

関係間の関係性を考慮した時間関係グラフ改善のためのグローバル反復辺編集器

牧野 晃平 三輪 誠 佐々木 裕
豊田工業大学

{sd21505, makoto-miwa, yutaka.sasaki}@toyota-ti.ac.jp

概要

時間関係抽出では、イベント間の相対的な時間を考慮する際に、時間関係間の関係性が重要となることがある。しかし、既存の手法ではこのような時間関係間の関係性を考慮せず、個別に時間関係を抽出しているものが多い。そこで、本研究では、既存手法に接続可能な、グローバルに時間関係間の関係性を考慮したモデルとして、既存手法の出力をもとに時間関係グラフを反復的に改善するグローバル反復辺編集器 GIEE を提案する。実験では、時間関係抽出の評価用データセットである MATRES について提案手法を評価し、提案手法が異なる2つの既存手法の性能を一貫して向上できることを確認した。

1 序論

文書中のイベント間の時間的な関係性を対象とし、文内および文をまたいだイベント間の時間関係を抽出する時間関係抽出は、自然言語で記述された文書中の情報を構造化するのに重要なタスクである。時間関係は関係がイベントを共有するフローグラフの様に表現されたり、事前の事前のイベントは事前のイベントであるというような時間的な制約が存在したりするなど、時間関係抽出は一般的な関係抽出 [1, 2] とは異なるため、時間関係を抽出するための特有な手法が提案されている [3, 4]。

時間関係抽出ではイベント間の相対的な時間を対象とするため、関係を個別に分類するだけでなく、時間関係間の関係性を考慮することが重要になる。実際に TempEval-3 [5] や MATRES [6] のような既存の時間関係抽出用コーパスでは、ニュース記事などの文章に記述されたイベントの絶対的な発生時間は一部しか記述されておらず、イベント同士の相対的な時間関係によって発生時間を表現している。

一方で、既存の高性能な手法は時間関係同士の関

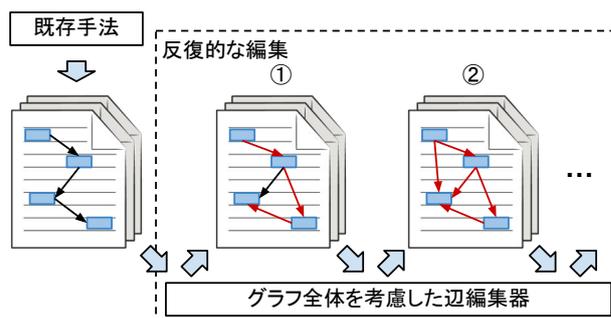


図1 反復的な辺編集

係を考慮できていない。Wen と Ji はイベント間の相対的な時間を直接予測している [7]。Mathur らは複数のグラフ表現を組み合わせ、時間とイベント間の関係性をモデル化したが、時間関係間の関係性は考慮できていない [4]。このような既存手法の利点を保持したまま、異なる構造・実装のモデルに対して関係間の関係性を直接導入するのはコストが高いため、関係間の関係性を既存手法に容易に導入できる手法が必要である。

以上のことから、本研究では、文書検索のリランキングのように [8]、既存の関係抽出手法の出力を全体を見ながら再評価し、関係を編集することで関係間の関連性を考慮するモデルとして、グローバル反復辺編集器 (Global Iterative Edge-Editor; GIEE) を提案する。本手法では、入力となる文書と時間関係グラフを用いて、文書の文脈表現に加え、時間関係グラフの表現を利用することで、全体の時間関係間の関係性を考慮した辺の編集を行う。更に、編集されたグラフを繰り返し編集することで、入力された辺間だけでなく、編集された辺同士の相互関係を含めてグラフ全体で考慮する。

本研究の貢献は以下の3点である。

- 関係間の関係性を考慮するため、既存手法の出力を編集するという、既存手法に容易に導入可能な問題設定を提案した。

- 時間関係グラフの表現を利用する，時間関係間全体の関係性を考慮した時間関係抽出モデルであるグローバル反復辺編集器 GIEE を提案した.
- 異なる 2 つの既存手法の後続に提案手法を接続することで，いずれの手法についても，もとの抽出性能を改善できることを示した.

2 提案手法

本研究では，既存の時間関係抽出手法に対して関係間の関係性の観点を導入するため，既存手法で抽出した時間関係グラフを編集するモデルとして，グローバル反復辺編集器 GIEE を提案する．時間関係グラフはイベントを節点，時間関係を辺として表現される．GIEE では図 1 のように，時間関係全体の関係間相互関係を考慮した辺編集器 (Edge-Editor; EE) を反復的に利用することで，既存手法で抽出したグラフの辺間のみならず，編集された辺も含めた，辺間の関係性をグラフ全体で考慮することを目指す．ここでは，この EE と GIEE についてそれぞれ説明し，その後，それらを使った予測と学習，既存手法との接続について説明する．

2.1 辺編集器

まず辺同士の相互関係を考慮して辺を編集する深層学習ベースの辺編集器 EE について説明する．EE では BigBird [9] や Longformer [10] のような文書単位の事前学習モデルを用いて得られる文脈表現と，グラフニューラルネットワーク (GNN) を用いて作成する入力的时间関係グラフ $G^{in} = \mathcal{G}(N, E^{in})$ の表現から対象の辺に接続している節点ペアを分類して辺を編集することで編集後のグラフ $G^{out} = \mathcal{G}(N, E^{out})$ を出力する．モデルのパラメタを θ とすると EE は以下のように表現できる．

$$G^{out} = \text{EE}(\text{doc}, G^{in}; \theta) \quad (1)$$

辺の編集は以下の 3 ステップで行う．(1) 事前学習モデルによって節点の初期表現としてイベントの文脈表現を作成する．(2) 関係間の関係性を考慮するために GNN を利用して入力グラフの構造情報を埋め込んだ節点の表現を作成する．(3) 節点ペアの表現と編集前の辺の表現から辺を分類して編集後のグラフを作成する．

イベントの初期表現は，イベントを構成するトークンの文脈表現を集約して作成する．文長 T の入力文書 doc を事前学習モデルによって埋め込んだト

Algorithm 1: グローバル反復辺編集器 GIEE

Input: doc
 $G^0 = \mathcal{G}(N, E^0)$: 初期グラフ
 $\Theta = [\theta_1, \theta_2, \dots, \theta_K]$: 反復ステップごとのパラメタのリスト
Output: $G^K = \mathcal{G}(N, E^K)$: 出力グラフ
while i in range(1, K) **do**
 $G^i = \text{EE}(\text{doc}, G^{i-1}; \theta_i)$
end while

クンの表現 $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T]$ の次元を線形層で減らしたものに対して， i 番目の節点の初期表現 \mathbf{n}_i^0 は始点のトークンと終点のトークンとその間の表現の平均を取って表現を作成する．

次に関係間の関係性をグラフ全体で考慮するために入力グラフの構造表現を GNN によって埋め込む．具体的には関係ラベルの違いを捉えるために，異種グラフが扱える Relational Graph Convolution Network (RGCN) [11] を L 層積み重ねて L -hop の関係性を考慮した表現を作成する． l 番目の層の節点の表現は， $\sigma(\mathbf{x})$ をシグモイド関数， \mathbf{W}^{self} を自己ループのためのパラメタ，RGCN の引数を節点の表現と辺すると，以下のように表現できる．

$$\mathbf{n}_i^l = \mathbf{n}_i^{l-1} \mathbf{W}^{self\top} + \sigma(\text{RGCN}(\mathbf{n}, E^{in})) \quad (2)$$

最後に，それぞれの節点ペア (N_i, N_j) 間の辺を，その表現 \mathbf{v}_{ij} から同時に分類する． \mathbf{v}_{ij} は以下のように，RGCN の L 層目の節点ペアの表現 \mathbf{n}_i^L と \mathbf{n}_j^L ，編集前の辺のクラスを埋め込んだ $\mathbf{u}_{ij} = \text{Embed}(E_{ij}^{in})$ を結合した表現とする．

$$\mathbf{v}_{ij} = [\mathbf{n}_i^L; \mathbf{n}_j^L; \mathbf{u}_{ij}] \quad (3)$$

関係なしを含むクラス全体 C の内，クラス c に対して用意した重み \mathbf{w}_c^{edit} による \mathbf{v}_{ij} の重み付き和によってスコア $z_{ij,c}$ を算出し，スコアが最大となるクラスを選択することで編集を行う．

$$z_{ij,c} = \mathbf{w}_c^{edit} \mathbf{v}_{ij}^\top \quad (4)$$

$$E_{ij}^{out} = \arg \max_{c \in C} p_{ij,c} \quad (5)$$

2.2 反復的な編集

EE により辺を一度のみ編集した場合には，全ての辺が入力されたグラフの表現のみに基づいて同時に編集されるため，編集結果の辺を含めた辺間の関係性を考慮出来ていない．そこで GIEE では，反復的に EE を利用して編集することで，グラフ全体で

