain}, DS)$
5: **end for** ▷ Disagreement computation (see Figure 4b)
6: $\overline{\phi}_\Gamma \leftarrow \texttt{mean}(\{\phi_\Gamma(s,o) : (s,o) \in DS\}), T \leftarrow \emptyset$
7: **while** $\overline{\phi}_\Gamma > \varepsilon$ **and** $|T| < b$ **do** ▷ Loop until stop condition is satisfied (see Figure 4c)
8: $S \leftarrow$ Sample k candidate long-tail triples with highest disagreement from $\{DDS^1, .., DDS^n\}$ ▷ Sampling and annotation (see Figure 4d)
9: $SA \leftarrow \texttt{annotate}(S)$
10: $T \leftarrow T \cup SA$
11: **for each** γ_i **in** Γ **do** ▷ Finetune the models (see Figure 4a)
12: $\gamma_i^{finetune} \leftarrow \texttt{finetune}(\gamma_i^{prev}, HA \cup T)$
13: $\gamma_i^{prev} \leftarrow \gamma_i^{finetune}$
14: $DDS^i \leftarrow \texttt{predict}(\gamma_i^{finetune}, DS)$
15: **end for** ▷ Disagreement computation (see Figure 4b)
16: $\overline{\phi}_\Gamma \leftarrow \texttt{mean}(\{\phi_\Gamma(s,o) : (s,o) \in DS\})$
17: **end while** ▷ Stop condition reached (see Figure 4c)
18: **for each** γ_i **in** Γ **do**
19: $DSS^i \leftarrow \texttt{predict}(\gamma_i^{finetune}, DS)$
20: **end for**
21: $DDS \leftarrow \texttt{merge}(DDS^1, .., DDS^n)$ ▷ Label aggregation (see Figure 4c)

models $\Gamma = \{\gamma_i\}_{i=1}^n$, a sample size ($k$), and an annotation budget (b).

In lines 1-5, each DocRE model γ_i is pretrained on the HA training set, stored, and used to label the DS data. The predictions are then used to compute the model disagreement for each entity pair (s, o), which in turn is used to calculate the preliminary mean disagreement $\overline{\phi}_\Gamma$ (line 6), averaging over all the pair disagreements between the models.

Iterations (lines 7-17) continue as long as the average disagreement among the models is above threshold ($\overline{\phi}_\Gamma > \varepsilon$) and annotation budget is available ($|T| < b$). Specifically, DOREMI samples k candidate long-tail triples based on the highest disagreement (line 8), which are then annotated and stored in the sample pool T (lines 9,10). Each model γ_i is fine-tuned on the combination of HA input data and newly annotated data T (line 12). The fine-tuned model is stored to serve as a starting point for the next iteration (line 13). Predictions are made on the DS data using the fine-tuned model, resulting in the set DSS^i (line 14). Once predictions are obtained from all models, they are used to compute the model disagreement for each entity pair, as well as the *current iteration* average disagreement (line 16).

After the loop concludes, each model γ_i makes a final round of predictions on DS data (line 19). These predictions are aggregated across models to generate the DDS (line 21).

4. Experimental Setup

This section describes the experimental setup exploited to evaluate DOREMI. In particular, Subsection 4.1 reports the core models used to predict the relations in the noisy DS data and finetuned at each iteration. Subsection 4.2 reports the dataset used for training and testing DOREMI. To conclude, Subsection 4.3 describes the baselines chosen to evaluate the effectiveness of DOREMI denoising.

The implementation of DOREMI is available in GitHub at: `https://github.com/mntlra/DOREMI`. Together with the source code, we release some Python scripts to reproduce the DOREMI iterative training procedure. Table 2 summarizes the hyperparameters used in DOREMI.

(a) Training with DocRED	
Sample size k	100
Annotation budget b	400
(b) Training with Re-DocRED	
Sample size k	300
Annotation budget b	1,200
(c) Common parameters	
Label aggregation threshold τ	0.7

Table 2: DOREMI experimental setting. The table summarizes the hyperparameters exploited in DOREMI.

4.1. Core Models

We chose five core models ($n = 5$) for DOREMI due to two primary reasons. Firstly, using an odd number of models avoids tie cases during aggregation. Secondly, we adopted the core models originally used in DocRED [1], the first benchmark dataset for DocRE. Specifically, DocRED employs CNN, LSTM, BiLSTM, and ContextAware (based on word2vec). the code to train the core models exploited in DocRED is publicly accessible in GitHub. [2] Given the widespread adoption of contextual embeddings, in particular BERT [21], and their strong capability to model long-range contextual dependencies for relation prediction, we incorporated it into DOREMI as the fifth core model. We developed our own core model exploiting BERT. the implementation is based on DREEAM [3] and ATLOP [4] GitHub repositories. A higher number of models serves a proxy to enhance diversity between the models, as in the case of $n = 5$. We decided to not include more recent models tailored for DocRE because most of them rely on either BERT or RoBERTa as the backbone transformer architecture [4, 5, 8, 9, 22, 23, 24, 25]. Thus, we do not expect these models to meaningfully increase diversity, as they are likely to exhibit score distributions similar to BERT (reported in Figure 6).

[2] `https://github.com/thunlp/DocRED/`
[3] `https://github.com/YoumiMa/dreeam`
[4] `https://github.com/wzhouad/ATLOP`

During training, we followed the same hyperparameters adopted in [1]. Table 3 reports the experimental setting used in DOREMI. All baselines were optimized using Adam. For the BERT core mode, we adopted the same hyperparameter settings of DREEAM [9].

(a) DocRED core models	
Batch size	40
Pre-training epochs	200
Finetuning epochs	70
Learning rate	0.001
CNN hidden size	200
CNN window size	3
CNN dropout rate	0.5
LSTM hidden size	128
LSTM dropout rate	0.2
Word embedding dimension	100
Entity type embedding dimension	20
Coreference embedding dimension	20
Distance embedding dimension	20
(b) BERT core model	
Batch size	4
Pre-training epochs	30
Finetuning epochs	10
Transformer learning rate	3e-5
Classifier learning rate	1e-4
Warmup ratio	0.06
Maximum norm of the gradients	1.0

Table 3: Core models hyperparameter settings. We employed the same training setting as DocRED [1] for CNN, BiLSTM, LSTM, and ContextAware. the hyperparameters for BERT are taken from DREEAM implementation [9].

4.2. Datasets

We used DOREMI with DocRED [1] and Re-DocRED [3], a polished version of the former. Despite known annotation issues in DocRED, it remains a widely used benchmark [7, 30, 31, 32, 33]. In the DocRED configuration, the core models are iteratively trained starting from the DocRED annotated training set, with the optimal iteration selected based on the highest long-tail F1 evaluated on the DocRED development set. The test dataset of DocRED is not publicly accessible; the dataset can be evaluated by submitting the models predictions to CodaLab [5], which solely reports the relation extraction F1 and evidence F1 on the test set. Thus, we cannot perform a long-tail evaluation exploiting it. Given this dataset limitation, previous works on DocRE report the final performance on both the DocRED development and test datasets [4, 5, 8, 9, 22, 23, 24, 25, 30, 33]. Thus, we decided to perform a detailed evaluation on the DocRED development dataset and not change the dataset splitting to ensure compatibility with previous works. In this way, DOREMI results can be compared to any existing DocRE model exploiting DocRED. We also evaluated our approach on the DocRED test dataset and included the micro-averaged F1 score on the full dataset.

We define long-tail relations as those with less than 100 instances in the training set (cf. Figure 2a), and set the sampling size $k = 100$. To keep annotation costs minimal, we annotated only the 0.001% ($b = 400$) of the distantly supervised dataset. Therefore, at each iteration, the fine-tuning dataset grows by 0.3%. For Re-DocRED, we define long-tail relations as those having less than 300 training instances (cf. Figure 2b). Since Re-DocRED contains roughly three times more triples than DocRED – 85,932 vs 38,180 – we scale both k and b proportionally, setting $k = 300$ and $b = 1,200$ (0.003%). This proportional scaling ensures that the fine-tuning dataset grows by 0.3% as for DocRED.

Figure 6 reports the score distribution of the distant triple predicted by the core models trained using DocRED (Figure 6a) and Re-DocRED (Figure 6b). The distribution is skewed towards high confidence for both datasets, with most models exhibiting a mean confidence score (μ) around 0.7. Based on this insight, we set the threshold $\tau = 0.7$ to retain only relations strongly supported by at least one model. This choice of τ ensures a robust balance between coverage and noise reduction, effectively acting as a precision-preserving filter without overly sacrificing recall.

Using this threshold during the aggregation step, DOREMI retains 65.5% of predicted relations on DocRED, producing a dataset with 1,704,471 triples. On Re-DocRED, it preserves 61.5% of relations, yielding 4,158,468 triples.

4.3. Baselines

To assess the effectiveness of our DDS, we train state-of-the-art DocRE models from scratch on three datasets: (i) the original DocRED DS dataset [1], (ii) the denoised UGDRE dataset [6], and (iii) the DOREMI DDS (ours). DOREMI aims to improve performance in long-tail relations; hence, it can complement approaches focusing on frequent relations, such as UGDRE. To demonstrate this, we evaluate each model on a hybrid dataset, denoted D+U, which combines DOREMI long-tail predictions with UGDRE frequent relations annotations.

We adopt two transformer-based architectures as baselines: ATLOP [8] and DREEAM [9]. Both approaches support multiple transformer backbones; we report results using BERT-base and RoBERTa-large. DREEAM follows a teacher–student paradigm, where the student model is first trained on DS data (called "*self-training*") and then fine-tuned on human-annotated data. Since our focus is on evaluating the quality of distantly supervised datasets, we report the performance of the DREEAM student model right after self-training. These models have been selected due to their strong empirical performance and the availability of open-source implementations (cf. Section 2). The implementation of ATLOP [6] and DREEAM [7] is publicly available in GitHub. Thus, we train the two models from

[5] https://codalab.lisn.upsaclay.fr/competitions/365

[6] https://github.com/wzhouad/ATLOP

[7] https://github.com/YoumiMa/dreeam

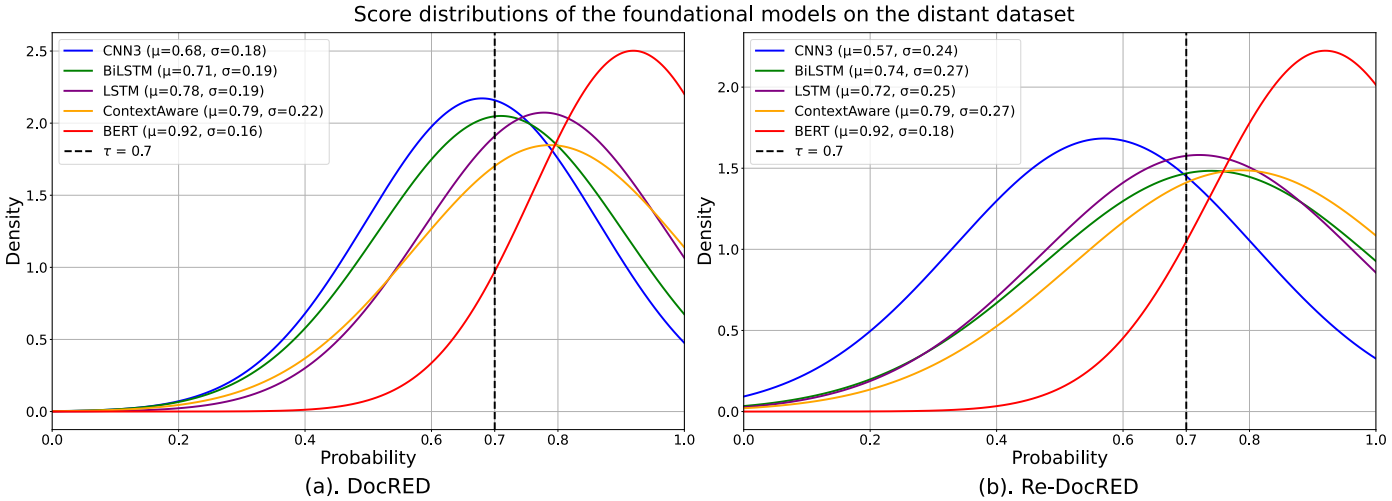


Figure 6: Score distributions of the core models trained using DocRED (a) and Re-DocRED (b) in the DocRED distant dataset.

scratch, leveraging the experimental setup and hyperparameter settings of the corresponding reference papers. We share the scripts to train DREEAM and ATLOP using different distant datasets in the `repro` directory of our GitHub repository. [8] Although UGDRE [6] also provides a competitive DocRE model along with the denoised dataset, we could not reproduce their results due to deprecated dependencies. Model performance is reported using micro-averaged metrics, as this is the standard evaluation approach commonly adopted in DocRE literature.

5. Experimental Results

This section reports DOREMI results compared to a state-of-the-art approach for denoising DS data. In particular, subsection 5.1 reports the performance of ATLOP and DREEAM evaluated on the DocRED development dataset. In this configuretion, DOREMI is trained and finetuned on the DocRED training dataset. Subsection 5.2 reports the performance of ATLOP and DREEAM evaluated on the Re-DocRED test dataset. In this configuretion, DOREMI is trained and finetuned on the Re-DocRED training dataset. Since Re-DocRED is three times larger than DocRED, we consider long-tail relations those having less than 300 training examples. Relations having less than 100 training examples represent the "*extreme long-tail*". Extreme long-tail performances are reported in Subsection 5.2.1.

5.1. DocRED

Table 4 reports the performance of ATLOP and DREEAM trained on different distant datasets and evaluated on the DocRED development (dev) dataset. We rely on the dev dataset because the test set is not publicly available and thus prevents long-tail analysis.

When trained on the DDS generated by DOREMI, all DocRE models exhibit superior performance – both overall and on long-tail relations. Compared to UGDRE, the current state-of-the-art in distant label denoising, DOREMI improves overall PRECISION by up to +8.2% and F1 by up to +0.7%. The improvement of DOREMI over UGDRE in terms of F1 score is statistically significant for three out of four models, as determined by a paired t-test ($p < 0.01$). The overall improvement of DOREMI over UGDRE in terms of F1 score is confirmed by the performance on the DocRED test dataset (column "Test-F1" of Table 4). In long-tail prediction, DOREMI achieves a PRECISION of 43.3%, representing a relative gain of +76.0% over UGDRE with ATLOP-RoBERTa, and +63.1% with DREEAM-RoBERTa. We further evaluate performance using IGNPRECISION and IGNF1, which exclude entity pairs observed during training. On these metrics, DocRED and UGDRE exhibit steep precision drops (nearly −50%) on unseen long-tail triples. In contrast, DOREMI delivers two to three times higher IGNPRECISION than UGDRE, scoring a performance gain of up to +137.6%.

Remarkably, these improvements are attained with 400 manual annotations – just **0.001%** of all candidate pairs – underscoring the efficiency of DOREMI in constructing a high-quality corpus with minimal human effort.

Finally, in a hybrid configuration (D+U) that combines DOREMI labels for rare relations with UGDRE labels for frequent ones, D+U matches or exceeds UGDRE in most metrics in both the DocRED dev and test dataset. These findings indicate that iterative, disagreement-driven labeling yields stronger denoising capabilities than existing approaches and can be seamlessly integrated with existing frequency-based techniques.

