逆翻訳を利用したデータ拡張による文間の修辞構造解析の改善

前川 在 [1]　小林 尚輝 [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, kobayasi, kamigaito, oku}@lr.pi.titech.ac.jp
tsutomu.hirao.kp@hco.ntt.co.jp

概要

十分な量の学習データを確保できないことにより，文間の修辞構造解析の性能は文内と比較して大幅に低く，下流タスクにとって大きな問題となっている．これを解決するため，本稿では，学習データを逆翻訳することで得た擬似正解データを用いて解析器を事前学習し，正解データを用いて追加学習する手法を提案する．シフト還元法による上向き解析器，スパン分割による下向き解析器に提案法を適用し，標準的ベンチマークデータセットである RST-DT，Instr-DT を用いて評価した結果，疑似正解データを用いることで Standard-ParsEval のスコアが約 1-2 ポイント向上することを確認した．

1　はじめに

修辞構造理論 [1] では，Elementary Discourse Unit (EDU) と呼ばれる，節に相当する単位を葉ノードとする構成素木として文書の構造を表す．木は完全 2 分木として表され，中間ノードには 1 つ以上の EDU からなるテキストスパンの役割である核 (Nucleus: N) あるいは衛星 (Satellite: S) と，S から N への修飾関係を表す修辞関係ラベルが与えられる．基本的に核と衛星は対 (S-N，N-S のどちらか) となるが，並列構造を表す場合に例外的に核と核が対 (N-N) となる．図 1 の左の木は 3 文，6 つの EDU からなる文書の構造を表した修辞構造木である．通常，修辞構造解析は与えられた文書を，EDU を葉とした修辞構造木へ変換するが，修辞構造木を要約や翻訳などの下流タスクで利用する場合には，図 1 の右の木のように文を葉とした木が用いられることが多く [2, 3]，文を葉とした解析の性能向上が望まれる．

一般的に修辞構造解析は隣接するテキストスパンを結合するか否か，あるいはどこで分割するかという一種の分類問題を解くことで実現される．よって，学習データの個々の事例は，テキストスパンとその正解/不正解の分割位置，結合される/されないテキストスパンの対のように，テキストスパンを基本単位とする．文を葉とした修辞構造木を構築する解析器を学習するには，EDU を葉とした木を，文を葉とした木へと変換する必要がある．すると，図 1 の右の木からも明らかなように 1 つの修辞構造木から得られるテキストスパンの数が大きく減るため学習データも大きく減ることとなる．修辞構造解析器の学習，評価に標準的に利用される最大規模のデータセットである RST Discourse Treebank (RST-DT) [4] ですら 385 文書しかなく，文間の修辞構造解析のための学習データ不足はより深刻な問題となる．

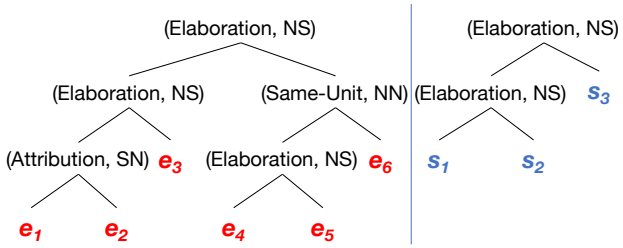

図 1　RST-DT [4] における修辞構造木の例 (WSJ_1100): s_1:[e_1:(Westinghouse Electric Corp.　said), e_2:(it will buy Shaw-Walker Co.)], s_2:[e_3:(Terms weren't disclosed.)], s_3:[e_4:(Shaw-Walker,), e_5:(based in Muskegon, Mich.,), e_6:(makes metal files and desks, and seating and office systems furniture.)] 図中 e は EDU，s は文を表す．

この問題を解決するため，本稿では，正解データの文書に対し逆翻訳を適用することで得た文書を擬似正解データとして活用する手法を提案する．逆翻訳で大量の擬似正解データを作成し，解析器の事前学習に用いることで汎化性能の向上を図る．現在の最高性能の解析器である Kobayashi ら [5] の上向き，下向き解析器に基づき，RST-DT，Instr-DT [6] の二つの異なる領域のデータセットを用いて提案法を評価したところ，擬似正解データを用いることで，RST-DT の場合には Standard-ParsEval で約 1 ポイント，Instr-DT の場合には約 2 ポイント性能が向上することを確認した．

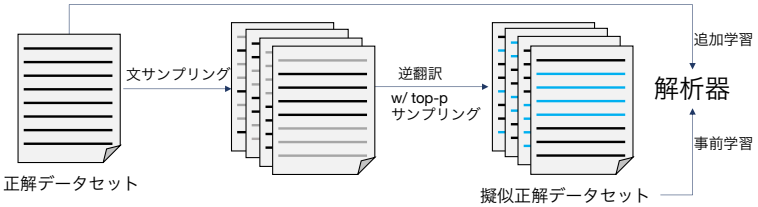

図 2　提案手法の概要

2　関連研究

本来の修辞構造解析は EDU を葉とした木を構築する．EDU は文よりも小さな単位であるため文書内の EDU の数は文の数よりも多く，RST-DT では 1 文書あたりの平均 EDU 数は約 57 である．いわゆる構文解析と比較すると木のサイズが非常に大きい．よって，解析中のエラーの伝搬が問題となる．これを抑制するため，文内の修辞構造と文間の修辞構造を独立に解析する手法が提案されている [7, 8, 9, 10]．Feng ら [7] は，CRF に基づく上向き解析法を提案し， Joty ら [8, 9] は Dynamic CRF に基づく上向き解析法を提案している．Lin ら [10] は，Feng らの手法に対し LSTM を導入することで人手の規則による特徴抽出を廃した．また，Kobayashi ら [11] は，EDU，文，段落という 3 つの粒度の単位を葉とする修辞構造を解析する手法を提案した．

修辞構造解析の最大規模のデータセットである RST-DT ですら文書数は 385 しかなく，十分な量があるとは言えない．よって，学習データとテストデータとの間で語彙が違う場合には解析器の性能が劣化する．これを解決するため擬似正解データを利用する手法が盛んに研究されている．Braud ら [12] はスペイン語，ドイツ語の修辞構造アノテーション済みデータセットを英語へと翻訳することで擬似正解データを作成した．しかし，英語以外のデータセットも十分な量があるとは言えず，大量の擬似正解データを用意することはできない．一方，Huber ら [13] は，distant supervision を用いてラベルなしデータから大規模な擬似正解データ MEGA-DT を作成した．そして，Guz ら [14] は，MEGA-DT を用いて解析器を事前学習，正解データを用いて追加学習することで修辞構造解析の性能が向上することを示した．ただし，Huber らの方法は，大規模な擬似正解データを作成可能であるが，修辞関係ラベルを与えることができない．一方，Kobayashi ら [15] は，複数の解析器の間で結果が一致する合意部分木を擬似正解データとする手法を提案した．MEGA-DT と同様，大量の擬似正解データを作成可能であるが，擬似正解データの信頼性を担保するためには解析器の性能がある程度高くなければならない．しかし，後述するように現在のトップレベルの解析器でも文間の修辞構造解析の性能は非常に低く解析結果の信頼性が低い．よって，複数の解析器の間で合意がとりにくく，それがとれたとしても非常に小さな部分木となってしまい擬似正解データとして役に立たない可能性がある．また，ラベルなしデータを用いる手法は全般に，正解データと同じ領域のラベルなしデータを利用できるとは限らないという限界もある．

3　提案手法

文間の修辞構造解析を改善するには，当然文間の修辞構造のアノテーション済みデータセットが必要となる．人手作成データをこれ以上集めることが困難であることを勘案すれば従来研究と同様に擬似正解データに頼ることが妥当であろう．ここで，我々が欲するデータが文を単位としていることに注目すると，逆翻訳を用いることにより，文書の領域にとらわれることなく語彙や構文の多様性を高めた擬似正解データを作成できる．そして，これを用いて解析器を事前学習することで頑健性の向上を図る (図 2 参照)．

3.1　逆翻訳を用いた擬似正解データ作成

逆翻訳を単純に適用すると元のデータセットと同じサイズの擬似正解データしか得ることができない．これを解決するため，本稿では，以下のように，原文書から逆翻訳対象とする文をサンプリングし，逆翻訳時に累積確率を基にした top-p サンプリング [16] を利用して原文に対して複数の逆翻訳結果を用意する．

1. 文書 d から，n%の文を選択する．具体的には，s_d を文書 d に含まれる文の数として，$\mathtt{int}(s_d \times n/100)$ 文をランダムに選択する．

2. 1 で選択した文を逆翻訳する．逆翻訳時には top-p サンプリングを用いて k 個の翻訳を獲得し，その中から 1 つの翻訳をランダムに選択する．

1, 2 を M 回繰り返すことで正解データの M 倍の擬似正解データを獲得する．なお，1 で選択されなかった文には逆翻訳を適用せずそのまま擬似正解データとして用いることに注意されたい．

3.2 修辞構造解析器

修辞構造解析器としては，現時点で世界最高性能の，Kobayashi ら [5] で用いられた，上向き，下向きのものを利用する．両方の手法ともテキストスパン (文の系列) に対するベクトル表現は事前学習済み言語モデルを経て得たスパンの左端の単語ベクトルと右端の単語ベクトルの平均ベクトルで表す．本稿では文献 [5] に従い事前学習済み言語モデルとして DeBERTa [17] を採用する．

上向き解析

Guz ら [18] のシフト還元法による上向き解析法を単純化した手法である．スタック S に解析済み部分木を格納し，キュー Q に未解析の文を格納し，以下のシフトと還元操作を繰り返すことで下から修辞構造木を構築する．

シフト Q の先頭の文を取り出し，S に積む，

還元 S の上 2 つの部分木をとりだし併合した後，再度 S に積む．

なお，還元操作の後，左右の部分木の核性 (N-S,S-N,N-N) と修辞関係ラベルの推定を独立に行う．それぞれの推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．

$$s_{*}=\mathrm{FFN}_{*}(\mathrm{Concat}(\mathbf{u}_{\mathrm{s}_0}, \mathbf{u}_{\mathrm{s}_1}, \mathbf{u}_{\mathrm{q}_0})) \tag{1}$$

ここで，$\mathbf{u}_{\mathrm{s}_0}$，$\mathbf{u}_{\mathrm{s}_1}$ はそれぞれ S の上 2 つの部分木が支配するスパンのベクトル表現，$\mathbf{u}_{\mathrm{q}_0}$ は Q の先頭の文のベクトル表現である．

下向き解析

Kobayashi ら [11] の再帰的スパン分割による下向き解析法を単純化した手法である．文書全体を表すスパンを貪欲に 2 分割することで下向きに修辞構造木を構築する．i 番目から j 番目の文からなるスパンを k 番目の文で分割するスコアは以下の式で定義される．

$$s_{\mathrm{split}}(i,j,k) = \mathbf{h}_{i:k}\mathbf{W}\mathbf{h}_{k+1:j} + \mathbf{v}_{\mathrm{left}}\mathbf{h}_{i:k} + \mathbf{v}_{\mathrm{right}}\mathbf{h}_{k+1:j} \tag{2}$$

ここで，$\mathbf{W}$ は重み行列，$\mathbf{v}_{\mathrm{left}}$, $\mathbf{v}_{\mathrm{right}}$ は重みベクトル，$\mathbf{h}_{i:k}$, $\mathbf{h}_{k+1:j}$ は以下の式で定義される．

$$\mathbf{h}_{i:k} = \mathrm{FFN}_{\mathrm{left}}(\mathbf{u}_{i:k}), \tag{3}$$

$$\mathbf{h}_{k+1:j} = \mathrm{FFN}_{\mathrm{right}}(\mathbf{u}_{k+1:j}) \tag{4}$$

なお，$\mathbf{u}$ はスパンのベクトル表現である．そして，以下の式でスパンの分割を決定する．

$$\hat{k} = \underset{i \le k < j}{\mathrm{argmax}}\, s_{\mathrm{split}}(i,j,k) \tag{5}$$

上向き解析と同様，核性と修辞ラベル推定は独立に，上のスパン分割と同様に行う．

	RST-DT	Instr-DT
# of Docs.	385	176
# of Sents./Doc.	22.5	19.5
# of Words/Doc.	458.1	276.0

表 1 2 つのデータセットの統計データ

4 実験

4.1 実験設定

評価実験には，標準的ベンチマークデータセットである RST-DT [4] と Instr-DT [6] を用いた．RST-DT は新聞記事 (Wall Street Journal) に対して修辞構造のアノテーションを与えたものであり，学習データ 347 文書，テストデータ 38 文書からなる．開発データは与えられていないので Heilman ら [19] に従い，学習データから 40 文書を選択した． Instr-DT は家の修理に関するマニュアルに対してアノテーションを与えたものであり，176 文書ある．RST-DT とは異なり，テストデータ，開発データの分割は与えられていない．本稿では Kobayashi ら [5] の設定に従った．それぞれのデータセットの統計データを表 1 に示す．

文区切りについては文献 [5] に従った．評価には文献 [20] に従い Standard-ParsEval を用いた．木構造のみを評価する Span，核性，修辞関係ラベルを含めて評価する Nuc.， Rel.，すべてのラベルを含めて評価する Full の計 4 通りの指標を用いた．

なお，各文書に対して n = 25, 50, 75, 100 として文をランダムに選択し，各文に対し p = 0.95 の top-p サンプリングで 4 件の逆翻訳を作成した．そ

