TTM-RE: Memory-Augmented Document-Level Relation Extraction

Anonymous ACL submission

Abstract

Document-level relation extraction aims to categorize the association between any two entities within a document. We find that previous methods for document-level relation extraction are ineffective in exploiting the full potential of large amounts of training data with varied noise levels. For example, in the ReDocRED benchmark dataset, state-of-the-art methods trained on the large-scale, lower-quality, distantly supervised training data generally do not perform better than those trained solely on the smaller, high-quality, human-annotated training data. To unlock the full potential of large-scale noisy training data for document-level relation extraction, we propose TTM-RE, a novel approach that integrates a trainable memory module, known as the Token Turing Machine, with a noisy-robust loss function that accounts for the positive-unlabeled setting. The trainable memory module enhances knowledge extraction from the large-scale noisy training dataset through an explicit learning of the memory tokens and a soft integration of the learned memory tokens into the input representation, thereby improving the model's effectiveness for the final relation classification. Extensive experiments on ReDocRED, a benchmark dataset for document-level relation extraction, reveal that TTM-RE achieves state-of-the-art performance (with an absolute F1 score improvement of over 3%). Ablation studies further illustrate the superiority of TTM-RE in other domains (the ChemDisGene dataset in the biomedical domain) and under highly unlabeled settings.

1 Introduction

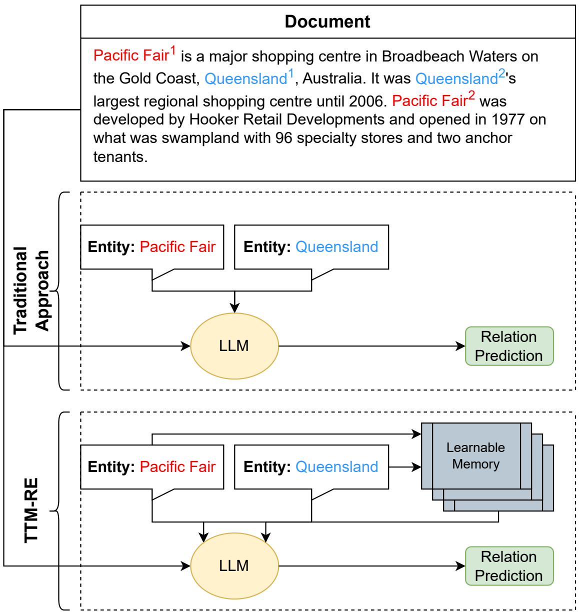
Document
Pacific Fair[1] is a major shopping centre in Broadbeach Waters on the Gold Coast, Queensland[1], Australia. It was Queensland[2]'s largest regional shopping centre until 2006. Pacific Fair[2] was developed by Hooker Retail Developments and opened in 1977 on what was swampland with 96 specialty stores and two anchor tenants.
Traditional Approach
Entity: Pacific Fair
Entity: Queensland
LLM
Relation Prediction
TTM-RE
Entity: Pacific Fair
Entity: Queensland
Learnable Memory
LLM
Relation Prediction

Figure 1: Differences between the generic document relation extraction approach and TTM-RE for document-level relation extraction. The memory module processes the input entities and outputs to the relation classifier. We investigate how adding the memory component affects performance (such as different datasets and memory sizes).

Relation extraction aims to classify the relationships between two specified entities into predefined categories. This task is pivotal in natural language processing, as it involves identifying and categorizing the connections between two entities (for example, "Pacific Fair" and "Queensland" in Figure 1) into predefined classes (for example, "located in" in Figure 1). Its importance spans across various downstream applications, encompassing question answering (Veena et al., 2017), knowledge graph construction (Distiawan et al., 2019), and the extraction of general patterns (Han et al., 2020).

Previous work for relation extraction mainly focuses on sentence-level relations (Alt et al., 2020). For example, Sainz et al. (2021) characterized each relation class using a label verbalizer and addressed the relation extraction task through a textual entailment model, as well as other models such as DeepStruct (Wang et al., 2022a). However, techniques that are primarily designed and evaluated for extracting relationships at the sentence level, face challenges that limit their suitability for datasets focused on document-level relation extraction such as ReDocRED (Tan et al., 2022c).

These challenges include a large label imbalance and a large number of possible combinations between possible head and tail entity pairs in each document, which is quadratic. Previous work has generally addressed DocRE's label imbalance with custom loss functions and its quadratic entity computation by minimizing document processing. Usually, a common processing step for the document inspired by Zhou et al. (2021a) is used. Initially, a pre-trained encoder processes the entire document as a whole. Subsequently, indexing over the entity mentions can be employed to retrieve the head and tail entities for classification. Recent work has mainly focused on novel loss functions (Tan et al., 2022a; Wang et al., 2022b) or additional inputs such as evidence (Ma et al., 2023).

However, less effort has been made in effectively leveraging the large amount of the distantly labeled data in ReDocRED and DocRED (Yao et al., 2019; Tan et al., 2022c). Most work (Tan et al., 2022b; Ma et al., 2023) use the distantly labeled dataset for knowledge distillation–that is, a model is first trained on the fully human annotated train data, and then used to obtain output logits on the distantly supervised data. These logits, along with the distant relation labels, are then used as guidance in training a secondary student model. However, previous work investigating fine-tuning on the distantly supervised dataset has failed to significantly boost performance (as we show later in our results). Previously, this could be explained by the lower quality and the lack of human annotation on the distantly supervised dataset. However, we assert that this is due to an architectural limitation on the prevailing framework.

Recently, numerous studies have highlighted the advantages of incorporating memory in both computer vision and NLP to acquire pertinent representations from past data points that facilitate improved classification performance. For instance, Barraco et al. (2023) showcased the enhanced performance of an encoder-decoder model for image captioning by integrating memory of past observations into the attention mechanism. Additionally, the integration of memory has been demonstrated to enhance performance in knowledge-intensive tasks such as long-form question answering and dialogue (Wu et al., 2022). More recently, Google's Token Turing Machine (TTM) (Ryoo et al., 2023) has showcased state-of-the-art performance in real-world long-sequential visual understanding using an autoregressive Transformer model equipped with memory.

`Pacific Fair`^{1} `is a major shopping centre in Broadbeach Water on the Gold Coast,` `Queensland`^{1}`, Australia. It was` `Queensland`^{2}`'s largest regional shopping centre until 2006.` `Pacific Fair`^{2} `was developed by Hooker Retail Developments and opened in 1977 on what was swampland with 96 specialty stores and two anchor tenants.`

Figure 2: Sample document relation extraction document from DocRED (Yao et al., 2019). Here, the head entity is related to the tail entity by "`P131: located in the administrative territorial entity`".

Drawing from recent advancements in memory-augmented models, we introduce TTM-RE, the inaugural memory-augmented architecture designed specifically for document-level relation extraction. Through empirical evidence, we demonstrate that this architecture enables notably enhanced fine-tuning on extensive distantly labeled data from empirical observations. Specifically, we show that adding memory tokens from TTM (Ryoo et al., 2023) empirically enhances downstream relation classification by allowing reprocessing of head and tail entities while jointly considering the learned memory tokens in mind (see Figure 1).

1. We propose TTM-RE, the first memory-augmented document-level relation extraction model. By incorporating pseudo entities, it significantly enhances downstream relation classification performance on datasets such as ReDocRED (+3 F1 score) and ChemDisGene (+5 F1 score).

2. We show that without any human-labeled data, TTM-RE achieves impressive relation extraction performance on unseen data (+9 F1 score). Furthermore, in an extremely unsupervised scenario (19% of training labels), TTM-RE outperforms the previous SOTA by an impressive margin (+12 F1 score).