5.2. Re-DocRED

Table 5 presents the performance of ATLOP and DREEAM trained on different distant datasets and evaluated on the Re-DocRED test dataset. Models trained with the DOREMI DDS consistently outperform those employing the original DocRED DS dataset and the UGDRE one in terms of long-tail PRECISION and *ignored* precision (IGNPRECISION) – regardless of model architecture (ATLOP or DREEAM) and transformer backbone

[8] `https://github.com/mntlra/DOREMI/tree/main/repro`

Table 4: Performance of DocRE models trained on distant datasets: DocRED, UGDRE, DOREMI **(ours)**, and D+U (DOREMI for long-tail, UGDRE for frequent relations). Results are on the **DocRED dev set**, except for column "Test-F1", which reports the micro-F1 for the full dataset on the **DocRED test set**. Long-tail refers to relations with < 100 training instances. Best and second-best results are bolded and underlined.

Model	Distant Data	Full Dataset						Long-Tail Triples				
		Test-F1	Precision	Ign Prec	Recall	F1	Ign F1	Precision	Ign Prec	Recall	F1	Ign F1
ATLOP BERT	DocRED	0.5326	0.5139	0.3346	0.5725	0.5416	0.4223	0.1917	0.0857	0.3839	0.2557	0.1401
	UGDRE	0.5797	0.5362	0.3711	**0.6725**	0.5967	0.4783	0.2332	0.1293	**0.4013**	**0.2950**	0.1956
	DOREMI	0.5883	**0.5722**	**0.4384**	0.6160	0.5933	**0.5123**	0.3634	0.2701	0.1946	0.2535	0.2262
	D+U	**0.5901**	0.5600	0.3930	0.6416	**0.5980**	0.4874	**0.3807**	**0.2744**	0.2336	0.2895	**0.2523**
ATLOP RoBERTa	DocRED	0.5409	0.5094	0.3345	0.5912	0.5472	0.4272	0.1913	0.0876	0.4067	0.2602	0.1442
	UGDRE	0.5840	0.5350	0.3746	**0.6810**	0.5992	0.4834	0.2387	0.1399	**0.4604**	**0.3144**	0.2146
	DOREMI	0.5909	**0.5788**	**0.4453**	0.6242	0.6006	**0.5198**	**0.4200**	**0.3324**	0.2362	0.3024	**0.2762**
	D+U	**0.5998**	0.5657	0.4063	0.6528	**0.6061**	0.5009	0.3713	0.2737	0.2188	0.2753	0.2432
DREEAM BERT	DocRED	0.5306	0.5602	0.3905	0.5036	0.5304	0.4399	0.1714	0.0862	0.2443	0.2013	0.1274
	UGDRE	0.5966	0.5915	0.4366	0.6138	0.6025	0.5102	0.2649	0.1652	**0.3154**	0.2880	0.2168
	DOREMI	**0.6095**	0.5765	0.4433	**0.6341**	0.6039	**0.5218**	0.3928	0.3058	0.2483	0.3043	0.2741
	D+U	0.6011	**0.6108**	**0.4578**	0.5991	**0.6049**	0.5190	**0.4387**	**0.3444**	0.2497	**0.3182**	**0.2895**
DREEAM RoBERTa	DocRED	0.5459	0.5582	0.3890	0.5370	0.5474	0.4512	0.1834	0.0965	0.3047	0.2289	0.1466
	UGDRE	0.6064	0.5853	0.4335	0.6481	0.6151	0.5195	0.2656	0.1723	**0.3544**	0.3036	0.2319
	DOREMI	**0.6198**	0.5904	**0.4564**	**0.6513**	**0.6194**	**0.5367**	**0.4332**	**0.3492**	0.2523	**0.3189**	**0.2930**
	D+U	0.6144	**0.6090**	**0.4564**	0.6287	0.6187	0.5289	0.4226	0.3261	0.2456	0.3107	0.2802

Table 5: Performance of DocRE models trained on distantly supervised datasets: DocRED, UGDRE, DOREMI **(ours)**, and D+U (DOREMI for long-tail, UGDRE for frequent relations). Results are on the **Re-DocRED test set**; long-tail refers to relations with < 300 training instances. Best and second-best results are bolded and underlined.

Model	Distant Data	Full Dataset					Long-Tail Triples				
		Precision	Ign Prec	Recall	F1	Ign F1	Precision	Ign Prec	Recall	F1	Ign F1
ATLOP BERT	DocRED	0.7353	0.5793	0.3000	0.4261	0.3953	0.3932	0.2610	0.2710	0.3209	0.2659
	UGDRE	0.8044	0.7445	**0.7007**	**0.7490**	**0.7220**	0.6034	0.5280	**0.5513**	**0.5762**	**0.5394**
	DOREMI	0.7480	0.6297	0.6914	0.7186	0.6591	**0.6741**	**0.6061**	0.4090	0.5091	0.4884
	D+U	**0.8340**	**0.7779**	0.6580	0.7356	0.7129	0.6087	0.5260	0.4699	0.5303	0.4963
ATLOP RoBERTa	DocRED	0.7412	0.6013	0.3241	0.4510	0.4212	0.4029	0.2885	0.3064	0.3480	0.2972
	UGDRE	0.8090	0.7502	**0.7087**	**0.7555**	**0.7288**	0.5994	0.5248	**0.5600**	**0.5791**	**0.5419**
	DOREMI	0.7486	0.6311	0.7023	0.7247	0.6648	**0.6715**	**0.6112**	0.4020	0.5029	0.4850
	D+U	**0.8264**	**0.7707**	0.6888	0.7513	0.7275	0.6049	0.5326	0.5182	0.5582	0.5253
DREEAM BERT	DocRED	0.8087	0.6791	0.2609	0.3945	0.3770	0.4429	0.3166	0.1939	0.2697	0.2405
	UGDRE	0.8265	0.7740	0.6966	**0.7560**	**0.7333**	0.6364	0.5729	**0.5182**	**0.5713**	**0.5442**
	DOREMI	0.7701	0.6576	**0.7037**	0.7354	0.6799	**0.7250**	**0.6640**	0.4210	0.5326	0.5153
	D+U	**0.8562**	**0.8086**	0.6433	0.7347	0.7166	0.6642	0.5992	0.4340	0.5250	0.5034
DREEAM RoBERTa	DocRED	0.8170	0.6964	0.2861	0.4238	0.4056	0.4672	0.3444	0.2357	0.3134	0.2799
	UGDRE	0.8464	0.7996	0.7252	**0.7811**	**0.7606**	0.6517	0.5901	**0.5530**	**0.5983**	**0.5709**
	DOREMI	0.7939	0.6894	**0.7267**	0.7588	0.7075	**0.7575**	**0.7032**	0.4498	0.5644	0.5486
	D+U	**0.8766**	**0.8354**	0.6738	0.7619	0.7459	0.6936	0.6349	0.4807	0.5679	0.5472

(BERT or RoBERTa). Compared to the UGDRE denoised dataset, training with the DOREMI DDS reaches a PRECISION of 75.75%, representing a gain of 10.58 points over UGDRE (65.17%) with DREEAM-RoBERTa. Pronounced improvements are observed in the *ignored* metrics, which exclude entity pairs observed during training. Compared to UGDRE, DOREMI achieves a +19.2% gain in long-tail IGNPRECISION, highlighting its superior capability of generalizing to new unseen entity pairs.

Considering the full dataset, DOREMI achieves the highest recall on DREEAM for both transformer backbones, registering a statistically significant improvement of up to +1.0% compared to UGDRE ($p < 0.01$). In the hybrid D+U setting, top performance is obtained for full precision-related metrics (+3.7% on UGDRE), indicating that DOREMI disagreement-driven supervision effectively complements existing denoising approaches.

These gains are achieved by annotating just **0.003%** of the distant dataset, underscoring the efficiency of DOREMI in producing high-quality labels at minimal annotation cost. The number of triples annotated to achieve these results is $1,200$. This amounts to roughly 20 annotator-hours of work, based on the annotation rate of one minute per triple that we observed during the annotation phase. This highlights its potential to sig-

Table 6: Micro-averaged performance on extreme long-tail prediction of DocRE models trained on distant datasets: DocRED, UGDRE, DOREMI **(ours)**, and D+U (DOREMI for long-tail, UGDRE for frequent relations). Results are on the **Re-DocRED test set**; extreme long-tail refers to relations with < 100 training instances. Best and second-best results are bolded and underlined.

