自動獲得された因果関係知識に基づく文間の因果関係の推定

山田 涼太
北陸先端科学技術大学院大学
yamada224@jaist.ac.jp

白井 清昭
北陸先端科学技術大学院大学
kshirai@jaist.ac.jp

1 はじめに

文間の因果関係の推定は，質問応答システムなど，自然言語処理における幅広い場面で応用できる技術である．ここで，文間の因果関係の推定とは，2 つの文 C と E が与えらたとき，C が E によって表される事象を引き起こす原因になっているか否かを推定するタスクとする．これまでの因果関係推定の研究は教師あり機械学習に基づく手法が主流であった．しかし，教師あり機械学習のためには因果関係の有無が付与された大量の文の組の集合が必要である．

本研究は，ブートストラップの手法によって因果関係が付与されたデータセットを自動構築し，それを元に文間の因果関係推定モデルを学習する手法を探究する [1]．データセットの構築，および因果関係推定モデルの学習には，近年自然言語処理の様々なタスクで良い成果が得られている Bidirectional Encoder Representationsfrom Transformers (BERT)[2] を用いる．また，自動獲得された因果関係知識の量と推定精度の相関を実験的に検証する．

2 関連研究

Kruengkrai らは，訓練データにおける原因文と結果文のそれぞれから得られた単語ベクトルや，質問応答システムの回答から得られた情報から，畳み込みニューラルネットワークを用いて因果関係を判定する手法を提案した [3]．実験の結果，この手法による判定の精度は最大で 55.13%であった．Hashimoto らは，文間の因果関係を推定する SVM(Support Vector Machine) を学習し，得られた原因-結果の因果関係を推移律によって繋げることで，先に起こることを予測するシナリオを生成した [4]．実験では，68%の精度で 50,000 のシナリオを作成できたと報告している．これらの研究では，因果関係を推定するモデルは人手で構築された訓練データから学習されている．

Abe らは，特定の関係が成立する単語の組とそれを抽出するためのパターンを同時に獲得するアルゴリズム [5] を応用し，因果関係の文の組とそれを抽出するパターンを自動獲得する手法を提案した [6]．例えば，同じ文内に出現しかつ同じ目的語を持つ 2 つの節を因果関係が成立する文の組として抽出するパターンが得られた．

本研究は，人手によるアノテーションを必要としない手法によって文間の因果関係を推定するモデルを学習する点に特徴がある．文献 [6] の手法でも因果関係が付与されたデータを必要としないが，コーパスから因果関係が成立する文の組を抽出することを目的としているのに対し，本研究は 2 文間の因果関係の推定モデルの学習を目的としている点が異なる．

3 提案手法

3.1 概要

提案手法の処理の流れを図 1 に示す．まず，コーパスから，ヒューリスティクスによって因果関係の有無のラベルが付与された初期のデータを作成する．作成した初期データは，訓練データ，開発データ，検証データに分割する．一方，コーパスから因果関係の有無のラベルが付与されていない文の組の集合をあらかじめ獲得する．

推定モデルの学習は通常のブートストラップ法にしたがう．まず，初期の訓練データと開発データから，因果関係の有無を判定するモデルを学習する．次に，ラベルなしデータに学習した因果関係推定モデルを適用し，判定結果が付与された新たなデータセットを得る．この中から判定の信頼度が高いデータを選別し，訓練データに追加する．この処理を繰り返すが，推定モデルを学習する度に，検証データを用いてモデルによる因果関係推定の正解率を測り，ひとつ前のステップで学習されたモデルから正解率が改善されなければ，学習を終了する (図 1 の

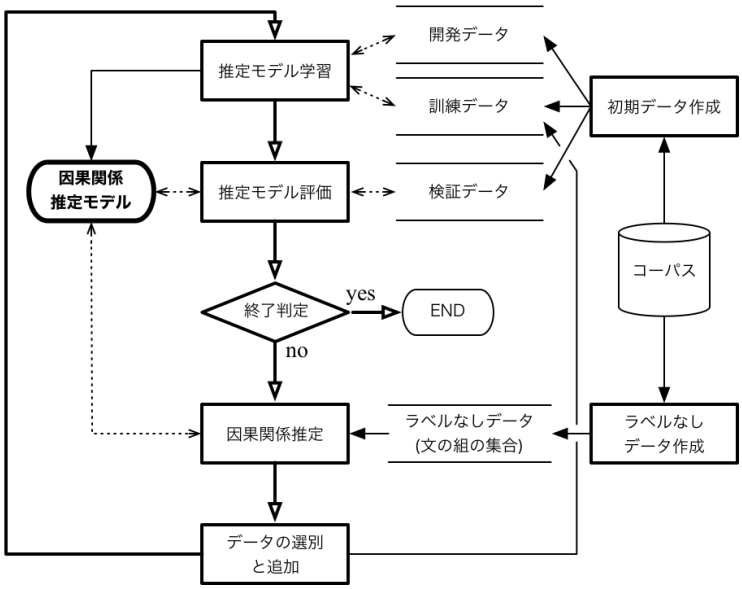

図 1　提案手法の概要

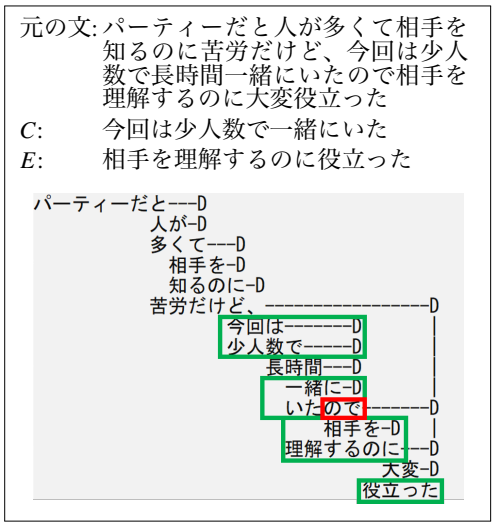

図 2　因果関係が成立する文の組の抽出例

「終了判定」). 最終的に得られる因果関係推定モデルは，初期データと自動拡張したデータを合わせたデータから学習されたものとなる．

3.2　初期データの作成

コーパスから因果関係が成立する可能性の高い文の組を収集する．具体的には，接続詞「から」「ので」を含む文を検出し，その前後に出現する文を抽出する．以下に示す例文のように，「から」「ので」で結ばれた文の間には因果関係が成立する可能性が高いと考える．

電車が止まったからバスが混む
雨が降ったので地面がぬかるんでいる

因果関係が成立する文の組 (C, E, yes) を以下の手続きで抽出する．

1. 接続詞「から」「ので」(以下，「因果関係キーワード」と記す) を含み，かつその直前が動詞または助動詞である文を抽出する．
2. CaboCha[7] を用いて文の文節の係り受け解析を行う．
3. 因果関係キーワードを含む文節に係り，かつその末尾が助詞である文節を抽出する．また，抽出した文節に係り，かつその末尾が助詞である文節も抽出する．この操作を再帰的に繰り返す．最後に，抽出した文節を連結し，「から」「ので」を削除して，文 C を得る．
4. 因果関係キーワードの後に最初に出現する動詞を検出する．同様に，動詞に係りかつ末尾が助詞である文節を再帰的に抽出する．抽出した文節を連結して文 E を得る．
5. C，E のいずれかの文字数が 7 未満のとき，これを除外する．
6. これらの文に因果関係が成立するというラベル「yes」をつけ，(C, E, yes) という組を抽出する．

上記の手続きによる抽出の例を図 2 に示す．

因果関係を推定するモデルを学習するためには，因果関係が成立する文の組 (正例) だけでなく，成立しない文の組 (負例) も必要である．負例は以下の手続きで獲得する．先の手続きで得られた正例のデータセットの集合を $\{ (C_i, E_i, yes) \}$ と記す．原因文 C_i に対し，他の組の結果文 E_j $(i \neq j)$ の中からランダムに 1 つを選択し，負例 (C_i, E_j, no) を生成する．この操作を全ての C_i について繰り返す．結果として，正例と同数の負例を得る．

初期データは，あらかじめ 8:1:1 の比率で，訓練データ，開発データ，検証データにランダムに分割する．

3.3　因果関係推定モデルの学習

原因文 C と結果文 E の組が与えられたとき，それらの間に因果関係が成立するか否かを判定するモデルを学習する．このモデルを BERT を用いて学習する．すなわち，入力を以下のような系列に変換する．

[CLS] $cw_1 \cdots cw_n$ [SEP] $ew_1 \cdots ew_m$ [SEP]

[CLS] は文の組の分類のための抽象表現を得るた

めのトークン，[SEP] は 2 つの文の境界を示すトークン，cw_i は原因文の単語，ew_i は結果文の単語を表す．

BERT による分類モデルの学習は，pre-training(事前学習) と fine-tuning の 2 つのステップから構成される．事前学習済みの言語モデルとして，日本語版 Wikipedia から事前学習され，京都大学によって公開されているモデル [8] を使用する．一方，fine-tuning は因果関係の有無のラベルが付与された訓練データと開発データを用いて行う．学習率 (learning rate) は 2^{-5}，バッチサイズは 32，エポック数は 10 と設定する．

3.4 訓練データの拡張

訓練データを拡張するために，ラベルなしデータをあらかじめ用意する．接続詞「ため」をキーワードとし，初期データの作成と同様の手続きで原因文 C と結果文 E の候補の組を抽出する．「ため」の前後に出現する文は，以下の文 1 のように因果関係が成立する場合もあれば，文 2 のように従属節が主節の目的を表す場合もあるため，正例と負例が混在したデータが得られることが期待できる．

文 1: 雪が降ったため遠足は中止になった

文 2: 学会で発表するため何回も練習した

n 回目の反復ステップで学習された因果関係推定モデルを M_n と記す．すなわち，M_{n-1} を用いて訓練データを拡張し，拡張後のデータで M_n を再学習する．初期データから学習された因果関係推定モデルは M_0 とする．ラベルなしデータの集合 $\{(C_i, E_i, ?)\}$(?は因果関係の有無が不明であることを表す) は，因果関係判定モデルの反復学習のたびに，別のものを用意する．以降，n 回目の反復で訓練データ拡張のために用いるラベルなしデータの集合を U_n と記す．

訓練データの拡張では，モデル M_{n-1} を用いて U_n 内の文の組に対して因果関係の有無を判定する．さらに，判定の信頼度も求める．判定の信頼度は，ここでは BERT による因果関係推定モデルにおける出力ノードの値とする．U_n の中から判定の信頼度の大きいデータを選別し，訓練データに追加する．

予備実験では，信頼度が上位のデータの多くが，因果関係推定モデルによって 2 つの文の間に因果関係が成立すると判定されていた．つまり，判定の信頼度が上位のデータのほとんどが正例であった．そのため，訓練データに追加するデータを作成する際に，正例と負例のバランスを取る．具体的には，追加データの数を N_{add} と設定するとき，信頼度の大きい順に正例の数が $N_{add}/2$ 件に到達するまで追加データを取得する．この中に含まれる負例の数が N_{neg} のとき，$N_{add}/2 - N_{neg}$ 件の負例を新たに作成する．この負例は，初期データの作成時と同様に，ラベルなしデータ U_n の中から原因文と結果文をランダムに組み合わせて作成する．最終的に正例と負例の数が等しい N_{add} 件のデータを拡張データとし，これを訓練データに追加する．以降，拡張した訓練データを用いて因果関係推定モデルを再学習する．

4 評価実験

4.1 実験データ

初期データは，毎日新聞の 2009 年から 2013 年の記事から獲得した．初期データの数は 2,796(正例，負例が 1,398 ずつ) であった．

ラベルなしデータは，同様に毎日新聞の新聞記事データから獲得した．U_1 は 2013 年，U_2 は 2012 年，U_3 は 2011 年の新聞記事から獲得した．提案手法では，検証データの正解率が向上しなくなった時点で訓練データの追加を停止するが，今回の実験では試験的に反復回数を 3 回と設定している．U_i は互いに重なりはなく，また抽出に用いた因果関係キーワードが異なるため，初期データとも重なりはない．

前述の検証データは自動構築されたものである．これとは別に，因果関係推定モデルの性能を正確に測るため，テストデータを人手で作成した．接続詞「から」をキーワードとして抽出した文の組を 50 件，「ので」について 50 件，「ため」について 100 件，合計 200 件の文の組を抽出した．これらに対して，著者 2 名が独立に因果関係の有無を判定した．二者の判定の一致率は 72%，κ 係数は 0.44 であった．一致率や κ 係数は低いが，これは因果関係が成立するかを判定する際に，どれだけ常識的知識を使って情報を補うかに関して見解が分かれることが主な原因であった．例えば，C=「２歳だった」，E=「原爆の記憶はない」のとき，1 名の判定者は 2 歳という幼ない年齢では記憶が残らないと判断し，因果関係があると判定したが，もう 1 名の判定者は 2 つの文に強い関連性がないと判断した．判定が異なる文の組については，著者 2 名の合議により最終的なラベルを決めた．テストデータの正例数は 69，負例数は 131 となった．

表 1　判定の信頼度と判定精度の関係

t	精度 P_t	正例:負例 (全体)	正例:負例 (正解のみ)
0.8	0.733 (88/120)	76:44	62:26
0.9	0.787 (70/89)	63:26	57:13
1.0	0.797 (55/69)	58:11	54:1
1.1	0.839 (47/56)	47:9	47:0
1.2	0.860 (43/50)	43:7	43:0
1.3	0.897 (35/39)	35:4	35:0
1.4	0.906 (29/32)	29:3	29:0
1.5	0.895 (17/19)	17:2	17:0

表 2　因果関係推定モデルの反復学習の結果

i	モデル	U_i	訓練	正解率
0	M_0	–	2,236	0.639
1	M_1	5,581	4,236	0.657
2	M_2	6,063	6,236	0.636
3	M_3	6,350	8,236	0.650

4.2　実験結果と考察

提案手法では，BERT モデルの出力ノードの値を判定の信頼度とし，これが高いデータを訓練データに追加する．この妥当性を検証するために，判定の信頼度が閾値 t 以上のデータに対する精度を P_t とし，t を変化させたときの P_t の変動を調べた．検証に用いた因果関係推定モデルは初期データで学習したもの (M_0)，P_t を調べるためのデータは (自動作成した) 検証データを用いた．結果を表 1 に示す．閾値 t が大きいほど判定の精度が高いことから，BERT モデルの出力ノードの値を判定の信頼度とすることは妥当であるといえる．また，判定の信頼度が高くなると，負例の数が正例の数よりもかなり少なくなる傾向も見られた．

表 2 は，それぞれの反復ステップ i について，ラベルなしデータ U_i の数 (3 列目)，訓練データの総数 (4 列目)，および検証データでの正解率 (5 列目) を示している．今回の実験では，一回の反復で追加するラベル付きデータの数 N_{add} を 2000 と設定した．検証データでの正解率は，1 回目の反復で向上し，0.657 となったが，それ以降は変動はあるものの，この正解率を越えることはなかった．

表 3 は学習したモデルを用いてテストデータの因果関係を判定した結果である．2 行目の「正解率」はモデルによる因果関係の有無の判定が正解と一致した割合である．また，本実験のタスクは因果関係

表 3　因果関係推定モデルの評価

モデル		M_0	M_1	M_2	M_3
正解率		0.475	0.515	0.520	0.495
関係あり	精度	0.368	0.383	0.378	0.364
	再現率	0.725	0.667	0.609	0.623
	F 値	0.488	0.487	0.467	0.460
関係なし	精度	0.703	0.713	0.697	0.683
	再現率	0.344	0.435	0.473	0.427
	F 値	0.462	0.540	0.564	0.526

が成立するか否の二値分類問題なので，因果関係ありのクラスとなしのクラスのそれぞれについて，精度，再現率，F 値を調べた．正解率は反復回数が 2 のときに最大で，0.520 となった．初期データのみから学習したモデル M_0 と比べて 0.045 ポイント上昇したことから，訓練データの拡張は効果があることが確認された．しかし，正解率自体は 5 割程度であり，二者の判定の一致率 72%と比べても決して高くなく，改善が必要である．

「因果関係あり」クラスの F 値は反復が進むにつれて低下するが，「因果関係なし」クラスの F 値は反復回数 2 回までは向上する．このことから，訓練データの拡張は，因果関係が成立しない文の組に対して正しく判定ができるようになる効果が大きいと言える．

正解率のピークは，表 2 の検証データでは反復回数が 1 のとき，表 3 のテストデータでは反復回数が 2 のときと，一致していない．検証データは自動作成されたものであり，テストデータと性質が異なることが原因のひとつと考えられる．反復をいつ停止するかは重要な研究課題である．

5　おわりに

本論文は，人手によるアノテーションなしに文間の因果関係を推定する手法を提案した．ブートストラップ法によって自動獲得された訓練データが推定モデルの正解率の向上に寄与することを確認した．今後の課題として，初期データの正例の中には誤りも少なからず含まれているため，真に因果関係が成立する文の組をより正確に選別するルールを開発することが挙げられる．また，本研究では負例はランダムに文を組み合わせて作成したが，2 つの文は明らかに無関係であり，因果関係推定モデルの学習にどれだけ寄与するか疑問が残る．より適切な負例の作成方法の探究も今後の重要な課題である．

参考文献

[1]山田涼太. 自動獲得された因果関係知識に基づく文間の因果関係の推定. 修士論文, 北陸先端科学技術大学院大学, 3 2021.

[2]Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186, 2019.

[3]Canasai Kruengkrai, Kentaro Torisawa, Chikara Hashimoto, Julien Kloetzer, Jong-Hoon Oh, and Masahiro Tanaka. Improving event causality recognition with multiple background knowledge sources using multi-column convolutional neural networks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 3466–3473, 2017.

[4]Chikara Hashimoto, Kentaro Torisawa, