Tracing Origins: Coreference-aware Machine Reading Comprehension

Baorong Huang[1,#], Zhuosheng Zhang[2,3,#], Hai Zhao[2,3,*]
[1]Institute of Corpus Studies and Applications, Shanghai International Studies University
[2]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[3]Key Laboratory of Shanghai Education Commission for Intelligent Interaction
and Cognitive Engineering, Shanghai Jiao Tong University
0214101730@shisu.edu.cn, zhangzs@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn

Abstract

Machine reading comprehension is a heavily-studied research and test field for evaluating new pre-trained language models (PrLMs) and fine-tuning strategies, and recent studies have enriched the pre-trained language models with syntactic, semantic and other linguistic information to improve the performance of the models. In this paper, we imitate the human reading process in connecting the anaphoric expressions and explicitly leverage the coreference information of the entities to enhance the word embeddings from the pre-trained language model, in order to highlight the coreference mentions of the entities that must be identified for coreference-intensive question answering in QUOREF, a relatively new dataset that is specifically designed to evaluate the coreference-related performance of a model. We use two strategies to fine-tune a pre-trained language model, namely, placing an additional encoder layer after a pre-trained language model to focus on the coreference mentions or constructing a relational graph convolutional network to model the coreference relations. We demonstrate that the explicit incorporation of coreference information in the fine-tuning stage performs better than the incorporation of the coreference information in pre-training a language model.

1 Introduction

Machine reading comprehension (MRC), a task that automatically identifies one or multiple words from a given passage as the context to answer a specific question for that passage, is widely used in information retrieving, search engines, and dialog systems. Several datasets on MRC that limit the answer to one single word or multiple words from the passage are introduced, including TREC (Harman, 1993), SQuAD (Rajpurkar et al., 2018), NewsQA (Trischler et al., 2017), SearchQA (Dunn et al., 2017), and QuAC (Choi et al., 2018), and intensive efforts were made to build new models that surpass the human performance on these datasets, including the pre-trained language models (Devlin et al., 2019; Liu et al., 2019; Yang et al., 2019a) or the ensemble models that outperform the human, in particular on SQuAD (Lan et al., 2020; Yamada et al., 2020; Zhang et al., 2021). More challenging datasets are also introduced, which require several reasoning steps to answer (Yang et al., 2018; Qi et al., 2021), the understanding of a much larger context (Kočiský et al., 2018) or the understanding of the adversarial content and numeric reasoning (Dua et al., 2019).

Context: ***Frankie Bono****, a mentally disturbed hitman from Cleveland, comes back to his hometown in New York City during Christmas week to kill a middle-management mobster, Troiano. ...First* ***he*** *follows his target to select the best possible location, but opts to wait until Troiano isn't being accompanied by his bodyguards. ... Losing his nerve,* ***Frankie*** *calls up his employers to tell them he wants to quit the job. Unsympathetic, the supervisor tells him* ***he*** *has until New Year's Eve to perform the hit.*
Question: *What is the first name of the person who has until New Year's Eve to perform a hit?* Answer: he ->Frankie
Question: *What is the first name of the person who follows their target to select the best possible location?* Answer: he ->Frankie

Table 1: An example from QUOREF: coreference resolution is required to extract the correct answer. We highlight the supporting text in teal color and the related deictic expressions in **bold**.

Human texts, especially long texts, are abound in deictic and anaphoric expressions that refer to the entities in the same text. These deictic and anaphoric expressions, in particular, constrain the generalization of the models trained without explicit awareness of the coreference. The QUOREF dataset (Dasigi et al., 2019) is specifically designed to validate the performance

[*]Corresponding author. # Equal contribution. This work was supported in part by the Key Projects of National Natural Science Foundation of China under Grants U1836222 and 61733011.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1281 - 1292
May 22-27, 2022 ©2022 Association for Computational Linguistics

of the models in coreferential reasoning, in that "78% of the manually analyzed questions cannot be answered without coreference" (Dasigi et al., 2019). The example in Table 1 shows that the answers to the two questions cannot be directly retrieved from the sentences because the word in the corresponding sentence of the context is an anaphoric pronoun ***he***, and to obtain the correct answers, tracing of its antecedent ***Frankie*** is required. The reasoning in coreference resolution is required to successfully complete the task in machine reading comprehension in the SQuAD-style QUOREF dataset.

Pre-trained language models, including BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and XLNet (Yang et al., 2019b), that are trained through self-supervised language modeling objectives like masked language modeling, perform rather poorly in the QUOREF dataset. We argue that the reason for the poor performance is that those pre-trained language models do learn the background knowledge for coreference resolution but may not learn adequately the coreference information required for the coreference-intensive reading comprehension tasks. In the human reading process, as shown in the empirical study of first-year English as a second language students during the reading of expository texts, "anaphoric resolution requires a reader to perform a text-connecting task across textual units by successfully linking an appropriate antecedent (among several prior antecedents) with a specific anaphoric referent" and "students who were not performing well academically were not skilled at resolving anaphors" (Pretorius, 2005) and the direct instruction on anaphoric resolution elevated the readers' comprehension of the text (Baumann, 1986). In addition, the studies on anaphor resolution in both adults using eye movement studies (Duffy and Rayner, 1990; van Gompel et al., 2004) and children (Joseph et al., 2015) evidenced a two-stage model of anaphor resolution proposed by Garrod and Terras(Garrod and Terras, 2000). The first stage is "an initial lexically driven, context-free stage known as bonding, whereby a link between the anaphor and a potential antecedent is made, followed by a later process known as resolution, which resolves the link with respect to the overall discourse context" (Joseph et al., 2015). The pre-trained language models only capture the semantic representations of the words and sentences, without explicitly performing such text-connecting actions in the specific coreference-intensive reading comprehension task, thus they do not learn adequate knowledge to solve the complex coreference reasoning problems.