Model	Distant Data	Precision	IgnPrec	Recall	F1	Ign F1
ATLOP BERT	DocRED	0.2532	0.1016	0.2977	0.2737	0.1515
	UGDRE	0.3333	0.1940	**0.4122**	0.3683	0.2639
	DOREMI	**0.5806**	**0.4902**	0.2748	**0.3731**	**0.3522**
	D+U	0.5161	0.4208	0.2443	0.3316	0.3097
ATLOP RoBERTa	DocRED	0.2711	0.1295	0.3435	0.3030	0.1881
	UGDRE	0.3253	0.1765	**0.4122**	0.3636	0.2471
	DOREMI	**0.5962**	**0.5435**	0.2366	0.3388	0.3297
	D+U	0.4756	0.3582	0.2977	**0.3662**	**0.3252**
DREEAM BERT	DocRED	0.3538	0.1923	0.1756	0.2347	0.1836
	UGDRE	0.3929	0.2917	**0.3359**	0.3621	0.3122
	DOREMI	0.4921	0.4483	0.2366	0.3196	0.3098
	D+U	**0.5606**	**0.4821**	0.2824	**0.3756**	**0.3562**
DREEAM RoBERTa	DocRED	0.3085	0.1667	0.2214	0.2578	0.1902
	UGDRE	0.3966	0.3069	**0.3511**	**0.3725**	0.3276
	DOREMI	**0.5862**	**0.4783**	0.2595	0.3598	**0.3365**
	D+U	0.5636	0.4545	0.2366	0.3333	0.3112

nificantly improve DocRE and KBC. The results for *extreme long-tail triples*, defined as relations with less than 100 examples in the Re-DocRED training dataset, reported in the next section, confirm the same trend.

5.2.1. Extreme Long-Tail Relations

Re-DocRED long-tail triples have less than 300 examples in the annotated training dataset. This section investigates the impact of the DOREMI DDS in the *extreme long-tail triples*, i.e., triples with less than 100 examples in the Re-DocRED annotated training dataset. As shown in Table 6, DOREMI shows superior results compared to the DocRED DS dataset and UGDRE, especially in PRECISION (+83.2% compared to UGDRE). When combined with UGDRE, i.e. dataset D+U, the dataset outperforms DOREMI when considering DREEAM with BERT-base as a backbone and reports superior performance compared to the DocRED DS dataset and UGDRE. The findings validate that DOREMI can be effectively integrated with various systems to enhance performance further.

DOREMI proves particularly effective on the *ignored* metrics, which evaluate only pairs unseen by the models during training. Compared to UGDRE on long-tail prediction, DOREMI achieves a +207.9% relative increase in IGNPRECISION and a +33.5% gain in IGNF1. ATLOP-BERT more than doubles UGDRE's IGNPRECISION on unseen long-tail predicitions, while ATLOP-RoBERTa more than triples it.

6. Discussion

This section investigates different strategies to select meaningful training examples to annotate, identifying the most suitable approach for our study (Subsection 6.1). Subsection 6.2 describes the effect of setting the annotation budget $b = 400$ for DocRED and motivates the infeasibility of a sensitivity analysis for DOREMI. In addition, we demonstrate the effectiveness of our disagreement-based sampling procedure by reporting the number of long-tail examples and N/A triples detected (Subsection 6.3).

6.1. Sampling Strategies

We compare our disagreement-based strategy with *Max-Entropy*, one of the most common sampling techniques [34]. To establish the best solution to represent the disagreement between n DocRE models, we also consider a few alternatives to Equation 5. We introduce a selection criterion based on the disagreement between relations called *Positive Probability Mean (PPM)*. Instead of a probabilistic approach, PPM considers the mean disagreement across all relations as a proxy to compute the disagreement between the models. In this scenario, the probability of disagreement on a given relation r is computed as one minus the product of the probability that a given model predicts relation r for the pair (s, o), which we defined as $p_{\gamma_i}(r|(s, o))$ in Section 3.1:

$$\phi_\Gamma(r|(s,o)) = 1 - \prod_{i=1}^{n} p_{\gamma_i}(r|(s,o)) \tag{6}$$

The sampling criterion selects the top-k entity pairs with the highest mean probability of disagreement across all relations:

$$\arg\max_{(s,o)} \frac{\sum_{r\in R} \phi_\Gamma(r|(s,o))}{|R|} \tag{7}$$

DOREMI defines the probability of disagreement on a given relation r using both the probability that a model predicts the relation and the probability that the model does not predict the relation (cfr. Section 3.1). Instead, in *Positive Probability Disagreement (PPD)* the probability of disagreement only considers the probability that a model predicts the relation. In this case, the probability of disagreement on a given relation r is defined as:

$$\phi_\Gamma(r|(s,o)) = 1 - \prod_{i=1}^{n} p_{\gamma_i}(r|(s,o)) \tag{8}$$

Following the same steps as Section 3.1, the overall disagreement on the relation set R for the entity pair (s, o) is:

$$\phi_\Gamma(s,o) = \prod_{r\in R} \phi_\Gamma(r|(s,o)) \tag{9}$$

We apply the same logarithmic transformation as Section 3.1 and define the disagreement for the entity pair (s, o) as:

$$\psi_\Gamma(s,o) = \sum_{r\in R} \log \phi_\Gamma(r|(s,o)) \tag{10}$$

Based on it, we define the sampling criterion as selecting the top-k entity pairs with the highest disagreement:

$$\arg\max_{(s,o)} \psi_\Gamma(s,o) \tag{11}$$

In DOREMI, the training data set is iteratively enriched with annotated triples selected according to the chosen criterion.

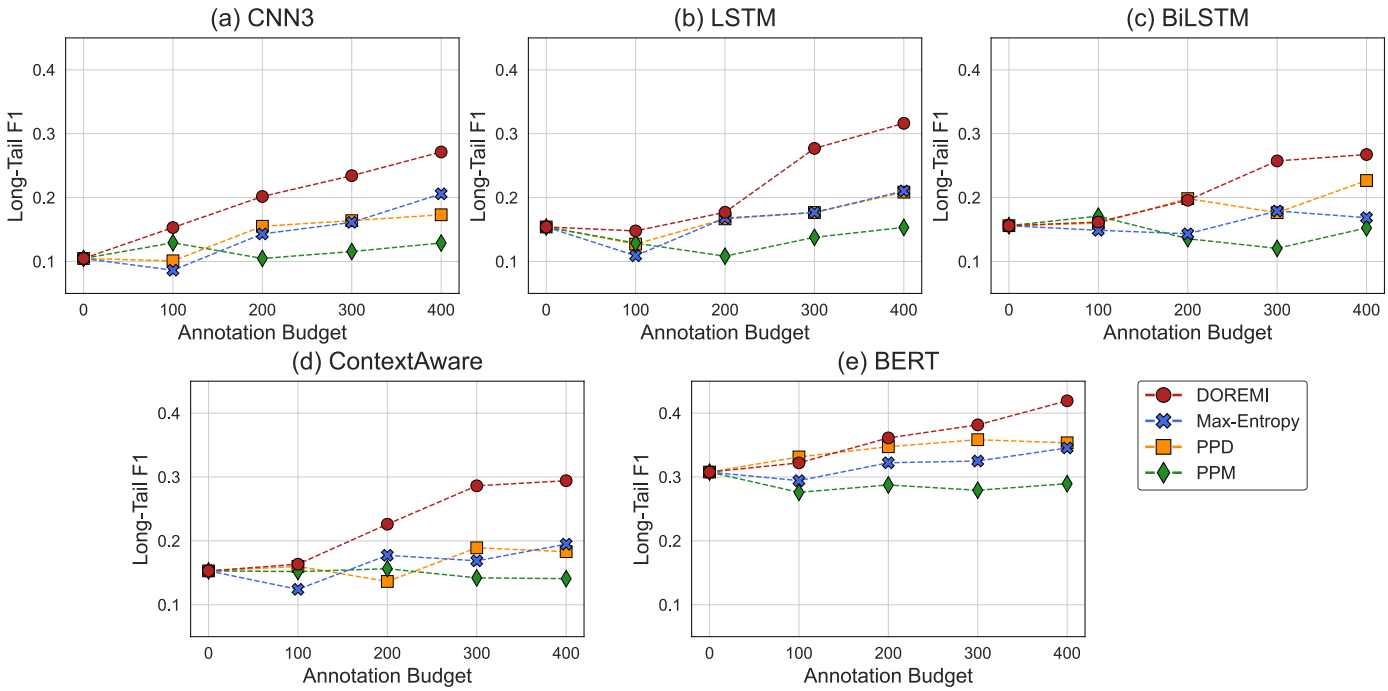


Figure 7: Micro-averaged F1 score of long-tail relations using different selection criteria computed on half of the DocRED dev dataset. Each subplot reports one of the five DocRE core models exploited during iterative training by DOREMI. Long-tail refers to relations with < 100 training instances. Annotation Budget "0" refers to the pre-training step, which is independent of the employed selection criteria.

Thus, testing different selection criteria requires multiple annotation rounds for each criterion, which can be costly. Therefore, instead of sampling training examples from the DocRED DS data and manually annotating the triples, we split the DocRED development dataset in half: one half serves for evaluation, while the other half is used as the sampling pool. We maintain the budget b and k used for the manual annotation, resulting in four iterations of DOREMI.

Figure 7 reports the micro-averaged F1 score of long-tail relations at each iteration for the DocRE core models using different selection criteria for the sampling module in DOREMI. The F1 score is computed on half of the DocRED development dataset and considers long-tail relations with less than 100 training instances in the DocRED manual training dataset. The disagreement-based selection criterion exploited in DOREMI shows a steady performance improvement across iterations for all core models. Between iterations 2 and 5, the steeper F1 improvement suggests more effective sampling. Other approaches yield minimal changes in the F1 score across iterations and, in some cases, result in performance degradation. Such a behaviour is evident in ContextAware and BERT when exploiting PPD or in BiLSTM when using Max-Entropy. Notably, employing PPM results in a lower F1 score than the pre-training step for ContextAware and BERT; hence, demonstrating it is not a good proxy to select training instances.

6.2. *Sensitivity Analysis*

Figure 7 reports DOREMI micro-averaged F1 scores of long-tail relation computed on half of the DocRED dev dataset using different annotation budgets (DOREMI is represented with red circled). All core models report a sizable improvement when increasing the number of annotations, motivating the need for the annotation budget for DocRED equal to 400. We capped b at 400 to limit the amount of human annotation and because the improvement curve exhibits an inflection between $b = 300$ and $b = 400$, suggesting plateau performance gains beyond this point.

The sensitivity analysis for the other hyperparameters is infeasible for DOREMI. Being a human-in-the-loop method, studying the effect of different values for each hyperparameter would require several human annotation rounds and a large annotation budget.

6.3. *Manual Annotation Analysis*

This section reports some statistics on the manual annotation process. We investigate whether the sampling strategy employed in DOREMI effectively identifies long-tail examples and the number of N/A triples detected. Table 7 reports the manual annotation statistics for both DocRED and Re-DocRED. In both datasets, the number of long-tail triples annotated grows at each iteration, demonstrating that models are getting better at predicting long-tail triples and the sampling strategy effectively samples them. Notably, for the DocRED dataset, the growth is more pronounced. During the first iteration, the ratio of frequent to long-tail triples is nearly equal, whereas by the final iteration, long-tail relations account for 60% of the annotated triples, while frequent triples make up only 14%.

The portion of N/A instances is marginal across all iterations and datasets. However, it grows at each iteration in Re-DocRED. A manual analysis of these cases revealed that the

Table 7: Manual annotations statistics. For each dataset and each iteration, we report the number and the percentage over the total of annotated long-tail triples, frequent relations, N/A, and the total number of annotated triples.

Iteration	Long-Tail	Frequent	N/A	Triples
(a) DOREMI iterative training with DocRED				
Iteration 1	34 (34 %)	42 (42%)	24 (24%)	100
Iteration 2	39 (39 %)	43 (43%)	18 (18%)	100
Iteration 3	49 (49 %)	36 (36%)	15 (15%)	100
Iteration 4	60 (60 %)	14 (14%)	26 (26%)	100
Total	182 (46 %)	135 (33%)	83 (21%)	400
(b) DOREMI iterative training with Re-DocRED				
Iteration 1	172 (57 %)	122 (41%)	6 (2%)	300
Iteration 2	178 (59 %)	99 (33%)	23 (8%)	300
Iteration 3	183 (61 %)	84 (28%)	33 (11%)	300
Iteration 4	206 (67 %)	50 (18%)	44 (15 %)	300
Total	739 (62 %)	355 (29%)	106 (9%)	1,200

majority of N/A instances in Re-DocRED samples stem from incorrect entity annotations in the DocRED DS dataset. This observation, coupled with the superior performance of models trained on Re-DocRED compared to those trained on DocRED, suggests that highly uncertain triples are often the result of such annotation errors. Nevertheless, incorporating N/A triples enhances model recall – as demonstrated in Section 5 – by helping the model learn to distinguish between the presence and absence of a relation.

While the sampling strategy is designed to emphasize long-tail relations, it cannot eliminate the presence of frequent triples and N/A instances. However, including such examples contributes to a more balanced training signal, ultimately enabling the models trained with the DOREMI denoised dataset to achieve strong performance.

7. Conclusions

We introduced DOREMI, a dataset enhancement framework that targets long-tail relation prediction while minimizing manual annotation. Leveraging iterative training, DOREMI identifies Hard-To-Classify examples by measuring disagreement across multiple models. Being optimized for long-tail predictions, DOREMI can complement existing denoising approaches, such as UGDRE, which are better suited for frequent relations. The resulting denoised distantly supervised dataset can be used to train any off-the-shelf DocRE model, yielding improved performance on long-tail relation prediction. Experiments on DocRED and Re-DocRED show that DOREMI significantly improves performance, especially in long-tail settings. Remarkably, annotating just 0.001% of the DocRED DS dataset yields precision improvements of up to a +8.2% overall and +76.0% in long-tail predictions over UGDRE – the current state-of-the-art in label denoising – as evaluated on the DocRED dev dataset. DOREMI also significantly boosts long-tail IGNPRECISION by +137.6%, indicating better generalization to novel entity pairs unseen during training. Similar findings hold for Re-DocRED, where labeling 0.003% of the data leads to a +16.2% gain in long-tail precision and +19.2% on IGN-PRECISION. On the *extreme long-tail triples*, DOREMI achieves a +83.2% relative increase in precision and +207.9% in IGN-PRECISION compared to UGDRE.

These results highlight the effectiveness of disagreement-driven annotation – enabling better generalization with negligible human effort.

Funding

This work has received funding from the HEREDITARY Project as part of the European Union's Horizon Europe research and innovation programme under grant agreement No GA 101137074. Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible.

Contributions

L.M. contributed to the design and development of DOREMI, creation of the distant denoised datasets, and manuscript preparation. S.M. contributed to the design and formalization of DOREMI and manuscript preparation. G.S. contributed to the design and formalization of DOREMI, supervision and coordination of the teamwork, and manuscript revision. All the authors contributed to the revision of the manuscript.

Competing Interests

The authors declare no competing interests.

Declaration of Generative AI use

During the preparation of this work, the authors used ChatGPT, Grammarly, and Comet as writing assistants, in particular to rephrase some parts of the writing. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

[1] Y. Yao, D. Ye, P. Li, X. Han, Y. Lin, Z. Liu, Z. Liu, L. Huang, J. Zhou, M. Sun, DocRED: A Large-Scale Document-Level Relation Extraction Dataset, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2019, pp. 764–777. URL: https://doi.org/10.18653/v1/P19-1074.

[2] Q. Huang, S. Hao, Y. Ye, S. Zhu, Y. Feng, D. Zhao, Does Recommend-Revise Produce Reliable Annotations? An Analysis on Missing Instances in DocRED, in: S. Muresan, P. Nakov, A. Villavicencio (Eds.), Proceedings of

the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 6241–6252. URL: https://doi.org/10.18653/v1/2022.acl-long.432.

[3] Q. Tan, L. Xu, L. Bing, H. T. Ng, S. M. Aljunied, Revisiting DocRED - Addressing the False Negative Problem in Relation Extraction, in: Y. Goldberg, Z. Kozareva, Y. Zhang (Eds.), Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 2022, pp. 8472–8487. URL: https://doi.org/10.18653/v1/2022.emnlp-main.580.

[4] R. Han, T. Peng, B. Wang, L. Liu, P. Tiwari, X. Wan, Document-level Relation Extraction with Relation Correlations, Neural Networks 171 (2024) 14–24. URL: https://doi.org/10.1016/j.neunet.2023.11.062.

[5] M. Choi, H. Lim, J. Choo, PRiSM: Enhancing Low-Resource Document-Level Relation Extraction with Relation-Aware Score Calibration, in: Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023 (Findings), Association for Computational Linguistics, 2023, pp. 39–47. URL: https://doi.org/10.18653/v1/2023.findings-ijcnlp.4.

[6] Q. Sun, K. Huang, X. Yang, P. Hong, K. Zhang, S. Poria, Uncertainty Guided Label Denoising for Document-level Distant Relation Extraction, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, 2023, pp. 15960–15973. URL: https://doi.org/10.18653/v1/2023.acl-long.889.

[7] F. Zhang, X. Jin, J. Cheng, H. Yu, H. Xu, Rethinking the Role of LLMs for Document-level Relation Extraction: a Refiner with Task Distribution and Probability Fusion, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 6293–6312. URL: https://doi.org/10.18653/v1/2025.naacl-long.319.

[8] W. Zhou, K. Huang, T. Ma, J. Huang, Document-Level Relation Extraction with Adaptive Thresholding and Localized Context Pooling, Proceedings of the AAAI Conference on Artificial Intelligence 35 (2021) 14612–14620. URL: https://doi.org/10.1609/aaai.v35i16.17717.

[9] Y. Ma, A. Wang, N. Okazaki, DREEAM: Guiding Attention with Evidence for Improving Document-Level Relation Extraction, in: Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2023, pp. 1971–1983. URL: https://doi.org/10.18653/v1/2023.eacl-main.145.

[10] L. Le, G. Zhao, X. Zhang, G. Zuccon, G. Demartini, CoLAL: Co-learning Active Learning for Text Classification, Proceedings of the AAAI Conference on Artificial Intelligence 38 (2024) 13337–13345. URL: https://doi.org/10.1609/aaai.v38i12.29235.

[11] B. Dalvi Mishra, N. Tandon, P. Clark, Domain-targeted, high precision knowledge extraction, Transactions of the Association for Computational Linguistics 5 (2017) 233–246. URL: https://doi.org/10.1162/tacl_a_00058.

[12] L. Menotti, S. Marchesin, G. Silvello, DOREMI Denoised Distantly Supervised Datasets, Zenodo, 2025. URL: https://doi.org/10.5281/zenodo.18170552.

[13] C. Quirk, H. Poon, Distant Supervision for Relation Extraction beyond the Sentence Boundary, in: M. Lapata, P. Blunsom, A. Koller (Eds.), Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, Association for Computational Linguistics, Valencia, Spain, 2017, pp. 1171–1182. URL: https://aclanthology.org/E17-1110.

[14] F. Christopoulou, M. Miwa, S. Ananiadou, Connecting the Dots: Document-level Neural Relation Extraction with Edge-oriented Graphs, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 4925–4936. URL: https://doi.org/10.18653/v1/D19-1498.

[15] S. Zeng, R. Xu, B. Chang, L. Li, Double Graph Based Reasoning for Document-level Relation Extraction, in: B. Webber, T. Cohn, Y. He, Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 1630–1640. URL: https://doi.org/10.18653/v1/2020.emnlp-main.127.

[16] B. Xu, Q. Wang, Y. Lyu, Y. Zhu, Z. Mao, Entity Structure Within and Throughout: Modeling Mention Dependencies for Document-Level Relation Extraction, Proceedings of the AAAI Conference on Artificial Intelligence 35 (2021) 14149–14157. URL: https://doi.org/10.1609/aaai.v35i16.17665.

[17] N. Zhang, X. Chen, X. Xie, S. Deng, C. Tan, M. Chen, F. Huang, L. Si, H. Chen, Document-level Relation Extraction as Semantic Segmentation, in: Z.-H. Zhou

(Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, 2021, pp. 3999–4006. URL: https://doi.org/10.24963/ijcai.2021/551, main Track.

[18] W. Xu, K. Chen, T. Zhao, Document-Level Relation Extraction with Reconstruction, Proceedings of the AAAI Conference on Artificial Intelligence 35 (2021) 14167–14175. URL: https://doi.org/10.1609/aaai.v35i16.17667.

[19] Y. Wang, H. Pan, T. Zhang, W. Wu, W. Hu, A Positive-Unlabeled Metric Learning Framework for Document-Level Relation Extraction with Incomplete Labeling, Proceedings of the AAAI Conference on Artificial Intelligence 38 (2024) 19197–19205. URL: https://doi.org/10.1609/aaai.v38i17.29888.

[20] M. Jain, R. Mutharaju, R. Kavuluru, K. Singh, Revisiting Document-Level Relation Extraction with Context-Guided Link Prediction, Proceedings of the AAAI Conference on Artificial Intelligence 38 (2024) 18327–18335. URL: https://doi.org/10.1609/aaai.v38i16.29792.

[21] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), 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), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://doi.org/10.18653/v1/N19-1423.

[22] Q. Tan, R. He, L. Bing, H. T. Ng, Document-Level Relation Extraction with Adaptive Focal Loss and Knowledge Distillation, in: Findings of the Association for Computational Linguistics: ACL 2022, Association for Computational Linguistics, 2022, pp. 1672–1681. URL: https://doi.org/10.18653/v1/2022.findings-acl.132.

[23] Z. Duan, T. Pan, Z. Li, X. Li, J. Wang, COMM: Concentrated Margin Maximization for Robust Document-Level Relation Extraction, Proceedings of the AAAI Conference on Artificial Intelligence 39 (2025) 23841–23849. URL: https://doi.org/10.1609/aaai.v39i22.34556.

[24] Y. Xiao, Z. Zhang, Y. Mao, C. Yang, J. Han, SAIS: Supervising and Augmenting Intermediate Steps for Document-Level Relation Extraction, in: M. Carpuat, M.-C. de Marneffe, I. V. Meza Ruiz (Eds.), Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Seattle, United States, 2022, pp. 2395–2409. URL: https://doi.org/10.18653/v1/2022.naacl-main.171.

[25] Y. Xie, J. Shen, S. Li, Y. Mao, J. Han, Eider: Empowering Document-level Relation Extraction with Efficient Evidence Extraction and Inference-stage Fusion, in: S. Muresan, P. Nakov, A. Villavicencio (Eds.), Findings of the Association for Computational Linguistics: ACL 2022, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 257–268. URL: https://doi.org/10.18653/v1/2022.findings-acl.23.

[26] Y. Du, T. Ma, L. Wu, Y. Wu, X. Zhang, B. Long, S. Ji, Improving long tailed document-level relation extraction via easy relation augmentation and contrastive learning, 2022. URL: https://arxiv.org/abs/2205.10511.

[27] C. Xiao, Y. Yao, R. Xie, X. Han, Z. Liu, M. Sun, F. Lin, L. Lin, Denoising Relation Extraction from Document-level Distant Supervision, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, 2020, pp. 3683–3688. URL: https://doi.org/10.18653/v1/2020.emnlp-main.300.

[28] L. Zhang, J. Su, Z. Min, Z. Miao, Q. Hu, B. Fu, X. Shi, Y. Chen, Exploring Self-Distillation Based Relational Reasoning Training for Document-Level Relation Extraction, Proceedings of the AAAI Conference on Artificial Intelligence 37 (2023) 13967–13975. URL: https://doi.org/10.1609/aaai.v37i11.26635.

[29] J. Li, Z. Jia, Z. Zheng, Semi-automatic data enhancement for document-level relation extraction with distant supervision from large language models, in: H. Bouamor, J. Pino, K. Bali (Eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, 2023, pp. 5495–5505. URL: https://doi.org/10.18653/v1/2023.emnlp-main.334.

[30] F. Zhang, Q. Miao, J. Cheng, H. Yu, Y. Yan, X. Li, Y. Wu, SRF: Enhancing Document-Level Relation Extraction with a Novel Secondary Reasoning Framework, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 15426–15439. URL: https://doi.org/10.18653/v1/2024.emnlp-main.863.

[31] S. Fan, Y. Wang, S. Mo, J. Niu, LogicST: A Logical Self-Training Framework for Document-Level Relation Extraction with Incomplete Annotations, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 5496–5510. URL: https://doi.org/10.18653/v1/2024.emnlp-main.314.

[32] K. Qi, J. Du, H. Wan, End-to-end Learning of Logical Rules for Enhancing Document-level Relation Extraction, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 7247–7263. URL: https://doi.org/10.18653/v1/2024.acl-long.391.

[33] F. Zhang, H. Yu, J. Cheng, H. Xu, Entity Pair-guided Relation Summarization and Retrieval in LLMs for Document-level Relation Extraction, in: L. Chiruzzo, A. Ritter, L. Wang (Eds.), Findings of the Association for Computational Linguistics: NAACL 2025, Association for Computational Linguistics, Albuquerque, New Mexico, 2025, pp. 4022–4037. URL: https://doi.org/10.18653/v1/2025.findings-naacl.224.

[34] I. Dagan, S. P. Engelson, Committee-Based Sample Selection for Probabilistic Classifiers, Journal Of Artificial Intelligence Research 11 (1999) 335–360. URL: https://doi.org/10.1613/jair.612.

[35] L. Xue, D. Zhang, Y. Dong, J. Tang, AutoRE: Document-Level Relation Extraction with Large Language Models, in: Proc. of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 211–220. URL: https://doi.org/10.18653/v1/2024.acl-demos.20.

[36] C. Gao, X. Wang, J. Sun, TTM-RE: Memory-Augmented Document-Level Relation Extraction, in: L.-W. Ku, A. Martins, V. Srikumar (Eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 443–458. URL: https://doi.org/10.18653/v1/2024.acl-long.26.

[37] A. Modarressi, A. Köksal, H. Schuetze, Consistent Document-level Relation Extraction via Counterfactuals, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2024, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 11501–11507. URL: https://doi.org/10.18653/v1/2024.findings-emnlp.672.

[38] Grattafiori, A. et al., The llama 3 herd of models, 2024. URL: https://arxiv.org/abs/2407.21783. arXiv:2407.21783.

[39] OpenAI, Achiam, J., et al., Gpt-4 technical report, 2024. URL: https://arxiv.org/abs/2303.08774. arXiv:2303.08774.

Table A.8: Macro-averaged performance of DocRE models trained on distant datasets: DocRED, UGDRE, DOREMI **(ours)**, and D+U (DOREMI for long-tail, UGDRE for frequent relations). Results are on the **DocRED** dev set. Best and second-best results are bolded and underlined.

Model	Distant Data	Precision	IgnPrec	Recall	F1	Ign F1
ATLOP BERT	DocRED	0.4499	0.2764	0.4456	0.4196	0.2917
	UGDRE	0.4490	0.2951	**0.5117**	**0.4572**	0.3358
	DOREMI	**0.4825**	**0.3693**	0.4057	0.3997	**0.3402**
	D+U	0.4576	0.3162	0.4486	0.4343	0.3351
ATLOP RoBERTa	DocRED	0.4497	0.2844	0.4696	0.4299	0.3054
	UGDRE	0.4497	0.3006	**0.5481**	**0.4682**	0.3495
	DOREMI	0.4502	0.3433	0.4264	0.4128	0.3505
	D+U	**0.4834**	**0.3479**	0.4663	0.4465	**0.3541**
DREEAM BERT	DocRED	0.4711	0.3277	0.3310	0.3566	0.2764
	UGDRE	0.4801	0.3408	0.4255	0.4279	0.3420
	DOREMI	0.4444	0.3430	**0.4519**	0.4260	0.3633
	D+U	**0.5199**	**0.3916**	0.4172	**0.4415**	**0.3673**
DREEAM RoBERTa	DocRED	0.4644	0.3200	0.3709	0.3851	0.2997
	UGDRE	0.4811	0.3364	**0.4626**	0.4477	0.3543
	DOREMI	0.4749	0.3742	0.4496	0.4312	0.3673
	D+U	**0.5325**	**0.3993**	0.4416	**0.4542**	**0.3741**

Table A.9: Macro-averaged performance of DocRE models trained on distant datasets: DocRED, UGDRE, DOREMI **(ours)**, and D+U (DOREMI for long-tail, UGDRE for frequent relations). Results are on the **Re-DocRED** test set. Best and second-best results are bolded and underlined.

Model	Distant Data	Precision	IgnPrec	Recall	F1	Ign F1
ATLOP BERT	DocRED	0.6614	0.5093	0.3016	0.3698	0.3075
	UGDRE	0.7088	0.6189	**0.5970**	**0.6264**	**0.5702**
	DOREMI	0.6922	0.5801	0.5354	0.5781	0.5181
	D+U	**0.7388**	**0.6578**	0.5490	0.6107	0.5641
ATLOP RoBERTa	DocRED	0.6691	0.5218	0.3354	0.3983	0.3393
	UGDRE	0.7064	0.6195	**0.6075**	**0.6358**	**0.5774**
	DOREMI	0.6951	0.5885	0.5275	0.5699	0.5150
	D+U	**0.7302**	**0.6500**	0.5760	0.6245	0.5760
DREEAM BERT	DocRED	0.7548	0.6040	0.2461	0.3236	0.2826
	UGDRE	0.7238	0.6528	**0.5813**	**0.6253**	**0.5843**
	DOREMI	0.7100	0.6110	0.5400	0.5918	0.5387
	D+U	**0.7628**	**0.6990**	0.5232	0.6013	0.5649
DREEAM RoBERTa	DocRED	0.7642	0.6453	0.2834	0.3633	0.3245
	UGDRE	0.7597	0.6929	**0.6164**	**0.6599**	**0.6216**
	DOREMI	0.7605	0.6591	0.5661	0.6260	0.5725
	D+U	**0.7963**	**0.7311**	0.5522	0.6306	0.5983

Appendix A. Macro-Averaged Results

Tables A.8 and A.9 report the macro-average performance of ATLOP and DREEAM trained on different datasets and tested on DocRED and Re-DocRED, respectively. Macro-averaging computes the metric independently for each class and then takes the average, treating all classes equally regardless of their frequency. This method is particularly useful for assessing performance on rare or underrepresented classes. Thus, in our case, it makes sense to display the macro-average performance on all relations.

In both datasets, the macro-averaged performance confirms our findings in Section 5, demonstrating how DOREMI alone or combined with UGDRE improves the *ignored* metrics – IGN-PRECISION and IGNF1 –, aiding models to generalize to unseen pairs. In particular, when DOREMI exploits DocRED for iterative training (Table A.8), DOREMI shows an improvement of up to +25.1% in IGNPRECISION and +6.2% in IGNF1 compared to

Table B.10: Performance of DocRE models trained on the DOREMI distant dataset and LLMs. Best results are bolded.

