大規模言語モデルによるシフト還元修辞構造解析の模倣

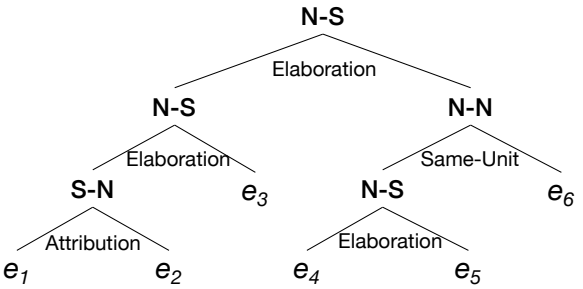
前川在 [1]　平尾努 [2]　上垣外英剛 [1]　奥村学 [1]
[1] 東京工業大学　[2]NTT コミュニケーション科学基礎研究所
{maekawa,kamigaito,oku}@lr.pi.titech.ac.jp　tsutomu.hirao@ntt.com

概要

デコーダのみからなる大規模言語モデル (LLM) の発展は目覚ましく，様々な自然言語処理タスクにおいて良好な結果を残している．一方，修辞構造解析におけるそれらの有効性はこれまで議論されていない．本稿では，今後の修辞構造解析の研究においてLLM を活用すべきかどうかを探ることを目的として，プロンプトを介してシフト還元動作を LLM で模倣する手法を提案し，その有効性を議論する．評価実験の結果，提案法は世界最高の解析性能を達成し，テキストドメインの汎化性においても優れていた．つまり，修辞構造解析においても LLM に注力すべきことが強調される結果を得た．

1　はじめに

事前学習済み言語モデルは，エンコーダのみのモデルである BERT [1]，DeBERTa [2]，エンコーダ・デコーダモデルである BART [3]，T5 [4]，デコーダのみのモデルである GPT-3 [5]，Llama 2 [6] など様々なモデルが提案されており，後者 2 つのモデルの学習データ量とパラメータ数の大規模化が顕著である．特にデコーダのみの事前学習済み言語モデルについては，数兆トークンのテキストデータを用い，数十，数百億のパラメータで学習されたモデル (以降，LLM) が公開され，言語生成を伴うタスクのみならず様々な自然言語処理タスクにおいてその有効性が実証されている．しかし，これまでのところ修辞構造解析という比較的困難なタスクにおける有効性を検証した研究はない．事前学習済み言語モデルの研究の焦点がほぼ LLM へとシフトした現状において，修辞構造解析への LLM 活用の模索は，今後の研究の方向性を考える上で興味深い．本稿では，従来から提案されているシフト還元法に基づく修辞構造解析を，プロンプトを介し LLM で模倣する手法[1)] を提案する．具体的にはシフト，還元という解析動作を，QLoRA [7] を用いて微調整を行った Llama 2 [6] で決定することで木構造を推定する．標準的ベンチマークデータセットである RST Discourse Treebank (RST-DT) [8] を用いた評価実験では，提案法が世界最高性能を達成し，GUM コーパス [9] を用いたテキストドメイン汎化性の検証においても提案法が優れていることを確認した．これらの結果は，修辞構造解析に LLM を利用する価値が十分あることを示しており，その活用を促進するものと考える．ただし，解析に要する時間が非常に長いという致命的な問題点も同時に浮き彫りとなった．



e_1: [Westinghouse Electric Corp. said], e_2: [it will buy Shaw-Walker Co.], e_3: [Terms weren't disclosed.], e_4: [Shaw-Walker,], e_5: [based in Muskegon, Mich.,], e_6: [makes metal files and desks, and seating and office systems furniture.].

図 1　修辞構造木の例 (RST-DT [8] の WSJ_1100 より)．

2　準備

2.1　修辞構造理論

修辞構造理論 (Rhetorical Structure Theory: RST) [10] によると，テキストは，葉が Elementary Discourse Unit (EDU) と呼ばれる節相当のユニット，中間ノードが 1 つ以上の EDU からなるテキストスパンの核性 (核: Nucleus，衛星: Satellite)，枝がテキストスパン間の修辞関係を表わす完全 2 分木[2)] として表現

1)　より単純には，系列変換タスクとして捉え，入力テキストを線形化した木へと変換する手法がある．しかし，文書を入力とすると LLM の入力トークン数の制限を超えることがしばしばあるのでこの方法は実際には利用できない．

2)　本来は，多分木であるが等価な完全 2 分木として変換可能なので，本稿では完全 2 分木として扱う．

される．図 1 に RST-DT から得た例を示す．図中，N-S と S-N (単核)，N-N (多核) は中間ノードが支配するテキストスパンの核性の組み合わせを示す．単核の場合には衛星から核へ向けて修辞関係が与えられ，多核の場合には等価な修辞関係が与えられる．例では，e_3 が e_1，e_2 からなるスパンを Elaboration という関係で結び，e_4，e_5 からなるスパンと e_6 が Same-Unit という等価な関係で結ばれていることを示す．修辞関係はデータセットによって異なり，RST-DT では 18 種が定義されている．

2.2 修辞構造解析の方法論

修辞構造木は構成素木であるので句構造解析で用いられる解析法をそのまま適用できることが多い．ただし，句構造解析と比較すると，木の葉の数が多いことから，比較的計算量の少ないアルゴリズムが好まれる．現在では，貪欲法によるトップダウン解析，シフト還元法によるボトムアップ解析が主流である [11, 12, 13, 14]．どちらのアプローチでも近年のニューラルネットワークに基づく解析法ではテキストスパンをベクトルに変換する特徴抽出層，特徴抽出層から得たベクトルに基づき解析動作を決定する解析動作決定層からなる．特徴抽出層には，エンコーダのみからなる事前学習済み言語モデルが用いられ，テキストスパンの両端のトークンの埋め込みベクトルの平均をそのスパンのベクトルとする場合が多い．トップダウン解析は，文書を EDU からなる 1 つの系列とみなし，これを再帰的に分割していくことで木を構築する．解析動作決定層は EDU 系列のどこで分割すべきかを FFN やポインターネットワークなどを用いて決定する [11, 12]．一方，ボトムアップ解析は，解析動作決定層がシフトか還元を決定し，EDU を結合しながら木を構築する [13, 14]．近年，Kobayashi ら [15] は，様々な事前学習済み言語モデルを用いて，トップダウン，ボトムアップ解析を比較した結果，解析法による違いよりも言語モデルによる違いが性能に大きく影響を及ぼすことを示し，DeBERTa を用いたボトムアップ解析が RST-DT における現在の世界最高性能であることを示した．

3 提案法

本研究では，前節で述べたエンコーダのみの事前学習済み言語モデルを利用したシフト還元法によるボトムアップ解析法を LLM により模倣する．

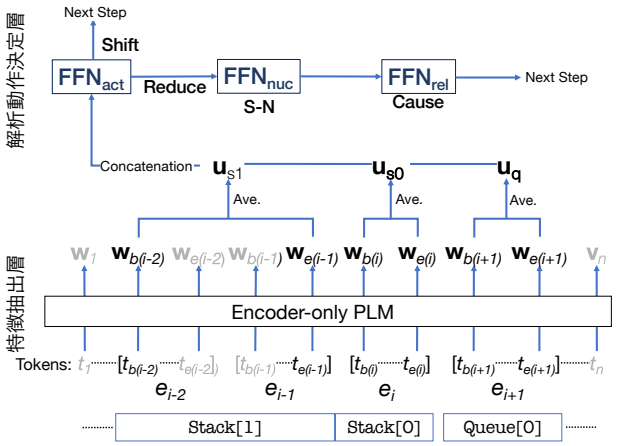


図 2 エンコーダのみの事前学習済み言語モデルを用いたシフト還元解析法．

3.1 従来のシフト還元解析法

解析済みの部分木[3] を格納するスタック，これから解析対象となる EDU を格納するキューを用いて，以下のシフト還元操作を繰り返し適用することで左から右に順に EDU 系列を読み込んで木を構築する (図 2 参照)．