3. We perform a thorough analysis of ablations examining the performance of TTM-RE across less/more frequent relation classes, assessing the impact of memory size, layer size, and the utilization of different base models.

2 Related Work

2.1 Document-level Relation Extraction

Document-level relation extraction stands as a critical endeavor in natural language processing, given that over 40.7% of relations necessitate the extraction of information spanning multiple sentences and multiple entity mentions (Yao et al., 2019; Tan et al., 2022c; Zhang et al., 2022a). In Figure 2, we illustrate an instance of document-level relation extraction. Here, the objective is to discern the relationship between a pair of entities ("Pacific Fair" and "Queensland") within the provided document. Each entity is referenced twice in the text (indicated by superscripts).

Thus, a document with n entities will have $n(n-1)$ possible relation predictions. Many works (Tan et al., 2022a; Wang et al., 2022b; Tan et al., 2022b) have proposed new loss functions on top of a single encoder pass (usually added on top of ATLOP (Zhou et al., 2021a)) to alleviate the combinatorial bottleneck and propose loss functions to tackle the class-imbalance problem in document-level relation extraction. Zhou et al. (2021a) introduced the ATLOP, which uses Roberta-large as the encoder and adaptive thresholding for the multilabel relation classification. introduced Tan et al. (2022a) introduced KD-DocRE, which combines axial attention over the entity mentions with adaptive focal loss and knowledge distillation. Wang et al. (2022b) introduced a loss that considered a positive-unlabeled nature of the label distribution. Ma et al. (2023) proposed using the evidence labels along with the distantly supervised labels to achieve the current SOTA.

2.2 Memory-based Models in NLP

Memory-based models have begun to see rising usage in the CV and image captioning areas. However, their usage in NLP has been surprisingly limited. Still, there are some interesting and relevant work to our application. De Jong et al. (2021) utilized ’mention memory’ to represent knowledge—a table of dense vector representations of each entity mention in the corpus, and demonstrated good performance on multiple open-domain tasks including claim verification, and entity-based QA. Zhong et al. (2022) introduced a training objective that directly utilizes memory sets from local, long-term, and external and showed reduced perplexity on WIKITEXT-103. Chen et al. (2022) introduced a question-answer augmented encoder-decoder model and pretraining strategy, demonstrating improved performance on single-hop and multi-hop QA datasets. Wu et al. (2022) learns keys and values that represent questions and corresponding answers respectively; at inference time, the model would retrieve information from the memory using maximum inner product search.

Inspired by Token Turing Machines, TTM-RE’s memory mechanism differs from these approaches in that it does not necessitate learning relevant portions of real text. It simply learns memory tokens (dense embeddings). This is uniquely suited to our application, as retrieving and reprocessing text requires additional LLM encoder calls for each entity-entity pair, which is quadratic in nature.

3 Methodology

We propose TTM-RE, a memory-augmented, document-level relation extraction method that adds. An illustration of the overall framework of TTM-RE is shown in Figure 3.

3.1 Problem Definition

In the task formulation, we examine a document D that comprises M sentences (s_1, s_2, ..., s_M), N entities (e_1, e_2, ..., e_N), and R relation classes. Given this document D and a specified pair of entities (e_{head}, e_{tail}), the goal is to forecast a set of positive relations ($\hat{r}_1, \hat{r}_2, ..., \hat{r}_p$) between the entity pair based on the information derived from the document. It’s important to note that each entity may appear multiple times within the document D and that each possible entity-entity pair needs to be considered (i.e. if n entities, we need to consider $R \times n \times (m-1)$ possible relations).

3.2 Token Turing Machines

The memory, denoted as $M \in \mathbb{R}^{m \times d}$, comprises a collection of m tokens each with a dimensionality of d. The input, consisting of n tokens represented by $I \in \mathbb{R}^{n \times d}$, is combined with the memory M. This concatenated input is then further processed to generate an output denoted as $O \in \mathbb{R}^{r \times d}$, where r represents the desired number of retrieved tokens. The outputs from this process, in conjunction with the preceding inputs and the current memory, constitute the output of the TTM. In our case, $I \in \mathbb{R}^{2 \times d}, O \in \mathbb{R}^{2 \times d}$ for the head and tail entities.

Token Turing Machines add support for external memory in the form of tokens (Figure 3 Memory Module). In Token Turing Machines (TTMs), the

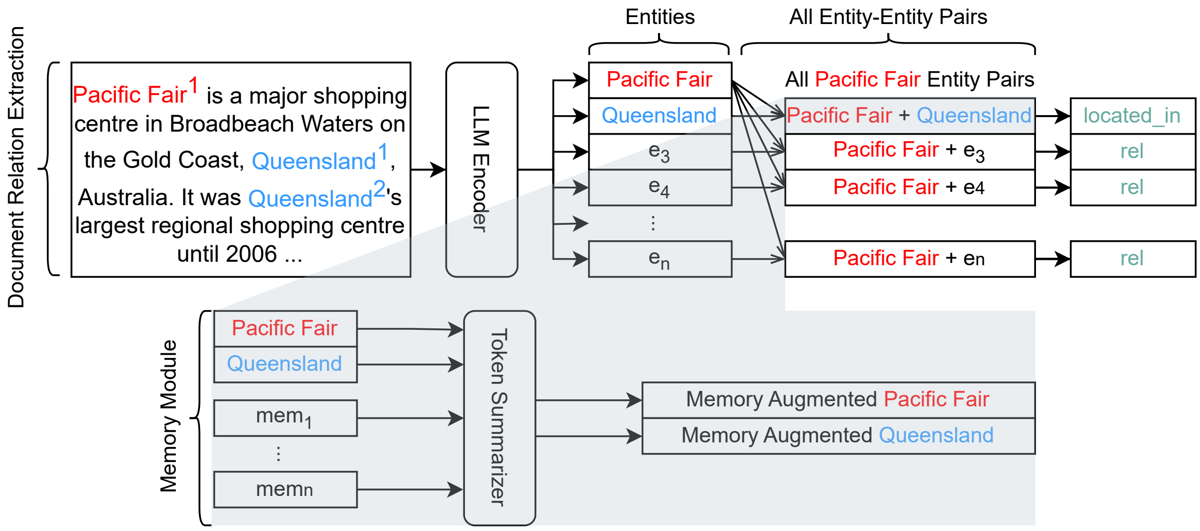
Document Relation Extraction
Pacific Fair[1] is a major shopping centre in Broadbeach Waters on the Gold Coast, Queensland[1], Australia. It was Queensland[2]'s largest regional shopping centre until 2006 ...
LLM Encoder
Entities
All Entity-Entity Pairs
Pacific Fair
Queensland
e3
e4
en
All Pacific Fair Entity Pairs
Pacific Fair + Queensland
Pacific Fair + e3
Pacific Fair + e4
Pacific Fair + en
located_in
rel
rel
rel
Memory Module
Pacific Fair
Queensland
mem1
memn
Token Summarizer
Memory Augmented Pacific Fair
Memory Augmented Queensland

Figure 3: Overall framework of TTM-RE. Given an example document and an expected relation distribution, we use an LLM (Roberta-Large) to encode the input tokens in a single pass and consider head and tail entities by their token representations, which are then fed into a memory module (in gray). The memory module then returns 2 memory-augmented versions of the head and tail entities for final relation classification.

interface between the processing unit and memory are done purely in terms of “read” and “write” operations. Note that in the original paper, the output from the processing unit is “written” to the memory, but in our case, since we are not applying this model sequentially, we can ignore this step and focus solely on the reading portion.

