Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

_____ _____

Liyan Xu Date

Enhancing Document Understanding through
the Incorporation of Structural Inference

By

Liyan Xu
Doctor of Philosophy

Computer Science

Jinho D. Choi, Ph.D.
Advisor

Liang Zhao, Ph.D.
Committee Member

Joyce C Ho, Ph.D.
Committee Member

Chenwei Zhang, Ph.D.
Committee Member

Accepted:

Kimberly Jacob Arriola, Ph.D.
Dean of the James T. Laney School of Graduate Studies

Date

Enhancing Document Understanding through the Incorporation of Structural Inference

By

Liyan Xu
B.S., Colorado State University, CO, 2016
B.S., East China Normal University, Shanghai, 2016

Advisor: Jinho D. Choi, Ph.D.

An abstract of
A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Computer Science
2023

Abstract

Enhancing Document Understanding through
the Incorporation of Structural Inference
By Liyan Xu

Towards resolving a variety of Natural Language Processing (NLP) tasks, pretrained language models (PLMs) have been incredibly successful by simply modeling language sequences, backed by their powerful sequence encoding capabilities. However, for document understanding tasks involving multi-sentence or multi-paragraph inputs, the model still needs to overcome the inherent challenge of processing scattered information across the entire document context, such as resolving pronouns or recognizing relations among multiple sentences.

To address the motivation of effectively understanding document context beyond sequence modeling, this dissertation presents an in-depth study on the incorporation of structural inference, utilizing intrinsic structures of languages and documents. Four research works are outlined in Chapters 3-6 that experiment with various structural inference approaches for improving performance on document-oriented tasks. Particularly, Chapter 3 proposes to integrate syntactic dependency structures into the document encoding process, capturing inter-sentence dependencies through designed graph encoding in self-attention, which is shown effective for the task of machine reading comprehension, especially under the multilingual setting. Chapter 4 investigates different methods to perform inference on the discourse structure that concerns coreference relations, allowing for higher-order decision making, thus higher quality predictions, in coreference resolution. Chapter 5 presents a novel formulation of structural inference to facilitate joint information extraction. It incorporates a knowledge specific structure that comprises entity relations, fusing multi-facet information of document entities in terms of both coreference and relations, boosting towards entity-centric information information. Lastly, Chapter 6 continues on the same task as chapter 5, and explores the potential of the sequence-to-sequence generation as an approach that performs implicit inference on linearized entity structures without specific decoder design, which is motivated by its unified encoder-decoder architecture and inherent abilities to perform higher-order inference.

The results of the experiments presented in the dissertation demonstrate that incorporating designed structural inference upon certain intrinsic structures of languages or documents can effectively enhance document understanding, showing improved performance on various benchmarks for document-oriented tasks. This dissertation highlights that modeling dependencies among different parts of the context can lead to more accurate and robust encoding and decoding process, where auxiliary information can be provided through modeling these structures, complementing the sequence modeling of PLMs. Overall, the dissertation makes insightful contributions to the field of natural language processing by investigating the potentials and benefits of leveraging different structures for advancing the state-of-the-art in document understanding.

Enhancing Document Understanding through the Incorporation of Structural Inference

By

Liyan Xu
B.S., Colorado State University, CO, 2016
B.S., East China Normal University, Shanghai, 2016

Advisor: Jinho D. Choi, Ph.D.

A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Computer Science
2023

Acknowledgments

I am deeply grateful for the support and encouragement of everyone who has accompanied me on my academic journey, and I extend my sincerest thanks to my committee for their invaluable mentorship.

I would like to first express my special gratitude to my advisor, Dr. Jinho Choi, who introduced me to the exciting research field of Natural Language Processing (NLP). When I first came to Emory, I had been away from academic research for two years and felt out of touch with the field, after working full-time as a Software Engineer in industry. Despite me being a novice in NLP with limited understanding of research, Dr. Choi inspired me to pursue serious NLP research with his passion, and his unwavering guidance and support have been instrumental in my progress over the past five years. Without him, I would not have thought to be able to accomplish this dissertation.

I am also fortunate to have had the opportunity to work closely with my committee member Dr. Liang Zhao, during my rotation project and the dissertation process, who has suggested insightful research ideas and directions with his expertise in data mining. Many of the constructive suggestions and feedbacks were from my committee member Dr. Joyce Ho; her machine learning course that I took in my first semester is still my favorite course at Emory that laid out the foundation of my machine learning knowledge. I am deeply appreciative of their contributions to my academic growth. I feel privileged to have them on my dissertation committee, and I cannot thank them enough for their exceptional mentorship and inspiration.

I spent my summer 2022 at Amazon, mentored by my external committee member Dr. Chenwei Zhang. Throughout the internship and dissertation, Dr. Zhang has been a wonderful mentor, offering tremendous research advice. In addition to our academic collaboration, I also enjoyed going on hikes together in Seattle, and I am truly grateful for the close connection developed since my internship.

Besides my dissertation committee, I would also like to express my gratitude to my academic collaborators: my mentors at Amazon during summer 2021, Dr. Yile Gu and Dr. Jari Kolehmainen, who made my first research internship experience both fruitful and joyful; my mentor for my externally collaborated research projects, Dr. Xuchao Zhang, who helped me push two top-tier publications with his advising. I am grateful for getting to know them both as collaborators and personal friends; without their support, it would be a lot harder for me to finish these research projects with the same level of quality.

Last but not least, I can't wait to say heartfelt thanks to my loved ones: my parents, and my significant other, Jie, who are always there for me; my life-long friends in China, Yuan, Bashan, Rui, Tong; my life-long friends in US, David, Francisco, Tim, Kye, Zhisheng; and my fellow PhD friends at Emory who made my last five years so memorable. The past five years have been a fun but challenging journey, especially with the unprecedented impact of COVID-19 that affected nearly half of my PhD experience. I feel so fortunate to have all the love and encouragement from my friends and important ones; your presence has enriched my life in so many ways, and I couldn't have asked for more.

Thank you all, for being together with me in this once-in-a-lifetime journey.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 NLP in Document Understanding

In recent years, Natural Language Processing (NLP) has made significant strides in addressing a wide range of document-related applications. In this dissertation, a broader concept of "document" is used that refers to any multi-sentence or multi-paragraph input, such as news articles, conversations or discussions, in lieu of other tasks operating on a single sentence (e.g. sentence parsing). These documents of vast varieties represent rich sources of information that require sophisticated processing and analysis, highlighting the need for advanced NLP techniques to move towards the goal of artificial intelligence for automatic text processing.

Given a document as input, one could ask the machine to perform different types of downstream tasks of interest. For example, given the following paragraph:

> "Dwight Tillery is an American politician of the Democratic Party who is active in local politics of Cincinnati, Ohio. ... He also holds a law degree from the University of Michigan Law School. Tillery served as mayor of Cincinnati from 1991 to 1993.",

one could be interested in which person is being mentioned and what information can

be extrapolated regarding this personal entity. Ideally, we would like the machine to recognize Dwight Tillery as the main entity covered within this text snippet, and to also identify other important facts such as what role and location he serves, and where he obtained his law degree from. Such questions examplify a classic task in NLP as called Information Extraction (IE), and one could also ask other questions of different types, e.g. classify the paragraph into certain categories, or summarize the long paragraph in a few sentences.

Nevertheless, the ultimate objective in driving machine intelligence on documents is to achieve semantic understanding of their content, as we refer as "document understanding", which is the foundation for any specific downstream tasks. Recent advancements in deep learning techniques have facilitated tremendous progress in this direction, empowering strong document encoding and task decoding capabilities. Especially, the developments on document encoding is mostly coupled by the sequence modeling from pretrained language models (PLMs) such as BERT [14], bringing enhancement to both the efficacy and simplicity of NLP approaches.

However, it could be argued that sequence modeling is not the sole solution: leveraging certain intrinsic structures of the document beyond its sequence form could bring additional benefits and insights. Throughout this dissertation, I will show that when we combine both - utilizing certain structural inference in addition to sequence modeling, could induce further improvement for certain document-oriented tasks.

While leveraging structural inference is the central theme of this study, it does not encompass the entirety of my doctoral research. In Section 1.2, three distinct challenges encountered in document understanding are outlined, and each of these facets has been investigated in my prior research endeavors. Nonetheless, the present dissertation will primarily delve into the first aspect - incorporating the optimization of structural inference, which can play a critical role in context encoding and task decoding, and is fundamental to achieve a more profound understanding of documents.

1.2 Challenges and Motivations

In this section, three unique challenges are delineated that are pertinent to document understanding, which detail the motivation behind the presented approaches to address each aspect. The first aspect, being the main theme of this dissertation, will be described and discussed in further details throughout this dissertation.

1.2.1 Scattered Information

In the document input, information is scattered across sentences and paragraphs, and it is often needed to capture the relationships between different parts of the document, and reason across sentences to gain a complete view of information. For instance, in the example from Section 1.1, in order to obtain the fact that Dwight Tillery received his law degree from University of Michigan, the system needs to understand that the pronoun "He" refers to the personal entity of Dwight Tillery, where the reasoning spans across two nonadjacent sentences.

> **Dwight Tillery** is an American politician of ...
>
> ...
>
> **He** also holds a law degree from the University of Michigan Law School.
>
> ...

Above example shows how the information extraction could benefit from incorporating coreference, a type of discourse structure across sentences. Without realizing the coreference structure from these two parts, the system would only perceive scattered and partial facts, while losing the inner logical connection that the document is trying to express.

Depending on the downstream tasks, the reasoning could utilize different language structures, either syntactic structure, discourse structure, or knowledge-specific structure (e.g. relation structure), such that these additional structural information could

further guide the document encoding and task decoding, rather than completely replying on the sequence modeling. A background introduction of different structures explored in this dissertation is covered in Chapter 2.2.

In essence, our underlying motivation and intuition is that, a document is more than a mere concatenation of sentences; rather, it embodies intrinsic structures. In light of this, research presented in this dissertation (Chapter 3-6) targets on how to utilize certain structures for document-input tasks, with the aim of augmenting the document understanding process.

1.2.2 Limited Supervisions

Another challenge in document understanding pertains to the limitations of available supervisions. Unlike sentence-level tasks that are relatively easy to annotate, document-level tasks often require specialized task knowledge and complex labeling procedures, as they usually require inference conditioned on the entire context, which limits the availability of labeled data. Furthermore, large-scale annotating is often not feasible due to time, budget, and personnel constraints. The scarcity of annotated supervisions is further exacerbated in low-resource languages. Consequently, developing effective solutions for document-level tasks under these constraints is also a critical aspect in document understanding research.

During my doctoral research, two settings under the challenge of limited supervisions are specifically addressed.

Cross-Lingual Transfer First, I target on improving zero-shot cross-lingual transfer on different tasks, especially for low-resource languages. The motivation is based on the observations that: 1) the zero-shot performance on low-resource languages is not on par with that on English that has relatively ample supervisions available for various tasks [11, 76]; 2) few annotated supervisions exist for those low-resource

languages, hindering the improvement by direct training.

To overcome this challenge, my work [73] proposes an iterative self-learning framework for various multilingual tasks that adopts a multilingual PLM as the backbone, while it iteratively grows the training set by adding predictions of low-resource languages as silver labels. An explicit uncertainty estimation phase is incorporated in this framework to select high-confidence predictions more accurately, as higher quality of silver labels should lead to higher self-learning efficacy, thus better cross-lingual transfer performance.

Weak Supervisions As noted above, large-scale annotating process could be unfeasible for certain tasks due to time and budget constraints. An example within this realm is information extraction on web corpus, as their expressions and properties of interest are always evolving. My work [75] specifically addresses the task of product attribute extraction on e-commerce corpus, where new types of products and attributes are constantly emerging in the real world, thereby making it untenable to have high-coverage human annotations that accurately capture the ever-changing attribute values and types.

To tackle the challenge of limited resources of supervisions, the proposed model is aimed to work under light supervision, by introducing only a relatively small seed attribute set in training. Since the seed set only provides sole semantic signals regarding seed attributes, the majority of the corpus lack proper supervision, as most of them are absent from the seed set. The model leverages additional signals by fully exploiting document context through self-supervised and unsupervised regularization, achieving discovery of new attribute values and types beyond the seed set.

1.2.3 Domain Adaptation

The last challenge addressed for document understanding is domain adaptation. The motivation is straightforward: since different forms of documents exist, it is desirable for the model to keep as much capability on a new domain of interest. Typically, the training resources often comprise articles, including news/magazine articles and Wikipedia articles, either news/magazine articles or Wikipedia articles, e.g. OntoNotes corpus [49], SQuAD corpus [53]. For document-input tasks, it is imperative to examine the performance when switching to a new domain, such as dialogues or medical domain. In my doctoral research, two settings of domain adaptation are investigated.

Dialogue Domain I study the adaptation of coreference resolution task trained mostly on articles to multi-party dialogues. First, dialogue-unique characteristics such as speaker interaction encoding are addressed by my work [70] that achieves state-of-the-art performance on four dialogue-domain test set. Second, online coreference resolution is specifically proposed by my work [71] for chatbot applications, where the system is expected to extract entities from the latest utterance turn-by-turn, and identifies their coreferent entities from the dialogue history.

Medical Domain Similar to dialogues, medical documents possess their own traits distinct from those of general domain articles. Especially, clinical notes from doctors or other healthcare professionals are frequently long and noisy without sufficient annotations. My work [72] overcomes this issue by employing reinforcement learning to trim out noisy paragraphs that are irrelevant to the task objective. Additionally, another work [39] demonstrates the effectiveness of domain adaptation by pretraining models on large-scale in-domain corpus through the self-supervised language model objective.

1.3 This Dissertation

The remainder of this dissertation focuses on the aspect of the first challenge, where various structural inference is introduced to strengthen the document understanding facing scattered information across context.

Chapter 2 introduces the technical foundations, in regards to the strong encoding capabilities of sequence modeling by pretrained language models, as well as introducing various language and document structures that are employed in this dissertation.

Chapter 3 presents the work of incorporating inference on syntactic structures in document reading comprehension [74]. Specifically, the structure comprises cross-linguistically consistent syntactic features from Universal Dependencies (UD) [47], such that any document can be represented as a syntactic dependency graph in a unified format regardless of languages, in additional to the original language sequence. This work then proposes to encode the graph structure in self-attention, particularly addressing both inner-sentence and inter-sentence graph structures via encoding one-hop and multi-hop dependencies explicitly. Evaluation is conducted on multiple datasets in different languages, showing that although the original raw document text of each language can exhibit its own unique linguistic traits, the transformation to syntactic structures can serve as the anchors across multiple languages, and the model benefits from a closer gap of cross-lingual structural representation.

Chapter 4 presents the work of leveraging discourse structures in coreference resolution [69]. Coreference resolution is an important step towards the context understanding, as shown by the example in Section 1.1. This work identifies the issue of error-prone local decisions existed in previous work, and proposes to alleviate this problem by performing higher-order inference that utilizes the discourse structure. In particular, the structure considers the coreference itself (a discourse relation that arises when two mentions refer to the same entity) between any two entity mentions in the document. It first creates the discourse structure from the input document, where

each node is an entity mention. It then focuses on different methods of higher-order inference, performing structural inference to obtain a less error-prone coreference resolution prediction.

Chapter 5 introduces novel inference on knowledge-specific structures in joint information extraction. Concretely, the structure concerns the (predefined) relations between any two entity mentions in the document. This work performs structural inference upon this graph, bridging multi-facet information of entities, including coreference resolution and relation extraction, which is shown to improve the performance on two document-level joint extraction datasets.

Chapter 6 continues on the topic of Chapter 5, but with a different paradigm for resolving the document-level joint information extraction problem. Unlike the approach in the previous chapter, which performs explicit structural inference, this work investigates sequence generation as an alternative method that implicitly considers entity structural interaction through a designed generation schema. By modeling the task as a sequence-to-sequence generation problem, this approach could potentially exploit complex dependencies between entity mentions, without relying on predefined decoding inference on graph structures.

Finally, chapter 7 concludes this dissertation, summarizes the research presented throughout the preceding chapters, and reiterates the importance to exploit intrinsic language structures in document understanding.

Chapter 2

Technical Foundations

Prior to the emergence of deep learning techniques in natural language processing (NLP), traditional language structures such as syntactic parsing trees, which were defined in the field of computational linguistics and related disciplines, were commonly utilized as fundamental features for modeling languages in downstream NLP tasks. In recent years, however, there has been a significant shift in academic focus from these traditional structures towards higher-level NLP tasks that are more closely aligned with real-world applications. This shift can be largely attributed to advances in deep learning, particularly pretrained language models (PLMs), which offer powerful encoding capabilities based on sequence modeling. Thus, they reduce the complexity to approach most NLP tasks, as modeling based on those traditional structures is no longer necessary.

In this chapter, the technical background of PLMs is first introduced in Section 2.1, highlighting their essential role in achieving language understanding in modern NLP approaches. However, it is important to note that sequence modeling may not always provide the optimal solution. In Section 2.2, several important structures of languages and documents are then introduced, which are later demonstrated in this dissertation on how they could be leveraged to provide complementary information beyond the

sole sequence modeling.

2.1 Sequence Modeling

Pretrained language models focus on modeling the given text sequence, which is the original form of natural languages. By encoding the sequence of lexical words or tokens into embedding space, PLMs can directly represent the context semantics without the need of extra features.

Transformers Most pretrained language models, as of 2023, adopt the Transformers architecture [60] or its variant, which is a stack of multiple Transformers layers. Each layer features the self-attention mechanism of multiple attention heads, followed by a fully connected feed-forward network.

Given a sequence of length n, and its input hidden states in embedding space $x_{1:n} := (x_1, \ldots, x_n)$, self-attention first transforms each hidden state into query, key, and value representation. Each position i then attends on every position $j = 1, \ldots, n$, acquiring a self-attention score α_{ij}. At each attention head, the attention distribution is computed by the softmax of scaled dot-product over the query and key representation:

$$e_{ij} = (x_i W_Q)(x_j W_K)^T \tag{2.1}$$

$$\alpha_{ij} = \frac{\exp(e_{ij}/\sqrt{d_k})}{\sum_{k=1}^{n} \exp(e_{ik}/\sqrt{d_k})} \tag{2.2}$$

$W_Q, W_K \in \mathbb{R}^{d_x \times d_k}$ are query and key parameters of each attention head, and d_x/d_k is the dimension of the input/key hidden state. The output of self-attention $z_{1:n} :=$

$(z_1, \ldots, z_n)$ is obtained by the weighted sum over the value representation:

$$z_i = \sum_{j=1}^{n} \alpha_{ij}(x_j W_V) \tag{2.3}$$

where $W_V \in \mathbb{R}^{d_x \times d_v}$ is the value parameter, with d_v being the hidden size of values.

$z_{1:n}$ then goes through the feedforward neural network, yielding a new sequence of hidden states $x'_{1:n} := (x'_1, \ldots, x'_n)$, as the output of one entire Transformers layer, which can be fed to another Transformers layer, repeatedly. Each layer can be viewed as a basic building block in PLMs.

Pretrained Language Models The first well-known PLMs that revolutionized NLP is BERT [14], which consists of 12 or 24 Transformers layers, depending on the model size. PLMs acquire their powerful sequence encoding capabilities through pre-training on large-scale corpus using self-supervised language model objectives. Specifically, BERT adopts the masked language model (MLM) objective, which enables it to encode bidirectional context. For generative models such as GPT [4], the causal language model (CLM) is used to encode unidirectional context. Certain PLMs also adopt other objectives, such as ELECTRA [10].

The primary strength of PLMs is their ability to provide contextualized embedding representations. As a result, the hidden states generated by these models encode the semantics of the entire context, making them ideal general encoders for a wide range of NLP tasks. This strong encoding capability is a crucial property that underpins the success of PLMs in many NLP applications.

PLMs can be categorized into three types, according to their usage paradigms.

- Encoder only: e.g BERT, which aims to obtain good embedding representation of the sequence, mainly for various classification-based tasks.

- Decoder only: e.g. GPT, which is common for language modeling and other

generative tasks.

- Encoder-Decoder: e.g. BART [32] and T5 [52], which supports generative tasks while strengthening the context understanding.

In this dissertation, all the experimental settings involve models of PLMs to encode the document context, especially the encoder model BERT. In Chapter 6, encoder-decoder models are also employed.

2.2 Structures within Documents

In this section, two linguistic-related structures are introduced, which can be used to represent context features prior to embedding-based representation. The relation structure, as a form of knowledge-specific structure, is briefly introduced in the end.

Syntactic Structure Syntactic structures refer to the hierarchical organization of words in a sentence based on their grammatical functions. In traditional NLP approaches, syntactic structures, such as Part-of-Speech (POS) tags and dependency relations, have been widely used as fundamental features for developing models in various downstream tasks. These structures can provide valuable insights into the underlying syntactic relationships between words in a sentence and can help models better understand the context of the text. An example of the syntactic dependency structure is shown in Figure 2.1, which is utilized in Chapter 3 for document-level context understanding.

Discourse Structure Discourse structures refer to the larger organization of sentences and paragraphs in a document. They describe how individual sentences or utterances relate to each other in terms of meaning and function. In this dissertation,

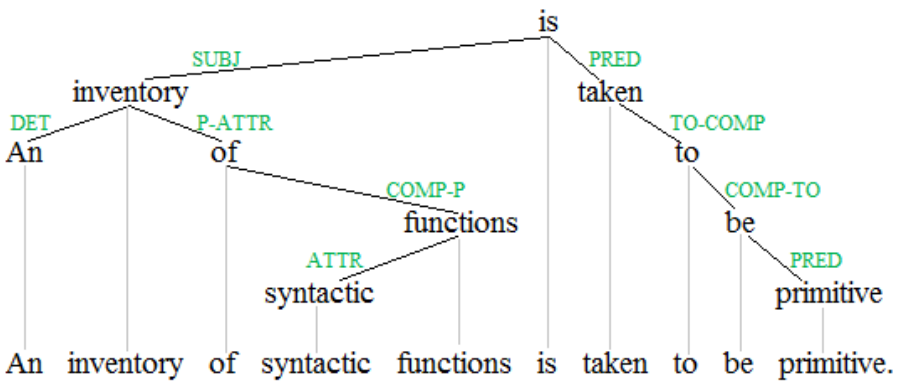

Figure 2.1: Example of the syntactic dependency structure. Source: `https://upload.wikimedia.org/wikipedia/commons/c/c3/Syntactic_functions_1.png`

the coreference, a discourse relation concerning mentions in different places of the document referring to the same entity, is considered in Chapter 4-6, to encode a more complete view of document context understanding.

Relation Structure In Chapter 5, a particular structure of the document is utilized - relation structure, which consists of relation instances in triples, and is common seen in knowledge base (KB)-related tasks [3]. Unlike structures that aim to preserve linguistic meaning, the relation structure is designed to represent specific knowledge of interest. As such, it is not intended to capture the nuances of language but rather to serve as a tool for representing structured knowledge conveyed in the document context. It provides a natural way to represent the relationships between entities given the context, and has been employed heavily in tasks such as entity linking and entity disambiguation.

Chapter 3

Syntactic Structures for Reading Comprehension

3.1 Introduction

In this chapter, I demonstrate that syntactic structures, despite their diminishing importance in deep learning era, can still present a meaningful impact to document understanding through designed structural inference, especially in a multilingual context.

As syntactic structures were once instrumental in language representation prior to the advent of word embedding, they have since been eclipsed by the impressive performance of pretrained language models. My research indicates that, while embedding representation from language models is powerful, incorporating syntactic structures of document input can serve as an anchor point to align diverse languages and provide additional guidance in document encoding. Experiments in this research work suggest that the proposed inference on syntactic structures is able to gain substantial improvement on certain languages, up to 11.2 Exact-Match for machine reading comprehension.

3.1.1 Universal Dependencies (UD)

Universal Dependencies (UD) is a unified framework for providing cross-linguistically consistent part-of-speech (POS) tags, morphological features, and syntactic dependencies across over 90 languages [47]. With over 100 treebanks now available thanks to extensive annotation efforts, several toolkits have been developed, such as Stanza [51] and UDPipe [58], which can provide state-of-the-art performance on obtaining universal syntactic features for multiple languages. The incorporation of UD features has significant potential for cross-lingual applications.

3.1.2 Motivations

In this study, I focus on incorporating UD features, specifically, syntactic dependency structures on a document-level, in an important document understanding task: machine reading comprehension (MRC). Especially, I adopt the multilingual setting of zero-shot cross-lingual transfer, leveraging the potential of UD to align multiple languages syntactically. The motivation is that while each language may possess unique linguistic traits in its raw text, cross-linguistically consistent syntax can serve as the anchor point to a more unified format. For instance, Figure 3.1 depicts parallel sentences in English and Japanese that differ significantly in sentence structure. By providing additional clues from universal syntactic dependencies, my model aims to reduce the gap in cross-lingual representation, benefiting from the explicit alignment provided by the dependency graph structure.

Past research has demonstrated the effectiveness of syntactically informed models in machine translation [6, 87] and other intra-sentence tasks such as Semantic Role Labeling (SRL) [59, 27]. Although the utilization of additional syntactic clues has become less common with the emergence of pretrained language models like BERT [14], which implicitly encode linguistic notions of syntax [19], the potential value of

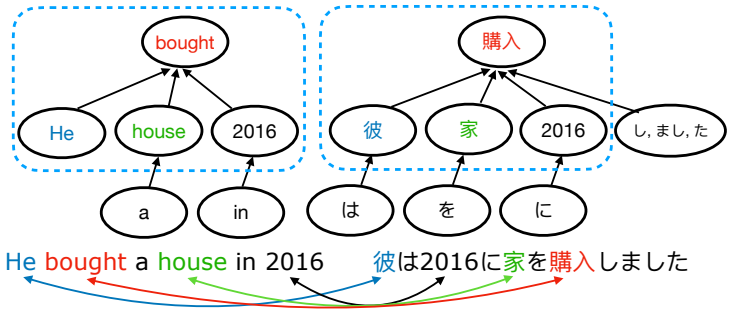

Figure 3.1: Syntactic dependency representation of parallel sentences in English and Japanese. The aligned verbs and nouns of the same meaning are marked by the same color. Two languages have quite different sentence structure, while the main components (verbs and nouns) have the same graph structure under syntactic dependencies, reducing the cross-lingual gap on the representation.

incorporating syntactic features for multilingual document understanding remains an open question. This work seeks to address this question by explicitly addressing inter-sentence relations during the inference on a cross-linguistically consistent syntactic graph, extending beyond the direct intra-sentence syntactic relations explored by previous monolingual MRC models such as SG-Net [88].

3.1.3 Problem Formulation

To model this particular research problem, my proposed approach builds upon multilingual pretrained language models as a backbone, featuring direct zero-shot transfer where the entire model is trained only on the source language and evaluated on test sets in multiple target languages directly. This approach aims to be an augmentation and can be further combined with other cross-lingual transfer techniques that involve training with target languages, such as adding translations to target languages during training [21, 31, 12, 82].

To address the challenge of utilizing syntactic structures in this document understanding task, Inter-Sentence Dependency Graph (ISDG) is firstly introduced in Section 3.2.2. ISDG is a document-level graph structure that connects the syntactic dependencies of each sentence. An ISDG encoder is then proposed that stacks upon a

pretrained language model, adapting self-attention [60] to encode the ISDG structure and relations. The encoder comprises two components: a "local" component that models the one-hop relations directly among graph nodes, and a "global" component that focuses on multi-hop relations by explicitly modeling the syntactic dependencies across sentences. In particular, to circumvent the giant graph matrix for a long document, "soft" paths are defined that approximate full paths between every node pair based on the unique characteristics of the ISDG, and inject these paths as new representations of keys and queries in self-attention.

My approach is evaluated on three multilingual MRC datasets (XQuAD [1], MLQA [33], TyDiQA-GoldP [7]) using three different pretrained language models, testing the generalizability of the approach across 14 test sets in 8 languages that are supported by UD. Empirical results demonstrate that my proposed structural inference improves zero-shot performance on all test sets in terms of either F1 or Exact-Match (EM), with the on-average performance on all three datasets being boosted by up to 3.8 F1 and 5.2 EM. My approach also achieved up to 5.2 F1/11.2 EM improvement on certain languages. These results validate our motivation that the zero-shot performance can benefit from cross-linguistically consistent syntactic structures for most of the experimented languages. My analysis shows that the proposed attention on the global inter-sentence syntactic dependencies plays an important role.

3.2 Approach

This section begins by a brief overview of the multilingual pretrained language model, which serves as the backbone as well as baseline in our experiments. It then introduces UD features and details how to encode syntactic structures using both local and global encoding components in the proposed ISDG encoder.

3.2.1 Multilingual Pretrained Models

Multilingual pretrained language models typically employ the Transformer architecture [60] for sequence encoding. In my approach, I utilize its direct zero-shot performance on target language sequences (documents in target languages) as the baseline.

Following the previous work on the span-extraction MRC task, the same input format is used where the question and context are packed in a single sequence. The same decoding scheme is also used in all our experiments, where two linear layers are stacked on the encoder to predict the start and end positions of the answer span respectively. The log-likelihoods of the gold start and end positions i_s, i_e are being optimized during training:

$$p^{s/e}(i) = \text{softmax}\big(W_L^{s/e} x_i + b_L^{s/e}\big) \tag{3.1}$$

$$\mathcal{L} = -\log p^s(i_s) - \log p^e(i_e) \tag{3.2}$$

$p^{s/e}(i)$ is the likelihood of token i being the start/end position, $W_L^{s/e}$ and $b_L^{s/e}$ are the parameters for the linear layers, and $\mathcal{L}$ is the loss function. The final selected prediction is the span with the highest sum of start and end likelihood.

3.2.2 Syntactic Features

Universal POS A learnable embedding layer is used for the 17 POS types defined by UD. For each subtoken, its POS embedding is concatenated along with its hidden state from the last layer of the pretrained models, serving as the new input hidden state for the following graph encoder.

Universal Syntactic Dependencies UD provides the syntactic dependency features for each word in a sentence, including its head word and the dependency relation to the head word. Each sentence contains one unique root word with no head word.

In this work, only the main relation types from UD are used, without considering subtypes. The syntactic dependency features are consumed by the proposed model as described below.

3.2.3 Inter-Sentence Dependency Graph (ISDG)

As MRC is a document-level task, the input typically includes multiple sentences for the context and question. While previous research has primarily focused on encoding raw syntactic dependencies within each sentence directly, I propose to further consider global syntactic structure across sentences to strengthen the document encoding. To achieve this, Inter-Sentence Dependency Graph (ISDG) is constructed, utilizing the dependency trees of each sentence to construct the global syntactic structure. An example of ISDG is shown in Figure 3.2.

To construct ISDG, the original dependency tree of each sentence is firstly obtained; the reserve relation from each head word to its child words is added. The tree is then adapted to the subtoken level by splitting each word into nodes of its corresponding subtokens, where each subtoken node shares the same relations as the word. Among all subtokens from the same word, they are fully connected by a special relation *subtoken*, and also self-connect each node by a special relation *self*. For special subtokens such as `[CLS]` and `[SEP]`, only the self-connections are assigned. For the rest of this paper, "nodes" refer to graph nodes on the subtoken level.

Next, all independent dependency trees are connected to construct the final ISDG structure. All root nodes within context sentences are fully connected with a special relation *cross-sentence*; another special relation *cross-type* is used to fully connect all root nodes between question and context sentences, distinguishing the dual input types. This enables each node in the ISDG to reach any other node through a one-hop or multi-hop dependency path, building the global syntactic relations. The design objective of ISDG is to keep all raw syntactic features as well as adding the visibility

of the cross-sentence input structure.

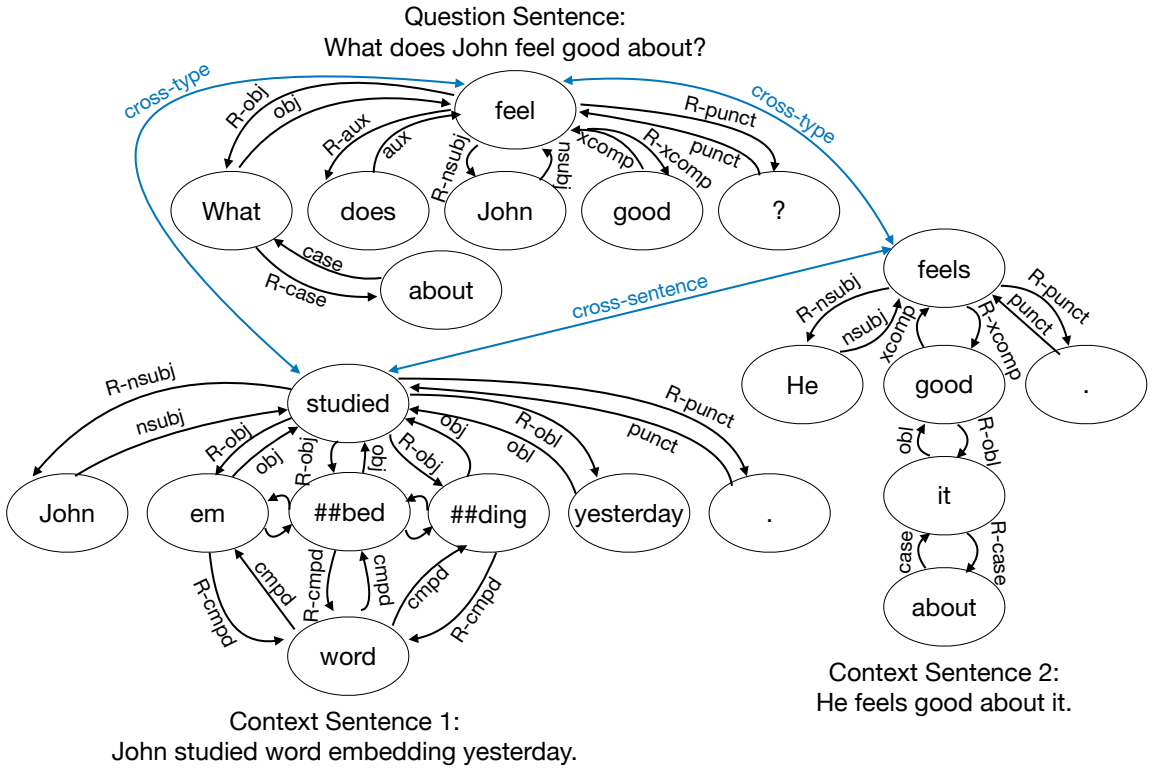

Figure 3.2: A simplified example of the ISDG is shown. Nodes are connected by syntactic dependency relations; reverse relations are prepended by "R-". Special types of *cross-sentence* and *cross-type* connect root nodes of the dependency trees, marked by the blue color. For simplicity, the self-connection on each node is omitted, as well as the *subtoken* relations among subtokens of "em", "##bed", "##ding".

3.2.4 ISDG Encoder: Local Encoding

For each input, the proposed ISDG encoder is dedicated to encode its ISDG obtained above, and it consists of two components: the local encoding component that focuses on the local one-hop relations directly (Section 3.2.4), and the global encoding component that further accounts for the global multi-hop syntactic relations across sentences (Section 3.2.5).

The local encoding component adapts the idea of relative position encoding that has been explored by several recent work [57, 13, 5]. Denote the hidden state of

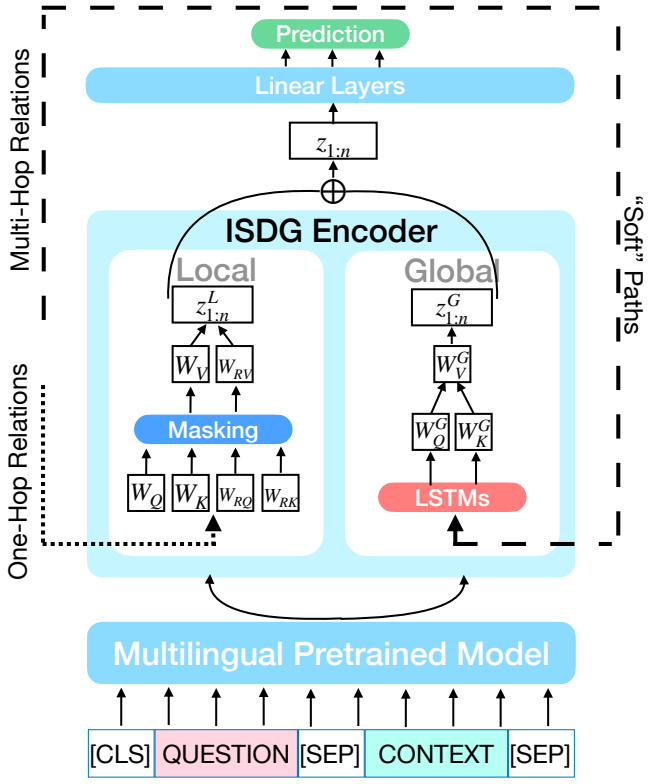

Figure 3.3: An overview of the model architecture is shown. The proposed ISDG encoder is stacked upon the pretrained language model, and encodes the local one-hop and global multi-hop dependency relations in the obtained multi-sentence graph structure.

each input node at sequence position i as x_i, which is the concatenation of its POS embedding and its hidden state from the pretrained model. The hidden state of the relation type from node i to node j is denoted as r_{ij}, which is obtained from a separate learnable embedding layer. The structure of one-hop relations are injected into the self-attention as follows:

$$
\begin{aligned}
e^{\mathrm{L}}_{ij} &= \big((x_i + r_{ij})W_Q\big)\big((x_j + r_{ji})W_K\big)^T \\
&= \underbrace{(x_i W_Q W_K^T x_j)}_{(a)} + \underbrace{(x_i W_Q W_K^T r_{ji})}_{(b)} \\
&+ \underbrace{(r_{ij} W_Q W_K^T x_j)}_{(c)} + \underbrace{(r_{ij} W_Q W_K^T r_{ji})}_{(d)}
\end{aligned}
\tag{3.3}
$$

e^{L}_{ij} is the raw attention score that takes into account the local one-hop relation type from node i to j in ISDG; W_Q and W_K are the query and key parameters. In particular, Eq (3.3) can be decomposed and interpreted by four parts. The term (a) is the same as the original self-attention; the term (b) and (c) represent the relation bias conditioned on the source/target node; the term (d) is the prior bias on the relation types.

However, the vanilla injection in Eq (3.3) cannot fit for ISDG directly, and two adaptations are made to address the following issues.

First, let d_x and d_r be the hidden size of nodes and relations; Eq (3.3) requires equal hidden sizes $d_x = d_r$. For each input sequence, the embedding matrices of nodes and relations have sizes nd_x and $n^2 d_r$ respectively. Therefore, it would be impractical to keep $d_x = d_r$ for the document-level task where n can be quite large. The first adaptation sets d_r to be much smaller than d_x and uses another set of key and query parameters for the relations. The relation matrix is also shared across attention heads to reduce the memory usage.

Second, since ISDG is not a complete graph, a *none* type is set for any r_{ij} with

no relations. However, this would introduce a non-trivial inductive bias in Eq (3.3), as *none* type can be prevalent in the graph matrix. Thus, attention masking $\mathcal{M}$ is applied on the attention scores by the *none* type specified in Eq (3.4) and (3.5), similar to Yao et al. [79], Zhang et al. [88], enforcing the inductive bias to be 0 among nodes that are not directly connected.

Lastly, the relations are injected into the value representation of self-attention as in Eq (3.6). The final normalized attention score α^{L} and output z^{L} are computed as:

$$\mathcal{M}_{ij} = \begin{cases} 1 & r_{ij} \neq none \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

$$\alpha_{ij}^{\mathrm{L}} = \frac{\exp(\mathcal{M}_{ij} \cdot e_{ij}^{\mathrm{L}} / \sqrt{d_x})}{\sum_{k=1}^{n} \exp(\mathcal{M}_{ik} \cdot e_{ik}^{\mathrm{L}} / \sqrt{d_x})} \tag{3.5}$$

$$z_i^{\mathrm{L}} = \sum_{j=1}^{n} \alpha_{ij}^{\mathrm{L}} (x_j W_V + r_{ij} W_{RV}) \tag{3.6}$$

$W_V \in \mathbb{R}^{d_x \times d_x}$ and $W_{RV} \in \mathbb{R}^{d_r \times d_x}$ are the query parameters for the nodes and relations. Note that multiple layers of the local encoding component can be stacked together to implicitly model the higher-order dependencies, however in practice, stacking multiple layers are constrained by the GPU memory, and quickly becomes impractical under the huge document-level graph matrix.

3.2.5 ISDG Encoder: Global Encoding

Next, the following global encoding component is proposed and integrated into the ISDG encoder, for the fact that each pair of nodes in ISDG always has a dependency path of relations, and making use of this multi-hop relations should further provide stronger sequence encoding. Previous work has addressed multi-hop relations by directly encoding the shortest path between two nodes for sentence-level tasks [91, 5]. However, this is not practical for the MRC task, as the sequence length n can be

much larger for the document-level input. Let l_p be the maximum path length, d_p be the hidden size for each path step. The size of the path matrix is $n^2 l_p d_p$ that includes each pair of nodes, which can easily consume all GPU memory.

To address the above challenge, my proposed global encoding component utilizes an approximated path between any two nodes, rather than the full path, referred as the "soft" path, which has a much lower space complexity than the full path matrix, making it possible for the model to encode the multi-hop relations give the long input sequence.

The rationale behind "soft" paths is the observation that the paths of many node pairs are heavily overlapped: for any cross-sentence node pairs, each of the node always goes through its root node. Denote $p_\dagger(i)$ as the outgoing path of hidden states from node i to its root node i_r:

$$p_\dagger(i) = (x_i, r_{ik_1}, x_{k_1}, r_{k_1k_2}, \ldots, r_{k_ii_r}, x_{i_r})$$

with $k_1, \ldots, k_i$ being the intermediate nodes in the path. Similarly, denote $p_\ddagger(i)$ as the incoming path from root node i_r to node i, which has the reverse order of $p_\dagger(i)$. The "soft" path τ_{ij} is then defined from node i to j as:

$$\begin{aligned} \tau_{ij} &= (x_i, \ldots, x_{ir}, x_{jr}, \ldots, x_j) \\ &= p_\dagger(i) \oplus p_\ddagger(j) \end{aligned} \tag{3.7}$$

x_{ir} and x_{jr} are the root nodes for i and j, $\oplus$ denotes the concatenation. τ_{ij} largely captures the true shortest paths of cross-sentence node pairs and only loses one intermediate relation $r_{i_rj_r}$ between the two root nodes; for within-sentence pairs, τ_{ij} can become non-shortest path, but still provides auxiliary information over the direct one-hop relations in the local encoding component. An illustration of the "soft" paths are shown in Figure 3.4.

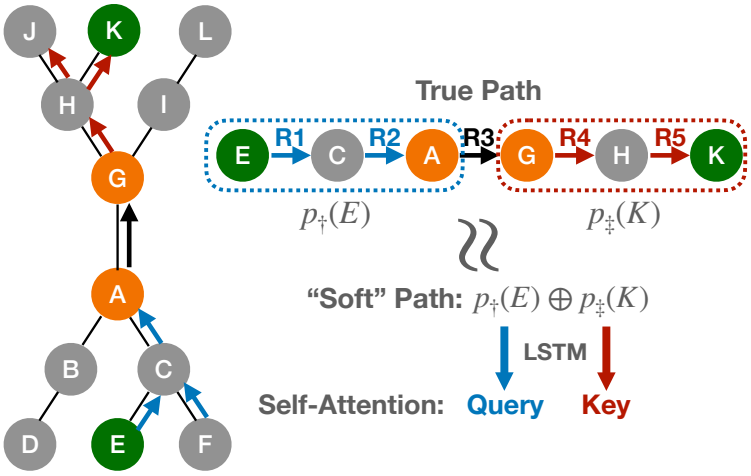

Figure 3.4: Illustration of the "soft" path. Two dependency trees are depicted with root nodes A and G. True paths of all node pairs are heavily overlapped, as each node needs to go through its root node. The "soft" path from node E to K is shown, which is the concatenation of the outgoing path of node E: $p_\dagger(E)$, and the incoming path of node K: $p_\ddagger(K)$, as an approximation of the true path.

As a result of the "soft" path trade-off, the approximated path of global multi-hop relations can now be fit into self-attention. The outgoing and incoming "soft" paths are encoded by long short-term memory (LSTM), of which hidden states are denoted by $\overrightarrow{h_{i,t}}$ and $\overleftarrow{h_{i,t}}$ at the step t for the node i:

$$\overrightarrow{h_{i,t}} \leftarrow \text{LSTM}(s^\dagger_{i,t}, \overrightarrow{h_{i,t-1}}; \theta^\dagger) \tag{3.8}$$

$$\overleftarrow{h_{i,t}} \leftarrow \text{LSTM}(s^\ddagger_{i,t}, \overleftarrow{h_{i,t-1}}; \theta^\ddagger) \tag{3.9}$$

where $s^\dagger_{i,t}$ and $s^\ddagger_{i,t}$ are the tth hidden states in the "soft" path $p_\dagger(i)$ and $p_\ddagger(i)$; $\theta^\dagger$ and $\theta^\ddagger$ are the parameters for LSTMs.

Two distinct representation for each node i, denoted by $\overrightarrow{g_i}$ and $\overleftarrow{g_i}$, can then be obtained, which are the last LSTM hidden states of the outgoing path $p_\dagger(i)$ and incoming path $p_\ddagger(i)$ respectively. The outgoing path representation $\overrightarrow{g_i}$ of node i is made as the query, and the incoming path representation $\overleftarrow{g_j}$ of node j is made as the

key, resembling the "soft" path τ_{ij} to be injected into the self-attention:

$$e_{ij}^{\mathrm{G}} = (\overrightarrow{g_i} W_Q^{\mathrm{G}})(\overleftarrow{g_j} W_K^{\mathrm{G}})^T \tag{3.10}$$

$$\alpha_{ij}^{\mathrm{G}} = \frac{\exp(e_{ij}^{\mathrm{G}}/\sqrt{d_x})}{\sum_{k=1}^{n} \exp(e_{ik}^{\mathrm{G}}/\sqrt{d_x})} \tag{3.11}$$

$$z_i^{\mathrm{G}} = \sum_{j=1}^{n} \alpha_{ij}^{\mathrm{G}} \big((\overrightarrow{g_i} + \overleftarrow{g_j}) W_V^{\mathrm{G}} \big) \tag{3.12}$$

$$z_i = z_i^{\mathrm{L}} \oplus z_i^{\mathrm{G}} \tag{3.13}$$

$W_Q^{\mathrm{G}}, W_K^{\mathrm{G}}, W_V^{\mathrm{G}} \in \mathbb{R}^{d_x \times d_x}$ are the query, key, value parameters for the global encoding component. The final output of the ISDG encoder z_i is the concatenation of the output from both local and global encoding components. To further strengthen inter-sentence interaction, additional layers of vanilla self-attention can optionally be stacked upon the ISDG encoder that takes the output sequence $z_{1:n}$ as input.

3.3 Evaluation and Analysis

Models are evaluated on three multilingual MRC benchmarks suggested by XTREME: XQuAD [1], MLQA [33], TyDiQA-GoldP [7]. For XQuAD and MLQA, models are trained on English SQuAD v1.1 [53] and evaluated directly on the test sets of each dataset in multiple target languages. For TyDiQA-GoldP, models are trained on its English training set and evaluated directly on its test sets. The evaluation scripts provided by XTREME are used, keeping the evaluation protocols identical. Standard metrics of F1 and Exact-Match (EM) are used.

As Stanza is used to obtain UD features, the experiments include languages that are supported by UD and have similar prediction performance as the source language English, to ensure consistent quality of the UD features across languages. Specifically, the dependency parsing performance per language are compared according to

	en	de	el	es	hi	ru	**avg**
mBERT*	83.5 / 72.2	70.6 / 54.0	62.6 / 44.9	75.5 / 56.9	59.2 / **46.0**	71.3 / 53.3	70.5 / 54.6
mBERT	83.8 / 73.0	71.7 / 55.8	63.6 / 45.8	**76.4 / 59.0**	58.2 / 44.0	71.5 / 55.1	70.9 / 55.5
+ ISDG	**84.1 / 73.1**	**74.1 / 57.6**	**64.4 / 48.2**	76.1 / 57.8	**59.3 / 46.0**	**72.2 / 55.3**	**71.7 / 56.3**
XLM-R*	86.5 / 75.7	80.4 / 63.4	79.8 / 61.7	82.0 / 63.9	76.7 / 59.7	80.1 / 64.3	80.9 / 64.8
XLM-R	87.4 / 76.3	80.8 / 63.9	80.6 / 63.4	82.2 / 63.0	76.4 / 60.0	80.9 / **65.1**	81.4 / 65.3
+ ISDG	**88.6 / 77.9**	**82.1** / **66.1**	**81.9** / **64.3**	**83.4** / **65.9**	**76.9** / **60.9**	**81.3** / 64.5	**82.4** / **66.6**
mT5*	88.4 / 77.3	80.0 / 62.9	77.5 / 57.6	81.8 / **64.2**	73.4 / 56.6	74.7 / 56.9	79.3 / 62.6
mT5	87.8 / 76.8	80.9 / 63.9	79.3 / 60.9	**82.4** / 64.0	75.7 / 58.7	78.6 / 62.2	80.8 / 64.4
+ ISDG	**88.7** / **78.2**	**82.5** / **65.4**	**80.5 / 61.3**	82.1 / 63.2	**76.9** / **60.3**	**80.5 / 64.2**	**81.9 / 65.4**

Table 3.1: XQuAD results (F1/EM) for each language. * denotes the results from original papers. Bold numbers are the best results per pretrained language model; underlined numbers are the best results across all models (same for Table 3.2 & 3.3).

	en	de	es	hi	**avg**
mBERT*	80.2 / 67.0	59.0 / 43.8	**67.4** / 49.2	50.2 / **35.3**	64.2 / 48.8
mBERT	**80.8** / 67.8	61.0 / 46.4	67.3 / 49.2	49.3 / 33.6	64.6 / 49.3
+ ISDG	80.7 / **67.9**	**62.3 / 48.1**	67.1 / **49.4**	**50.3** / 35.1	**65.1 / 50.2**
XLM-R*	83.5 / 70.6	70.1 / 54.9	74.1 / **56.6**	70.6 / 53.1	74.6 / 58.8
XLM-R	84.5 / 71.5	71.1 / 56.1	74.2 / 56.4	71.4 / 53.6	75.3 / 59.4
+ ISDG	**84.9** / **71.9**	**71.2** / **56.2**	**74.4** / 56.2	**71.8** / **54.0**	**75.6** / **59.6**
mT5*	**84.9** / 70.7	68.9 / 51.8	73.5 / 54.1	66.9 / 47.7	73.6 / 56.1
mT5	84.5 / 71.7	69.0 / 53.9	73.8 / 56.2	69.2 / 51.8	74.1 / 58.4
+ ISDG	**84.9** / **71.9**	**69.6 / 54.4**	**74.7** / **56.7**	**70.4 / 52.2**	**74.9 / 58.8**

Table 3.2: MLQA results (F1/EM) for each language.

	en	fi	ko	ru	**avg**
mBERT*	**75.3 / 63.6**	59.7 / **45.3**	**58.8 / 50.0**	60.0 / 38.8	**63.5 / 49.4**
mBERT	74.3 / 61.8	60.3 / 44.0	57.3 / 46.7	**62.5** / 42.3	63.6 / 48.7
+ ISDG	74.4 / 63.2	**61.1** / 43.5	52.5 / 44.2	61.3 / **43.7**	62.3 / 48.7
XLM-R*	73.6 / 61.3	74.2 / 58.2	59.4 / 47.8	69.5 / 46.8	69.2 / 53.5
+ ISDG	**76.2** / **64.5**	**75.3** / **59.4**	**64.0 / 52.5**	**70.7 / 51.2**	**71.6 / 56.9**
mT5*	71.6 / 58.9	64.6 / 48.8	47.6 / 37.3	58.9 / 36.8	60.7 / 45.5
mT5	73.3 / 60.9	71.5 / 54.5	60.8 / 51.1	68.1 / 44.8	68.4 / 52.8
+ ISDG	**76.3** / **64.5**	**73.1 / 55.1**	**66.0** / **56.5**	**73.3** / **56.0**	**72.2** / **58.0**

Table 3.3: TyDiQA-GoldP results (F1/EM) for each language.

Labeled Attachment Score (LAS, the main evaluation metric for dependency parsing) provided by Stanza[1], and any languages that currently have LAS score above 80 are included. The resulting evaluation includes a total of 8 languages and 14 test sets in

[1] `https://stanfordnlp.github.io/stanza/performance.html`

the experiments. More languages and higher feature quality in the near future can be expected with the ongoing development of the UD project.

Evaluation results The evaluation results for XQuAD are shown in Table 3.1, and Table 3.2 & 3.3 show the results for MLQA and TyDiQA-GoldP respectively. In particular, mBERT*, XLM-R* and mT5* denote the results reported from the original papers of XTREME and mT5; all other results are obtained from re-implemented baselines and proposed models. Three different multilingual pretrained language models are experimented on all three datasets, and "+ISDG" shows the results of adding the ISDG encoder on the corresponding pretrained model.

The entire evaluation consists of 14 test sets in 8 languages. The best result for every test set, denoted by the underlined score of each column, is achieved by the ISDG encoder in terms of either F1 or EM. The ISDG encoder also establishes the best on-average performance on all three datasets using either one of the three multilingual pretrained models, except for mBERT on TyDiQA-GoldP. Specifically, the best on-average results of both XQuAD and MLQA are achieved by the ISDG encoder with XLM-R, while the encoder with mT5 shows the best results for TyDiQA-GoldP, improving upon its corresponding baseline by 3.8 F1 / 5.2 EM on average. Notably, on certain test sets, the improvement can be substantial, such as a 5.2 F1 / 11.2 EM improvement using mT5 on the TyDiQA-GoldP test set in Russian (ru).

The language-specific results suggest that although UD is designed to provide consistent features across languages, different languages do not equally benefit from the syntactic features, possibly due to intrinsic linguistic differences and differences in feature quality obtained from Stanza. Nonetheless, most languages do show a consistent performance boost. Specifically, English (en), German (de), Greek (el), Hindi (hi), Russian (ru), and Finnish (fi) consistently benefit from UD features across different datasets using any of the pretrained models (with improvement up to 5.2

F1). Spanish (es) benefits overall but may be dataset-specific and does not outperform the baseline on XQuAD using mBERT or mT5. Korean (ko) exhibits a significant improvement on TyDiQA-GoldP using XLM-R or mT5 (up to 5.2 F1 / 5.4 EM), but the performance drops when using mBERT, likely due to the incompatibility between the wordpiece tokenizer of mBERT and Stanza tokenization on the segmentation of text in Korean.

An ablation study is conducted to evaluate the impact of local and global graph encoding in the ISDG encoder. The evaluation includes languages that consistently benefit from UD features on XQuAD so to provide more explicit insights. The results of the study are presented in Table 3.4, which reports the F1 score differences for three settings: using only POS features (which skips graph encoding altogether and is similar to the baselines, but with UD tokenization and POS features), adding the local encoding component (+ L), and adding both local and global components (+ L&G).

	en	de	el	hi	ru
mBERT + POS	83.9	71.8	63.8	58.3	71.7
+ L	+0.1	+1.2	+0.3	+0.5	+0.3
+ L&G	+0.2	+2.3	+0.6	+0.9	+0.5
XLM-R + POS	87.6	81.3	81.1	76.5	81.1
+ L	+0.6	+0.5	+0.4	+0.2	+0.2
+ L&G	+1.0	+0.8	+0.8	+0.4	+0.2
mT5 + POS	87.9	81.0	79.4	75.8	78.8
+ L	+0.5	+0.8	+0.7	+0.6	+0.8
+ L&G	+0.8	+1.5	+1.1	+1.1	+1.7

Table 3.4: Ablation study of the ISDG encoder. Results (F1) are shown on XQuAD, collected from five runs on average. The improvement from the local and global components is largely consistent across the experimented languages.

The ablation study shows that the improvement from both components of the ISDG encoder is consistent across the evaluated languages. On average, the global encoding component contributes around 40% of the improvement, demonstrating the

effectiveness of encoding the approximated "soft" paths to address global multi-hop syntactic relations across sentences. Furthermore, even using only POS features can still provide around 0.1 - 0.2 F1 improvement over the corresponding baseline, indicating that the UD tokenization and POS features also contribute to the final performance, albeit to a lesser extent.

3.4 Discussion

The utilization of syntactic structures has been of great importance in NLP traditionally , especially in language representation, prior to the emergence of word embedding. However, the importance of syntactic structures has diminished since the advent of pretrained language models. Despite this, my research shows that the incorporation of syntactic structures, with the designed structural inference on Inter-Sentence Dependency Graph (ISDG), can provide useful auxiliary information for multilingual MRC tasks, and lead to substantial performance improvements for certain languages.

Experiments demonstrate that the ISDG encoder, with both local and global encoding components, outperforms the baseline multilingual pretrained language models on three benchmarks, XQuAD, MLQA, and TyDiQA-GoldP, and establishes the best on-average performance. Additionally, the results show that the global encoding component is the most impactful component, indicating that the global multi-hop syntactic relations across sentences play an important role in document understanding. Furthermore, the improvement is consistent across the experimented languages, with most languages showing positive impact from the UD features. However, different languages do not benefit equally from the syntactic features, possibly due to intrinsic linguistic differences and different feature quality across languages.

In summary, my work provides evidence for the continued importance of syntactic structures, particularly in the multilingual setting. By explicitly modeling the global

syntactic relations in a document-level graph, the proposed ISDG encoder enhances the document understanding capabilities of the machine reading comprehension task. I anticipate that my findings will inspire further research on incorporating syntactic structures into language models and developing more effective graph encoding techniques.

Chapter 4

Discourse Structures for Coreference Resolution

4.1 Introduction

This chapter investigates the incorporation of discourse structure in coreference resolution, a challenging task in NLP central to achieving full document understanding, as shown by the example in Chapter 1. Coreference resolution involves the semantic interpretation of entity mentions in a text, with the goal of grouping together those that refer to the same entity, particularly in the case of pronoun mentions which are more ambiguous than proper nouns. To achieve this, this work [69] focuses on performing different structural inference techniques that allow for higher-order decision making, as opposed to relying solely on local pairwise mention scoring. Empirical findings indicate that certain techniques from previous works do not lead to significant improvements using their designed discourse structures, while the clustering merging technique proposed in this research achieves state-of-the-art performance in 2020.

Figure 4.1: Illustration of the coreference resolution task: the system is expected to interpret the semantic meaning of entity mentions, and groups mentions of the same entity together.

4.1.1 Background and Motivations

Despite recent advancements in coreference resolution due to the use of contextualized embedding encoders such as ELMo and BERT, this task has remained challenging because of its demand on document-level understanding. As shown in Figure 4.2, the state-of-the-art model in 2020 shows a considerable improvement of 12.4% over the model introduced 2.5 years earlier, with representation learning playing a major role in this improvement [65, 66, 8, 9, 29, 30, 16, 26, 23, 25].

While previous models have typically relied on pairwise mention scoring for inference, with final decisions based on local pairwise decisions rather than global optimization, some have explored the use of higher-order inference for global optimization of coreference links. However, the gains reported from higher-order inference have been marginal, suggesting the need for further investigation into the methodology and impact of higher-order inference upon discourse structures in modern coreference resolution models, and pointing the way to future research directions.

4.1.2 Problem Formulation

In Section 4.2, the end-to-end coreference resolution system based on local pairwise decisions proposed by Lee et al. [29] is firstly introduced, which has served as the

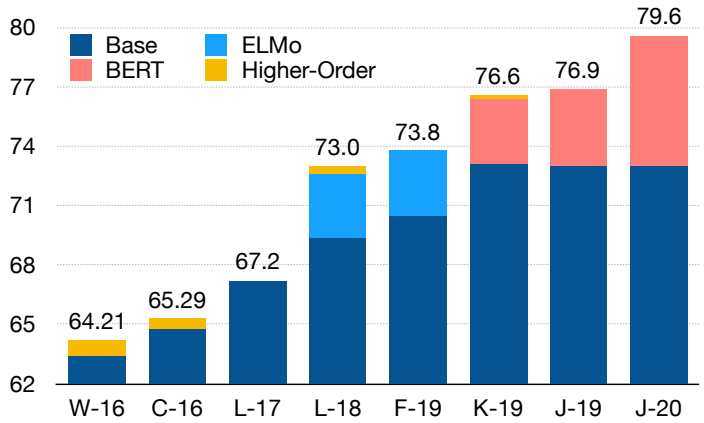

Figure 4.2: Performance of the recent state-of-the-art models on the CoNLL 2012 shared task. W-16: Wiseman et al. [66], C-16: Clark and Manning [9], L-17: Lee et al. [29], L-18: Lee et al. [30], F-19: Fei et al. [16], K-19: Kantor and Globerson [26], J-19: Joshi et al. [23], J-20: Joshi et al. [25].

foundation for many subsequent coreference resolution models [30, 26, 16, 23]. In Section 4.3, the use of Higher-Order Inference (HOI) based on discourse structure is investigated, a long-standing research goal that seeks to improve performance by conditioning the final clustering process on more than just local pairwise decisions. Four HOI approaches, as a form of structural inference, are implemented and experimented on top of the end-to-end model, two of which are original methods developed in this work. Emperical results indicate that certain HOI methods can indeed improve performance, although the gains are relatively small; while others are not able to bring improvement.

To enable a thorough comparison of different approaches, the end-to-end coreference system is implemented in PyTorch and experimented with two Transformer encoders, BERT and SpanBERT, to assess the effectiveness of different HOI methods when used in conjunction with these high-performing encoders. My study represents the first comprehensive analysis of multiple HOI approaches that leverage discourse structure side-by-side for the coreference resolution task.

4.2 Approach: Local Inference

The end-to-end coreference resolution system proposed by [29] serves as the backbone and baseline, as it is responsible for both extracting mentions and identifying entity clusters from a document input. The system operates by first performing a span enumeration stage, where candidate spans are enumerated and scored by the same model that performs entity resolution, enabling an end-to-end solution for coreference resolution. This approach represents a departure from previous state-of-the-art models [65, 66], which relied on a pipeline-like system that first extracted mentions using a separate model before resolving entities based on the mentions.

Since its introduction, the end-to-end system has been adapted in various follow-up works either directly [26] or as part of the design module [16]. Later, Joshi et al. [23] further improved the system by incorporating Transformers-based pretrained language models as encoders, such as BERT [14], which replace the LSTM-based encoder [20]. This approach has been shown to provide a significant performance boost, an the resulting Transformers-based end-to-end model is described in this section, as it serves as the foundation for this research into higher-order inference.

Span Enumeration Given an input document with T tokens, the model first enumerates all possible spans, and scores every span for being a likely mention, denoted by the mention score s_m. The model then greedily prunes spans by selecting top λT spans (ranked by s_m) as mention candidates that may appear in the final coreference clusters, discarding the rest of spans, with $\lambda \in (0, 1]$ being a hyperparameter. Let $\mathcal{X} = (x_1, \ldots, x_{\lambda T})$ be the list of all mention candidates in the document after the pruning, ordered by their appearance in the document.

Mention-Ranking For each mention candidate $x_i \in \mathcal{X}$, the model follows the mention-ranking (or mention-linking) strategy, by selecting a single coreferent an-

tecedent from all its preceding mention candidates $\mathcal{Y}_i = (\epsilon, x_1, \ldots, x_{i-1})$, with ϵ being a "dummy" antecedent that may be selected when x_i is not anaphoric (no antecedents should be selected).

The antecedent selection is performed by the pairwise scoring process accordingly, between the current mention candidate x_i and each of its preceding candidate $y \in \mathcal{Y}_i$. The final pairwise score $s(x_i, y)$ consists of three scores: how likely each candidate being a mention, measured by the two mention scores s_m; and how likely they refer to the same entity, measured by the coreference score s_c. The final score $s(x_i, y)$ can be denoted as follows:

$$
\begin{aligned}
s(x_i, y) &= s_m(x_i) + s_m(y) + s_c(x_i, y, \phi(x_i, y)) \qquad (4.1)\\
s_m(x_i) &= w_m \texttt{FFNN}_m(g_{x_i})\\
s_c(x_i, y) &= w_c \texttt{FFNN}_c(g_{x_i}, g_y, \phi(x_i, y))
\end{aligned}
$$

Both s_m and s_c are scalars computed by learnable FeedForward Neural Network (`FFNN`), and g_{x_i}/g_y is the embedding representation of the corresponding span. $\phi(x_i, y)$ represents additional meta features, such as the speaker and genre information.

Span Representation Following [29], the span representation g_{x_i} for x_i consists of the attended token representation of this span, where the token representation comes from the Transformers-based pretrained language model (PLM), and the attention is computed by another FeedForward network:

$$
\begin{aligned}
g^{\alpha}_{x_i} &= \sum_{k=\mathrm{START}_i}^{\mathrm{END}_i} \alpha_k \cdot h_k \qquad (4.2)\\
\alpha_k &= \mathrm{Softmax}_k\big(w_\alpha \texttt{FFNN}_\alpha(h_t)\big) \qquad (4.3)\\
h_k &= \mathrm{PLM}(t_k \mid t_1, ..., t_T) \qquad (4.4)
\end{aligned}
$$

$t_1, ..., t_T$ is the input token sequence, and h_k is the kth token embedding from the Transformers encoder. Additionally, the start and end token representation are also concatenated together, and the final span representation is described as:

$$g_{x_i} = h_{\text{START}_i} \oplus h_{\text{END}_i} \oplus g^{\alpha}_{x_i}, \tag{4.5}$$

where $\oplus$ denotes the concatenation.

Optimization For training, the marginal log-likelihood of all gold antecedents $\hat{\mathcal{Y}}_i \subseteq \mathcal{Y}_i$ for each $x_i \in \mathcal{X}$ is optimized, denoted by the coreference loss $\mathcal{L}_c$:

$$P(y) = \frac{e^{s(x_i,y)}}{\sum_{y' \in \mathcal{Y}_i} e^{s(x_i,y')}} \tag{4.6}$$

$$\mathcal{L}_c = -\log \prod_{x_i \in \mathcal{X}} \sum_{\hat{y} \in \hat{\mathcal{Y}}_i} P(\hat{y}) \tag{4.7}$$

$\mathcal{L}_c$ is constructed in a way such that for each span x_i, the model essentially learns a coreference distribution over its antecedents $\mathcal{Y}_i$.

Inference For inference, the selected coreferent antecedent for each span x_i is the one preceding candidate with the most pairwise score, denoted by $\text{argmax}_{y' \in \mathcal{Y}_i} s(x_i, y')$. The "dummy" antecedent could be selected if the span has no coreferent antecedents. The final entity clusters can be obtained by sequentially linking the selected antecedents together, illustrated by Figure 4.3. Mentions in the same cluster should ideally refer to the same entity.

4.3 Approach: Higher-Order Inference (HOI)

Although the end-to-end system has proven to be effective, one limitation of the mention-ranking approach is that the final clustering process is based solely on local

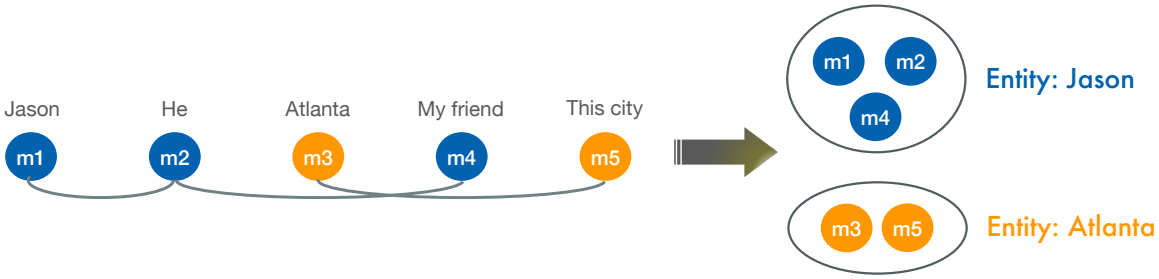

Figure 4.3: The inference process of mention-ranking: each node represents a span in the document, and is linked to the best coreferent antecedent based on the pairwise score from Eq (4.1). The final entity clusters are thus constructed by transitivity.

pairwise links. Specifically, a span is added to a cluster if there exists a link between that span and another span in the cluster, without considering the entire cluster's features. This approach may be suboptimal since coreference resolution is a clustering problem that naturally benefits from integrating more global features into the final clustering process.

Previous work has discussed the need for global features, and one example that illustrates this need is the pronoun problem described in Wiseman et al. [66]. In a simplified example in Figure 4.4, if there is a link between "he" and "you" and another link between "you" and "they," the mention-ranking system would form a cluster ["he", "you", "they"]. However, this clustering would be incorrect because "he" is a singular pronoun and "they" is a plural pronoun, even though the independent link between "you" and "they" could seem plausible since "you" could be either singular or plural in terms of grammar. In such cases, adding higher-order inference that considers more global features could potentially improve performance and avoid these types of errors.

4.3.1 HOI via Span Refinement

Two HOI methods presented by recent coreference work are based on span refinement that aggregates non-local features to enrich the span representation with more

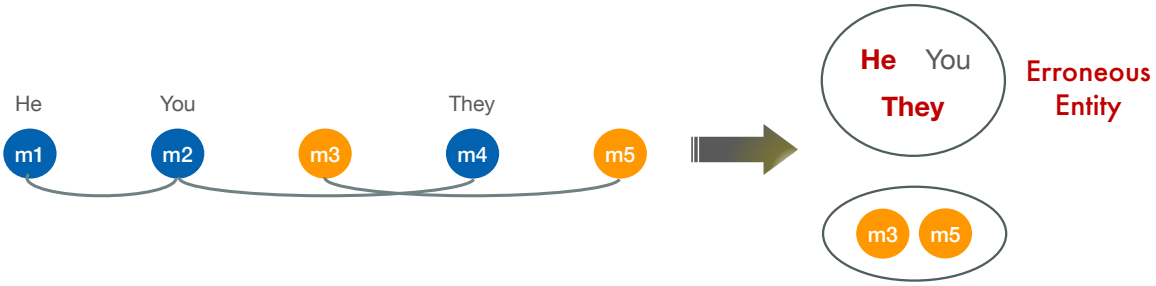

Figure 4.4: Error-prone local decisions of mention ranking.

"global" information. The updated span representation g'_x can be derived as Eq (4.8), where g'_x is the interpolation between the current span representation g_x and a refined representation a_x, controlled by the f_x which is a learned gating factor. g'_x is then used to perform another round of antecedent scoring in replacement of g_x.

$$g'_x = f_x \circ g_x + (1 - f_x) \circ a_x \tag{4.8}$$

$$f_x = \sigma(W_f[g_x \oplus a_x]) \tag{4.9}$$

σ denotes the sigmoid function, $\circ$ is the element-wise multiplication, and W_f is the learnable gate parameter. The following two methods share the same updating process for g'_x, but with different ways to obtain the refined span representation a_x, by regarding local decisions among spans as specific discourse structures.

Figure 4.5: Higher-order inference utilizes specific discourse structures constructed from local pairwise decisions between spans. Different methods employ their own structure and inference.

Attended Antecedent (AA) takes the antecedent information to enrich g'_x [30]. The refined span a_x is the attended antecedent representation over the current antecedent distribution $P(y)$ obtained from Eq (4.6), where $\mathcal{Y}(x)$ are the antecedents of x:

$$a_x = \sum_{y \in \mathcal{Y}(x)} P(y) \cdot g_y \tag{4.10}$$

For AA, the discourse structure taken is the antecedent pairwise coreference score of each span, resembling a diagonal score matrix for a document.

Entity Equalization (EE) takes the clustering relaxation as in Eq (4.11) to model the entity distribution [26]; $Q(x \in E_{y'})$ is the probability of the span x referring to an entity $E_{y'}$ in which the span y' is the first mention.

$$Q(x \in E_{y'}) = \begin{cases} \sum_{k=y'}^{x-1} P(y=k) \cdot Q(k \in E_{y'}) & y' < x \\ P(y=\epsilon) & y' = x \\ 0 & y' > x \end{cases} \tag{4.11}$$

The refined span a_x is the attended entity representation, where $e_y^{(x)}$ is the entity representation to which the span y belongs till the span x:

$$e_x^{(t)} = \sum_{y=1}^{t} Q(y \in E_x) \cdot g_y \tag{4.12}$$

$$a_x = \sum_{y=1}^{x} Q(x \in E_y) \cdot e_y^{(x)} \tag{4.13}$$

EE takes the discourse structure as clusters built upon the diagonal score matrix in AA, which is one step closer towards the final clustering objective, albeit the cluster

representation is in a relaxed form.

Span Clustering (SC) is a HOI method proposed in this work that also fits the span refinement paradigm. It constructs the actual clusters and obtains the "true" predicted entities by mention-ranking using $P(y)$ instead of modeling the "soft" entity clusters through the relaxation as in EE. This way, although the clustering process is not differentiable, the obtaining of true entities with the same empirical inference time as EE has made SC desirable.

The entity representation e_i for an entity cluster C_i is given by the attended spans in this cluster, and attention is :

$$e_i = \sum_{k \in C_i} \alpha_{i,k} \cdot g_k \tag{4.14}$$

$$\alpha_{i,k} = \text{Softmax}_{k \in C_i}\big(w_{\text{sc}}\text{FFNN}_{\text{sc}}(g_k)\big) \tag{4.15}$$

The entity clusters C_i are constructed in the same way as in the final cluster prediction. The refined span a_x is then equal to the representation of entity e_i to which it belongs.

SC regards the final entity clusters as the discourse structure directly; unlike EE, the clusters in SC are not in relaxed form, but rather represent the true clustering prediction.

4.3.2 HOI via Maintaining Clusters

Cluster Merging (CM) is another HOI method proposed in this work that performs sequential antecedent ranking combining both antecedent and entity information to gradually build up the entity clusters, distinguished from the previous span refinement methods that simply have another round of scoring.

Algorithm 1 describes the scoring process for CM. g_i is the ith span, $\mathcal{Y}(i)$ is the

indices of g_i's antecedents, and C_i is the cluster that g_i belongs to. The final score $s_x(y)$ now consists of both the pairwise score f_a as Eq (4.1) between two spans, and a new cluster score f_c that checks the compatibility of a span and a cluster. To avoid overlapping between f_a and f_c, f_c is set as 0 if the cluster is the initial cluster (L6). Thus, f_c becomes another source of consultation such that when $f_c > 0$, the span g_x is regarded likely to match with the cluster C_y, and vice versa. f_c is computed by `FFNN` similar to f_a, and $\phi(C_y)$ is the meta-feature such as the cluster size.

Algorithm 1 Antecedent Ranking for `CM`

1: **procedure** RANKING($g_1, \cdots, g_N$)
2: $C_{i=1,\cdots,N} \leftarrow g_i$
3: $R \leftarrow$ ranking_order($g_1, \cdots, g_N$)
4: **for** $x = R_1 \cdots R_N$ **do**
5: **for** $y \in \mathcal{Y}(x)$ **do** ▷ Parallelized
6: $f_c(g_x, C_y) \leftarrow 0$ if $C_y = g_y$
7: $s_x(y) \leftarrow f_a(g_x, g_y) + f_c(g_x, C_y, \phi(C_y))$
8: $y' \leftarrow \text{argmax}_{y \in \mathcal{Y}(x)} s_x(y)$
9: **if** $y' \neq \epsilon$ **then**
10: merge C_x and $C_{y'}$
11: **return** $s_1, \cdots, s_N$

The `CM` approach can be configured in two simple ways. The first option is the sequential left-to-right ranking order, while the second option is the easy-first order (L3). In the easy-first order, the sequence is determined based on each span's maximum antecedent score, allowing the system to build the most confident clusters first [44, 9]. Additionally, there is the choice between element-wise mean or max-reduction for the spans in the two merging clusters (L10).

`CM` takes the discourse structure as partial clusters, which are maintained and incremented during the inference process, unlike the previous three span refinement methods that build the discourse structure at once.

4.4 Evaluation and Analysis

All models are evaluated on CoNLL 2012 English shared task [50], the standard benchmark for coreference resolution. Six models are developed as follows:

- `BERT`: BERT [14] as the encoder
- `SpanBERT`: SpanBERT [25] as the encoder
- `+AA`: `SpanBERT` with attended antecedent
- `+EE`: `SpanBERT` with entity equalization
- `+SC`: `SpanBERT` with span clustering
- `+CM`: `SpanBERT` with cluster merging

	MUC			B^3			$CEAF_{\phi_4}$			
	P	R	F1	P	R	F1	P	R	F1	Avg. F1
`L-17`	78.4	73.4	75.8	68.6	61.8	65.0	62.7	59.0	60.8	67.2
`L-18`	81.4	79.5	80.4	72.2	69.5	70.8	68.2	67.1	67.6	73.0
`F-19`	85.4	77.9	81.4	77.9	66.4	71.7	70.6	66.3	68.4	73.8
`K-19`	82.6	84.1	83.4	73.3	76.2	74.7	72.4	71.1	71.8	76.6
`J-19`	84.7	82.4	83.5	76.5	74.0	75.3	74.1	69.8	71.9	76.9
`J-20`	85.8	84.8	85.3	78.3	77.9	78.1	76.4	74.2	75.3	79.6
`BERT`	85.0	82.5	83.8	77.3	74.0	75.6	74.9	70.7	72.8	77.4
`SpanBERT`	85.7	85.3	85.5	78.6	78.6	78.6	**76.8**	74.8	75.8	79.9
+ `AA`	**86.1**	84.8	85.4	**79.3**	77.3	78.3	76.0	74.7	75.4	79.7
+ `EE`	85.7	84.5	85.1	78.5	77.4	77.9	76.7	73.4	75.0	79.4
+ `SC`	85.5	85.2	85.4	78.4	78.5	78.4	76.5	74.1	75.2	79.7
+ `CM`	85.9	**85.5**	**85.7**	79.0	**78.9**	**79.0**	76.7	**75.2**	**75.9**	**80.2**

Table 4.1: Results on the test set of CoNLL 2012 English shared task. The averaged F1 of MUC, B^3, $CEAF_{\phi_4}$ is the main evaluation metric. Note that `BERT` and `SpanBERT` completely rely on only local decisions without any HOI. Particularly, `+AA` is equivalent to Joshi et al. [25]. See Figure 4.2 for acronyms of the previous works.

Table 4.1 presents a comparison of HOI models to previous state-of-the-art systems. The results show that SpanBERT outperforms BERT, with an average improvement of 2.4%. `CM` model brings improvement over `SpanBERT`, achieving the best

Avg-F1 score of 80.2. Both the encoder and the `CM` HOI mothod are able to contribute positively to the final performance. To understand each model better, a more thorough analysis is performed as follows.