の整合性がとれた辺編集を目指す。具体的には、アルゴリズム 1 のように、EE を繰り返し利用し、同じ辺を複数回にわたって繰り返し編集することで、以前に編集した辺との関係性も考慮する。このとき、EE のパラメタを全ての反復で同一のパラメタとするか ($\theta_1 = \theta_2 = \dots = \theta_l$)、全ての反復で別のパラメタを使うかは、チューニングによって決定する。

2.3 アンサンブルを用いた予測

GIEE では同一の辺を複数回予測した結果が編集履歴として得られるため、その履歴を有効利用し、予測結果のアンサンブルを行い、抽出性能向上を目指す。それぞれの反復では異なるグラフを入力として予測をするため、アンサンブルするのに適していると言える。アンサンブルの方法は、反復ごとの $z_{ij,c}$ の最大値・平均値・投票・アンサンブルなし (最後の反復の出力を使用) の 4 種類の中からチューニングで最も高い性能が得られたものを利用する。

2.4 学習

モデルの学習は、各反復で予測されたグラフの対数尤度を最大化すると同時に、アンサンブルで得られた予測の対数尤度を最大化するための損失 L^{ens} に対してもパラメタを最適化する。反復的な編集に対する損失を L^{itr} とすると損失関数は $(1 - \lambda)L^{itr} + \lambda L^{ens}$ として学習する。

2.5 既存手法との接続

既存手法と本提案手法を接続し、時間関係抽出において関係間の関係性を考慮するために、提案手法の入力 G^{in} の辺 E^{in} を既存の手法で抽出した時間関係として、パラメタを最適化する。このようにすることで、追加の実装を必要なしに、既存手法の出力を活かしつつ、容易に関係間の関係性を導入することが可能になる。

3 評価

3.1 評価設定

提案手法の評価は時間関係抽出の評価におけるデファクトスタンダードとなっている MATRES コーパス [6] を用いて評価する。MATRES では時間関係のラベルが BEFORE, AFTER, EQUAL, VAGUE の 4 つのラベルが存在し、既存研究 [12] と同様に VAGUE は負例として扱い、評価には含めないもの

表 1 MATRES に対する既存手法との比較

論文	再現		+GIEE	
	開発	評価	開発	評価
スクラッチ	-	-	75.7	70.9
TIMERS [4]	82.3	-	-	-
SMTL [12]	81.6	-	-	-
Wen と Ji[7]	81.7	82.1	80.7	82.5
Zhao ら [13]	79.6	81.1	78.2	81.4
				81.1
				78.5

とする。データは Wen と Ji [7] と同様に分割して、訓練、開発、評価用データはそれぞれ 234, 21, 20 件で、それぞれ含まれる関係数は 10,884, 1,852, 818 件である。評価は関係に対する F 値で、3 回実験したときの平均値を用いる。

評価は既存手法の出力をモデルの入力として最適化したものを既存手法の性能と比較する。最先端の手法の内、抽出結果が得られるもの [7, 13] に限定し、再現実験によって得られた出力をモデルの入力として利用した。また、入力グラフの辺を空とした場合 (スクラッチ) に対する抽出性能も確認する。モデルの最適化には Adam [14] を用いた。モデルで利用する事前学習モデルは、文書単位の前学習モデルの中でも高い性能である BigBird [9] の重みを固定して利用する。ハイパーパラメタチューニングは Optuna [15] を用いて、Wen と Ji[7] の手法の出力に対して行い、その詳細は付録 A に示した。

3.2 結果

表 1 の結果では、スクラッチの場合は他の手法と比較しても大幅に低い性能しか得られていないにもかかわらず、既存の手法に対して GIEE を適用すると、編集前と比べて性能が一貫して向上していることが確認できる。一方で、論文で報告されているスコアや最先端の結果と比較すると劣っているが、これは編集前の時点での性能が低かったことに起因すると考えられる。ベースラインとなるモデルのチューニングおよび選定は今後の課題である。以上のことから、GIEE は入力されたグラフ上で辺間の関係性をグラフ全体で考慮することで、抽出性能を向上させる効果を持つと言える。

4 解析

提案モデルの動作メカニズムを解明するため、ハイパーパラメタチューニングで得られた特徴的なパラメタの考察と、反復ごとに出力されるグラフの

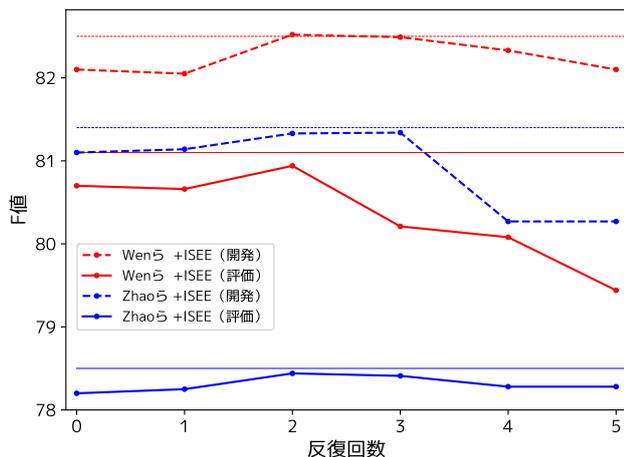


図2 反復ごとの抽出性能. 水平線はアンサンブルしたときのF値, 反復回数0は編集前のF値を示している.

抽出性能について議論する. アブレーションの結果を表2に示した. ハイパーパラメータチューニングで得られたパラメータとして特徴的だったのは, 全ての θ を同一にする点である. これは, 全く同じパラメータのモデルで自己回帰してその出力をアンサンブルするのが最も良い性能となる, ということを示しており, モデルのパラメータ数を増やすことなく性能を向上させることができる. また, アブレーションによって確認された, モデルに最も寄与していた点は, u_{ij} による入力されたグラフの情報を保持する機構であり, これを除くと3%ポイント以上性能が低下した. 次に影響があったのは時間関係全体の相互関係を確認するためのRGCNの層の数 L で, RGCNを利用しない場合においては約1%ポイント性能が低下した. この結果から, 既存手法に対して関係間の関係性を導入することで性能が向上したということが確認でき, 関係間の関係性を考慮する重要性が確認できた. アブレーションに関する解析の詳細は付録Bに示す.

図2のWenとJi+GIEEにおける最高性能のモデルの反復ごとの結果を確認すると, どの場合においても編集前に比べて徐々に抽出性能が向上し, その後低下していることが確認できる. アブレーションの結果から $I < 5$ の場合には性能が低下していることから, 前半の反復では抽出出来ていない部分を後半の反復で抽出しようとして反復単体の性能は低下しているのではないかと考えられる. 最終的なアンサンブルとそれぞれの反復を比較すると, ほとんどの反復で下回っているが, 同じパラメータのモデルに対して少し異なるグラフを入力してアンサンブルしたことで性能が向上した. これは性能が低くなってい

表2 Ablation study

	開発	評価
Wen と Ji[7]	82.1	80.7
+GIEE	82.5	81.1
$I = 1$	82.4	80.7
$I = 2$	82.3	80.6
$I = 3$	82.4	81.0
$I = 4$	82.4	80.8
$-u_{ij}$	81.0	77.9
$\lambda = 0$	82.4	80.8
$\lambda = 1$	82.5	80.8
$L = 0$	82.2	79.8
$L = 1$	82.5	80.5
Gold+GIEE	100.0	100.0

る後半の反復も有意義な情報を提供していることを示している.

5 関連研究

文単位の関係抽出では関係グラフ上での表現を利用した手法が提案されているが, 文書単位に適用するのは計算コスト上難しいという問題がある[16]. 文書単位の関係抽出では文書のグラフを構築して用語間の関係性をモデル化した手法が提案されている[17, 18]. しかしながら, 関係を辺としてグラフを構築していない. このような中で, 文書単位の関係抽出において順次的に関係グラフの辺を編集する手法[19]が提案されたが, 編集された辺同士の関係性が考慮できておらず, 時間関係抽出にも適用されていない.

6 結論

本研究では時間関係抽出において, 既存手法に対して容易に接続可能で時間関係グラフ全体の関係間の関係性を考慮可能なモデルとして, 既存手法が出力した時間関係グラフをグローバル反復辺編集器GIEEによって編集する方法を提案した. 実験では2つの既存手法に提案手法を適用し, いずれの手法についても性能を向上させることができ, 関係間の関係性を導入する有効性が確認できた. 今後の課題は, 本研究で提案した編集する機構がどのように作用しているのか更なる調査を行い, 単一のコーパスのみでなく, 他のコーパスでの調査を進める. また, 本提案のように, 既存の抽出手法に対して接続するような設定において, 更に効果的な方策がないか調査する.