Category	Model	Full Dataset					Long-Tail Triples				
		Precision	Ign Prec	Recall	F1	Ign F1	Precision	Ign Prec	Recall	F1	Ign F1
(a) Evaluated on DocRED dev dataset (long-tail relations with <100 training instances)											
LLMs	GPT-4.1	0.1986	- - -	0.0528	0.0834	- - -	0.0654	- - -	0.0470	0.0547	- - -
	Llama-3.3-70B	0.0236	- - -	0.0025	0.0046	- - -	0.000	- - -	0.0000	0.0000	- - -
BERT	ATLOP-DOREMI	0.5722	0.4384	0.6160	0.5933	0.5123	0.3634	0.2701	0.1946	0.2535	0.2262
	DREEAM-DOREMI	0.5765	0.4433	0.6341	0.6039	0.5218	0.3928	0.3058	0.2483	0.3043	0.2741
RoBERTa	ATLOP-DOREMI	0.5788	0.4453	0.6242	0.6006	0.5198	0.4200	0.3324	0.2362	0.3024	0.2762
	DREEAM-DOREMI	**0.5904**	**0.4564**	**0.6513**	**0.6194**	**0.5367**	**0.4332**	**0.3492**	**0.2523**	**0.3189**	**0.2930**
(b) Evaluated on Re-DocRED test dataset (long-tail relations with <300 training instances)											
LLMs	GPT-4.1	0.1612	- - -	0.0301	0.0579	- - -	0.0093	- - -	0.0153	0.0116	- - -
	Llama-3.3-70B	0.0198	- - -	0.0015	0.0028	- - -	0.000	- - -	0.0000	0.0000	- - -
BERT	ATLOP-DOREMI	0.7480	0.6297	0.6914	0.7186	0.6591	0.6023	0.5004	0.3628	0.4299	0.3795
	DREEAM-DOREMI	0.7701	0.6576	0.7037	0.7354	0.6799	0.7250	0.6640	0.4210	0.5326	0.5153
RoBERTa	ATLOP-DOREMI	0.7486	0.6311	0.7023	0.7247	0.6648	0.6232	0.5315	0.3489	0.4206	0.3784
	DREEAM-DOREMI	**0.7939**	**0.6894**	**0.7267**	**0.7588**	**0.7075**	**0.7575**	**0.7032**	**0.4498**	**0.5644**	**0.5486**

UGDRE.

When DOREMI exploits Re-DocRED for iterative training (Table A.9), combining DOREMI with UGDRE (dataset D+U) consistently outperform UGDRE in terms of precision and IGN-PRECISION, reporting a relative of up to +5.4% and +7.1%, respectively.

Appendix B. Large Language Models Performance

Recent works explored LLMs for DocRE [7, 33, 35]. To enhance their performance, logical reasoning [30, 32] and data augmentation [31, 36, 37] have been proposed to integrate additional filtering. Although this is a promising avenue, LLMs still proves to be ineffective for DocRE. The primary reason is that LLMs often overestimate positive relations between entities, as they struggle to handle the large proportion of negative examples present in DocRE datasets. A secondary factor is their difficulty with multi-label classification. Indeed, LLMs frequently predict a single relation even when multiple relations are expressed for the same entity pair within a document [7].

This section compares the performance of models trained with DOREMI DDSs and LLMs. To obtain the LLMs predictions, inspired by [31], we exploit a zero-shot prompt-based approach using the same prompt for all LLMs. We use two LLMs: Llama 3.3 70B Instruct [38] and GPT-4.1 [39]. These two models were chosen primarily for their availability and compatibility with our computational resources. To run GPT-4.1, we used the Azure OpenAI Batch API in the Azure AI Foundry Portal and set the LLM temperature to 0.75. To run Llama 3.3 70B Instruct, we developed a Python script that exploit Ollama.[9] We empirically set `num_predict` to $1,000$, `temperature` to 0.7, and `top_p` to 0.1. Due to limited computational resources, we opted for `llama3.3:70b-instruct-q4_0`, [10] a 4-bit quantized version of Llama 3.3 70B Instruct. Llama 3.3 70B Instruct was run on a Mac Studio 2025 with 512 GB of memory and the Apple M3 Ultra chip. We report the prompt template we used in Figure B.8.

Table B.10 reports the micro-averaged results for the LLMs compared with two state-of-the-art models trained with the DOREMI DDS. Table B.10 (a) reports the performance of all the models evaluated in the DocRED dev dataset. ATLOP and DREEAM are trained on the DOREMI DDS produced using the DocRED dataset during training. Table B.10 (b) evaluates the models in the Re-DocRED test dataset. In this configuration, ATLOP and DREEAM are trained on the DOREMI DDS constructed with iterative training on the Re-DocRED dataset during training.

Both LLMs underperform relative to state-of-the-art models, exhibiting a substantial drop in performance. GPT outperforms LLama, achieving a PRECISION of nearly 20% and an F1 score of 8% on the DocRED development set, while on the Re-DocRED test dataset PRECISION reaches 16% and F1 is approaching 6%. GPT struggles in RECALL, obtaining only 5% on the DocRED development set. Indeed, GPT predicts $3,422$ triples for the documents in the DocRED development dataset – regardless of correctness – while the ground truth contains $12,275$ triples. Llama confirms this pattern, with RECALL being ten times lower than PRECISION. In this case, the LLM returned only $1,351$ triples in the DocRED development documents, irrespective of whether they were correct. Furthermore, GPT and Llama predicted 11 and 3 relations, respectively, outside of DocRED relations, highlighting the tendency of LLMs to hallucinate.

Results indicate that LLMs remain less effective than current approaches for DocRE. Furthermore, these findings justify our

[9] `https://github.com/ollama/ollama`

[10] `https://ollama.com/library/llama3.3:70b-instruct-q4_0`

You will be given a document from Wikidata and a list of entities identified in the document. For each entity, you will be given its identifier and a list of all the mentions of the entity in the document. For each mention, you will be given the textual mention, the type of the entity, the position in the text, and the sentence in which the mention appears.

For all possible entity pairs, your task is to predict whether the document contains any relation between the entity pair. You must return a list of dictionary with keys 'h', 't', and 'r'. 'h' contains the id of the subject entity, 't' contains the id of the object entity, and 'r' contains a list of the predicted relation(s). The first entity (key 'h') is the subject of the relation, while the second entity in the pair (key 't') is the object of the relation. You will also be given a set of relation to choose from.

The set of relations is formatted as a dictionary where the key is the identifier of the relation and the value is a dictionary with the name of the relation (key 'name') and a brief textual description (key 'description'). You must predict what relation is expressed in the document between the two entities by choosing one of more relations from the given dictionary of relations. You must reply with the identifier of the one or more relations identified.
If you think no relation is described in the document between the two entities, reply with '999'.

DOCUMENT
[input_document]

ENTITIES
[list_of_entities]
Format: Entity e_id: [{'pos': [x, y], 'type': ..., 'sent_id': z, 'name': ...}]
...

POSSIBLE RELATIONS
[list_of_relations]
Format: ['Pxyz': {'name': ..., 'description': ...}, ...]

QUERY
For the given document, predict one or more relations for all possible entity pairs. You must choose relations from the list provided in ''POSSIBLE RELATIONS''. You must return a list of dictionary with keys 'h', 't', and 'r'.'h' contains the id of the subject entity, 't' contains the id of the object entity, and 'r' contains a list of the predicted relation(s).

Figure B.8: Prompt template for DocRE. For each document, information between square brackets was integrated accordingly. The same prompt is used for both LLMs.

decision not to use LLMs to annotate new data for inclusion in the DOREMI loop, as their predictions are unreliable and may introduce even more noise than DS.