4.4.1 HOI Impact

Three HOI methods based on span refinement, `AA`, `EE`, and `SC`, show negative impact upon local decisions. It is suspected that error propagation from antecedent-ranking may downgrade the quality of refinement. On the other hand, `CM` shows improvement, suggesting that maintaining entity clusters can be superior to span refinement, though at a cost of more inference time from the sequential ranking process.

Direct Impact To evaluate the direct impact of HOI, the trained models of each HOI method is evaluated on the test set while turning off HOI, making them compatible with `SpanBERT`. This analysis shows that the average performance drop with respect to Avg-F1 after turning off HOI is less than 0.2 for all methods, implying that none of the HOI methods have a significant direct impact on the final performance of the model using SpanBERT.

Link Changes Furthermore, the change of coreferent links with respect to correctness are examined. Specifically, Table 4.2 shows the four types of link changes before and after HOI. The results demonstrate that the benefits from HOI are diminished because the effects are two-sided: there are roughly equal amounts of links (about 1%) becoming correct or incorrect after HOI, thereby diminishing the positive effects from HOI. Therefore, none of the HOI methods lead to significant improvement overall.

Despite the improvement being relatively marginal, the utilization of discourse structures, especially the cluster merging method, still shows performance gain. I suggest this direction for future research to further leverage the power of structural

inference for coreference resolution.

	W2C	C2W	C2C	W2W
+ AA	240.8 (1.3)	241.2 (1.3)	16262.2	2168.4
+ EE	244.1 (1.3)	245.3 (1.3)	16183.3	2136.3
+ SC	248.2 (1.3)	262.0 (1.4)	16184.4	2146.0
+ CM	226.4 (**1.2**)	235.0 (**1.2**)	16446.0	2180.0

Table 4.2: Averaged statistics on the test set prediction of four HOI approaches. W2C represents the number of mentions that are linked to a **W**rong antecedent before HOI and are linked to a **C**orrect antecedent after HOI; vice versa for C2W. C2C/W2W is the number of mentions that are both linked to **C**orrect/**W**rong antecedents before and after HOI. Parentheses indicate the percentage of corresponding numbers per row.

It is important to note that the impact of HOI extends beyond global decisions. During training, HOI serves as a form of regularization that indirectly affects local decisions, as HOI and local ranking are mutually dependent. This indirect influence of HOI makes it challenging to accurately assess its true impact, which could be investigated further in future research.

Personal Pronouns In addition, I want to emphasize the impact of HOI on the personal pronoun issue discussed in Section 4.3.

Direct Inference Table 4.3 presents the numbers of links where one pronoun incorrectly selects another pronoun with a different plurality as its antecedent (SP/PS). The findings show that adopting HOI has a slightly greater impact than switching to a more advanced encoder. `AA` reinforces the pronoun representation to bias towards singularity, resulting in lower SP error and higher PS error, while the difference between `BERT` and `SpanBERT` is negligible on SP/PS.

The general types of coreferent errors involving two pronouns are also examined, namely False Link (FL) and Wrong Link (WL). FL falsely links a non-anaphoric

	SP	PS	FL	WL	BC
BERT	2.3	6.5	213.8	186.3	48.8 (3.5)
SpanBERT	2.8	6.6	218.3	168.0	**43.8** (2.7)
+ AA	**1.8**	8.8	214.2	**159.4**	44.8 (2.4)
+ EE	**1.8**	**5.5**	210.0	165.3	44.0 (2.5)
+ SC	3.8	7.2	223.6	170.0	45.4 (3.0)
+ CM	3.0	6.6	**208.0**	162.2	**43.8** (2.6)

Table 4.3: Averaged statistics on the test set prediction of different approaches. SP is the number of coreferent links from **S**ingular to **P**lural personal pronouns; vice versa for PS. FL (False Link) and WL (Wrong Link) is the number of conreferent link errors that involve two personal pronouns. BC is the number of clusters that contain both singular and plural pronouns, and the parentheses indicate the numbers of BC that contain ambiguous pronouns such as "you".

pronoun to another pronoun as antecedent, while WL links an anaphoric pronoun to another incorrect pronoun as antecedent. Table 4.3 indicates that EE and CM reduce FL errors by over 4%, suggesting that the aggregation of non-local features leads to more conservative linking decisions. However, adopting an advanced encoder has a greater impact on WL errors, with SpanBERT reducing these errors by almost 10% compared to BERT, implying that representation learning is still more critical for semantic matching in current research stage.

Indirect Inference Table 4.3 shows the number of erroneous clusters in predictions that contain both singular and plural pronouns. Surprisingly, few of these clusters include ambiguous pronouns such as "you" in either approach, further moderating the long-standing motivation for HOI. Additionally, changing the representation from BERT to SpanBERT has a significantly greater impact, reducing the number of erroneous clusters by 10%. In contrast, the four HOI methods do not bring a significant difference.

4.5 Discussion

In this study, I introduce the incorporation of discourse structure in coreference resolution, and focus on leveraging different structural inference upon it to make higher-order decisions, rather than relying solely on local pairwise mention scoring. Through empirical evaluation, I found that certain techniques from previous works were not able to bring meaningful improvements using the designed discourse structure, while the clustering merging technique that I proposed was able to achieve state-of-the-art performance in 2020.

This work highlights the importance of utilizing discourse structure in coreference resolution. Previous research has shown that coreference resolution is a challenging task that requires document-level understanding, and this study suggests that incorporating discourse structure can be meaningful to improve the performance of coreference resolution systems. Moreover, this study indicates that higher-order inference, which considers more global features in the final clustering process, can potentially benefit the personal pronoun issue in coreference resolution and avoid errors caused by local pairwise decisions.

These findings also shed light on the impact of advanced encoders and HOI methods in coreference resolution. It is observed that SpanBERT, a Transformer-based pretrained language model, outperformed BERT, and that none of the HOI methods showed significant improvements over SpanBERT. However, the clustering merging technique could bring marginal improvements, suggesting that maintaining entity clusters can be superior to span refinement.

Furthermore, this study demonstrates that the indirect influence of HOI is challenging to assess, as it serves as a form of regularization that affects both global and local decisions. This observation highlights the importance of evaluating the true impact of HOI in future research.

In summary, my study contributes to the ongoing effort to improve coreference

resolution systems and highlights the importance of incorporating discourse structure and higher-order inference techniques in this task. My findings suggest that future research in this area should focus on exploring more effective ways to leverage discourse structure and investigate the true impact of HOI in coreference resolution.

Chapter 5

Relation Structures for Information Extraction

5.1 Introduction

Chapter 3 and 4 discuss the structural inference on two linguistic-related structures: syntactic and discourse structure, and demonstrate how they could contribute positively to different document understanding tasks. In this chapter, I expand the scope of utilized structures from linguistic-related structures to a knowledge-specific structure, referred as relation structure, which concerns the expressed relations from a predefined relation set between two entities in a document. The relation structure is commonly employed in various knowledge bases. This research work aims to fuse multi-facet information of document entities, including their coreference and relation information, through inference on constructed relation graphs. Especially, this work targets on document-level joint information extraction, and proposes a novel formulation of structural inference that bridges multi-task learning in this problem. Experimental results suggest that the inference on the relation structure is effective towards the motivation, and achieves the state-of-the-art performance on two

datasets.

5.1.1 Entity-Centric Relation Extraction

Recently, document-level relation extraction has become an area of growing interest, particularly since the introduction of large-scale datasets such as DocRED [80]. This task requires inter-sentence reasoning over global entities and involves classifying relation instances at the entity-level. Each entity of this task is an entity cluster of coreferent mentions across the document. In contrast to sentence-level relation extraction, document-level extraction considers entity-to-entity interactions in the entire context of document, which is referred as entity-centric information extraction (entity-centric IE).

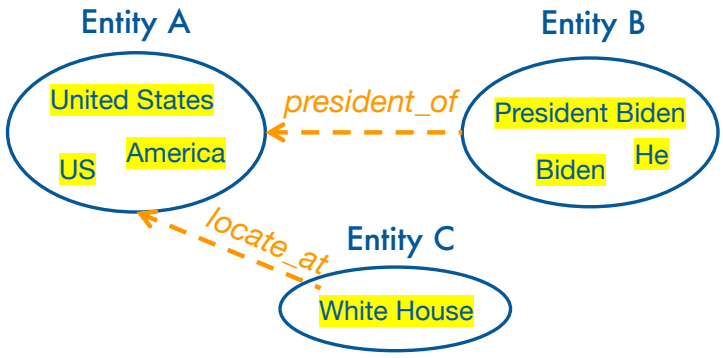

Figure 5.1: Example of the document-level information extraction task.

Recent research in entity-centric studies has made significant advancements in global reasoning while considering the entities as given [41, 90, 67, 54]. However, the more practical end-to-end setting that involves jointly extracting global entities and relations has received less attention. This setting imposes an additional burden on the model as it must resolve mentions, coreference, and relations simultaneously.

In this study [68], I address this end-to-end setting by targeting the extraction of all gold triples (e_h, e_t, r) from a given document. A triple instance is evaluated as correct only if both the head/tail entity clusters (e_h/e_t) as well as the relation r are

correct. In this setting, the model must handle three tasks at once: mention recognition, entity resolution, and entity-level relation extraction, which requires strong document understanding from the modeling.

5.1.2 Background and Motivations

To address this multi-task learning problem, recent span-extraction-based models have employed two popular methods. One is to share the encoder, and therefore the mention representation, in multi-task learning while decoding separately in a pipeline manner [37, 55]. The other is to use graph propagation to enrich mention representation with task-specific decisions, such as in DyGIE [38].

However, these task interactions only occur at the representation level and still employ pipeline-like decoding, with no explicit interactions that directly influence the decisions of different tasks, which yield independent decisions of each decoder without fusing information of different tasks. Furthermore, recent research [63, 69, 83] has shown that under strong encoders like BERT [23], the benefits of graph propagation are diminished as they can model long-range dependencies effectively.

In this work, I aim to further improve performance by focusing on task interactions and proposing the formulation of explicit interactions that utilize unique task characteristics. This approach mitigates negative effects such as error propagation from pipeline decoding and leverages the potentials of relation structures to enhance performance, fusing multi-facet information of different tasks together.

5.1.3 Problem Formulation

To address the aforementioned motivation, a second source of coreference scores is incorporated based on the predicted relation structures from the model, in addition to the regular scoring on mention pairs for coreference resolution that is independent of relation extraction. My formulation exploits the observation that for a pair of

mentions (m_x, m_y) referring to the same entity, their relation scores s^r should be similar when paired with any other mentions m_k, such that $s^r(m_x, m_k) \approx s^r(m_y, m_k)$, whereas for non-coreferent pairs, their relation scores towards other mentions tend to be divergent.

To implement this, the relation scores s^r for each mention are formulated as a local graph, and the model learns a distance metric as the secondary coreference score that checks the compatibility of local graphs of a mention pair. This added term acts as a bridge between coreference and relations, providing explicit task interactions that circumvent independent decoding of each task, and achieves multi-facet information fusing.

Experiments are conducted in five multi-task settings, ranging from the pipeline approach to three different interaction methods. These experiments assess the impact of task interactions for document-level joint information extraction, providing a comprehensive evaluation of the proposed approach leveraging relation structures.

5.2 Approach

5.2.1 Independent Decoding

This subsection introduces models of different settings that decode each task independently.

For coreference resolution (COREF), the Transformers-based end-to-end architecture [23] is adopted as described in Section 4.2 that resolves both mention extraction and coreference, with two slight modifications.

First, the pairwise mention scoring is simplified, by only keeping the lightweight bilinear scoring and discarding the slow antecedent scoring, as no noticeable degradation is observed in preliminary experiments, likely due to the fact that COREF in current IE datasets is easier (e.g. pronouns are not considered in DocRED). Second,

the prediction of singleton entities (entity with only one mention) is supported by optimizing mention scores, described as below.

Singleton Recognition The original mention-ranking system does not support the extraction of singleton entities, due to the fact that each cluster needs at least a pair of linked spans (see Section 4.2). However, singletons can be desirable under many usage scenarios, which is the case for entity-centric IE.

Several previous work has addressed the singleton problem from different perspectives [81, 84]. My model is built upon the end-to-end system described in Section 4.2, and further recognizes singletons based on the simple strategy as follows: I make use of the mention score s_m in the final linking process, and create a singleton cluster for any candidates with $s_m > 0$ that have not yet found any antecedents, which now poses an additional requirement on the mention score, such that only valid mentions should have $s_m > 0$.

Let $\Psi^+ \subseteq \mathcal{X}$ be the set of gold mention candidates, and $\Psi^- = \mathcal{X} \setminus \Psi^+$ be the set of other mention candidates. The mention score is optimized with the binary cross-entropy loss $\mathcal{L}_m$ and joinly train with the coreference loss $\mathcal{L}_c$ from Eq (4.7):

$$\begin{aligned} \mathcal{L}_m = &- \sum_{x_i \in \Psi^+} \log \sigma(s_m(x_i)) \\ &- \sum_{x_j \in \Psi^-} \log(1 - \sigma(s_m(x_j))) \end{aligned} \tag{5.1}$$

$$\mathcal{L} = \mathcal{L}_c + \alpha_m \cdot \mathcal{L}_m \tag{5.2}$$

σ is the sigmoid function, and α_m is a hyperparameter. $\mathcal{L}$ is the final loss composed of two tasks. In practice, negative sampling is also performed on Ψ^- dynamically, so that Ψ^+ and Ψ^- are of similar sizes ($|\Psi^+| \approx |\Psi^-|$), to alleviate the negative effects from the skewed class distribution.

In the new selection process, the selected non-dummy antecedent y is still regarded

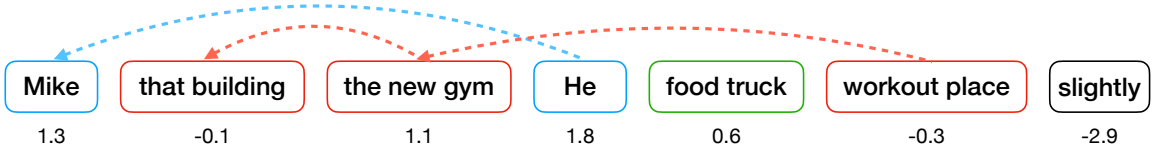

Figure 5.2: Example of the new antecedent selection process that support singletons. Each arrow indicates the selected antecedent (the dummy antecedent is excluded), and the mention score s_m is shown below each mention. Mentions of the same predicted clusters are marked in the same color. Although no antecedent is selected for "food truck", it will still be assigned as a singleton cluster because of $s_m = 0.6 > 0$. "that building" and "workout place" are still assigned to the corresponding cluster even though their $s_m < 0$, to allow some slacks on the mention score prediction. "slightly" will not be assigned to any clusters.

as valid by $y = \text{argmax}_{y' \in \mathcal{Y}_i} s(x_i, y')$, even though the mention score of either candidate can be negative ($s_m(x_i) < 0$ or $s_m(y) < 0$). This is to allow certain slacks on the mention score prediction which could help with the mention recall. Figure 5.2 shows three different cases of the predicted clusters by the `SR` model.

For relation extraction (RE), the recent model ATLOP [90] is adopted that takes a document and its entities as input, and produces relation triples on the entity-level, by learning adaptive thresholds for relation scores. One minor modification is made that the localized context pooling is not utilized, as the task interactions are aimed to be encoder-agnostic without using BERT-specific features. For both models, the concatenated embedding of mention boundary is used as mention representation.

Pipeline The first setting is the pipeline approach that trains COREF and RE models separately, and decodes in the naive pipeline manner, where the extracted entities (entity clusters) are first obtained by the COREF model, and then fed to the RE model that produces the final relation triples (Figure 5.3).

Joint The second setting features the common joint paradigm adopted in most related work [38, 83, 15] that shares the same encoder and mention representation

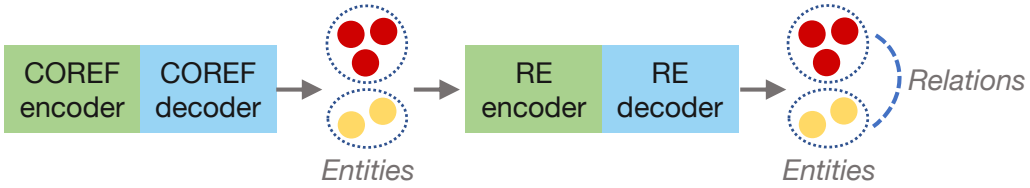

Figure 5.3: Pipeline setting: no task interactions.

for all tasks, while keeping independent decoders for COREF and RE that are jointly trained in a multi-task manner (adding two losses). This and later settings employ "shared representation" as the first type of task interactions (Figure 5.4).

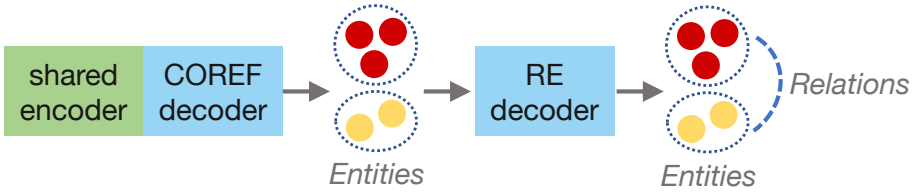

Figure 5.4: Joint setting: shared encoder.

5.2.2 Shallow Task Interactions

Joint-M As the COREF model operates on the mention-level but ATLOP scores between entities directly, another joint model is proposed that unifies all scoring on the mention-level, allowing more straightforward inter-task interference later.

Same as the baseline, the COREF module in Joint-M still generates a set of mention candidates $(m_1, .., m_n)$ and their pairwise coreference scores $s^c(m_x, m_y)$ indexed by $x, y \in [1, n]$. Different from ATLOP that obtains entity representation first and performs relation scoring among entities, the RE module in Joint-M simply obtains mention-level pairwise relation scores s^r through a lightweight biaffine scoring, di-

rectly on the same set of mention candidates. More formally:

$$s^c(m_x, m_y) = g_x W^c g_y^T + s^m(g_x) + s^m(g_y)$$
$$s^{r_i}(m_h, m_t) = g_h W^{r_i} g_t^T + s^{h_i}(g_h) + s^{t_i}(g_t)$$

g denotes the embedding of the corresponding mention; W^c/W^{r_i} are learned parameters for COREF scoring and RE scoring of the ith relation type. $s^m/s^{h_i}/s^{t_i}$ are additional prior scores predicted by separate feed-forward networks on how likely the mention span is a gold mention (s^m) or a head/tail mention for the ith relation type (s^{h_i}/s^{t_i}).

Though the original relation labels are on the entity-level, the labels are transferred to the mention-level by letting any mention pair (m_h, m_t) express the same relations as their belonging entities (e_h, e_t), with $m_h \in e_h$ and $m_t \in e_t$. By doing so, the model is forced to learn more inter-sentence reasoning implicitly in the encoding stage to aggregate different local context of mentions belonging to the same entity.

Similar mention-level decoding is also adopted in previous work [83, 15]. In particular, Eberts and Ulges [15] applies multi-instance learning on mentions; nevertheless, their approach regards mention-level labels as latent variables and still needs to formulate the entity representation, while Joint-M offers a simpler paradigm that discards entities in the model completely, and yields similar performance as multi-instance learning in preliminary experiments.

Joint-M is trained similar to Joint and still employs the same task interaction as "shared representation". For inference, the entity-level relation labels are obtained by simply averaging the mention-level relation scores from the cartesian product of the predicted entity clusters, denoted as below:

$$s^{r_i}(e_h, e_t) = \text{MEAN}\{s^{r_i}(m_h, m_t)\}, \quad \forall (m_h, m_t) \in e_h \times e_t \tag{5.3}$$

+GP In this setting, **G**raph **P**ropagation is applied upon Joint-M, which has the similar formulation as DyGIE++ [63]. Distinguished from the original DyGIE++ that only extracts intra-sentence relations, it is adapted for document-level graph propagation as follows.