		RST-DT				Instr-DT			
		Span	**Nuc.**	**Rel.**	**Full**	**Span**	**Nuc.**	**Rel.**	**Full**
上向き	Baseline	56.4 (2.0)	43.4 (1.4)	31.1 (1.3)	28.8 (1.5)	70.3 (1.6)	54.3 (1.8)	42.9 (2.3)	39.0 (2.2)
	BT (n=25)	55.6 (1.6)	43.5 (1.9)	30.4 (0.9)	28.3 (1.0)	72.3 (0.9)	56.2 (1.3)	45.7 (1.5)	41.6 (1.6)
	BT (n=50)	56.9 (1.3)	44.3 (1.5)	31.9 (1.2)	29.6 (1.2)	72.2 (1.9)	56.4 (2.2)	45.8 (1.3)	41.8 (1.5)
	BT (n=75)	56.4 (1.4)	43.9 (1.7)	31.0 (1.4)	28.8 (1.5)	72.2 (0.9)	56.6 (1.8)	44.9 (1.0)	40.7 (1.1)
	BT (n=100)	55.5 (1.1)	43.1 (0.9)	30.5 (0.8)	28.5 (0.7)	72.4 (1.4)	56.9 (1.7)	45.2 (1.4)	41.9 (1.2)
下向き	Baseline	57.8 (0.6)	44.4 (1.2)	30.6 (1.2)	28.3 (1.1)	69.4 (1.3)	54.1 (1.0)	42.3 (0.8)	38.3 (0.6)
	BT (n=25)	57.8 (0.6)	44.7 (0.8)	31.2 (0.8)	29.1 (0.6)	68.4 (1.4)	54.1 (1.1)	43.0 (1.2)	39.3 (1.2)
	BT (n=50)	58.0 (0..4)	44.5 (0.7)	30.6 (0.5)	28.5 (0.6)	69.6 (1.0)	55.1 (1.1)	43.5 (1.2)	40.0 (1.2)
	BT (n=75)	57.2 (1.3)	44.2 (1.6)	30.7 (1.2)	28.7 (1.3)	70.1 (1.4)	55.1 (1.5)	43.3 (1.4)	39.8 (0.9)
	BT (n=100)	58.1 (0.9)	45.1 (0.6)	31.5 (0.5)	29.5 (0.4)	69.9 (1.4)	53.8 (1.6)	43.0 (1.5)	39.1 (1.7)

表 2　Standard-ParsEval を用いた評価結果. Baseline は正解データのみを用いた解析器.

して，$M = 10$ として原文書の文数の 10 倍の擬似正解データを得た．

翻訳器としては，Marian-NMT [21] を用いて 1.5 億文のパラレルコーパス[1)]で学習された OPUS-MT [22] を用いた．また，逆翻訳を行うためには，原言語を一旦任意の目標言語へと翻訳しなければならない．目標言語として，スペイン語，イタリア語，ドイツ語，フランス語，ベトナム語を用い，RST-DT の学習データを逆翻訳した文と逆翻訳前の文との間の BLEU 値 [23] を計算したところ，スペイン語が 42.7 と最も高かったため，以降の実験の逆翻訳にはスペイン語を利用した．なお，Instr-DT の学習データに対して同様に BLEU 値を計算したところ 40.3 であった．

4.2　結果と考察

表 2 に評価結果を示す．スコアは異なるシードを用いた 5 回の試行の平均値であり，カッコ内はその標準偏差を表す．表より，RST-DT のスコアは全般に Instr-DT よりも低い．RST-DT は，1 文書あたりの文の数が多く，正解の修辞構造木の多様性も高い．よって，学習がより困難であったと考える．

逆翻訳による擬似正解データを用いることで，RST-DT において Span で 0.3，Nuc.，Rel. で 0.9，Full で 1.3 ポイントの性能向上が得られている．上向きと下向きを比較するとやや下向きが良く，逆翻訳対象として選択する文のパラメタ n については，上向きでは $n = 50$，下向きでは $n = 100$ が良い．

一方，Instr-DT においては，すべての指標において RST-DT よりも大きく性能向上しており，その値は約 2 ポイントである．特に上向き解析ではすべての性能向上が 2 ポイントを上回っている．これは，Instr-DT が RST-DT よりもデータサイズが小さいことから，疑似正解データがより効果を発揮したものと考える．上向きと下向きを比較すると明らかに上向きが良い．逆翻訳のパラメタ n については $n = 50$ が良い．両方のデータセットで最適な n は異なるものの擬似正解データを利用する効果は明らかである．