Explicitly injecting external knowledge such as linguistics and knowledge graph entities, has been shown effective to broaden the scope of the pre-trained language models' capacity and performance, and they are often known as X-aware pre-trained language models (Zhang et al., 2020; Liu et al., 2020; Kumar et al., 2021). It is plausible that we may imitate the anaphoric resolution process in human's two-stage reading comprehension of coreference intensive materials and explicitly make the text-connecting task in our fine-tuning stage as the second stage in the machine reading comprehension.

As an important tool that captures the anaphoric relationship between words or phrases, coreference resolution that clusters the mentions of the same entity within a given text is an active field in natural language processing (Chen et al., 2011; Sangeetha, 2012; Huang et al., 2019; Joshi et al., 2020; Kirstain et al., 2021), with neural networks taking the lead in the coreference resolution challenges. The incorporation of the coreference resolution results in the pre-training to obtain the coreference-informed pre-trained language models, such as CorefBERT and CorefRoBERTa (Ye et al., 2020), has shown positive improvements on the QUOREF dataset, a dataset that is specially designed for measuring the models' coreference capability, but the performance is still considerably below the human performance.

In this paper, we make a different attempt to leverage the coreference resolution knowledge and complete the anaphoric resolution process in reading comprehension. We propose a fine-tuned coref-aware model that directly instructs the model to learn the coreference information[1]. Our model can be roughly divided into three major components: 1) pre-trained language model component. We use the contextualized representations from the pre-trained language models as the token embeddings for the downstream reading comprehension tasks. 2) coreference resolution component. NeuralCoref, an extension to the spaCy, is applied here to extract the mention clusters from the context. 3)

[1] Our codes are publicly available at `https://github.com/bright2013/CorefAwareMRC`.

coreference enrichment component. We apply three methods in incorporating the coreference knowledge: additive attention enhancement, multiplication attention enhancement, and relation-enhanced graph-attention network + fusing layer.

In this paper, we show that by simulating the human behavior in explicitly connecting the anaphoric expressions to the antecedent entities and infusing the coreference knowledge our model can surpass that of the pre-trained coreference language models on the QUOREF dataset.

2 Background and Related Work

2.1 Models and Training Strategies

Recent studies on machine reading comprehension mainly rely on the neural network approaches. Before the prevalence of the pre-trained language models, the main focus was to guide and fuse the attentions between questions and paragraphs in their models, in order to gain better global and attended representation (Huang et al., 2018; Hu et al., 2018; Wang et al., 2018).

After the advent of the BERT (Devlin et al., 2019), there were two trends in solving the machine reading comprehension. The first trend was to develop better pre-trained language models that captured the representation of contexts and questions (Liu et al., 2019; Yang et al., 2019a; Lewis et al., 2020), and more datasets on question answering were introduced to increase the difficulty in this task, including NewsQA (Trischler et al., 2017), SearchQA (Dunn et al., 2017), QuAC (Choi et al., 2018), HotpotQA (Yang et al., 2018), NarrativeQA (Kočiský et al., 2018), DROP (Dua et al., 2019), and BeerQA (Qi et al., 2021).

However, the raw pre-trained language models, being deprived of the in-domain knowledge, the structures and the reasoning capabilities required for the datasets, often perform unsatisfactorily in the hard datasets, being significantly below the human performance. Efforts had been made to boost the model performance by enriching the pre-trained language models with specific syntactic information (Ye et al., 2020) or semantic information. Another trend was to fine-tune the pre-trained language model and added additional layers to incorporate task-specific information for better representation, in particular, the coreference information (Ouyang et al., 2021; Liu et al., 2021). For some questions that have multi-span answers, in other words, a single answer contains two or more discontinuous entities in the context, the BIO (B denotes the start token of the span; I denotes the subsequent tokens and O denotes tokens outside of the span) tagging mechanism is used to identify these answers and improve the model performance (Segal et al., 2020).

Recent studies also explored the possibilities of prompt-based learning in machine reading comprehension, including a new pre-training scheme that changed the question answering into a few-shot span selection model (Ram et al., 2021) and a new model that fine-tuned the prompts with knowledge (Chen et al., 2021). The performance of the models using prompt-based learning is significantly higher than the baseline models, but is still below that of the fine-tuned models (Chen et al., 2021).

2.2 Graph Neural Network in Machine Reading Comprehension

Graph neural network (GNN) captures the relations among the entities in the text by modeling the entities as nodes in the graph and learning the weights via the message passing between the nodes of the graph (Kipf and Welling, 2017; Velickovic et al., 2018). As the dependencies in the natural language text, the relations among entities and knowledge-base triples can be relatively easily modeled in a graph structure, graph neural networks are used for numeric reasoning (Ran et al., 2019), for multi-document question answering by connecting mentions of candidate answers (De Cao et al., 2019), and for multi-hop reasoning by adding the edges with co-occurrence relations(Qiu et al., 2019), or with contextual sentences as embeddings (Tu et al., 2020), or with a hierarchical paragraph-sentence-entity graph (Fang et al., 2020), but none of them had attempted to connect the anaphoric expressions and their antecedents as a coreference resolution strategy in a graph neural network for machine reading comprehension.