After the RE scoring in Joint-M, each mention candidate is regarded as a graph node and their relation scores as weighted graph edges. Instead of propagating on one graph as DyGIE++, each relation type inherently forms its own directed subgraph that only consists of edges of a specific type. In +GP, subgraph propagation is performed respectively, then the final node representation is obtained by aggregating nodes from each subgraph.

More formally, let R be the set of relation types. $|R|$ heterogeneous relation subgraphs can thus be constructed after the RE scoring. Graph Attention Network (GAT)-like propagation [61] is then applied on each subgraph:

$$\alpha_{ht}^{r_i} = \frac{\exp\left(\mathrm{ReLU}\left(s^{r_i}(m_h, m_t)\right)\right)}{\sum_{k \in \mathcal{N}_h} \exp\left(\mathrm{ReLU}\left(s^{r_i}(m_h, m_k)\right)\right)} \tag{5.4}$$

$$g_h^{r_i} = \tanh\left(\sum_{t \in \mathcal{N}_h} \alpha_{ht}^{r_i} \cdot g_t W^{r_i}\right) \tag{5.5}$$

$$\hat{g}_t = g_t + \sum_{r_i \in R} g_h^{r_i} / |R| \tag{5.6}$$

$\hat{g}_t$ is the new tail embedding after the propagation that will replace g_t; $\mathcal{N}_h$ is the set of neighboring nodes of m_h, which in this case are all the mention candidates. W^{r_i} is the learned matrix for type-specific node transformation. The new head embedding $\hat{g}_h$ will also be obtained accordingly.

With the new node embedding that fuses the RE decisions, +GP performs the COREF scoring as in Joint-M but using the updated mention representation, accomplishing implicit task interactions. No further propagation is performed on COREF graphs as it is shown little effects by previous work [63, 69].

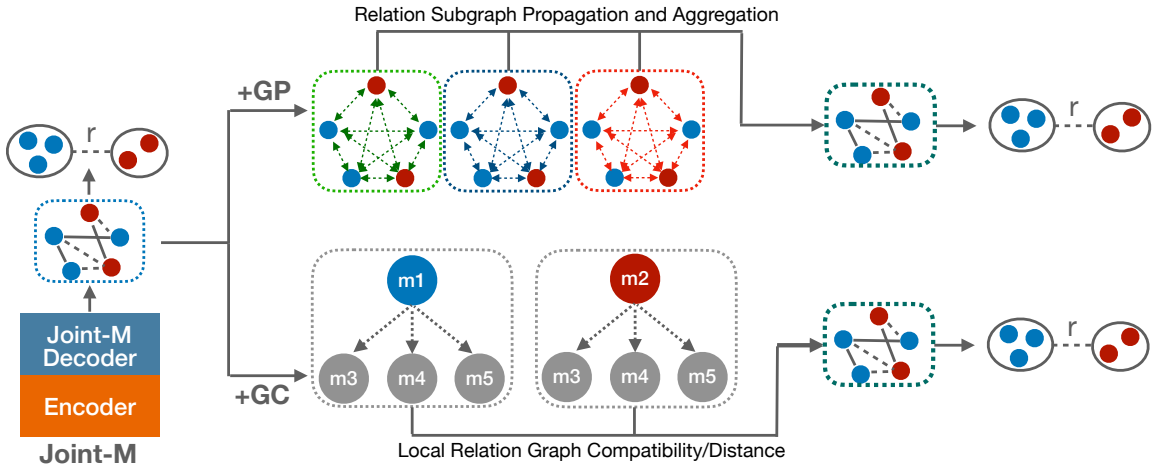

Figure 5.5: Proposed approach with different task interactions. Each node represents an extracted mention in the document.

5.2.3 Fuse Multi-Task Decoding

Although the previous +GP model achieves shallow task interaction through propagation on mention representation, each task decoding is still independent. In this subsection, the proposed +GC is described that conducts structural inference on relation graphs to achieve deeper task interactions and fuses multi-facet information of entity mentions in the document.

+GC As above interactions are all implicit, I propose to leverage task characteristics between COREF and RE to design explicit task interactions, dubbed **G**raph **C**ompatibility as a new setting upon Joint-M. Specifically, each node after RE scoring can be regarded as a local graph that connects to all other nodes with weighted edges (relation scores). If two mention nodes are from the same entity cluster, their local graphs should be similar, since they are forced by Joint-M to have the exact same relations to other nodes; vice versa, if two nodes do not refer to the same entity, their relations (weighted edges) to other mentions are likely to be distant from each other.

Therefore, +GC model learns a distance metric to check the "compatibility" of local relation graphs, as an additional clue of how likely two mentions are coreferent.

An illustration of this process that converts coreference into graph distances on relation structures is shown in Figure 5.6.

More formally, this second source of coreference scores $\hat{s}^c$ can be denoted as:

$$d^{r_i}_{x,y} = \sum_{k \in \mathcal{N}_{x,y}} |s^{r_i}(m_x, m_k) - s^{r_i}(m_y, m_k)| \tag{5.7}$$

$$\hat{s}^c(m_x, m_y) = \sum_{r_i \in R} \beta^{r_i} \cdot d^{r_i}_{x,y} \tag{5.8}$$

$$\widetilde{s}^c(m_x, m_y) = s^c(m_x, m_y) - \lambda \hat{s}^c(m_x, m_y)$$

$d^{r_i}_{x,y}$ is the raw L1 distance between the two local graphs by all neighboring edges of the r_i relation type. $\hat{s}^c$ is the final distance/compatibility of two local graphs, weighted by the learned parameter β^{r_i} that determines the importance of each r_i; higher $\hat{s}^c$ indicates more diverging graphs. The final coreference score $\widetilde{s}^c$ interpolates the original s^c and the new distance $\hat{s}^c$, with λ being a hyperparameter.

Overall, +GC enables explicit multi-task fusing through task interactions that bridge COREF and RE together: RE can affect COREF directly, while COREF also pushes similar RE scores for coreferent pairs during back-propagation. The final distance $\hat{s}^c$ is optimized by a contrastive loss as in Eq (5.9) that is commonly used in Siamese Network [28]. For simplicity, denote $D = \hat{s}^c(m_x, m_y)$, $Y = 1$ when (m_x, m_y) is from the same entity, and $Y = 0$ elsewise. m is the margin as a hyperparameter. $\hat{\mathcal{L}}$ is added as the third loss in Joint-M's training.

$$\hat{\mathcal{L}} = Y \cdot D^2 + (1 - Y) \cdot \max(0, m - D)^2 \tag{5.9}$$

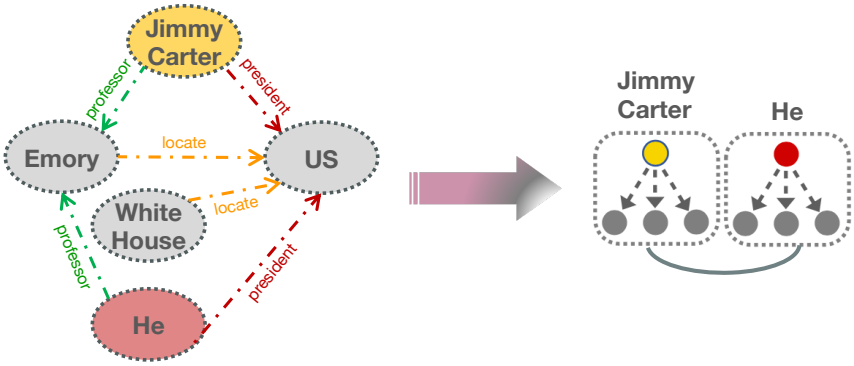

Figure 5.6: Illustration of fusing coreference and relation through the relation graph (the +GC formulation). If two mentions refer to the same entity, their local relation structures should be similar; vice versa, these two relation structures tend to have larger graph distance.

5.3 Evaluation and Analysis

Five proposed settings are evaluated on two datasets: DocRED [80], which comprises Wikipedia documents, and DWIE [83], which comprises news articles. The evaluation protocol and metrics are consistent with previous work on the end-to-end joint setting [15, 62], and are identical for both datasets. For DocRED, the provided split is used to submit test set predictions to its official Codalab competition. For DWIE, I randomly hold out 10% of the training set for model tuning and use the entire training set for the final evaluation to ensure consistency with previous work.

The baseline implementation is adapted from the PyTorch COREF model by [69] and the ATLOP RE model by [90]. The proposed Joint-M, +GP, and +GC models are implemented in PyTorch. For all experiments, SpanBERT-Base [24] is used as the encoder, which performs slightly better than BERT according to the findings.

Evaluation results Table 5.1 reports the evaluation results on both datasets with three metrics for mention extraction (ME), coreference resolution (COREF), and relation extraction (RE), with RE being the main point of interest for the end-to-end evaluation. The table also shows the performance of three previous works that employ

		DocRED				DWIE		
		ME	COREF	RE	RE Ign	ME	COREF	RE
LSTM	Verlinden et al. [62]	-	83.6*	25.7*	-	-	91.5*	52.1*
BERT	Zaporojets et al. [83]	-	-	-	-	-	91.1	50.4
	Eberts and Ulges [15]	92.99*	82.79*	40.38*	-	-	-	-
	Pipeline	92.56	84.09	38.29	35.88	96.09	92.80	57.76
	Joint	93.34	84.79	38.94	36.64	96.16	92.87	59.32
	Joint-M	93.33	84.83	39.65	37.17	96.47	92.91	61.01
	+GP	**93.38**	84.85	40.12	38.09	96.37	93.05	61.95
	+GC	93.35	**84.96**	**40.62**	**38.28**	**96.57**	**93.47**	**62.85**

Table 5.1: Evaluation results on the test set of DocRED and DWIE. Three metrics are included: (1) Mention Extraction (ME) in mention-level F1 score (2) Coreference Resolution (COREF) in averaged F1 score of MUC, B^3, and $CEAF_{\phi_4}$ (3) Relation Extraction (RE) in entity-level F1 score. DocRED also provides a F1 score (RE Ign) that excludes shared relational facts between training and evaluation. Three related work with the same end-to-end objective are shown, and they all employ certain mention-level decoding similar to our Joint-M. Note that Verlinden et al. [62] also utilizes external knowledge; Eberts and Ulges [15] is not directly comparable as their reported numbers are on a self-split development set instead of the official test set.

the same end-to-end evaluation approach. Note that [15] is not directly comparable as they do not use the official test set. All previous works adopt "shared representation" as a basic task interaction, with [83] also utilizing DyGIE-like graph propagation as an additional interaction, similar to the +GP setting.

My approach yields improvements in COREF of 1.4/2.0 F1 on DocRED/DWIE, respectively, compared to previous works. Furthermore, it achieves the best performance on RE for both datasets, with up to 10.8 F1 improvement for DWIE.

Comparing within the five multi-task settings, the Pipeline model has no interactions and performs the worst. By simply sharing the encoder, Joint consistently outperforms Pipeline on both datasets. Joint-M improves over Joint by 0.7 F1 on both datasets, indicating that forcing mention-level decoding while retaining the same relation labels as entities can be an effective strategy. Adding either +GP or +GC on top of Joint-M leads to further improvements in RE of up to 1.0/1.8 F1 on the two datasets, bringing the total RE improvement over Pipeline to 2.3/5.1 F1. No-

tably, +GC consistently outperforms +GP on both datasets, demonstrating that task-specific design for explicit interactions and multi-facet information fusing is more effective than general but implicit interactions.

Table 5.1 also indicates that although +GC achieves the best performance in both COREF and RE, the improvement in COREF is not as significant. As the effect of +GC goes both ways, with RE directly affecting COREF during inference and COREF regularizing RE during training, additional analysis is performed to demonstrate that regularization plays a more significant role in improving RE performance.

COREF			RE		
P	R	F	P	R	F
+0.2	+0.9	+0.6	+2.0	+0.6	+1.7

Table 5.2: Deltas of performance on the test set of DWIE applying +GC upon Joint-M. COREF and RE are evaluated separately (RE are given gold entities at evaluation). P/R/F is the precision/recall/F1 score.

Based on the dataset statistics, a majority of entities in both DocRED and DWIE are singletons. This characteristic creates an inductive bias in COREF towards non-linking decisions, leaving less room for improvement in COREF performance using graph distance $\hat{s}^c$. To further understand the impact of +GC, the performance changes of individual COREF and RE modules are examine on the DWIE test set, as shown in Table 5.2.

It is observed that +GC improves the RE module alone by 2% precision and an overall 1.7 F1 score, indicating that the regularization power from the graph distance is effective. However, the improvement in COREF is less significant, with an overall 0.6 F1 score, suggesting that although the graph distance brings two-way interactions between COREF and RE, RE benefits more while the direct contribution to COREF is trivial. Further analysis could focus on studying task interactions in-depth through this explicit interaction setting.

+GC improves the RE module alone by 2% precision and by an overall 1.7 F1 score, indicating that the regularization power from the graph distance is effective. By contrast, COREF improves much less by an overall 0.6 F1 score, suggesting that although the graph distance brings two-way interactions between COREF and RE, RE actually benefits more while the direct contribution to COREF is more trivial. More analysis can be a follow-up research that studies task interactions in-depth through this explicit interaction setting.

5.4 Discussion

This research work continues on the topic to improve document understanding tasks through structural inference upon relation structures. While previous work has focused on more linguistic-related structures, such as syntactic and discourse structures, this work underscores the importance of leveraging knowledge-specific structures - concretely, relation structures, for the document understanding task of joint information extraction. Further research in this area has the potential to unlock new avenues for improving the performance of models in more related tasks.

As a recap, the proposed structural inference leverages a constructed relation graph to fuse multi-facet information of document entities, including their coreference and relation information. One of the key contributions of this work is the formulation of explicit task interactions that utilize unique task characteristics to mitigate negative effects such as error propagation from pipeline decoding. It is showed that this approach brings significant improvements over previous work that employs shallow decoding interactions such as shared representation or graph propagation.

This work also presents the implication for future research in document understanding, such that through task-specific inference design, structural information that incorporates multi-facet task perspectives could be further leveraged towards more

related task setting in the multi-task learning manner, such as event extraction. I do hope more future research could delve into the promising paradigm of this work.

Chapter 6

Implicit Structural Inference through Sequence Generation

6.1 Introduction

This chapter continues on the same document understanding task objective as Chapter 5: joint entity-centric information extraction, but focus on a different methodology that is intrinsically capable of multi-task learning and global inference: autoregressive sequence generation, where the output of each step is conditioned on the entire sequence generated thus far. In contrast to Chapter 5 that necessitated multiple decoders for different tasks, generation-based approach employs a unified encoder-decoder framework without requiring task-specific architecture design, fitting multiple tasks into a single generation process. Each decision in the generation process can be viewed as a higher-order inference step on the input and previous decisions together.

The chapter begins with a discussion of prior works that explored different joint extraction paradigms, including early works on generation-based approaches for various IE tasks. My approach is subsequently introduced that models entity-centric sequence generation with a schema designed to enable implicit inference on the entity

structure throughout the autoregressive generation process, which has not been well explored in previous research.

6.1.1 Background: Joint Extraction Paradigms

The task of joint information extraction requires a model to address multiple tasks simultaneously. For example, end-to-end relation extraction involves both mention extraction and relation classification. Previous research has extensively studied joint extraction paradigms, which can be categorized into two directions.

The first direction is to employ multiple task decoders in the model and to perform multi-task learning. Within this direction, different perspectives can be addressed to improve the multi-task learning process. Many previous work has focused on entity dependencies and interactions under multiple tasks [17, 34]; label dependencies are also explored by [45, 46]. Apart from entity interactions, Yan et al. [78] investigates the two-way interactions between tasks within the encoder to obtain better feature representation. Xu and Choi [68] specifically models the interactions between decoders directly, as in Chapter 5. Nevertheless, multiple decoders for each task need to be in place and some level of "interaction" is focused.

The second direction is to use one module for decoding multiple tasks. Within this direction, Zheng et al. [89] proposes a novel tagging scheme to jointly decode entity and relation extraction. Miwa and Sasaki [40], Wang et al. [64], Shang et al. [56] adopt two-dimensional tagging (a.k.a table filling) that can further handle overlapping entities. Recently, generation-based approaches have attracted attention, and Zeng et al. [86, 85], Nayak and Ng [43] have investigated lexical generation or copy-based generation on sentence-level extraction tasks, where the structured output is linearized to a sequence for generation based on designed templates. Lu et al. [36] further incorporates instructions as part of the input that achieves dynamic templates and enables zero-shot extraction. For document-level extraction tasks, previous work

takes the same way as sentence-level tasks: Paolini et al. [48] uses augmented natural language as target output for coreference resolution, Huguet Cabot and Navigli [22], Giorgi et al. [18] design output templates that can fit multiple relation triple instances in the documents. However, these approaches merely regard a document as a long sentence, without utilizing its intrinsic structures.

In this research work, sequence generation is the focus, combined with implicit inference on the designed entity structure in the generation process. Section 6.1.2 describes the main motivation of this research, aiming to contribute to the development of joint information extraction models, by exploring the potentials and effectiveness of sequence generation.

6.1.2 Sequence Generation

Auto-regressive sequence generation has been successfully applied in various natural language processing tasks, such as language modeling, machine translation, and text summarization. In these tasks, the model generates a sequence of tokens or words, one at a time, based on the previously generated tokens and the input context.

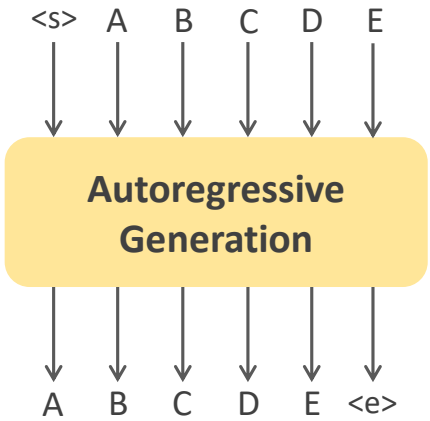

Figure 6.1: Autoregressive generation: the inference of each step is conditioned on the entire previously generated sequence.

One of the advantages behind using auto-regressive sequence generation for joint entity-centric information extraction is that it provides a natural framework for multi-task learning. Unlike other joint extraction paradigms discussed in Section 6.1.1, such as multi-task learning with shared representations or pipeline-based approaches, where different tasks are separately modeled and combined in a later stage, generation-based approaches can handle multiple tasks in a unified way through designed schema, and does not require task-specific architectures, making it more flexible and easier to implement compared to other joint extraction paradigms.

Another advantage is based on the property that each generation step conditions on the entire previously generated sequence, featuring higher-order inference based on previous decisions. In the context of joint information extraction, the input context consists of a sentence or a multi-sentence document, and the output sequence consists of entities and their corresponding attributes and relations. In previous related work introduced in Section 6.1.1, the model typically generates each entity mention and its attributes and relations in a designed sequential order, conditioned on the previously generated entities and their attributes and relations, as well as the input text.

For document-level extraction, the second advantage is especially appealing, as it enables implicit inference on a entity structure of the document, through the schema design, which is introduced in Section 6.1.3. It allows the model to incorporate global information about the entity structure throughout the generation process, rather than making local decisions for each entity in isolation. This is particularly useful for tasks such as document-level information extraction, where the entities and their relations could form a complex structure based on the entire document context.

Overall, auto-regressive sequence generation provides a powerful and flexible framework for joint entity-centric information extraction, and can handle multiple tasks in a unified way while allowing for global inference on the entity structure.

6.1.3 Problem Formulation

Although generation-based approaches have been proven effective for primarily sentence-level information extraction by previous work, however, two significant challenges still remain for document-level information extraction.

Firstly, document-level extraction differs from sentence-level extraction in an important aspect: the extracted output can have an entity-level structure, as mentions referring to the same entity should participate in the same roles or relations. Therefore, it is imperative to model the interactions in an entity-centric way, distinguished from the mention-level interactions in sentence-level extraction. For example, the popular document-level relation extraction dataset DocRED [80] requires the model to resolve relations between a pair of entities, rather than a pair of mentions.