シフト キューの先頭の EDU を取り出し，スタックに積む，

還元 スタックの 2 番目に格納された部分木を左の子，トップに格納された部分木を右の子とする木を構築し，再度スタックに積む．

なお，還元操作を行った後，左右の部分木の核性とそれらの間の修辞関係をそれぞれ異なる分類器を用いて決定する．核性ラベルは N-S，S-N，N-N の 3 種のいずれか，修辞関係ラベルはデータセットによって異なるが RST-DT の場合は 18 種である．解析動作，核性ラベル，修辞関係ラベルの決定は，以下の順伝播型ニューラルネットワーク $\mathrm{FFN}_{\mathrm{act}}$，$\mathrm{FFN}_{\mathrm{nuc}}$，$\mathrm{FFN}_{\mathrm{rel}}$ を用いて行う．

$$s_* = \mathrm{FFN}_*(\mathrm{Concat}(\mathbf{u}_{s_0}, \mathbf{u}_{s_1}, \mathbf{u}_{q_0}, \mathbf{u}_{\mathrm{org}})). \quad (1)$$

ここで，$\mathbf{u}_{s_0}$，$\mathbf{u}_{s_1}$ はスタックの上 2 つに格納される部分木に対応するテキストスパンのベクトル表現，$\mathbf{u}_{q_0}$ はキューの先頭に格納されている EDU のベクトル表現である．

3.2 LLM によるシフト還元解析法

スタックとキューは LLM 外部に用意しておき，その情報を LLM に与えることでシフト還元動作を

3) 部分木であるが，実際にはテキストスパンとして扱う．

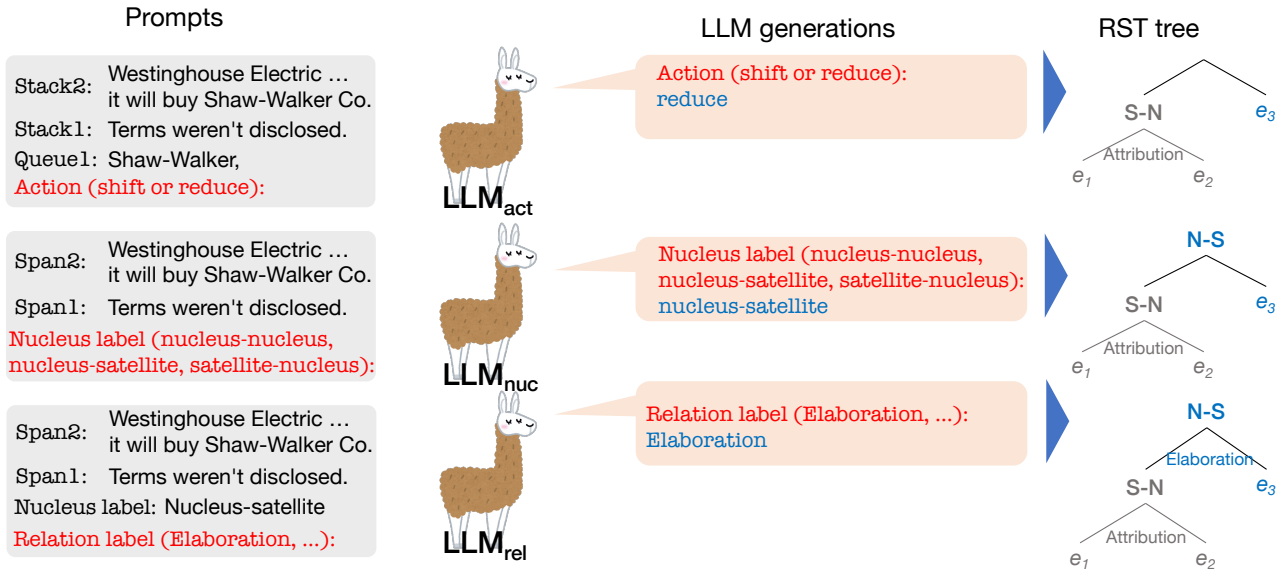


図 3 プロンプトと木の構築過程．スタックの 2 番目には e_1 と e_2 からなる部分木，トップには e_3，キューの先頭には e_4 が格納されている状態で LLM が還元を選択した場合．