3 Coreference-aware Machine Reading Comprehension

Our model, inspired by the anaphoric connecting behavior in the human reading comprehension process, consists of four parts, namely, a pre-trained language model, a coreference resolution component, a graph encoder and a fusing layer. Context in the machine reading comprehension task is first processed by a coreference resolution model to identify the underlying coreference clusters,

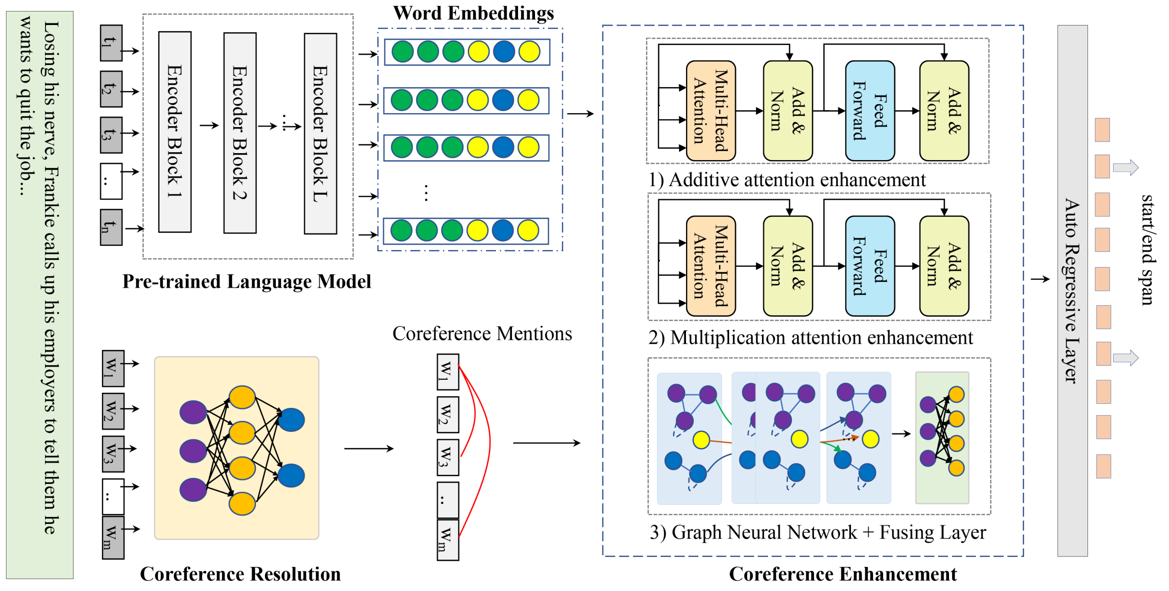

Figure 1: Coref-aware fine-tuning for machine reading comprehension. The text is tokenized and fed into a pre-trained language model to obtain the embeddings, and into a coreference resolution model to obtain coreference information. Both the embeddings and the coreference information are used in the fine-tuning stage to 1) enhance cross attentions with additive operations; 2) enhance cross attentions with multiplication operations, or; 3) construct a coreference graph neural network with the coreference relations as edges.

which are formed by dividing the entities and anaphoric expressions in the context into disjoint groups on the principle that the mentions of the same entity should be in the same group. Then we use the coreference clusters to construct a coreference matrix that labels each individual cluster and identifies each element in the same cluster with the same cluster number. Meanwhile, the context is tokenized by the tokenizer defined in the pre-trained language model and the embeddings for each token are retrieved from that model. We propose three methods for connecting the anaphoric expressions and their antecedent entity: 1) adding the coreference matrix with each attention head in the additional coreference encoder layer; 2) multiplying the coreference matrix with each attention head in the additional coreference encoder layer; 3) constructing a graph neural network based on the coreference matrix with the edges corresponding to the coreference relations and then fusing the graph representation in the graph neural network with the embeddings of the context, as shown in Figure 1. The final representations from either one of the three methods are fed into the classifier to calculate the start/end span of the question.

3.1 Coreference Resolution

Coreference resolution is the process that identifies all the expressions of the same entity in the text, clusters them together as coreference clusters, and locates their spans. For example, after coreference resolution for the text *Losing his nerve, Frankie calls up his employers to tell them he wants to quit the job.*, we obtained two mention clusters *[Frankie: [his, Frankie, his, he], his employers: [his employers, them]]*, where *Frankie* is the head entity and *his, Frankie, his, he* are all the expressions referring to this entity, as shown in Figure 2.

As pre-trained language models use subwords in their tokenization and the coreference resolution uses word in the tokenization, a mapping is required to establish the relations. For the input sequence $X = \{x_1, ...x_n\}$ of length n, the words $W = \{w_1, ..., w_m\}$ obtained from the coreference tokenization are mapped to the corresponding subwords (tokens) $T = \{t_1, ..., t_k\}$ from the tokenizer in the pre-trained language model, with one word contains one or more than one subwords. Then we constructed a coreference array with the