However, none of the previous generation-based models in document-level extraction have explicitly formulated the entity-centric interactions throughout the generation process, as they all regard coreference resolution as a special mention-level relation that gets resolved along with other relations.

Secondly, as the document input can be much longer than a sentence, the extracted output can also have a long sequence, which can lead to instability and error propagation in generation.

In this research, instead of generation words or tokens, I adopt a pointer-based mechanism, generating a concise sequence that avoids longer output. Specifically, at each step, the model infers a pointer instead of a token, referring to either an input position, or a special position of a label vocabulary. Thus, only two steps are needed to indicate an entity mention (start and end positions), regardless of its length in the input context. In addition, to leverage the entity structure for the entity-centric task objective, a schema is designed to resolve document entities first, then to resolve entity-level interactions (relations), based on the generated entity structures.

6.2 Approach

The proposed generation-based approach can be broken down into three distinct parts: output schema, pointer-based inference, and training strategy. Each of these parts is described in detail in the following three subsections, respectively.

6.2.1 Generation Schema

Schema: entities The generation schema first resolves global entities in the document, modeling the entity-structure directly. Therefore, the model effectively performs an end-to-end entity resolution before reasoning any entity-level interactions such as entity relations. Instead of regarding coreference as a special mention-level relation as in previous works [22, 18], I adopt a simple strategy that models the entity structure directly and is also straightforward to view, visually as follows:

$$\underbrace{\mathrm{m}_{11}\ \mathrm{m}_{12}\ ...\ \mathrm{m}_{1N_1}\textcircled{1}}_{\text{1st entity}}\ \underbrace{\mathrm{m}_{21}\ \mathrm{m}_{22}\ ...\ \mathrm{m}_{2N_2}\textcircled{2}}_{\text{2nd entity}}\ \underbrace{...\textcircled{3}}_{\text{3rd entity}}\ \ ... \tag{6.1}$$

m_{ij} represents the jth mention of ith entity, and N_i is the total number of mentions in ith entity cluster. ⓘ is a special symbol, serving as a separator to mark the end of an entity cluster, as well as the identifier for the ith entity. This strategy is more efficient to represent coreferent entities of a document in terms of length, compared to generating coreference as mention-level relations [22, 18], as each mention only appears once in the schema, rather than multiple times as in previous works. Additionally, it is also interpretable that directly presents the entity structures of the document.

This schema is also flexible and efficient to incorporate entity types (e.g. *person*, *location*, etc.), by simply adding a special symbol $l^{(i)}$ as the type of ith entity to the

end of its entity sequence, illustrated as follows:

$$\underbrace{\mathrm{m}_{11}\ \mathrm{m}_{12}\ ...\ \mathrm{m}_{1N_1}\ l^{(1)}①}_{\text{1st entity}}\ \underbrace{\mathrm{m}_{21}\ \mathrm{m}_{22}\ ...\ \mathrm{m}_{2N_2}\ l^{(2)}②}_{\text{2nd entity}}\ \underbrace{...\ l^{(3)}③}_{\text{3rd entity}}\ ... \tag{6.2}$$

Schema: entity interactions After resolving all document entities in the output sequence, the schema then starts to resolve the entity interactions. In this work, only the entity relations are focused, but more interactions such as events could also be integrated in future work.

To model the entity interactions, the model continues the generation with the following schema, where the inference of each step is now conditioned on the previously generated entity structure:

$$\underbrace{\mathrm{m}_{11}\ \mathrm{m}_{12}\ ...①\ \ \mathrm{m}_{21}\ \mathrm{m}_{22}\ ...②\ \ ...}_{\text{entity structure}}\ \underbrace{≀①② r_1^{(1,2)}, r_2^{(1,2)}, ...\ \ ≀③⑤ r_1^{(3,5)}...}_{\text{entity interactions}} \tag{6.3}$$

≀ is a special symbol indicating the start of an interaction, followed by a pair of entities ⓘ ⓙ, and their expressed relations $r^{(i,j)}$. Note that:

1. The special symbol ⓘ is the identifier of the ith entity from the entity structure. It is used to represent the entity in the interaction sequence, such that specific mentions do not need to appear again in the schema, which is efficient for document-level extraction where the number of mentions can be quite large.

2. Each interaction expresses the relations between a pair of entities, where the number of relations can be one to many. Each step is conditioned on the generated entity sequence and previous interactions, implicitly performing structural inference that leverages the entity structure.

3. If a pair of entities do not express any relations, they are excluded from the schema, and should not appear in the interaction sequence.

Schema 6.3 can be used to model the full entity-centric extraction, or only the entities, effectively as the end-to-end coreference resolution.

6.2.2 Pointer-based Inference

Given the introduced schema, a pointer-based inference is used for the generation process, to efficiently represent which mentions are being referred to by the schema. It is inspired by previous work from Yan et al. [77], where the pointer-based generation to is proven effective for various tasks of Named Entity Recognition (NER).

Concretely, the logit space at each generation step is pointers, or positions, covering the input context and a special symbol vocabulary. The logit space can be denoted as $[S \oplus I]$, where $S = \{①, ②, ...\} \cup \{l_1, l_2, ...\} \cup \{r_1, r_2, ...\} \cup \{\wr\}$ is the vocabulary of all **S**pecial symbols including entity identifiers, types and relations, and $I = \{t_0, t_1, ...\}$ represents the exact tokens of **I**nput context. If there are N_S special symbols and N_I input tokens, then the output logit space constitutes $N_S + N_I$ pointers. At each generation step, the model predicts a distribution on this logit space, and each index of this space represents either a specific special symbol, or a specific input position.

To represent a mention, the model only needs two steps to generate its start token position and end token position from the input I. Therefore, the sequence directly generated by the model can be illustrated as:

$$\underbrace{\mathrm{s}_{11}\mathrm{e}_{11}\ \mathrm{s}_{12}\mathrm{e}_{12}\ ...①\ \ \mathrm{s}_{21}\mathrm{e}_{21}\ ...②\ \ ...}_{\text{entity structure}}\ \underbrace{\wr①②\ r_1^{(1,2)}, r_2^{(1,2)}, ...\ \ \wr\ ...}_{\text{entity interactions}} \tag{6.4}$$

$\mathrm{s}_{ij}/\mathrm{e}_{ij}$ represents the logit space index for the start/end position of the mention m_{ij}. Other special symbols such as ① and $r_1^{(1,2)}$ are also the corresponding indices of the logit space.

6.2.3 Decoder

In this research, BART [32] is used as the encoder-decoder framework. Let $\mathbf{H} \in \mathbb{R}^{N_I \times d}$ be the hidden states of input sequence of length N_I from the last layer of encoder, d being the hidden state dimension. At each generation step t, denote the previously generated pointer output as $Y_{t-1} = (y_1, ..., y_{t-1})$; the hidden state of this step from the decoder h_t is conditioned on the input and previous output:

$$h_t = \text{Decoder}(\mathbf{H}; Y_{t-1}) \tag{6.5}$$

With h_t, the model predicts a distribution over the pointer-based logit space, and picks an index, representing either a special symbol or an input position, based on the logit space probability distribution. Let $\mathbf{G} \in \mathbb{R}^{N_S \times d}$ be the learnable embeddings of N_S special symbols. The distribution over the pointer-based logit space is then:

$$\hat{\mathbf{H}} = \text{MLP}(\mathbf{H}) \tag{6.6}$$

$$P(y_t) = \text{softmax}\big((\mathbf{G} \cdot h_t) \oplus (\hat{\mathbf{H}} \cdot h_t)\big) \tag{6.7}$$

A learnable MLP (multi layer perceptron) is employed to transform raw input hidden states $\mathbf{H}$ for decoding inference, and $\oplus$ denotes the concatenation, so that the distribution is of dimension size $N_S + N_I$.

Note that although the decoder generates a pointer at each step, when obtaining h_t in Eq (6.5), the decoder takes the previous output Y_{t-1} and converts to their corresponding embeddings of the special symbols or input tokens.

During inference, the standard beam-search is adopted, and constrained decoding is applied to generate sequences of valid format, according to the schema.

6.2.4 Training Strategy

During training, the teaching-forcing method is applied, and the model is optimized by the standard cross-entropy loss:

$$\mathcal{L} = -\sum_{t=1}^{T} \log P(\hat{y}_t | \mathbf{H}, Y_{t-1}) \tag{6.8}$$

T is the gold output length, and $\hat{y}_t$ is the gold index position at step t, which points to either a special symbol or an input position.

Although the model is able to learn through the naive training strategy, under the context of this task and schema, two particular issues stand out. Further strategies to accommodate these issues are proposed as below.

Inductive bias In Schema 6.3, entities in the interaction sequence are referred by the ordinal special symbols ①, ②, etc. This could pose an inductive bias that certain ordinal symbols are biased against certain relations, especially when the train data is not sufficient. For example, if gold training label sequences contain many instances of ①④ r', then during inference, the model could predict towards the relation r' for any entities at the position 1 and 4, regardless what semantic meaning these entities actually express.

To alleviate this issue, in the training, the ordinal symbols are not fixed for each entity; rather, for a small probability, these special symbols are dynamically shuffled, as long as the same symbol is used for a entity in an training sequence. For instance, the ordinal symbols ① and ② are now shuffled with ⑦ and ⑤.

$$\underbrace{\mathrm{m}_{11}\ \mathrm{m}_{12}\ ...⑦\ \ \mathrm{m}_{21}\ \mathrm{m}_{22}\ ...⑤\ \ ...}_{\text{entity structure}}\ \ \underbrace{\wr ⑦③\ r_1^{(7,3)}, r_2^{(7,3)}, ...\ \ \wr\ ...}_{\text{entity interactions}} \tag{6.9}$$

The strategy of dynamic shuffling effectively eliminates the inductive bias of ordi-

nal symbols, and is shown to bring significant improvement according to the ablation study.

False tolerance Since teaching-forcing is used for training, there is a gap between training and inference, as the model has not been trained to handle inference with potentially erroneous sequences. Previous work UIE [36] employs noisy token injection that learns to correct wrong predictions during generation. Injection is performed to the interaction sequence based on negative sampling: if a pair of entities do not have any relations, per a small probability, an interaction with "NA" relation is injected into the gold sequence, forcing the model to recognize entity pairs without relations, which promotes false tolerance during inference, when a pair of entities are generated but no relations are actually expressed. When the "NA" relation is generated during inference, this particular interaction is simply discarded.

6.3 Evaluation and Analysis

The described sequence generation approach is evaluated on three datasets:

- CoNLL 2012 shared task [49], to specifically evaluate the capability to identify document entities. This is the same evaluation dataset used in Chapter 4.
- DocRED [80] and DWIE [83], to further evaluate the end-to-end modeling of joint relation extraction upon the entity structure. These are the two datasets adopted in Chapter 5.

For all experimental settings, the same BART-Large [32] is employed as the encoder-decoder backbone. A beam size of 2 is used for all inference.

Evaluation results Table 6.1 shows the results on CoNLL 2012 shared task for coreference resolution, comparing with the non-generation counterpart - two mod-

els from Chapter 4. In Table 6.2, evaluation results on the two relation extraction datasets are presented, which are compared against the non-generation counterpart introduced in Chapter 5, keeping the same evaluation protocol.

	MUC			B^3			$CEAF_{\phi_4}$			
	P	R	F1	P	R	F1	P	R	F1	Avg. F1
`BERT`	85.0	82.5	83.8	77.3	74.0	75.6	74.9	70.7	72.8	77.4
`SpanBERT`	**85.7**	**85.3**	**85.5**	**78.6**	**78.6**	**78.6**	**76.8**	**74.8**	**75.8**	**79.9**
`Gen`	81.3	84.0	82.6	72.1	76.4	74.2	71.8	72.8	72.3	76.4

Table 6.1: Results on the test set of CoNLL 2012 English shared task. The averaged F1 of MUC, B^3, $CEAF_{\phi_4}$ is the main evaluation metric. `BERT` and `SpanBERT` are two settings from the span-based model in Chapter 4; `Gen` is the sequence generation model described in this chapter.

	DocRED				DWIE		
	ME	COREF	RE	RE Ign	ME	COREF	RE
`Pipeline`	92.56	84.09	38.29	35.88	96.09	92.80	57.76
`Joint-M`	93.33	84.83	**39.65**	**37.17**	**96.47**	**92.91**	**61.01**
`Gen`	**93.98**	**85.44**	37.08	34.72	96.13	92.58	56.30

Table 6.2: Evaluation results on the test set of DocRED and DWIE. Three metrics are included: (1) Mention Extraction (ME) in mention-level F1 score (2) Coreference Resolution (COREF) in averaged F1 score of MUC, B^3, and $CEAF_{\phi_4}$ (3) Relation Extraction (RE) in entity-level F1 score. DocRED also provides a F1 score (RE Ign) that excludes shared relational facts between training and evaluation. `Gen` is the sequence generation approach in this chapter, while `Pipeline` and `Joint-M` are the two approaches presented in Chapter 5.

For coreference resolution, the results are divergent on different datasets. Table 6.2 shows that for DocRED, the sequence generation model `Gen` attains the best performance on COREF; for DWIE, `Gen` also shows trivial degradation (0.3 F1) than the non-generation counterpart. For CoNLL-2012, the gap indeed becomes larger, being 1% lower than `BERT` and 3.5% lower than `SpanBERT`. This outcome can be attributed to the dataset characteristics: CoNLL-2012 requires a higher level of reasoning, as it

involves annotated entities that comprise not only proper nouns, but also pronouns and numerous long noun phrases. On the other hand, DocRED and DWIE primarily comprise proper nouns, thus making the task more manageable. Nevertheless, it is still encouraging that without complex decoder design and feature engineering, the sequence generation is able to effectively resolve document entities with almost no degradation on easier datasets, and without significant declining on the harder dataset.

With respect to end-to-end relation extraction, current sequence generation models exhibit larger room for improvement, with `Gen` yielding an F1 score that is 2.6% lower than its non-generation counterpart for DocRED and 4.7% for DWIE. Especially, since `Gen` displays comparable performance in entity recognition (COREF) to previous models, it can be concluded that the current sequence generation technique needs stronger inference on the entity interactions, conditioned on the extracted entity structure.

Interaction Reasoning To better understand the performance of implicit inference on the linearized entity structure, the entity interactions are bifurcated from the end-to-end evaluation: during sequence generation model inference, the gold sequence of entity structures is fed to the decoder, such that the model performs inference of entity interactions using always the correct document entities. The evaluation results are compared with a state-of-the-art relation extraction model from 2021 that takes gold entities as input: `ATLOP` [90], which employs RoBERTa-Large [35] as the encoder and is not generation-based.

Table 6.3 illustrates the performance outcomes on the dev set of DocRED and DWIE. By using the gold sequence of entity structure, `Gen (GOLD)` achieves 52.73 on DocRED and 70.60 F1 on DWIE. Compared with `ATLOP (GOLD)`, there is indeed

	DocRED		DWIE
	RE	RE Ign	RE
`Gen (E2E)`	37.08	34.72	56.30
`Gen (GOLD)`	52.73	50.22	70.60
`ATLOP (GOLD)`	**62.03**	**60.21**	**82.87**

Table 6.3: Evaluation on the entity interactions (relation extraction) by regarding entities as given, using `Gen (GOLD)` and the non-generation counterpart `ATLOP (GOLD)`. For comparison, `Gen (E2E)` denotes the end-to-end results from Table 6.2.

a large performance gap of 9.3 F1 for DocRED and 12 F1 for DWIE. It shows that the reasoning ability on entity interactions falls behind on entity resolution, due to the requisite for entity interactions to refer to each entity via a unique ordinal symbol. Accordingly, the subsequent research efforts could concentrate on enhancing the performance of entity interactions.

Context Length As error propagation could happen during the sequence generation process, especially for long documents with many entities, the performance by different context length is explicitly examined, as shown in Figure 6.2 and Figure 6.3.

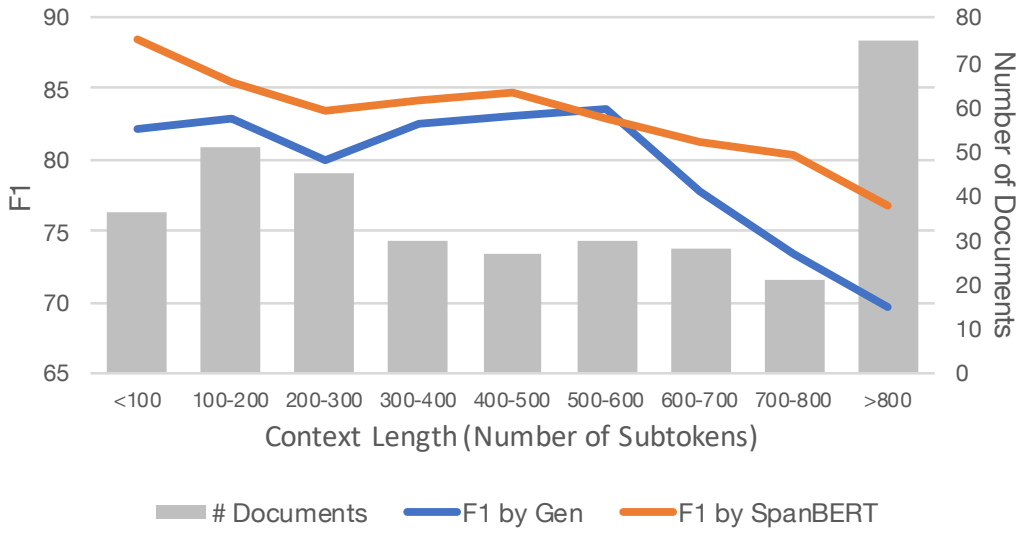

Figure 6.2: Entity resolution performance on the dev set of CoNLL 2012 shared task by different context length, for the approach of both `Gen` and `SpanBERT`. The distribution of context length is also shown.

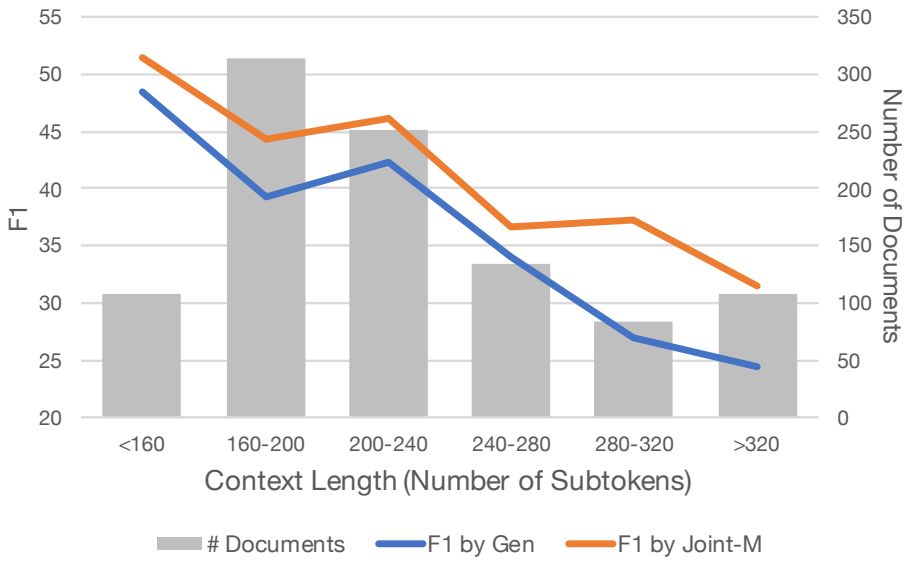

Figure 6.3: End-to-end relation extraction performance on the dev set of DocRED by different context length, for the approach of both `Gen` and `Joint-M`. The distribution of context length is also shown.

The assessment of entity resolution on CoNLL-2012 is presented in Figure 6.2, wherein `Gen` and `SpanBERT` sustain acceptable performance until the input context exceeds 600 subtokens, after which their performance declines rapidly. Meanwhile, `Gen` exhibits a greater performance gap compared to `SpanBERT` as the input context becomes lengthier.

Figure 6.3 illustrates the evaluation outcomes for end-to-end relation extraction on DocRED. A similar trend emerges, as both `Gen` and `Joint-M` demonstrate significant performance deterioration when the input context exceeds 240 subtokens.

The above observations for both datasets indicate that neither the conventional model nor the current sequence generation approach can handle long context while maintaining high performance. This emphasizes the need for a future research direction that seeks to mitigate error propagation when the document context gets long.

Ablation Study Further analysis is conducted to examine the effectiveness of proposed training strategies to handle inductive bias and false tolerance explicitly. Ta-

ble 6.4 shows the evaluation results with two additional settings, where both strategies are shown effective. Notably, the handling of false tolerance contributes 0.9 F1 to DocRED.

	DocRED		DWIE
	RE	RE Ign	RE
Gen	**37.08**	**34.72**	**56.30**
-IB	36.91	34.56	55.43
-FT	36.18	33.80	55.34

Table 6.4: Ablation study on the training strategies described in Section 6.2. `-IB` denotes the setting without handling the Inductive Bias issue; `-FT` denotes the setting without handling False Tolerance.

6.4 Discussion

This chapter conducts an exploratory study by using sequence generation for the task of entity-centric relation extraction, with the motivation of a unified architecture design with intrinsic higher-order inference on the entity structure. Though the current approach needs further research work to serve as a viable alternative to those traditional span-based models, it has already demonstrates promising results on entity resolution, exhibiting superior or comparable performance on simpler datasets like DocRED. These results underscore the efficacy of the proposed pointer-based inference and the designed schema.

Two research directions can be further conducted based on the presented analysis. First, the current generation shows a large performance gap in reasoning entity interactions. As the schema uses special ordinal symbols to refer to the entities, this issue could be potentially addressed by designing pretraining on a larger-scale corpus to enhance the model's capacity to learn this referral mechanism. Second, although false

tolerance is handled in the current approach, additional methods could be employed to mitigate the effects of error propagation, particularly when handling long context inputs.

Chapter 7

Conclusion

This dissertation presents four distinct research works that center around the overarching theme of performing various structural inference to enhance document understanding. This concluding chapter serves to provide a summary of the contributions made by the preceding chapters, and to reiterate the key objective - exploring the potentials of incorporating structures that could offer complementary information to sequence modeling. Additionally, a brief discussion regarding potential avenues for future work is provided in the end.

7.1 Research Contributions

In this dissertation, I have presented the utilization of different language or document structures for document understanding, and demonstrated how these structure can be formulated to improve the performance for various NLP tasks, through designed structural inference outlined in Chapter 3-6. Ultimately, they are leveraged to fulfill the main motivation described in Chapter 1, which is to overcome the inherent scattered information issue in the document encoding and task decoding process. By incorporating certain structures, either syntactic, discourse, or other knowledge-specific structures, they can be shown to augment the sequence modeling capabil-

ity of pretrained language models, leading to improved performance on a range of document-oriented tasks.

Overall, this dissertation makes insightful contributions to the research community on the incorporation of various structures under the development of deep learning in NLP. Experimental results by my proposed structural inference demonstrate that it can effectively enhance document understanding tasks, and benefit from modeling dependencies among different parts of the context.

7.2 Future Work

Task-specific Structures As there are more task-specific structures that could be utilized in more document understanding tasks, it would be interesting to investigate the potentials of structures that are not covered in this dissertation, such as event and temporal structures, and other discourse relations, which are all important aspects in general document understanding. These structures could initiate new insights into the relationships between different parts of a document, and potentially improve the performance, and bring improvement to certain downstream tasks.

General Structures Rather than task-specific structures, adopting a general task-agnostic structure to represent the document semantics is especially appealing. On the sentence-level, Abstract Meaning Representation (AMR) [2], a semantic representation framework proposed to capture the underlying meaning of a sentence in a structured and unambiguous way, has been largely utilized since its proposal. Recently, document-level AMR [42] has gained attraction as well. A future research that investigates how much these general semantic representation can assist the document understanding on a wide range of NLP tasks can also be particularly interesting.

Bibliography

[1] Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.421. URL `https://www.aclweb.org/anthology/2020.acl-main.421`.

[2] Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL `https://aclanthology.org/W13-2322`.

[3] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL `https://proceedings.neurips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf`.

[4] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan,

Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020. URL `https://arxiv.org/abs/2005.14165`.

[5] Deng Cai and Wai Lam. Graph transformer for graph-to-sequence learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7464–7471, apr 2020. doi: 10.1609/aaai.v34i05.6243. URL `https://ojs.aaai.org/index.php/AAAI/article/view/6243`.

[6] Kehai Chen, Rui Wang, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. Syntax-directed attention for neural machine translation. In *AAAI Conference on Artificial Intelligence*, 2018. URL `https://aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16060`.

[7] Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8: 454–470, 2020. doi: 10.1162/tacl_a_00317. URL `https://www.aclweb.org/anthology/2020.tacl-1.30`.

[8] Kevin Clark and Christopher D. Manning. Entity-centric coreference resolution with model stacking. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1405–

1415, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-1136. URL https://www.aclweb.org/anthology/P15-1136.

[9] Kevin Clark and Christopher D. Manning. Improving coreference resolution by learning entity-level distributed representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 643–653, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1061. URL https://www.aclweb.org/anthology/P16-1061.

[10] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1xMH1BtvB.

[11] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.747. URL https://www.aclweb.org/anthology/2020.acl-main.747.

[12] Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. Cross-lingual machine reading comprehension. 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 1586–1595, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1169. URL https://www.aclweb.org/anthology/D19-1169.

[13] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1285. URL https://www.aclweb.org/anthology/P19-1285.

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423.

[15] Markus Eberts and Adrian Ulges. An end-to-end model for entity-level relation extraction using multi-instance learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3650–3660, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.319. URL https://aclanthology.org/2021.eacl-main.319.

[16] Hongliang Fei, Xu Li, Dingcheng Li, and Ping Li. End-to-end deep reinforcement learning based coreference resolution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 660–665, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1064. URL https://www.aclweb.org/anthology/P19-1064.

[17] Tsu-Jui Fu, Peng-Hsuan Li, and Wei-Yun Ma. GraphRel: Modeling text as relational graphs for joint entity and relation extraction. In *Proceedings of the*

57th Annual Meeting of the Association for Computational Linguistics, pages 1409–1418, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1136. URL https://aclanthology.org/P19-1136.

[18] John Giorgi, Gary Bader, and Bo Wang. A sequence-to-sequence approach for document-level relation extraction. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 10–25, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.bionlp-1.2. URL https://aclanthology.org/2022.bionlp-1.2.

[19] John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1419. URL https://aclanthology.org/N19-1419.

[20] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[21] Tsung-Yuan Hsu, Chi-Liang Liu, and Hung-yi Lee. Zero-shot reading comprehension by cross-lingual transfer learning with multi-lingual language representation model. 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 5933–5940, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1607. URL https://www.aclweb.org/anthology/D19-1607.

[22] Pere-Lluís Huguet Cabot and Roberto Navigli. REBEL: Relation extraction by end-to-end language generation. In *Findings of the Association for Com-*

putational Linguistics: EMNLP 2021, pages 2370–2381, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.204. URL https://aclanthology.org/2021.findings-emnlp.204.

[23] Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. BERT for coreference resolution: Baselines and analysis. 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 5803–5808, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1588. URL https://www.aclweb.org/anthology/D19-1588.

[24] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020. doi: 10.1162/tacl_a_00300. URL https://doi.org/10.1162/tacl_a_00300.

[25] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020. doi: 10.1162/tacl_a_00300. URL https://doi.org/10.1162/tacl_a_00300.

[26] Ben Kantor and Amir Globerson. Coreference resolution with entity equalization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 673–677, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1066. URL https://www.aclweb.org/anthology/P19-1066.

[27] Jungo Kasai, Dan Friedman, Robert Frank, Dragomir Radev, and Owen Rambow. Syntax-aware neural semantic role labeling with supertags. 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 701–709, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1075. URL `https://www.aclweb.org/anthology/N19-1075`.

[28] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning workshop*, 2015.

[29] Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1018. URL `https://www.aclweb.org/anthology/D17-1018`.

[30] Kenton Lee, Luheng He, and Luke Zettlemoyer. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2108. URL `https://www.aclweb.org/anthology/N18-2108`.

[31] Kyungjae Lee, Sunghyun Park, Hojae Han, Jinyoung Yeo, Seung-won Hwang, and Juho Lee. Learning with limited data for multilingual reading comprehension. 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 2840–2850, Hong Kong, China,

November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1283. URL https://www.aclweb.org/anthology/D19-1283.

[32] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 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, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL https://aclanthology.org/2020.acl-main.703.

[33] Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.653. URL https://www.aclweb.org/anthology/2020.acl-main.653.

[34] Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.713. URL https://aclanthology.org/2020.acl-main.713.

[35] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. URL http://arxiv.org/abs/1907.11692.

[36] Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun,

and Hua Wu. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.395. URL https://aclanthology.org/2022.acl-long.395.

[37] Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1360. URL https://aclanthology.org/D18-1360.

[38] Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. A general framework for information extraction using dynamic span graphs. 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 3036–3046, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1308. URL https://aclanthology.org/N19-1308.

[39] Taejin L. Min, Liyan Xu, Jinho D. Choi, Ranliang Hu, Jason W. Allen, Christopher Reeves, Derek Hsu, Richard Duszak, Jeffrey Switchenko, and Gelareh Sadigh. Covid-19 pandemic-associated changes in the acuity of brain mri findings: A secondary analysis of reports using natural language processing. *Current Problems in Diagnostic Radiology*, 51(4):529–533, 2022. ISSN 0363-0188. doi: https://doi.org/10.1067/j.cpradiol.2021.11.001. URL https://www.sciencedirect.com/science/article/pii/S0363018821001894.

[40] Makoto Miwa and Yutaka Sasaki. Modeling joint entity and relation extraction

with table representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1858–1869, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1200. URL https://aclanthology.org/D14-1200.

[41] Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. Reasoning with latent structure refinement for document-level relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1546–1557, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.141. URL https://aclanthology.org/2020.acl-main.141.

[42] Tahira Naseem, Austin Blodgett, Sadhana Kumaravel, Tim O'Gorman, Young-Suk Lee, Jeffrey Flanigan, Ramón Astudillo, Radu Florian, Salim Roukos, and Nathan Schneider. DocAMR: Multi-sentence AMR representation and evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3496–3505, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.256. URL https://aclanthology.org/2022.naacl-main.256.

[43] Tapas Nayak and Hwee Tou Ng. Effective modeling of encoder-decoder architecture for joint entity and relation extraction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8528–8535, Apr. 2020. doi: 10.1609/aaai.v34i05.6374. URL https://ojs.aaai.org/index.php/AAAI/article/view/6374.

[44] Vincent Ng and Claire Cardie. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 104–111, Philadelphia, Pennsylvania, USA,

July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073102. URL https://www.aclweb.org/anthology/P02-1014.

[45] Minh Van Nguyen, Viet Dac Lai, and Thien Huu Nguyen. Cross-task instance representation interactions and label dependencies for joint information extraction with graph convolutional networks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 27–38, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.3. URL https://aclanthology.org/2021.naacl-main.3.

[46] Minh Van Nguyen, Bonan Min, Franck Dernoncourt, and Thien Nguyen. Joint extraction of entities, relations, and events via modeling inter-instance and inter-label dependencies. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4363–4374, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.324. URL https://aclanthology.org/2022.naacl-main.324.

[47] Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL https://www.aclweb.org/anthology/L16-1262.

[48] Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto.

Structured prediction as translation between augmented natural languages. In *9th International Conference on Learning Representations, ICLR 2021*, 2021.

[49] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, pages 1–40, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W12-4501.

[50] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, pages 1–40, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W12-4501.

[51] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-demos.14. URL https://www.aclweb.org/anthology/2020.acl-demos.14.

[52] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

[53] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 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, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL https://www.aclweb.org/anthology/D16-1264.

[54] Dongyu Ru, Changzhi Sun, Jiangtao Feng, Lin Qiu, Hao Zhou, Weinan Zhang, Yong Yu, and Lei Li. Learning logic rules for document-level relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1239–1250, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.95. URL https://aclanthology.org/2021.emnlp-main.95.

[55] Victor Sanh, Thomas Wolf, and Sebastian Ruder. A hierarchical multi-task approach for learning embeddings from semantic tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:6949–6956, Jul. 2019. doi: 10.1609/aaai.v33i01.33016949. URL https://ojs.aaai.org/index.php/AAAI/article/view/4673.

[56] Yu-Ming Shang, Heyan Huang, and Xianling Mao. Onerel: Joint entity and relation extraction with one module in one step. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11285–11293, Jun. 2022. doi: 10.1609/aaai.v36i10.21379. URL https://ojs.aaai.org/index.php/AAAI/article/view/21379.

[57] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 464–468, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2074. URL https://www.aclweb.org/anthology/N18-2074.

[58] Milan Straka. UDPipe 2.0 prototype at CoNLL 2018 UD shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207, Brussels, Belgium, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/K18-2020. URL https://www.aclweb.org/anthology/K18-2020.

[59] Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. Linguistically-informed self-attention for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5027–5038, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1548. URL https://www.aclweb.org/anthology/D18-1548.

[60] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[61] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

[62] Severine Verlinden, Klim Zaporojets, Johannes Deleu, Thomas Demeester, and Chris Develder. Injecting knowledge base information into end-to-end joint entity and relation extraction and coreference resolution. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1952–1957, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.171. URL https://aclanthology.org/2021.findings-acl.171.

[63] David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1585. URL https://aclanthology.org/D19-1585.

[64] Yijun Wang, Changzhi Sun, Yuanbin Wu, Hao Zhou, Lei Li, and Junchi Yan. UniRE: A unified label space for entity relation extraction. 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 220–231, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.19. URL https://aclanthology.org/2021.acl-long.19.

[65] Sam Wiseman, Alexander M. Rush, Stuart Shieber, and Jason Weston. Learning anaphoricity and antecedent ranking features for coreference resolution. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1416–1426, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-1137. URL https://www.aclweb.org/anthology/P15-1137.

[66] Sam Wiseman, Alexander M. Rush, and Stuart M. Shieber. Learning global features for coreference resolution. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 994–1004, San Diego, California, June

2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1114. URL https://www.aclweb.org/anthology/N16-1114.

[67] Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. 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, May 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/17665.

[68] Liyan Xu and Jinho Choi. Modeling task interactions in document-level joint entity and relation extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5409–5416, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.395. URL https://aclanthology.org/2022.naacl-main.395.

[69] Liyan Xu and Jinho D. Choi. Revealing the myth of higher-order inference in coreference resolution. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8527–8533, Online, November 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.emnlp-main.686.

[70] Liyan Xu and Jinho D. Choi. Adapted end-to-end coreference resolution system for anaphoric identities in dialogues. In *Proceedings of the CODI-CRAC 2021 Shared Task on Anaphora, Bridging, and Discourse Deixis in Dialogue*, pages 55–62, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.codi-sharedtask.6. URL https://aclanthology.org/2021.codi-sharedtask.6.

[71] Liyan Xu and Jinho D. Choi. Online coreference resolution for dialogue pro-

cessing: Improving mention-linking on real-time conversations. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pages 341–347, Seattle, Washington, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.starsem-1.30. URL https://aclanthology.org/2022.starsem-1.30.

[72] Liyan Xu, Julien Hogan, Rachel E. Patzer, and Jinho D. Choi. Noise pollution in hospital readmission prediction: Long document classification with reinforcement learning. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 95–104, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.bionlp-1.10. URL https://aclanthology.org/2020.bionlp-1.10.

[73] Liyan Xu, Xuchao Zhang, Xujiang Zhao, Haifeng Chen, Feng Chen, and Jinho D. Choi. Boosting cross-lingual transfer via self-learning with uncertainty estimation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6716–6723, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.538. URL https://aclanthology.org/2021.emnlp-main.538.

[74] Liyan Xu, Xuchao Zhang, Bo Zong, Yanchi Liu, Wei Cheng, Jingchao Ni, Haifeng Chen, Liang Zhao, and Jinho D. Choi. Zero-shot cross-lingual machine reading comprehension via inter-sentence dependency graph. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):11538–11546, Jun. 2022. doi: 10.1609/aaai.v36i10.21407. URL https://ojs.aaai.org/index.php/AAAI/article/view/21407.

[75] Liyan Xu, Chenwei Zhang, jingbo Shang, Xian Li, and Jinho Choi. Towards open-world product attribute mining: A lightly-supervised approach. *Submission*

to The 61st Annual Meeting of the Association for Computational Linguistics, 2023.

[76] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2021.

[77] Hang Yan, Tao Gui, Junqi Dai, Qipeng Guo, Zheng Zhang, and Xipeng Qiu. A unified generative framework for various NER subtasks. 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 5808–5822, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.451. URL https://aclanthology.org/2021.acl-long.451.

[78] Zhiheng Yan, Chong Zhang, Jinlan Fu, Qi Zhang, and Zhongyu Wei. A partition filter network for joint entity and relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 185–197, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.17. URL https://aclanthology.org/2021.emnlp-main.17.

[79] Shaowei Yao, Tianming Wang, and Xiaojun Wan. Heterogeneous graph transformer for graph-to-sequence learning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7145–7154, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.640. URL https://www.aclweb.org/anthology/2020.acl-main.640.

[80] Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. DocRED: A large-scale document-level relation extraction dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1074. URL https://aclanthology.org/P19-1074.

[81] Juntao Yu, Alexandra Uma, and Massimo Poesio. A cluster ranking model for full anaphora resolution. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 11–20, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://www.aclweb.org/anthology/2020.lrec-1.2.

[82] Fei Yuan, Linjun Shou, Xuanyu Bai, Ming Gong, Yaobo Liang, Nan Duan, Yan Fu, and Daxin Jiang. Enhancing answer boundary detection for multilingual machine reading comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 925–934, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.87. URL https://www.aclweb.org/anthology/2020.acl-main.87.

[83] Klim Zaporojets, Johannes Deleu, Chris Develder, and Thomas Demeester. DWIE: An entity-centric dataset for multi-task document-level information extraction. *Information Processing & Management*, 58(4):102563, 2021. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2021.102563. URL https://www.sciencedirect.com/science/article/pii/S0306457321000662.

[84] Klim Zaporojets, Johannes Deleu, Chris Develder, and Thomas Demeester. Dwie: An entity-centric dataset for multi-task document-level information extraction. *Information Processing & Management*, 58(4):102563, 2021.

[85] Daojian Zeng, Haoran Zhang, and Qianying Liu. Copymtl: Copy mechanism for joint extraction of entities and relations with multi-task learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9507–9514, Apr. 2020. doi: 10.1609/aaai.v34i05.6495. URL https://ojs.aaai.org/index.php/AAAI/article/view/6495.

[86] Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 506–514, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1047. URL https://aclanthology.org/P18-1047.

[87] Meishan Zhang, Zhenghua Li, Guohong Fu, and Min Zhang. Syntax-enhanced neural machine translation with syntax-aware word representations. 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 1151–1161, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1118. URL https://www.aclweb.org/anthology/N19-1118.

[88] Zhuosheng Zhang, Yuwei Wu, Junru Zhou, Sufeng Duan, Hai Zhao, and Rui Wang. Sg-net: Syntax-guided machine reading comprehension. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9636–9643, apr 2020. doi: 10.1609/aaai.v34i05.6511. URL https://ojs.aaai.org/index.php/AAAI/article/view/6511.

[89] Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of the 55th Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 1227–1236, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1113. URL https://aclanthology.org/P17-1113.

[90] Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 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, May 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/17717.

[91] Jie Zhu, Junhui Li, Muhua Zhu, Longhua Qian, Min Zhang, and Guodong Zhou. Modeling graph structure in transformer for better AMR-to-text generation. 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 5459–5468, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1548. URL https://www.aclweb.org/anthology/D19-1548.