Initializing Memory Tokens: We follow the original implementation of TTM and initialize memory tokens from scratch, with one major difference. While the original code initialized tokens from zeros, we found that this led to a lack of gradient updates. Instead, we initialize from a normal distribution. Note that we cannot simply use entity text embeddings since the memory layer is after all processing steps and only before the final classification layer. We found this setup to work the best empirically, but further research is needed on the placement of the memory mechanism.

Reading from Memory: While the memory is intended to encapsulate condensed information deemed significant by the model, not all of this data may be relevant. Additionally, redundancies in the input, denoted as I, can arise due to information already stored in our memory, M, or inherent within the data itself. Selective reading where only a smaller subset of tokens is considered should encourage the model to create a memory repository containing relevant information over the entire relation classification task.

We summarize a token set $I \in \mathbb{R}^{n\times d}$ by deriving an importance weight vector, $w_i \in \mathbb{R}^n$, which we utilize to compute a weighted aggregation across the n tokens. Notably, each output token, indexed as $i \in 1, \dots, k$, possesses its corresponding weight w_i, computed using a learnable function that takes the input I itself, denoted as $\alpha_i(I)$. This importance weighting function is implemented either through a Multi-Layer Perceptron (MLP) determined as $w_i = \alpha_i(I) = \text{softmax}(\text{MLP}(I))$.

Subsequently, these weights facilitate a weighted summation of the inputs: $z_i = w_i \cdot V = \alpha_i(V) \cdot V$, where each token z_i effectively condenses all tokens from the complete set V, guided by the dynamic weighting $w_i = \alpha_i(V)$. As the model learns to summarize p tokens into k tokens, it generates a matrix $W = [w_1, \cdots, w_k]$ comprising importance weights relative to the memory tokens. Also, to allow the model to read from a location, take advantage of memory token position, and distinguish them from input tokens, there is a learnable positional embedding (Dosovitskiy et al., 2020) before each read module. All of this is captured in a memory reading function defined as: $Z = \text{Read}(M, I) = S_r([M||I])$. Where $[M||I]$ denotes the concatenation of memory M and input I and $S_r \in \mathbb{R}^{(n+m)\times d} \rightarrow \mathbb{R}^{r\times d}$. In our application, $r = n = 2$, we retrieve the memory-augmented head S_1 and tail S_2 entities that are the most relevant for the head and tail entity relation classification problem.

3.3 Processing of Head and Tail Entities

After retrieving our memory-augmented head S_1 and tail S_2 entities, we use the group bilinear approach as specified in Zhou et al. (2021b); Tang et al. (2019) to reduce the number of parameters to enable more efficient learning. Let $e_{head} = S_1, e_{tail} = S_2$. Each entity is split into k sections of dimension d/k, $e_{head} = [e^1_{head}|| \dots ||e^k_{head}], e_{tail} = [e^1_{tail}|| \dots ||e^k_{tail}]$

$$p(r|e_{head}, e_{tail})s = \sigma\left(\sum_{i=1}^{k} e^i_{head}\boldsymbol{W}^i_r e^i_{tail}\right)$$

$\boldsymbol{W}^i_r \in \mathbb{R}^{d/k \times d/k}$ denotes bilinear layers, and the sum of the products represents the grouped bilinear layer. This reduces parameters from $d^2 \rightarrow d^2/k$ and enables much better performance empirically.

Furthermore, the final output is a prediction vector of dimension $R + 1$ (number of all relations + 1 to learn the threshold value), as we adopt the adaptive thresholding approach implemented by Zhou et al. (2021b), which most other recent Document RE work as done as well (Ma et al., 2023; Tan et al., 2022a).

3.4 Noise-Robust Loss Function (SSR-PU)

There exists a large number of false negatives in the labeled relation triples. Gao et al. (2023) demonstrated difficulty in learning to ignore the false negatives for zero-shot prompting, revealing the difficulty of prompting LLMs for document relation extraction. To address this problem, we adopt Positive Unlabeled (PU) learning with the prior shift for each class as in Wang et al. (2022b) (Plessis et al., 2015; du Plessis et al., 2014).

Ordinary PU learning assumes that the overall distribution is the same as the distribution of the unlabeled data, which may not be true in our case. To address this problem, PU learning under the prior shift of training data needs to be considered (Charoenphakdee and Sugiyama, 2019). For each class, assume that the original prior $\pi_i = p(y_i = +1)$. Let $\pi_{labeled,i} = p(s_i = +1)$ and $(1 - \pi_{labeled,i}) = (1 - p(s_i = +1)) = p(s_i = -1)$ where $s_i = +1$ or $s_i = -1$ mean that the i-th class is labeled or unlabeled, respectively.

The conditional probability of a positive sample under unlabeled data is:

$$\begin{aligned}\pi_{u,i} &= p(y_i = 1 \mid s_i = -1)\\ &= \frac{\pi_i - \pi_{labeled,i}}{1 - \pi_{labeled,i}}\end{aligned}$$

Table 1: Statistics of the Re-DocRED dataset (Train, Dev, and Test are fully reprocessed from DocRED for improved accuracy and completeness). In total, there are 96 relations. The distantly supervised dataset is the same as in DocRED and is created with no human supervision.

Statistics	Distant	Train	Dev	Test
# Docs	101873	3053	500	500
Avg. # Entities	19.29	19.4	19.4	19.6
Avg. # Labeled Triples	14.79	28.1	34.6	34.9
Avg. # Sentences	8.13	7.9	8.2	7.9

The non-negative risk estimator under class prior shift of training data is obtained as follows (Kiryo et al., 2017; Wang et al., 2022b):

$$\begin{aligned}\widehat{R}_{\mathrm{S-PU}}(f) &= \sum_{i=1}^{K}\Big(\frac{\pi_i}{n_{\mathrm{P}_i}}\sum_{j=1}^{n_{\mathrm{P}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{P}_i}), +1)\\ &+ \max(0, \Big[\frac{1}{n_{\mathrm{U}_i}}\frac{1-\pi_i}{1-\pi_{u,i}}\sum_{j=1}^{n_{\mathrm{U}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{U}_i}), -1)\\ &- \frac{1}{n_{\mathrm{P}_i}}\frac{\pi_{u,i} - \pi_{u,i}\pi_i}{1-\pi_{u,i}}\sum_{j=1}^{n_{\mathrm{P}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{P}_i}), -1)\Big]\Big)\Big)\end{aligned}$$

where $\pi_i = p(y_i = +1)$ denotes probability of positive prior for relation class i. n_{P_i} are the number of positive and n_{U_i} are the unlabelled samples of class i, respectively. ℓ is a convex loss function, and $f_i(\cdot)$ is a score function that predicts class i. $\boldsymbol{x}_j^{\mathrm{P}_i}$ and $\boldsymbol{x}_j^{\mathrm{U}_i}$ denotes that the j-th sample of class i is positive and unlabeled as class i respectively. Please see Appendix F for more details.

4 Experimental Settings

4.1 Datasets

ReDocRED To evaluate our methodology, we primarily use ReDocRED (Tan et al., 2022c), an open-access, document-level relation extraction dataset that improves upon the popular DocRED dataset (Yao et al., 2019) by resolving incompleteness, addressing logical inconsistencies, and correcting coreferential errors. Table 1 shows the amount of training data available for all data splits as well as the average number of entities. Note that we primarily use the Dev set of ReDocRED for our experiments for computational practicality.