[2] Image generated from https://huggingface.co/coref/

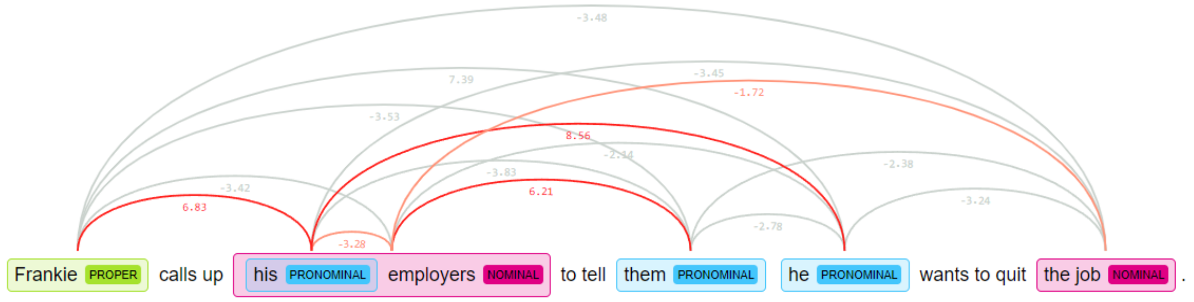

Figure 2: Coreference resolution: the red curves connecting the mentions of the same entity and marking the coreference relations. [2]

following rule:

$$coref(i) = \begin{cases} 0 & \text{if tokens[i]} \notin \mathrm{S_m}, \\ n & \text{if tokens[i]} \in \mathrm{S_m}, \end{cases} \tag{1}$$

where i is the position of the token in the token array, S_m is an array of all words in the coreference mention clusters, n is the sequence number of the mention cluster and $n \geq 1$. Tokens in the same mention cluster have the same sequence number n in the coreference array.

3.2 Graph Neural Network

We use the standard relational graph convolutional network (RGCN) (Sejr Schlichtkrull et al., 2018) to obtain the graph representation of the context enriched with coreference information. We use the coreference matrix and the word embeddings to construct a directed and labeled graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$, with nodes (subwords) $v_i \in \mathcal{V}$, edges(relations) $(v_i, r, v_j)) \in \mathcal{E}$, where $r \in \mathcal{R}$ is one of the two relation types (1 indicates coreference relation and self-loop; 2 indicates global relation), as shown in Figure 3 .

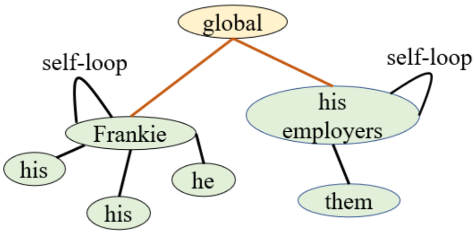

Figure 3: Coreference graph. We connect the entities with their coreference mentions to form a graph, and the nodes are connected to the global node to form global representations.

The constructed graph is then fed into the RGCN, with the differentiable message passing and the basis decomposition to reduce model parameter size and prevent overfitting:

$$h_i^{l+1} = \sigma\big(W_0^{(l)} h_i^{(l)} + \sum_{r \in R} \sum_{j \in N_i^r} \frac{1}{c_{i,r}} W_r^{(l)} h^{(l)}\big),$$
$$W_r^{(l)} = \sum_{b=1}^{B} a_{rb}^{(l)} V_b^{(l)}, \tag{2}$$

where N_i^r denotes the set of neighbor indices of node i under the relation $r \in \mathcal{R}$, $c_{i,r}$ is the normalization constant, and $W_r^{(l)}$ is a linear combination of basis transformation $V_b^{(l)}$ with coefficient $a_{rb}^{(l)}$.

3.3 Coreference-enhanced Attention

In addition to the Graph Neural Network method, we also explore the possibility of using the self-attention mechanism (Vaswani et al., 2017) to explicitly add an encoder layer and incorporate the coreference information into the attention heads of that layer, so as to guide the model to identify the mentions in the cluster as the same entity.

We use two methods to fuse the coreference information and the original embeddings from the pre-trained language model: additive attention fusing and dot product attention fusing (multiplication). Given the coreference array $A = \{m_1, 0, m_1, m_2, 0, m_2, m_3, 0, m_3, m_1 ...\}$, where m_n denotes the nth mention cluster, and 0 denotes no mentions, the enriched attention for additive attention fusing is formulated as:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}} + M_A)V,$$
$$head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V), \tag{3}$$

where M_A is a coreference matrix constructed from the coreference array A with the element value in the matrix calculated by adding (for additive model) or multiplying (for multiplication model) the coreference hyper-parameter $coref_{weight}$ with the original attention weight if the element belongs to the coreference array, Q, K, V are the query, key and value respectively, d_k is the dimension of the keys, and W_i is trainable parameter. For dot product (multiplication) fusing, it is formulated as:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}} \odot M_A)V,$$
$$head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V), \tag{4}$$

where we calculate the dot product of $\frac{QK^T}{\sqrt{d_k}}$ and a coreference matrix M_A constructed from the coreference array A.

3.4 Integration

A machine reading comprehension task expects the model to output the start and end positions of the answer. For the RCGN method, we fuse the hidden state of nodes v_i in the last layer of RCGN and the embeddings from the pre-trained language model with a fully-connected (FC) layer , and then calculate the start/end positions of the answer.

$$\begin{aligned} E &= FC(E_{prLM} || E_{gnn}), \\ P_s &= argmax(softmax(W_s S)), \end{aligned} \tag{5}$$

where E_{prLM} denotes the embeddings from the pre-trained language model, E_{gnn} denotes the embeddings from the graph encoder, P_s denotes the predicted start positions, W_s denotes the weight matrix and S denotes the text feature.