参考文献

- [1] 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**, pp. 764–777, Florence, Italy, July 2019. Association for Computational Linguistics.
- [2] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In **Proceedings of the 5th International Workshop on Semantic Evaluation**, pp. 33–38, Uppsala, Sweden, July 2010. Association for Computational Linguistics.
- [3] Rujun Han, Xiang Ren, and Nanyun Peng. ECONET: Effective continual pretraining of language models for event temporal reasoning. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, pp. 5367–5380, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.
- [4] Puneet Mathur, Rajiv Jain, Franck Dernoncourt, Vlad Morariu, Quan Hung Tran, and Dinesh Manocha. TIMERS: Document-level temporal 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 2: Short Papers)**, pp. 524–533, Online, August 2021. Association for Computational Linguistics.
- [5] Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. SemEval-2013 task 1: TempEval-3: Evaluating time expressions, events, and temporal relations. In **Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)**, pp. 1–9, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics.
- [6] Qiang Ning, Hao Wu, and Dan Roth. A multi-axis annotation scheme for event temporal relations. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 1318–1328, Melbourne, Australia, July 2018. Association for Computational Linguistics.
- [7] Haoyang Wen and Heng Ji. Utilizing relative event time to enhance event-event temporal relation extraction. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, pp. 10431–10437, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.
- [8] Qiang Sheng, Juan Cao, Xueyao Zhang, Xirong Li, and Lei Zhong. Article reranking by memory-enhanced key sentence matching for detecting previously fact-checked claims. 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)**, pp. 5468–5481, Online, August 2021. Association for Computational Linguistics.
- [9] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. **Advances in Neural Information Processing Systems**, Vol. 33, , 2020.
- [10] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. **arXiv:2004.05150**, 2020.
- [11] Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In Aldo Gangemi, Roberto Navigli, Maria-Esther Vidal, Pascal Hitzler, Raphaël Troncy, Laura Hollink, Anna Tordai, and Mehwish Alam, editors, **The Semantic Web**, pp. 593–607, Cham, 2018. Springer International Publishing.
- [12] Miguel Ballesteros, Rishita Anubhai, Shuai Wang, Nima Pourdamghani, Yogarshi Vyas, Jie Ma, Parminder Bhatia, Kathleen McKeown, and Yaser Al-Onaizan. Severing the edge between before and after: Neural architectures for temporal ordering of events. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 5412–5417, Online, November 2020. Association for Computational Linguistics.
- [13] Xinyu Zhao, Shih-Ting Lin, and Greg Durrett. Effective distant supervision for temporal relation extraction. In **Proceedings of the Second Workshop on Domain Adaptation for NLP**, pp. 195–203, Kyiv, Ukraine, April 2021. Association for Computational Linguistics.
- [14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, **3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings**, 2015.
- [15] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In **Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**, 2019.
- [16] 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**, pp. 1409–1418, Florence, Italy, July 2019. Association for Computational Linguistics.
- [17] Feni Christopoulou, Makoto Miwa, and Sophia Ananiadou. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. 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)**, pp. 4925–4936, Hong Kong, China, November 2019. Association for Computational Linguistics.
- [18] 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**, pp. 1546–1557, Online, July 2020. Association for Computational Linguistics.
- [19] Kohei Makino, Makoto Miwa, and Yutaka Sasaki. A neural edge-editing approach for document-level relation graph extraction. In **Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021**, pp. 2653–2662, Online, August 2021. Association for Computational Linguistics.
- [20] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. **The Journal of Machine Learning Research**, Vol. 15, No. 1, pp. 1929–1958, 2014.
- [21] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Q. Weinberger, editors, **Advances in Neural Information Processing Systems**, Vol. 24, pp. 2546–2554. Curran Associates, Inc., 2011.
- [22] Liam Li, Kevin Jamieson, Afshin Rostamizadeh, Ekaterina Gonnina, Jonathan Ben-tzur, Moritz Hardt, Benjamin Recht, and Ameet Talwalkar. A system for massively parallel hyperparameter tuning. In I. Dhillon, D. Papailiopoulos, and V. Sze, editors, **Proceedings of Machine Learning and Systems**, Vol. 2, pp. 230–246, 2020.