ChemDisGene ChemDisGene (Zhang et al., 2022a) is a biomedical multi-label document RE dataset. Entity mentions were obtained using PubTator Central (Wei et al., 2019), and the relation-

Table 2: Statistics of the ChemDisGene dataset. In total, there are 14 relations. The distantly supervised training is created with no human supervision.

Statistics	Train (Distant)	Test
# Docs	76942	523
Avg # Entities	7.5	10
Avg # Labeled Triples	2.1	7.2

ships are based on the Comparative Toxicogenomics Database (Davis et al., 2021). Table 2 shows the stats for the data. It comprises 523 abstracts meticulously curated by a team of biologists. Our training uses the larger distantly supervised training set, while evaluation is conducted using the fully expert-labeled test set. The average number of relations per document in the test set across both datasets significantly exceeds the average number of relations per document in the training set. This indicates the incomplete labeling phenomenon in the training set with a large number of false negatives, much like DocRED, before the updated ReDocRED.

4.2 Baselines

For ReDocRED, we compared baselines ranging from fully supervised to distantly supervised. We will compare three settings, all evaluated on the same human-annotated test and development set. All models were chosen using the best scores on the dev set.

1. **Human Annotated Only**: Denotes training only on the 3053 training dataset
2. **Distantly Supervised Only**: Denotes training only on the 101,873 distant dataset per DocRED, as ReDocRED does not revise this dataset.
3. **Human Annotated + Distantly Supervised**: Combines these two datasets.

Baseline models do this in a variety of different ways, with some using knowledge distillation (i.e. teacher training on human-annotated, student training from teacher output on distantly supervised). In TTM-RE, we fine-tune on distantly supervised training data via the regular loss, freeze the memory tokens, and then fine-tune on the training set.

For fully supervised (Human Annotation Only setting), we compare against ATLOP (Zhou et al., 2021b), DREEAM (Ma et al., 2023), KD-DocRE (Tan et al., 2022b), and SSR-PU (Wang et al., 2022b). For the distantly supervised setting, we only compare against SSR-PU, as it is shown to be better than ATLOP. Furthermore, DREAM and KD-DocRE both primarily use knowledge distillation to achieve their improvements over ATLOP, only using the distantly supervised data to create the teacher logits to supervise the student model. Therefore, we believe that our "fine-tuning on both the distantly and fully supervised training data" would not maintain the spirit of the baseline method. Finally, since the main focus of many of the previous baselines combines distantly supervised work with human annotations, we also evaluate all models on the combined human+distantly supervised datasets.

For ChemDisGene, we compared against baselines BRAN (Verga et al., 2018), PubmedBert (Gu et al., 2021), PubmedBert + BRAN (Zhang et al., 2022b), ATLOP, and Wang et al. (2022b) with (Positive-negative, positive-unlabelled, and the final SSR-PU variants).

In our experiments, we use precision, recall, and F1 scores as the evaluation metrics for the performance comparison. All standard deviations were calculated with 5 runs with different random seeds. More details about these evaluation metrics can be found in Appendix B.

5 Results

Main Results Table 3 shows the main results of our experiments. We see here that the Human Annotated Only dataset performs on the same level as the current SOTA results, DREEAM and SSR-PU. However, we see that on the 2 other settings, Distantly Supervised and the combined Human + Distantly Supervised, TTM-RE outperforms other methods by a significant margin, when considering the standard deviation (+9f1 and +3 F1 respectively). This implies that our model is much more effective with larger-scale training data even with noise in it, such as the distantly supervised training datasets. This intuitively makes sense since the memory tokens are initialized from scratch, and would benefit much more from larger-scale training data. Further research should seek to improve the initialization of the memory tokens, which could lead to faster training and further performance gains.

Finally, we observe that other baselines do not generally improve significantly even after training with distantly supervised and human-annotated data, which could be caused by architectural limits. Notably, this implies that TTM-RE's memory

Table 3: We compare all results against strong document relation extraction baselines: DREEAM (Ma et al., 2023), ATLOP (Zhou et al., 2021b), KD-DocRE (Tan et al., 2022a), and SSR-PU (Wang et al., 2022b). Bold denotes best performance or within 1 standard deviation.

	Dev				Test			
Model	F1	Ign F1	Precision	Recall	F1	Ign F1	Precision	Recall
Human Annotation Only								
DREEAM	**79.42 ± 0.18**	**78.36 ± 0.19**	74.74 ± 0.64	85.15 ± 0.25	**80.20 ± 0.45**	**78.56 ± 0.39**	75.74 ± 0.65	83.89 ± 0.61
ATLOP	76.15 ± 0.23	75.88 ± 0.23	69.62 ± 0.81	84.26 ± 0.97	77.81 ± 0.71	76.13 ± 0.28	67.76 ± 0.23	85.35 ± 0.62
KD-DocRE	77.88 ± 0.42	77.12 ± 0.49	**85.16 ± 0.58**	71.30 ± 0.79	78.28 ± 0.72	77.60 ± 0.25	**89.76 ± 0.14**	69.40 ± 0.03
SSR-PU	78.58 ± 0.11	78.08 ± 0.14	75.59 ± 0.27	**86.89 ± 0.51**	**80.18 ± 0.31**	**78.61 ± 0.46**	69.43 ± 0.43	**90.50 ± 0.53**
TTM-RE	78.13 ± 0.12	78.05 ± 0.17	83.28 ± 0.29	76.28 ± 0.61	**79.95 ± 0.33**	**78.20 ± 0.44**	85.81 ± 0.55	76.68 ± 0.22
Distantly Supervised Only								
ATLOP	**53.42 ± 0.61**	**50.14 ± 0.41**	**41.11 ± 0.66**	60.30 ± 0.81	53.42 ± 0.73	51.14 ± 0.66	51.11 ± 0.69	55.95 ± 0.91
SSR-PU	39.35 ± 0.46	35.04 ± 0.38	23.35 ± 0.27	72.63 ± 0.47	54.46 ± 0.48	53.26 ± 0.20	48.02 ± 0.34	**62.89 ± 0.42**
TTM-RE	42.21 ± 0.15	39.79 ± 0.37	27.68 ± 0.12	**81.70 ± 0.19**	**63.00 ± 0.29**	**61.55 ± 0.41**	**67.56 ± 0.24**	59.02 ± 0.30
Human Annotation + Distantly Supervised								
DREEAM	79.29 ± 0.23	78.89 ± 0.33	74.61 ± 0.24	**85.15 ± 0.30**	81.67 ± 0.35	78.95 ± 0.31	75.72 ± 0.35	84.88 ± 0.42
ATLOP	79.29 ± 0.23	74.83 ± 0.16	70.81 ± 0.73	80.67 ± 0.92	77.31 ± 0.65	75.72 ± 0.22	69.92 ± 0.26	84.10 ± 0.55
KD-DocRE	78.62 ± 0.56	77.15 ± 0.31	80.89 ± 0.24	73.02 ± 0.26	80.62 ± 0.45	80.32 ± 0.42	83.58 ± 0.21	75.06 ± 0.32
SSR-PU	80.09 ± 0.74	78.26 ± 0.30	74.51 ± 0.25	84.83 ± 0.30	80.52 ± 0.43	78.84 ± 0.31	74.24 ± 0.44	**87.96 ± 0.51**
TTM-RE	**83.56 ± 0.42**	**83.01 ± 0.35**	**88.09 ± 0.31**	81.78 ± 0.27	**84.01 ± 0.21**	**83.11 ± 0.37**	**86.03 ± 0.34**	82.09 ± 0.27

Table 4: F1 on ChemDisGene test dataset (all relationships). The models without standard deviations are taken from Wang et al. (2022b) accordingly. Standard deviations are shown with 5 random seed runs. Note that all baseline results are from Wang et al. (2022b) and Zhang et al. (2022b).

Model	F1	Precision	Recall
BRAN	32.5	41.8	26.6
PubmedBert	42.1	64.3	31.3
BRAN	43.8	70.9	31.6
ATLOP	42.73 ± 0.36	**76.17 ± 0.54**	29.7 ± 0.36
PN	44.25 ± 0.24	73.46 ± 0.95	31.67 ± 0.16
PU	44.6 ± 0.70	46.56 ± 1.17	42.8 ± 0.35
SSR-PU	48.56 ± 0.23	54.27 ± 0.40	43.93 ± 0.32
TTM-RE	**53.59 ± 0.27**	53.83 ± 0.85	**53.34 ± 0.15**

Table 5: Performance comparison of Top-K most common labels on the test dataset of ReDocRED. All but Top-K indicates the remainder of the $96 - K$ labels.

Model	F1	Ign F1	Precision	Recall
Top 10 Labels (62% of Data)				
ATLOP	62.12 ± 0.50	58.53 ± 0.62	50.70 ± 0.61	80.19 ±
SSR-PU	64.28 ± 0.31	60.87 ± 0.68	53.41 ± 0.35	80.72 ±
TTM-RE	68.21 ± 0.20	64.52 ± 0.43	57.94 ± 0.53	86.40 ±
All But Top 10 Labels				
ATLOP	39.47 ± 0.59	37.42 ± 0.34	27.34 ± 0.72	70.97 ±
SSR-PU	39.37 ± 0.41	37.39 ± 0.45	27.62 ± 0.62	68.51 ±
TTM-RE	44.04 ± 0.41	40.97 ± 0.49	32.01 ± 0.69	76.07 ±
Top 5 Labels (51% of Data)				
ATLOP	55.79 ± 0.42	51.34 ± 0.29	42.41 ± 0.56	81.48 ±
SSR-PU	58.32 ± 0.32	54.01 ± 0.36	45.04 ± 0.79	82.69 ±
TTM-RE	62.03 ± 0.71	56.68 ± 0.28	48.78 ± 0.69	87.77 ±
All But Top 5 Labels				
ATLOP	47.60 ± 0.47	45.63 ± 0.60	35.63 ± 0.47	71.69 ±
SSR-PU	47.35 ± 0.54	45.46 ± 0.37	35.99 ± 0.54	69.20 ±
TTM-RE	53.02 ± 0.53	49.97 ± 0.31	41.17 ± 0.56	76.97 ±

module adds processing capability that is actually *significantly useful* (Section 5 shows us an example where adding more parameters does not help).

ChemDisGene Results Table 4 shows that TTM-RE does indeed translate to other domains beyond the general task, with a 5 F1 point improvement over the best baseline. We observe TTM-RE performs well on the human-annotated training data. This is presumably because ChemDisGene has a larger dataset for training, so the memory tokens can be learned more effectively and, therefore it does not negatively affect performance as compared to the ReDocRED fully supervised setting.

Classifying Frequent/Infrequent Labels Previous work has shown that adding a memory component yields better performance on long-tailed or imbalanced class classification problems. We do generally see this phenomenon in Table 5, as the difference is around 4 F1 and 4.5 F1 on the top 10 labels and the rest of the data. This difference is slightly more pronounced in the top 5, as we see a difference of 3.5 F1 and 5.5 F1 respectively. This indicates that baseline models perform slightly worse on the infrequent classes, whereas TTM-RE's memory component can help alleviate this performance drop.

Extremely Unlabeled Setting Wang et al. (2022b) introduced an "extremely unlabeled" scenario, that reduced the training labels to a mere 19% of the original labels. We also evaluate our

Table 6: Extremely unlabeled scenario (with less than 19% of the original training labels as proposed by Wang et al. (2022b)). Standard deviations are shown with 5 random seed runs.

Model	F1	Ign F1	Precision	Recall
Human Annotation Only				
ATLOP	18.17 ± 0.66	18.14 ± 0.33	91.67 ± 3.39	11.16 ± 1.38
SSR-PU	52.78 ± 0.46	51.53 ± 0.41	46.12 ± 0.57	61.69 ± 0.78
TTM-RE	52.60 ± 0.42	51.30 ± 0.43	43.97 ± 0.66	65.45 ± 0.58
Human Annotation + Distantly Supervised				
ATLOP	19.16 ± 0.23	19.01 ± 0.46	96.36 ± 3.41	8.51 ± 2.29
SSR-PU	54.34 ± 0.21	54.12 ± 0.50	87.40 ± 0.61	39.43 ± 1.01
TTM-RE	66.47 ± 0.18	66.04 ± 0.46	81.40 ± 0.85	56.17 ± 0.94

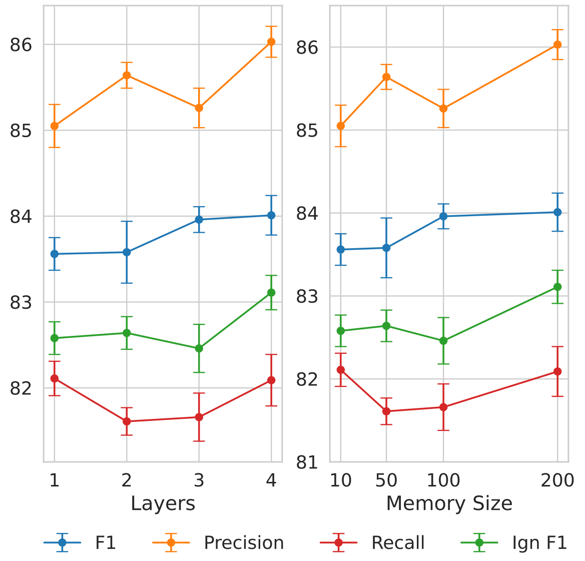

Figure 4: **Left Figure**: Effect of the size of the number of layers in the memory encoder. More layers imply a more powerful memory module. **Right Figure**: Effect of the number of memory tokens (Memory Size) available to be used in TTM-RE on the test dataset of ReDocRED.

model on the extremely unlabeled setting (19%) of the original training triples in ReDocRED (Wang et al., 2022b) (Table 6). We again see that TTM-RE does not work better than baselines on fully supervised, yet increases to 12 F1 points over the best baseline when allowed to train on distantly supervised data. We hypothesize that this is due to the better learning of the infrequent classes as shown in Table 5.

Memory Token Size In Figure 4, we generally see an improvement in model performance (F1, Precision, and Ign F1) when increasing the READ module as well as the memory token size of the Token Turing Machine. While we halted at 4 layers and 200 tokens due to computational constraints, this trend is promising as it suggests that there are potential performance improvements awaiting exploration in future endeavors, albeit requiring increased computational resources.

Table 7: Comparison of RoBERTa-large vs the more recent DeBERTaV3-large on the test dataset of ReDocRED.