決定する．そして，決定に基づきスタックとキューを更新する手続きを繰り返すことで木を構築する．LLM でシフト還元操作を行うためにはそれらを実行するためのプロンプトが必要となる．本稿では以下のプロンプトを用いる．

Stack2: スタックの 2 番目に格納された部分木に対応するテキストスパン

Stack1: スタックのトップに格納された部分木に対応するテキストスパン

Queue1: キューの先頭に格納された EDU

Action (shift or reduce): shift か reduce のいずれか．

なお，実行結果が還元であった場合には引き続き以下のプロンプトを用いて核性を決定する．

Span2: スタックの 2 番目に格納された部分木に対応するテキストスパン

Span1: スタックのトップに格納された部分木に対応するテキストスパン

Nucleus label (nucleus-nucleus, nucleus-satellite, satellite-nucleus): 3 種のいずれか

また，修辞関係も同様のプロンプトを用いて決定する．ただし，3 行目以降を以下のように変更し，前のステップで推定した核性ラベルも利用する．

Nucleus label: 推定した核性ラベル

Relation label ($\mathbf{rel}_1, \mathbf{rel}_2, \ldots, \mathbf{rel}_n$): n 個の関係のなかのいずれか

図 3 に実際のプロンプトを用いた解析例を示す．

4 実験設定

LLM としては Llama 2 [6][4] を利用した．パラメータサイズは 7B，13B，70B のすべてを用いて比較した．前節で説明したプロンプトは zero-shot あるいは few-shot の場合にも適用できるが，事前実験で試したところほとんどの場合で無効な出力により木が構築できなかった．そこで，プロンプトに正解を与え，微調整を行なうこととした．ただし，パラメータ数が膨大なため，GPU メモリ，学習時間の制約によりすべてのパラメータの微調整は行えない．よって，パラメータ行列の低ランク行列を用いたアダプタである LoRA [16] を量子化した QLoRA [7] を用いて微調整を行なった．

提案法の有効性を検証するため，修辞構造解析の標準的ベンチマークデータセットである RST-DT [8] に加えそれと同等規模のデータセットである GUM コーパス [9] を用いた．RST-DT は Wall Street Journal から得た 385 文書からなり，1 文書あたりの平均 EDU 数は約 57 である．修辞関係ラベルは 18 種が用いられる．RST-DT には公式の検証データセットがないため，Kobayashi ら [17] と同様に Heilman [18] の分割に従い，訓練/検証/テストをそれぞれ 338/40/38 とした．一方，GUM コーパスは，会話，インタビュー，ニュースなど多様な 12 のジャンルの 213 文書からなり，1 文書あたりの平均 EDU 数は

4) https://huggingface.co/meta-llama/Llama-2-{7,13,70}b-hf

		Span	Nuc	Rel	Full
RST-DT	Kobayashi et al.	77.8	68.0	57.3	55.4
	Llama 2 (7B)	78.2	67.5	57.6	55.8
	Llama 2 (13B)	78.3	68.1	57.8	56.0
	Llama 2 (70B)	**79.8**	**70.4**	**60.0**	**58.1**
GUM	Kobayashi et al.	73.4	60.9	50.3	48.5
	Llama 2 (7B)	74.4	63.0	53.4	52.1
	Llama 2 (13B)	74.8	63.4	54.0	52.8
	Llama 2 (70B)	**76.4**	**64.7**	**56.4**	**55.2**