付録

A チューニングの詳細

モデルの学習時には正則化のためにドロップアウト [20] を加える。ドロップアウトは入力 X 、出力層の前の v_{ij} に対するもの、それ以外の隠れ層全てに対するものの3種類に分けて、それぞれのドロップアウト率をハイパーパラメータとする。チューニング時には提案したモデルに追加して、EEの代わりに順次的な編集 [19] を反復的に行えるように、 d_{max} というパラメータを追加し、 $d_{max} = 1$ の場合にはEEと同一、 $d_{max} > 2$ の場合には文内の節点ペア、隣接文の節点ペアと徐々に離れたものを順次的に編集できるようにした。また、アンサンブルについては、どの反復を重要視して優先的に利用するか重みづけできるパラメータとして、 $\sum_{k=1}^K w_k$ の条件で k 番目の反復の重みを w_k とした。重みづけを利用する場合は、例えば平均でのアンサンブルは学習可能な重みを利用した重み付き平均に変化する。

ハイパーパラメータチューニングは Optuna [15] を用いて、枝刈りされたものも含めて 1,000 回の試行で開発データに対する F 値が最大となるパラメータを探索した。パラメータのサンプリングには木構造 Parzen 推定 [21]、探索の枝刈りアルゴリズムは論文中のパラメータを $s = 1, \eta = 2, r = 0.15$ とした Successive Halving アルゴリズム [22] を利用した。チューニングは複数のマシンで分散して行い、NVIDIA 社の GTX 1080Ti, RTX 3090, Tesla V100, TITAN V の GPU が搭載されたマシンを利用した。

探索空間とチューニングによって得られたパラメータは表 3 に示した。チューニングしたパラメータは表の上から、Adam の学習率、3 種類のドロップアウト率、RGCN の層の数 L 、RGCN に逆方向の辺を追加する双方向 RGCN とするかどうか、全体的な隠れ層の次元数、編集前の辺の埋め込み u_{ij} の次元数、反復回数 I 、アンサンブルの種類、アンサンブルの重み w_k の有無、アンサンブルの損失の重み λ 、 θ および順次的な編集の順次毎の θ をすべて同じものにするかどうか、順次の最大値 d_{max} の計 15 個である。チューニングの結果から、順次的な編集は必要なく、すべての辺を同時に繰り返し編集する方がよいということが確認できた。

表 3 ハイパーパラメータチューニングの探索空間と結果

パラメータ	範囲	数値
学習率	$[10^{-5}, 10^{-2}]$	0.0015
ドロップアウト (入力)	[0, 1]	0.13
ドロップアウト (出力)	[0, 1]	0.88
ドロップアウト (隠れ層)	[0, 1]	0.086
RGCN の層の数	[1, 5]	2
双方向 RGCN	T/F	F
隠れ層の次元数	[2, 200]	153
u_{ij} の次元数	[1, 32]	21
反復回数 I	[2, 6]	5
アンサンブルの種類	最大値, 平均値 投票, なし	平均値
w_k の有無	T/F	F
λ	[0, 1]	0.91
θ_k を同一にするか	T/F	F
θ を順次ごとに同一にするか	T/F	F
d_{max}	[1, 5]	1

B Ablation study

性能向上がモデルの何に起因しているのか確認するために、表 2 に示したようなアブレーションを行った。アブレーションは最も高い性能を示した Wen と Ji+GIEE に対して行った。また、別途入力に完全に正解を与えた場合 (Gold) についても検証した。アブレーションにおいても 3 回実験を行った際の平均の F 値で評価した。

まず反復回数 I を減らした場合には、全ての場合に対して開発・評価ともに僅かに性能が低下していることが確認できる。編集前の辺の埋め込み u_{ij} を消去した場合には編集前と比べても性能が大きく低下しており、編集前の情報を損なわないために重要な役割であると確認できる。アンサンブルの損失 L^{ens} の重み λ を変更して L^{ens} の有無を変えた際にはどちらも評価データに対する性能が低下しており、この損失の必要性が確認できる。RGCN の層の数 L については、グラフの構造情報が欠損する $L = 0$ のケースにおいては編集前より性能が低下している。 $L = 1$ としたケースでも依然として性能が低いままである。また、Gold の場合では完全に入力を再現できており、入力を完全に保持することが可能であると確認できた。以上の結果から、我々の提案に大きく寄与しているのは、編集前の入力グラフを保持する機構とグローバルに関係間の関係性を考慮する点で、既存手法に対して関係間の関係性を導入する重要性が確認できる。