Model	F1	Ign F1	Precision	Recall
Human Annotation Only (RoBERTa-large)				
SSR-PU	80.18 ± 0.31	78.61 ± 0.46	69.43 ± 0.43	90.50 ± 0.53
TTM-RE	79.95 ± 0.13	78.20 ± 0.34	85.81 ± 0.55	76.68 ± 0.22
Human Annotation Only (DeBERTaV3-large)				
SSR-PU	78.73 ± 0.25	77.05 ± 0.12	74.25 ± 0.16	83.79 ± 0.10
TTM-RE	78.88 ± 0.22	77.29 ± 0.19	75.63 ± 0.21	82.42 ± 0.18
Human + Distant (RoBERTa-large)				
SSR-PU	80.52 ± 0.43	78.84 ± 0.31	74.24 ± 0.44	87.96 ± 0.51
TTM-RE	84.01 ± 0.21	83.11 ± 0.37	86.03 ± 0.34	82.09 ± 0.27
Human + Distant (DeBERTaV3-large)				
SSR-PU	79.65 ± 0.27	78.34 ± 0.23	82.65 ± 0.27	78.49 ± 0.19
TTM-RE	80.56 ± 0.16	79.49 ± 0.21	83.44 ± 0.24	77.88 ± 0.25

Using DebertaV3 as the Base Model: Interestingly, all baselines generally rely on Roberta-large as the base model. We also explored using DebertaV3-large, which is presented as a more recent and powerful model due to its larger parameter count and higher performance on the GLUE benchmark (improvements include disentangling attention, an enhanced decoding layer (He et al., 2020), and Electra-style pretraining (He et al., 2021)). However, from Table 7, we see that for document RE, it surprisingly *does not improve* performance. Because of this observation, the TTM-RE also uses Roberta-large. Additionally, this demonstrates a case where naively adding parameters *does not* help improve relation classification performance, whereas adding the memory mechanism *does*.

6 Conclusion

In this paper, we investigated TTM-RE, integrating TTMs relation classification and evaluated our model on ReDocRED and ChemDisGene RE datasets. We observed that adding this memory component improves performance overall when trained on a large dataset but specifically improves learning over infrequent relation classes. We also observed that TTMs necessitate either fine-tuning on a large distantly-labeled training dataset or a significantly large human-annotated training dataset (ChemDisGene) to optimize memory vector initialization. Finally, we see that adding the TTM memory module is much better than simply using a larger base model (Deberta V3), which is highly encouraging. We hope that our findings will pave the way for future exploration of memory-augmented techniques in large language models for information extraction tasks.

Limitations

Although we investigated multiple different LLMs and parameters and the type relation distribution for relation prediction as well as addressing the false positives, the performance we attained is still limited compared to supervised methods on the same task. Relation prediction still requires a large amount of data, despite TTM-RE's ability to use distantly supervised data. Future work should seek to tackle this approach that combines labeled data creation with SOTA document relation extraction models for maximum efficiency on human annotators.

Ethical Statement

Based on the methodology we have currently employed, we do not foresee any significant ethical concerns. All the documents and models utilized in our study were obtained from open-source domains, ensuring a transparent and accessible source of information. Additionally, TTM-RE is trained on purely open-source document relation extraction data, eliminating the risk of privacy leakage. Additionally, the task of relation extraction is a widely recognized and well-studied problem across various natural language processing applications.

However, it is crucial to acknowledge a minor factor, namely the presence of potential hidden biases within the pre-trained language models used in our analysis. These biases may stem from the data on which the models were trained, which could have inadvertently introduced implicit human biases. While our usage of these pre-trained language models enables us to identify relationships between arbitrary entities, it is conceivable that biases may emerge if one were to explore sensitive relation classes and entities.

ChatGPT and Grammarly were used for parts of the writing.

In total, training took more than 70 hrs on NVIDIA RTX A6000 for pretraining in total. The main roadblock was the distantly supervised finetuning portion for all of the models, due to the size of the dataset.

Code will be released at https://anonymous.github.com/ttm-re

Derivatives of data accessed for research purposes should not be used outside of research contexts.

References

Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. 2020. Tacred revisited: A thorough evaluation of the tacred relation extraction task. *arXiv preprint arXiv:2004.14855*.

Manuele Barraco, Sara Sarto, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. 2023. With a little help from your own past: Prototypical memory networks for image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3021–3031.

Nontawat Charoenphakdee and Masashi Sugiyama. 2019. Positive-unlabeled classification under class prior shift and asymmetric error. In *Proceedings of the 2019 SIAM International Conference on Data Mining*, pages 271–279. SIAM.

Wenhu Chen, Pat Verga, Michiel De Jong, John Wieting, and William Cohen. 2022. Augmenting pre-trained language models with qa-memory for open-domain question answering. *arXiv preprint arXiv:2204.04581*.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. *arXiv preprint arXiv:1909.00228*.

Allan Peter Davis, Cynthia J Grondin, Robin J Johnson, Daniela Sciaky, Jolene Wiegers, Thomas C Wiegers, and Carolyn J Mattingly. 2021. Comparative toxicogenomics database (ctd): update 2021. *Nucleic acids research*, 49(D1):D1138–D1143.

Michiel De Jong, Yury Zemlyanskiy, Nicholas FitzGerald, Fei Sha, and William Cohen. 2021. Mention memory: incorporating textual knowledge into transformers through entity mention attention. *arXiv preprint arXiv:2110.06176*.

Bayu Distiawan, Gerhard Weikum, Jianzhong Qi, and Rui Zhang. 2019. Neural relation extraction for knowledge base enrichment. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 229–240.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Marthinus C du Plessis, Gang Niu, and Masashi Sugiyama. 2014. Analysis of learning from positive and unlabeled data. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Chufan Gao, Xulin Fan, Jimeng Sun, and Xuan Wang. 2023. Promptre: Weakly-supervised document-level relation extraction via prompting-based data programming. *arXiv preprint arXiv:2310.09265*.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Zhijiang Guo, Yan Zhang, and Wei Lu. 2019. Attention guided graph convolutional networks for relation extraction. *arXiv preprint arXiv:1906.07510*.

Xu Han, Tianyu Gao, Yankai Lin, Hao Peng, Yaoliang Yang, Chaojun Xiao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2020. More data, more relations, more context and more openness: A review and outlook for relation extraction. *arXiv preprint arXiv:2004.03186*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Yi Yao Huang and William Yang Wang. 2017. Deep residual learning for weakly-supervised relation extraction. *arXiv preprint arXiv:1707.08866*.

Jing Jiang. 2009. Multi-task transfer learning for weakly-supervised relation extraction. ACL.

Ryuichi Kiryo, Gang Niu, Marthinus C du Plessis, and Masashi Sugiyama. 2017. Positive-unlabeled learning with non-negative risk estimator. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Qi Li, Xuan Wang, Yu Zhang, Fei Ling, Cathy H Wu, and Jiawei Han. 2018. Pattern discovery for wide-window open information extraction in biomedical literature. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 420–427. IEEE.

Youmi Ma, An Wang, and Naoaki Okazaki. 2023. Dreeam: Guiding attention with evidence for improving document-level relation extraction. *arXiv preprint arXiv:2302.08675*.

Guoshun Nan, Zhijiang Guo, Ivan Sekulić, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. *arXiv preprint arXiv:2005.06312*.

Van-Thuy Phi, Joan Santoso, Masashi Shimbo, and Yuji Matsumoto. 2018. Ranking-based automatic seed selection and noise reduction for weakly supervised relation extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 89–95.

Marthinus Du Plessis, Gang Niu, and Masashi Sugiyama. 2015. Convex formulation for learning from positive and unlabeled data. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1386–1394, Lille, France. PMLR.

Meng Qu, Xiang Ren, Yu Zhang, and Jiawei Han. 2018. Weakly-supervised relation extraction by pattern-enhanced embedding learning. In *Proceedings of the 2018 World Wide Web Conference*, pages 1257–1266.

Michael S Ryoo, Keerthana Gopalakrishnan, Kumara Kahatapitiya, Ted Xiao, Kanishka Rao, Austin Stone, Yao Lu, Julian Ibarz, and Anurag Arnab. 2023. Token turing machines. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19070–19081.