表 1 RST-DT と GUM コーパスでの評価結果

124 と RST-DT よりも多い．訓練/検証/テストは公式の分割に従い，165/24/24 とした．GUM コーパスの修辞関係ラベルは RST-DT のそれとは異なるが，Liu ら [19] の変換ルールを用いて RST-DT と同様のものへと変換した．

また，評価指標には，Standard-Parseval [20] に基づき，ラベルなし (Span)，核性ラベル付き (Nuc)，関係ラベル付き (Rel)，全ラベル付き (Full) スパンの一致のマイクロ平均値で評価した．

5 結果と考察

表 1 に提案法と Kobayashi らの手法の評価結果を示す．提案法間を比較すると，パラメータ数が増えるにつれ性能が向上する．特に Rel と Full ではそれが顕著である．Kobayashi らの手法と比較すると，RST-DT の場合には 7B，13B は Full でやや良い程度であるが，GUM の場合には Nuc，Rel，Full で大きな差がある．70B は，Kobayashi らの手法に対しすべての指標において顕著な差で優れている．特に，GUM の Full では 7 ポイント近い差がある．Kobayashi らの手法と提案法の違いは特徴抽出層を経た解析動作決定層においてシフト還元動作を決定するか，LLM でそれら全体を代替して決定するかにあるが，実験結果より後者の有効性は明らかである．DeBERTa が 140M のパラメータであるのに対し，Llama 2 は 70B のパラメータであり約 500 倍程度の差がある．当然，パラメータ数が多いほうが良い結果が得られる可能性が高いので，もし 70B の DeBERTa が使えるのならば提案法より高い性能を示すかもしれない．しかしながら，そもそもそれが可能かは自明でないし，エンコーダのみの事前学習済み言語モデルの開発が現状ではほぼ止まっていることを考えれば，これらの実験結果は，我々が今後 LLM を用いた修辞構造解析に注力すべきであることを強く促す．ただし，提案法の学習にはハイエンド GPU が必要であり，微調整には数日を要する．さらに，1 文書の解析に数分を要する．これは，Kobayashi らの手法よりも多大なコストであり，提案法の実用上の大きな課題である．

	Span	Nuc	Rel	Full
Kobayashi et al.	72.0 (-5.2)	58.5 (-9.5)	46.3 (-11.0)	44.3 (-11.1)
Llama 2 (7B)	77.4 (-0.8)	63.6 (-3.9)	51.3 (-6.3)	49.0 (-6.8)
Llama 2 (13B)	77.4 (-0.9)	64.5 (-3.6)	52.2 (-5.6)	50.3 (-5.7)
Llama 2 (70B)	79.7 (-0.1)	66.5 (-3.9)	53.2 (-6.8)	51.1 (-7.0)

表 2 GUM コーパスで訓練し，RST-DT で評価した結果．カッコ内は表 1 との差分を示す．

LLM を用いることの利点をさらに探るため，コーパス横断の評価実験を行った．GUM コーパスはニュースを含む多様なジャンルのデータセットであるので，これで学習した解析器がジャンルとして GUM コーパスに包含される RST-DT でどの程度の性能かを調べた．結果を表 2 に示す．表より，Kobayashi らの手法は訓練データを変更することで性能が大きく劣化している．Span では 5 ポイント程度，Rel，Full では 10 ポイントを超える劣化である．一方，提案法は Span ではほぼ劣化せず，Nuc で 4 ポイント，Rel と Full で 6-7 ポイントの劣化にとどまっている．さらに，70B の場合には RST-DT で学習した Kobayashi らの手法に Rel，Full では及ばぬものの Span では勝り，Nuc ではやや劣る程度の性能である．これらより，LLM を用いることでテキストドメイン汎化性が向上していることがわかる．これは，LLM が大規模なテキストデータを用いて大規模なパラメータを事前学習したことが貢献していると考える．この結果も LLM を活用することの利点を示しており，今後は，LLM を活用した修辞構造解析を発展させていくべきであることを示す．