For the two methods that add one additional encoder layer for additive or multiplication attention enrichment, we directly used the output of that encoder layer for the follow-up processing.

Following the practice of CorefRoBERTa (Ye et al., 2020) in handling multiple answers for the same question, we use the cross entropy to calculate the losses for each answer if the question has multiple answers:

$$\begin{aligned} E_n &= FC(E_{prLM}, n), \\ L_s &= \sum_i^n H(p_s i, q_s i), \\ L_e &= \sum_i^n H(p_e i, q_e i), \\ L_{total} &= avg(L_s + L_e + L(E_n, n)), \end{aligned} \tag{6}$$

where n denotes the answer count as a hyper parameter for handling multiple answers, E_n denotes the results after the linear transformation of the embeddings for the answer count and then we obtains the predicted start positions and end positions from that embeddings, $L(E_n, n)$ denotes the cross-entropy loss between the transformed embeddings and the answer count, L_s denotes the total loss of the start positions, L_e denotes the total loss of the end positions and L_{total} denotes the combined total loss.

4 Experiments

4.1 Model Settings

We developed three models based on the sequence-to-sequence Transformer architecture. The pre-trained RoBERTa-large was used as the base model and then we used the following three methods to fine-tune it: 1) Coref$_{\text{GNN}}$: feeding the coreference information into a graph neural network and then fuse the representations; 2) Coref$_{\text{AddAtt}}$: adding the coreference weights with the self-attention weights; 3) Coref$_{\text{MultiAtt}}$: calculating the dot product of the coreference weights with the self-attention weights. We used the results from CorefRoBERTa (Ye et al., 2020) as our baselines.

4.2 Setup

Our coreference resolution was implemented in spaCy (Honnibal and Montani, 2017) and NeuralCoref. NeuralCoref is an extension for spaCy that is trained on the OntoNotes 5.0 dataset based on the training process proposed by Clark and Manning (Clark and Manning, 2016), which identifies the coreference clusters in the text as mentions. In particular, spaCy 2.1.0 and NeuralCoref 4.0 are used, because the latest spaCy version 3.0+ has compatibility issues with NeuralCoref and extra efforts are required to solve the issues.

The neural network implementation was implemented in PyTorch (Paszke et al., 2019) and Hugging Face Transformers (Wolf et al., 2020). We used the embeddings of the pre-trained language model RoBERTa$_{\text{LARGE}}$, with the relational graph convolutional network implemented in Deep Graph Library (DGL) (Wang et al., 2020). We used Adam (Kingma and Ba, 2015) as our optimizer, and the learning-rate was {1e-5, 2e-5, 3e-5}. We trained each model for {4, 6} epochs and selected the best checkpoints on the development dataset with Exact match and F1 scores. All experiments were run on

Model	Dev		Test	
	EM	F1	EM	F1
QANet*	34.41	38.26	34.17	38.90
QANet + $\text{BERT}^*_{\text{BASE}}$	43.09	47.38	42.41	47.20
$\text{BERT}^+_{\text{BASE}}$	61.29	67.25	61.37	68.56
$\text{CorefBERT}^+_{\text{BASE}}$	66.87	72.27	66.22	72.96
$\text{BERT}^+_{\text{LARGE}}$	67.91	73.82	67.24	74.00
$\text{CorefBERT}^+_{\text{LARGE}}$	70.89	76.56	70.67	76.89
$\text{RoBERTa}^+_{\text{LARGE}}$	74.15	81.05	75.56	82.11
$\text{CorefRoBERTa}^+_{\text{LARGE}}$	74.94	81.71	75.80	82.81
$\text{Coref}_{\text{GNN}}$	79.23	85.89	78.60	85.15
$\text{Coref}_{\text{AddAtt}}$	**80.02**	**86.13**	**79.11**	**85.86**
$\text{Coref}_{\text{MultiAtt}}$	79.85	86.02	78.52	85.27

Table 2: Exact Match and F1 scores of baselines and our proposed models. Results with *, + are from Dasigi et al. (2019) and Ye et al. (2020) respectively.

two NVIDIA TITAN RTX GPUs, each with 24GB memory.

4.3 Tasks and Datasets

Our evaluation was performed on the QUOREF dataset (Dasigi et al., 2019). The dataset contains a train set with 3,771 paragraphs and 19,399 questions, a validation set with 454 paragraphs and 2,418 questions, and a test set with 477 paragraphs and 2,537 questions.

4.4 Results

We quantitatively evaluated the three methods and reported the standard metrics: exact match score (EM) and word-level F1-score (F1) (Rajpurkar et al., 2016).

As shown in Table 2, compared with the baseline model CorefRoBERTa, the performance of our models improves significantly. In particular, $\text{Coref}_{\text{AddAtt}}$ performs best with 5.08%, 4.42% improvements over the baseline model in Exact Match and F1 score respectively on the QUOREF dev set, and 3.05% (F1) and 3.31% (Exact Match) improvements on the QUOREF test set. $\text{Coref}_{\text{GNN}}$ and $\text{Coref}_{\text{MultiAtt}}$ also outperform the baseline model by 2.34% (F1) and 2.80% (Exact Match), and 2.46% (F1) and 2.72% (Exact Match) respectively on the test set. Compared with the $\text{RoBERTa}_{\text{LARGE}}$ that does not use any explicit coreference information in the training or the $\text{CorefRoBERTa}_{\text{LARGE}}$ that uses the coreference information in the training, the improvements of our model are higher, which proves the effectiveness of the explicit coreference instructions in our strategies.

5 Analysis

5.1 Model Efficiency

As shown in Table 2, compared with $\text{RoBERTa}_{\text{LARGE}}$, our methods added only one component that explicitly incorporates the coreference information, and the three methods we used all exhibit considerable improvements over the baselines. Compared with $\text{RoBERTa}_{\text{LARGE}}$ which has 354M parameters, $\text{Coref}_{\text{AddAtt}}$ and the $\text{Coref}_{\text{MultiAtt}}$ add an encoder layer, which adds over 12M parameters. For the $\text{Coref}_{\text{GNN}}$ method, we added one hidden layer in GNN and two linear layers to transform the feature dimensions, with around 68.7K parameters in total. Our predictions are that intuitively with more focuses on the coreference clues, the models perform better on the task that requires intensive coreference resolution, as we have explicitly increased the attention weights to connect the words in the same coreference mention clusters. However, the overall performance of the models is also limited by the performance of the coreference component we use, namely, NeuralCoref.

5.2 Case Studies

To understand the model's performance beyond the automated metrics, we analyze our predicted answers qualitatively. Table 3 compares the representative answers predicted by our models and $\text{CorefRoBERTa}_{\text{LARGE}}$. These examples require that the models should precisely locate the entity from several distracting entities for the anaphoric expression that directly answers the questions. Our model demonstrates that, after resolving the anaphoric expression with the antecedents in the context and enhancing with the coreference information by connecting the anaphoric expression with its antecedents, such as the connection from ***her*** to ***Henrietta*** in the first example and the connection from ***she*** to ***Rihanna*** in the second example, our model accurately locates the entity name among several names in the context, which the $\text{CorefRoBERTa}_{\text{LARGE}}$ fails to uncover.

We further explored the effects of the anaphoric connections on the attention weights by comparing the attention weights of the sample in the first row in Table 3 between our $\text{Coref}_{\text{AddAtt}}$ and $\text{CorefRoBERTa}_{\text{LARGE}}$ model, as shown in Figure 4. It is clear that the anaphoric expressions are not connected in the $\text{CorefRoBERTa}_{\text{LARGE}}$ model,

Coref-resolved Context (Abbreviated)	Question	Answers
Henrietta *take an immediate liking to her, and she asks if Luce can sit by* ***her*** *during the wedding. Rachel arrives with her father and the ceremony begins. As Rachel is walking down the aisle, her eyes wander and she makes eye contact with Luce.*	*Rachel makes eye contact with a woman sitting next to whom?*	*Henrietta* (Golden) Rachel (CorefR) Henrietta (C$_{\text{AddAtt}}$)
After the song was completed, they wanted to play it to ***Rihanna****, but Blanco was skeptical about the reaction towards the song because of its slow sound. After StarGate played it to her, they called Blanco from London and told him that* ***she*** *liked the song: "She's flippin' out.*	*Who liked a song?*	*Rihanna* (Golden) Blanco (CorefR) Rihanna (C$_{\text{AddAtt}}$)

Table 3: Comparison of the predictions for two questions in QUOREF dev set. The blue and bold words indicate the mentions in the same coreference cluster obtained from coreference resolution. In the Answers column, Golden indicates the golden answer; CorefR indicates the prediction made by CorefRoBERTa$_{\text{LARGE}}$ model; C$_{\text{AddAtt}}$ indicates the prediction made by Coref$_{\text{AddAtt}}$ model.

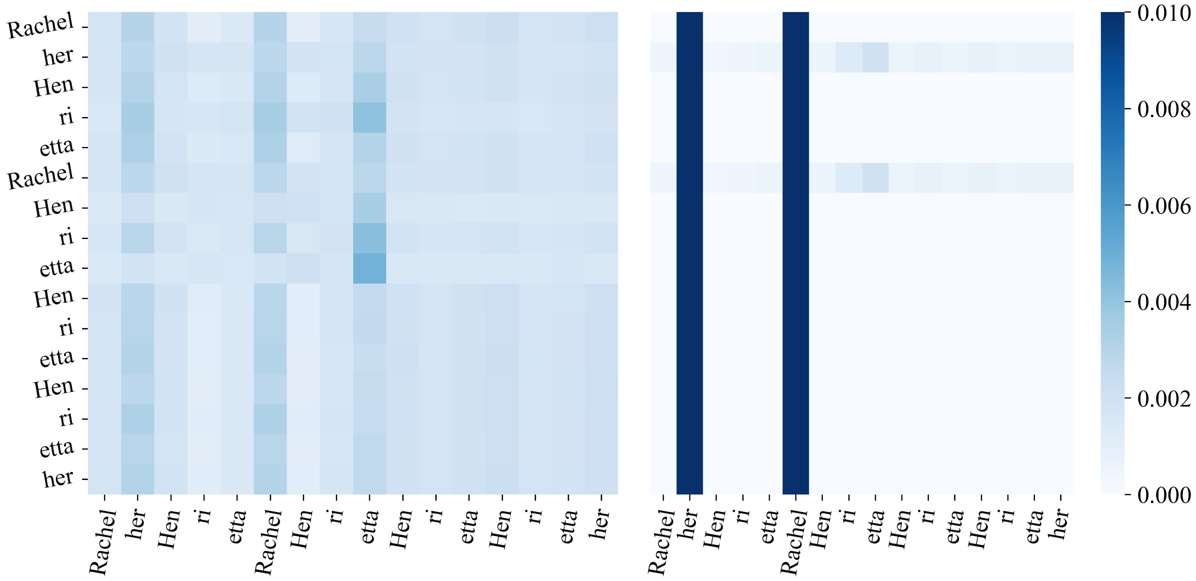

Figure 4: Sample average cross attentions for all heads from Coref$_{\text{AddAtt}}$ model (left) and CorefRoBERTa$_{\text{LARGE}}$ model (right). The cross attentions among the anaphoric expressions and the entities of our model (Coref$_{\text{AddAtt}}$) are visibly much more distinctive than those of the baseline model (CorefRoBERTa$_{\text{LARGE}}$).