Sunil Kumar Sahu, Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Inter-sentence relation extraction with document-level graph convolutional neural network. *arXiv preprint arXiv:1906.04684*.

Oscar Sainz, Oier Lopez de Lacalle, Gorka Labaka, Ander Barrena, and Eneko Agirre. 2021. Label verbalization and entailment for effective zero- and few-shot relation extraction. *arXiv preprint arXiv:2109.03659*.

Qingyu Tan, Ruidan He, Lidong Bing, and Hwee Tou Ng. 2022a. Document-level relation extraction with adaptive focal loss and knowledge distillation. *arXiv preprint arXiv:2203.10900*.

Qingyu Tan, Ruidan He, Lidong Bing, and Hwee Tou Ng. 2022b. Document-level relation extraction with adaptive focal loss and knowledge distillation. *arXiv preprint arXiv:2203.10900*.

Qingyu Tan, Lu Xu, Lidong Bing, Hwee Tou Ng, and Sharifah Mahani Aljunied. 2022c. Revisiting docred-addressing the false negative problem in relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8472–8487.

Hengzhu Tang, Yanan Cao, Zhenyu Zhang, Jiangxia Cao, Fang Fang, Shi Wang, and Pengfei Yin. 2020. Hin: Hierarchical inference network for document-level relation extraction. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 197–209. Springer.

Yun Tang, Jing Huang, Guangtao Wang, Xiaodong He, and Bowen Zhou. 2019. Orthogonal relation transforms with graph context modeling for knowledge graph embedding. *arXiv preprint arXiv:1911.04910*.

G Veena, S Athulya, Salma Shaji, and Deepa Gupta. 2017. A graph-based relation extraction method for question answering system. In *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 944–949. IEEE.

Patrick Verga, Emma Strubell, and Andrew McCallum. 2018. Simultaneously self-attending to all mentions for full-abstract biological relation extraction. *arXiv preprint arXiv:1802.10569*.

Chenguang Wang, Xiao Liu, Zui Chen, Haoyun Hong, Jie Tang, and Dawn Song. 2022a. Deepstruct: Pretraining of language models for structure prediction. *arXiv preprint arXiv:2205.10475*.

Xuan Wang, Yu Zhang, Qi Li, Yinyin Chen, and Jiawei Han. 2018. Open information extraction with meta-pattern discovery in biomedical literature. In *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 291–300.

Ye Wang, Xinxin Liu, Wenxin Hu, and Tao Zhang. 2022b. A unified positive-unlabeled learning framework for document-level relation extraction with different levels of labeling. *arXiv preprint arXiv:2210.08709*.

Chih-Hsuan Wei, Alexis Allot, Robert Leaman, and Zhiyong Lu. 2019. Pubtator central: automated concept annotation for biomedical full text articles. *Nucleic acids research*, 47(W1):W587–W593.

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

Yuxiang Wu, Yu Zhao, Baotian Hu, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2022. An efficient memory-augmented transformer for knowledge-intensive nlp tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5184–5196.

Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. 2021. Entity structure within and throughout: Modeling mention dependencies for document-level relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14149–14157.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. Docred: A large-scale document-level relation extraction dataset. *arXiv preprint arXiv:1906.06127*.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential reasoning learning for language representation. *arXiv preprint arXiv:2004.06870*.

Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. 2020. Double graph based reasoning for document-level relation extraction. *arXiv preprint arXiv:2009.13752*.

Dongxu Zhang, Sunil Mohan, Michaela Torkar, and Andrew McCallum. 2022a. A distant supervision corpus for extracting biomedical relationships between chemicals, diseases and genes. In *Proceedings of The 13th Language Resources and Evaluation Conference*, Marseille, France. European Language Resources Association.

Dongxu Zhang, Sunil Mohan, Michaela Torkar, and Andrew McCallum. 2022b. A distant supervision corpus for extracting biomedical relationships between chemicals, diseases and genes. *arXiv preprint arXiv:2204.06584*.

Zexuan Zhong, Tao Lei, and Danqi Chen. 2022. Training language models with memory augmentation. *arXiv preprint arXiv:2205.12674*.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 2021a. Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14612–14620.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 2021b. Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14612–14620.

A Parameter Settings

All models were run on an NVIDIA A6000 with 48 gigabytes of VRAM. Still, around 10 days were required to fully run the experiments. For particularly expensive computations, like $Logits_{SR}$, only the fastest model–UnifiedQA-large–could be feasibly run.

All models were downloaded from Huggingface (Wolf et al., 2019). We used the default setup of the pre-trained models and did not do further fine-tuning. All the step mentioned in the methodology section works on the output of the pre-trained models.

Supervised results DREEEAM (Ma et al., 2023) and KD-DocRE (Tan et al., 2022a) were taken from the original source papers.

B Evaluation Metrics

To keep in tradition with existing document relation extraction work, we report both F1 and Ign_F1 as computed by the official metrics from ReDocRED. F1 refers to micro-averaged F1 score that combines precision P and recall R

$$F1 = \frac{2PR}{P+R}$$

$$P = \frac{\texttt{length of correct (h,t,rel) preds}}{\texttt{length of all (h,t,rel) preds}}$$

$$R = \frac{\texttt{length of correct (h,t,rel) preds}}{\texttt{length of correct (h,t,rel)}}$$

Where `(h,t,rel)` denotes a tuple of the predicted head, tail, and relation. Ign_F1 is computed similarly to above but ignores the samples in the DocRED's distantly supervised training set. (Note that we do not use any distantly labeled data).

C Ablation Tables

Tables 8 and 9 for Figure 4.

Table 8: Ablation regarding the number of layers in the memory encoder. More layers imply a more powerful memory module. Results are evaluated from the test dataset of ReDocRED.

Num Layers	F1	Ign F1	Precision	Recall
1	83.56 ± 0.19	82.58 ± 0.19	85.05 ± 0.25	82.11 ± 0.20
2	83.58 ± 0.36	82.64 ± 0.19	85.64 ± 0.15	81.61 ± 0.16
3	83.96 ± 0.15	82.46 ± 0.28	85.26 ± 0.23	81.66 ± 0.28
4	84.01 ± 0.23	83.11 ± 0.20	86.03 ± 0.18	82.09 ± 0.30

Table 9: Effect of the size of the number of memory tokens available to be used in `TTM-RE` on the test dataset of ReDocRED.

Mem. Size	F1	Ign F1	Precision	Recall
10	83.20 ± 0.23	82.24 ± 0.16	85.31 ± 0.12	81.18 ± 0.13
50	83.52 ± 0.21	82.59 ± 0.14	85.65 ± 0.31	81.50 ± 0.17
100	83.67 ± 0.19	82.82 ± 0.22	87.01 ± 0.24	80.57 ± 0.20
200	84.01 ± 0.17	83.11 ± 0.20	86.03 ± 0.19	82.09 ± 0.13

D PCA Plots of Memory Tokens

Shown in Figure 5, we see that the token embeddings lie in a scattered space around the head entities. This makes intuitive sense as the prototypical tokens should capture a diverse set of different types of head tokens.

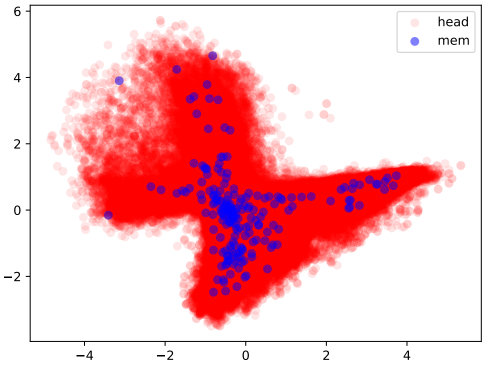

Figure 5: Plot of PCA-transformed head entities along with (200) memory entities. Tail entities are omitted due to redundancy.