解析器の性能を左右するスパンのベクトル表現が単語のベクトル表現に依存することに注意すると，学習データの語彙とテストデータの語彙に乖離がないことが望ましい．そこで，テストデータに対する学習データ，擬似正解データの語彙の被覆率を調べたところ，RST-DT では，学習データのみで 76.2%，擬似正解データを加えると 80.9%，Instr-DT では，学習データのみで 83.2%，擬似正解データを加えると 89.6%であった．両方ともに語彙の被覆率が向上しており，これが解析性能の向上に寄与したと考える．また，逆翻訳の BLEU にほぼ差がないにもかかわらず，Instr-DT の方が被覆率のゲインが大きいことから，逆翻訳の効果は Instr-DT の方がより顕著であったと考える．

5　おわりに

本稿では，文間の修辞構造解析の性能を改善するため，学習データに対して逆翻訳を適用することで大量の擬似正解データを作成し，これを用いて解析器を事前学習，正解データを用いて追加学習する手法を提案した． Kobayashi ら [5] の世界最高性能の上向き，下向き解析器に基づき，ベンチマークデータセット RST-DT，Instr-DT を用い，擬似正解データを用いないベースラインと提案法を比較したところ，RST-DT では最大で 1.3 ポイント，Instr-DT では最大で 2.8 ポイント，Standard-ParsEval スコアが向上し，逆翻訳による擬似正解データの有効性を確認できた．

1)　https://opus.nlpl.eu/

謝辞

本研究の一部は JSPS 科研費 JP21H03505 の助成を受けている．

参考文献

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

[2] Yuta Kikuchi, Tsutomu Hirao, Hiroya Takamura, Manabu Okumura, and Masaaki Nagata. Single document summarization based on nested tree structure. In **Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 315–320, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[3] Jingun Kwon, Naoki Kobayashi, Hidetaka Kamigaito, and Manabu Okumura. Considering nested tree structure in sentence extractive summarization with pre-trained transformer. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, pp. 4039–4044, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

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

[5] 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. 6754–6766, 2022.

[6] Rajen Subba and Barbara Di Eugenio. An effective discourse parser that uses rich linguistic information. In **Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics**, pp. 566–574, Boulder, Colorado, June 2009. Association for Computational Linguistics.

[7] Vanessa Wei Feng and Graeme Hirst. Text-level discourse parsing with rich linguistic features. In **Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 60–68, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[8] Shafiq Joty, Giuseppe Carenini, Raymond Ng, and Yashar Mehdad. Combining intra- and multi-sentential rhetorical parsing for document-level discourse analysis. In **Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 486–496, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[9] Shafiq Joty, Giuseppe Carenini, and Raymond T. Ng. CODRA: A Novel Discriminative Framework for Rhetorical Analysis. **Computational Linguistics**, Vol. 41, No. 3, pp. 385–435, 09 2015.

[10] 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.

[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] Chloé Braud, Barbara Plank, and Anders Søgaard. Multi-view and multi-task training of RST discourse parsers. In **Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers**, pp. 1903–1913, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[13] Patrick Huber and Giuseppe Carenini. MEGA RST discourse treebanks with structure and nuclearity from scalable distant sentiment supervision. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 7442–7457, Online, November 2020. Association for Computational Linguistics.

[14] Grigorii Guz, Patrick Huber, and Giuseppe Carenini. Unleashing the power of neural discourse parsers - a context and structure aware approach using large scale pretraining. In **Proceedings of the 28th International Conference on Computational Linguistics**, pp. 3794–3805, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[15] Naoki Kobayashi, Tsutomu Hirao, Hidetaka Kamigaito, Manabu Okumura, and Masaaki Nagata. Improving neural RST parsing model with silver agreement subtrees. In **Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 1600–1612, Online, June 2021. Association for Computational Linguistics.

[16] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In **8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020**. OpenReview.net, 2020.

[17] Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. **CoRR**, Vol. abs/2111.09543, , 2021.

[18] 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.

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

[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.

[21] Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. Marian: Fast neural machine translation in C++. In **Proceedings of ACL 2018, System Demonstrations**, pp. 116–121, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[22] Jörg Tiedemann and Santhosh Thottingal. OPUS-MT – building open translation services for the world. In **Proceedings of the 22nd Annual Conference of the European Association for Machine Translation**, pp. 479–480, Lisboa, Portugal, November 2020. European Association for Machine Translation.

[23] Matt Post. A call for clarity in reporting BLEU scores. In **Proceedings of the Third Conference on Machine Translation: Research Papers**, pp. 186–191, Brussels, Belgium, October 2018. Association for Computational Linguistics.