as indicated by the obtrusive attentions on **Rachel** and **Her** in the heatmap on the right of the figure. For the Coref$_{\text{AddAtt}}$, the varying colors on the left heat-map indicate the connection strength among the anaphoric expressions and evidence the effects of explicit coreference addition that smooth and strength the attentions for anaphoric expressions, which contributes to the higher performance of our models.

5.3 Error Analysis

Despite the improvements made by our model, it still fails to predict the correct answers for some questions. We analyzed and summarized several error cases as follows.

Table 4 shows three representative types of errors. The first type of errors is caused by the limitations of the coreference resolution component, NeuralCoref, as its performance had not reached 80% in F1 for MUC, B^3 or CEAF$_{\phi 4}$ (Clark and Manning, 2016), which is evidenced by the failure in resolving the antecedent of the anaphoric expression ***its*** as ***the academy*** in the first sample, and the failure in clustering the anaphoric expressions ***her*** with the entity ***Beyoncé*** in the second sample, despite the success in resolving the second ***Gilman*** to its antecedent ***Rockwell "Rocky" Gilman***. The second type of errors is more complicated, which involves multi-step reasoning that cannot be handled by simply adding the coreference information. To correctly answer the second question, the model should perform two successive tasks successfully: 1) it should understand that ***Mathew Knowles*** is the father

Coref-resolved Context (Abbreviated)	Question	Answers
West Point cadet ***Rockwell "Rocky" Gilman*** *is called before a hearing brought after an influential cadet, Raymond Denmore, Jr., is forced to leave the academy...Denmore's attorney, Lew Proctor, attacking* ***the academy*** *and* ***its*** *Honor Code system, declares that* ***Gilman*** *is unfit and possibly criminally liable.*	*Who's honor code system does Proctor attack?*	*the academy* (Golden) West Point ($\mathrm{C_{AddAtt}}$)
Following a career hiatus that reignited ***her*** *creativity,* ***Beyoncé*** *was inspired to create a record with a basis in traditional rhythm and blues that stood apart from contemporary popular music...Severing professional ties with* ***father*** *and manager* ***Mathew Knowles***, ***Beyoncé*** *eschewed the music of* ***her*** *previous releases*	*What is the last name of the person who went on a career hiatus?*	*Knowles* (Golden) Beyoncé ($\mathrm{C_{AddAtt}}$)
When the prosecutor suggests that the crime would have still happened if the owner were a woman, ***Christine, Andrea***, ***Annie***, *Janine and the other women who witnessed the crime all laugh and exit the courtroom.*	*What are the names of the women Janine has to determine are sane or crazy?*	*Christine, Andrea, Annie* (Golden) Christine, Andrea ($\mathrm{C_{AddAtt}}$)

Table 4: Errors in predictions for three questions in QUOREF dev set. The blue and bold words indicate the mentions in the same coreference cluster. The bold words in red or magenta indicate the failure of our model in making necessary reasoning. In the Answers column, Golden indicates the golden answer; $\mathrm{C_{AddAtt}}$ indicates the prediction made by $\mathrm{Coref_{AddAtt}}$ model.

of ***Beyoncé***; 2) it should understand the world knowledge that the last name of ***Beyoncé*** is the same as her father's, which should be ***Knowles***. This type of errors shows that our model performs poorly on the questions that require multi-step reasoning. The third type of errors is caused by the questions that have multiple items in an answer. A hyperparameter that limits the total number of items in an answer is used in our models and this parameter is set to 2 in the training, thus when the number of total items in the answer exceeds 2, our models fail to predict the exact items, and the third item ***Annie*** is ignored.

6 Conclusion

In this paper, we present intuitive methods to solve coreference-intensive machine reading comprehension tasks by following the reading process of human in which people connect the anaphoric expressions with explicit instructions. We demonstrate that all our three fine-tuning methods, including $\mathrm{Coref_{GNN}}$, $\mathrm{Coref_{AddAtt}}$ and $\mathrm{Coref_{MultiAtt}}$, are superior to the pre-trained language models that incorporate the coreference information in the pre-training stage, such as $\mathrm{CorefRoBERTa_{LARGE}}$. As the fine-tuning methods rely on the coreference resolution models supplied by other researchers, their performance is also constrained by the accuracy of those coreference resolution models. In addition, the questions that require multi-step reasoning, span multiple entities or contain multiple answer items also pose the challenges to our models. In the future, with more in-depth study on human reasoning in reading comprehension and more progress in graph neural networks, the GNN-based coreference graph can be enriched with more edge types and diverse structures to leverage more linguistic knowledge and gain better performance.

Acknowledgement

We would like to thank Yuchen He for the help of this work. We also appreciate the valuable feedback from the anonymous reviewers.