E Literature Review Continued

Pre-trained language models, such as BERT-based architectures (Xu et al., 2021), have shown considerable efficacy in document-level relation extraction. BERT-based methodologies have integrated approaches like hierarchical inference networks (Tang et al., 2020), enhanced co-reference reasoning (Ye et al., 2020), and adaptive thresholding. Furthermore, graphical neural networks (GNNs) (Zeng et al., 2020) have been leveraged for document-level relation extraction, employing techniques such as feature learning on a coreference graph (Sahu et al., 2019), edge-oriented learning strategies (Christopoulou et al., 2019), attention mechanisms (Guo et al., 2019), and iterative refinement methods to aggregate multi-hop information (Nan et al., 2020).

E.1 Weakly Supervised Document Relation Extraction

Past studies in document-level relation extraction have heavily depended on human annotation to create training datasets, a process known for its time-consuming and labor-intensive nature. There has been minimal exploration into document relation extraction methods that do not necessitate human annotation.

Various weakly supervised methods have been extensively investigated for relation extraction (Jiang, 2009; Huang and Wang, 2017; Qu et al., 2018; Wang et al., 2018; Li et al., 2018). For instance, Huang and Wang (2017) employed residual connections and convolutional neural networks (CNNs) to identify pertinent candidates, thereby enhancing supervised relation classification. In a similar vein, Qu et al. (2018) extracted textual patterns from initial examples to offer supplementary supervision. Introducing a ranking-based approach for seed selection, Phi et al. (2018) improved bootstrapping and distantly supervised relation extraction. Additionally, Sainz et al. (2021) proposed representing each relation class using a label verbalizer and tackled the relation extraction task with a textual entailment model.

Moreover, Wang et al. (2022b) showed an "extremely unlabeled" scenario wherein each relation type comprised only one instance, consequently reducing the training set to a smaller number of labeled relation triplets. However, this scenario does not help in *improving* performance on the fully supervised test data overall.

Qu et al. (2018) derived textual patterns from initial samples and employed them as weak signals for relation extraction. Gao et al. (2023) investigated purely weakly-supervised prompting methods devoid of human labels and revealed significant limitations in relying solely on weak supervision, particularly in handling a high incidence of hallucinations when predicting no-relation entity-entity pairs.

F SSR-PU Loss

F.1 Class-shift Adjusted Positive Unlabeled Loss Function (SSR-PU)

Previous supervised document-level RE methods only treated unlabeled relations as negative samples, which may lead to low recall in the presence of a large number of false negatives. To address this problem, we adopt PU learning with prior shift similar to Wang et al. (2022b) for each class (Plessis et al., 2015; du Plessis et al., 2014).

PU learning assumes that unlabeled data can reflect the true overall distribution, that is, $p_{\mathrm{U}_i}(\boldsymbol{x}) = p_i(\boldsymbol{x})$. The expected classification risk formulation can be defined in a form that can be approximated using the data like so:

$$\begin{aligned}\widehat{R}_{\mathrm{PU}}(f) = \sum_{i=1}^{K}(&\frac{\pi_i}{n_{\mathrm{P}_i}}\sum_{j=1}^{n_{\mathrm{P}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{P}_i}),+1)\\ &+\max(0,[\frac{1}{n_{\mathrm{U}_i}}\sum_{j=1}^{n_{\mathrm{U}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{U}_i}),-1) \quad (1)\\ &-\frac{\pi_i}{n_{\mathrm{P}_i}}\sum_{j=1}^{n_{\mathrm{P}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{P}_i}),-1)]))\end{aligned}$$

where $\pi_i = p(y_i = +1)$ denotes probability of positive prior for relation class i. n_{P_i} are the number of positive and n_{U_i} are the unlabelled samples of class i, respectively. ℓ is a convex loss function, and $f_i(\cdot)$ is a score function that predicts class i. $\boldsymbol{x}_j^{\mathrm{P}_i}$ and $\boldsymbol{x}_j^{\mathrm{U}_i}$ denotes that the j-th sample of class i is positive and unlabeled as class i respectively. Note that without the max function, the second term in Eq.1 can be negative and can be prone to overfitting (and therefore highly negative) when using a highly flexible model. Thus, a non-negative risk component (Kiryo et al., 2017) is used to solve the overfitting problem. Note that n_{U_i} is essentially a hyperparameter that one assumes before model training (n_{P_i}, π_i can be directly calculated by counting, and everything else is learned via backprop).

While the original method additionally corrected for the heavy class imbalance problem via multiplying $\gamma_i = \frac{1-\pi_i}{\pi_i})^{0.5}$ before positive risk estimations as the class weight, we found that this was unnecessary and that tuning the other hyper-parameters was sufficient in reproducing the original paper results.

Prior Shift: Ordinary PU learning requires an assumption that the overall distribution needs to be the same as the distribution of the unlabeled data. In contrast, with the document-level RE dataset constructed by a *recommend-revise* scheme, where Wang et al. (2022b) found that there existed a prior shift in the unlabeled data of the training data vs the test data. When these priors are different, ordinary PU learning will yield a biased result.

To address this problem, inspired by the method (Charoenphakdee and Sugiyama, 2019) for handling a prior shift between the test set and the training set, a correction term is added. For each class, assume that the original prior $\pi_i = p(y_i = +1)$. Let $\pi_{labeled,i} = p(s_i = +1)$ and $(1 - \pi_{labeled,i}) = (1 - p(s_i = +1)) = p(s_i = -1)$ where $s_i = +1$ or $s_i = -1$ mean that the i-th class is labeled or unlabeled, respectively.

The conditional probability of a positive sample under unlabeled data is:

$$\begin{aligned}\pi_{u,i} &= p(y_i = 1 \mid s_i = -1)\\ &= \frac{p(y_i = 1, s_i = -1)}{p(s_i = -1)}\\ &= \frac{p(y_i = 1) - p(s_i = +1)}{p(s_i = -1)}\\ &= \frac{\pi_i - \pi_{labeled,i}}{1 - \pi_{labeled,i}}\end{aligned} \tag{2}$$

where Step 3 is true because positive samples are assumed to be a superset of the unlabelled data and negative data has no overlaps with the labeled data. I.e. $p(y_i = -1, s_i) = 0$. Finally, the non-negative risk estimator (Kiryo et al., 2017) under class prior shift of training data is obtained as follows:

$$\begin{aligned}\widehat{R}_{\mathrm{S-PU}}(f) &= \sum_{i=1}^{K}\Big(\frac{\pi_i}{n_{\mathrm{P}_i}}\sum_{j=1}^{n_{\mathrm{P}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{P}_i}), +1)\\ &+ \max(0, [\frac{1}{n_{\mathrm{U}_i}}\frac{1-\pi_i}{1-\pi_{u,i}}\sum_{j=1}^{n_{\mathrm{U}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{U}_i}), -1)\\ &- \frac{1}{n_{\mathrm{P}_i}}\frac{\pi_{u,i} - \pi_{u,i}\pi_i}{1-\pi_{u,i}}\sum_{j=1}^{n_{\mathrm{P}_i}}\ell(f_i(\boldsymbol{x}_j^{\mathrm{P}_i}), -1)])\Big)\end{aligned} \tag{3}$$

The proof is shown in Theorem 1 in Wang et al. (2022b).