6 おわりに

本稿では，LLM を用いてシフト還元操作を模倣することによる修辞構造解析法を提案した．Llama 2 を採用し，RST-DT と GUM コーパスを用いて評価実験を行なった結果，提案法は，世界最高性能を達成し，テキストドメインの汎化性にも優れていることが明らかになった．これらの結果は，事前学習済み言語モデルの研究の焦点がデコーダのみの LLM へと移り変わる現状で，修辞構造解析においても今後 LLM を活用すべきであるという新しい知見をもたらした．一方，提案法は学習，推論にかかるコストが多大であるという実用上の課題が明確となった．

謝辞

本研究の一部は JSPS 科研費 JP21H03505 の助成を受けたものです.

参考文献

[1] 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)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[2] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced BERT with disentangled attention. In **Proceedings of the International Conference on Learning Representations**, 2021.

[3] 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**, pp. 7871–7880, Online, July 2020. Association for Computational Linguistics.

[4] 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**, Vol. 21, No. 140, pp. 1–67, 2020.

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris 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. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, **Advances in Neural Information Processing Systems**, Vol. 33, pp. 1877–1901. Curran Associates, Inc., 2020.

[6] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

[7] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. **CoRR**, Vol. abs/2305.14314, , 2023.

[8] Mary Ellen Okurowski Lynn Carison, Daniel Marcu. **RST Discourse Treebank**. Philadelphia: Linguistic Data Consortium, 2002.

[9] Amir Zeldes. The GUM corpus: Creating multilayer resources in the classroom. **Language Resources and Evaluation**, Vol. 51, No. 3, pp. 581–612, 2017.

[10] W.C. Mann and S.A Thompson. Rhetorical structure theory: A theory of text organization. Technical Report ISI/RS-87-190, USC/ISI, 1987.

[11] Naoki Kobayashi, Tsutomu Hirao, Hidetaka Kamigaito, Manabu Okumura, and Masaaki Nagata. Top-down rst parsing utilizing granularity levels in documents. In **Proceedings of the AAAI Conference on Artificial Intelligence**, pp. 8099–8106, Apr. 2020.

[12] Xiang Lin, Shafiq Joty, Prathyusha Jwalapuram, and M Saiful Bari. A unified linear-time framework for sentence-level discourse parsing. In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**, pp. 4190–4200, Florence, Italy, July 2019. Association for Computational Linguistics.

[13] Grigorii Guz and Giuseppe Carenini. Coreference for discourse parsing: A neural approach. In **Proceedings of the First Workshop on Computational Approaches to Discourse**, pp. 160–167, Online, November 2020. Association for Computational Linguistics.

[14] Nan Yu, Meishan Zhang, Guohong Fu, and Min Zhang. RST discourse parsing with second-stage EDU-level pre-training. In **Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 4269–4280, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[15] Naoki Kobayashi, Tsutomu Hirao, Hidetaka Kamigaito, Manabu Okumura, and Masaaki Nagata. A simple and strong baseline for end-to-end neural RST-style discourse parsing. In **Findings of the Association for Computational Linguistics: EMNLP 2022**, pp. 6725–6737, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[16] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In **International Conference on Learning Representations**, 2022.

[17] Naoki Kobayashi, Tsutomu Hirao, Hidetaka Kamigaito, Manabu Okumura, and Masaaki Nagata. A simple and strong baseline for end-to-end neural RST-style discourse parsing. In **Findings of the Association for Computational Linguistics: EMNLP 2022**, pp. 6725–6737, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[18] Michael Heilman and Kenji Sagae. Fast rhetorical structure theory discourse parsing. **CoRR**, Vol. abs/1505.02425, , 2015.

[19] Yang Janet Liu and Amir Zeldes. Why can't discourse parsing generalize? a thorough investigation of the impact of data diversity. In **Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics**, pp. 3112–3130, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics.

[20] Mathieu Morey, Philippe Muller, and Nicholas Asher. How much progress have we made on RST discourse parsing? a replication study of recent results on the RST-DT. In **Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing**, pp. 1319–1324, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.