References

James Baumann. 1986. Teaching third-grade students to comprehend anaphoric relationships: The application of a direct instruction model. *Reading Research Quarterly - READ RES QUART*, 21.

Bin Chen, Jian Su, Sinno Jialin Pan, and Chew Lim Tan. 2011. A unified event coreference resolution by integrating multiple resolvers. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 102–110, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Xiang Chen, Ningyu Zhang, Xin Xie, Shumin Deng, Yunzhi Yao, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2021. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. *ArXiv preprint*, abs/2104.07650.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on*

Empirical Methods in Natural Language Processing, pages 2174–2184, Brussels, Belgium.

Kevin Clark and Christopher D. Manning. 2016. Deep reinforcement learning for mention-ranking coreference models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2256–2262, Austin, Texas. Association for Computational Linguistics.

Pradeep Dasigi, Nelson F. Liu, Ana Marasović, Noah A. Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5925–5932, Hong Kong, China.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2019. Question answering by reasoning across documents with graph convolutional networks. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2306–2317, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota.

Susan Duffy and Keith Rayner. 1990. Eye movements and anaphor resolution: Effects of antecedent typicality and distance. *Language and speech*, 33 (Pt 2):103–19.

Matthew Dunn, Levent Sagun, Mike Higgins, V. Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. 2020. Hierarchical graph network for multi-hop question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8823–8838, Online.

S.C. Garrod and Melody Terras. 2000. The contribution of lexical and situational knowledge to resolving discourse roles: Bonding and resolution. *Journal of Memory and Language*, 42.

Donna Harman. 1993. Overview of TREC-1. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993.*

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. 2018. Reinforced mnemonic reader for machine reading comprehension. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4099–4106.

Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. 2018. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Yin Jou Huang, Jing Lu, Sadao Kurohashi, and Vincent Ng. 2019. Improving event coreference resolution by learning argument compatibility from unlabeled data. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 785–795, Minneapolis, Minnesota.

Holly Joseph, Georgina Bremner, Simon Liversedge, and Kate Nation. 2015. Working memory, reading ability and the effects of distance and typicality on anaphor resolution in children. *Journal of Cognitive Psychology*, 27.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Yuval Kirstain, Ori Ram, and Omer Levy. 2021. Coreference resolution without span representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 14–19, Online.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Avinash Kumar, Vishnu Teja Narapareddy, Pranjal Gupta, Veerubhotla Aditya Srikanth, Lalita Bhanu Murthy Neti, and Aruna Malapati. 2021. Adversarial and auxiliary features-aware bert for sarcasm detection. In *8th ACM IKDD CODS and 26th COMAD*, pages 163–170.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online.

Tao Liu, Xin Wang, Chengguo Lv, Ranran Zhen, and Guohong Fu. 2020. Sentence matching with syntax- and semantics-aware BERT. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3302–3312, Barcelona, Spain (Online).

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv preprint*, abs/1907.11692.

Zhengyuan Liu, Ke Shi, and Nancy Chen. 2021. Coreference-aware dialogue summarization. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 509–519, Singapore and Online.

Siru Ouyang, Zhuosheng Zhang, and Hai Zhao. 2021. Dialogue graph modeling for conversational machine reading. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3158–3169, Online.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Elizabeth Pretorius. 2005. English as a second language learner differences in anaphoric resolution: Reading to learn in the academic context. *Applied Psycholinguistics*, 26:521 – 539.

Peng Qi, Haejun Lee, Tg Sido, and Christopher Manning. 2021. Answering open-domain questions of varying reasoning steps from text. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3599–3614, Online and Punta Cana, Dominican Republic.

Lin Qiu, Yunxuan Xiao, Yanru Qu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. 2019. Dynamically fused graph network for multi-hop reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6140–6150, Florence, Italy. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Ori Ram, Yuval Kirstain, Jonathan Berant, Amir Globerson, and Omer Levy. 2021. Few-shot question answering by pretraining span selection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3066–3079, Online.

Qiu Ran, Yankai Lin, Peng Li, Jie Zhou, and Zhiyuan Liu. 2019. NumNet: Machine reading comprehension with numerical reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2474–2484, Hong Kong, China.

Sangeetha Sangeetha. 2012. Event coreference resolution using mincut based graph clustering. *Computer Science & Information Technology*, 2:253–260.

Elad Segal, Avia Efrat, Mor Shoham, Amir Globerson, and Jonathan Berant. 2020. A simple and effective model for answering multi-span questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3074–3080, Online.

Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *2018 European Semantic Web Conference*, pages 593–607.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada.

Ming Tu, Kevin Huang, Guangtao Wang, Jing Huang, Xiaodong He, and Bowen Zhou. 2020. Select, answer and explain: Interpretable multi-hop reading comprehension over multiple documents. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:9073–9080.

Roger P. G. van Gompel, Simon P. Liversedge, and Jamie Pearson. 2004. Antecedent typicality effects in the processing of noun phrase anaphors.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*.

Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, Tianjun Xiao, Tong He, George Karypis, Jinyang Li, and Zheng Zhang. 2020. Deep graph library: A graph-centric, highly-performant package for graph neural networks.

Wei Wang, Ming Yan, and Chen Wu. 2018. Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1705–1714, Melbourne, Australia. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019a. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019b. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential Reasoning Learning for Language Representation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7170–7186, Online.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020. Semantics-aware BERT for language understanding. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9628–9635.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2021. Retrospective reader for machine reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 14506–14514.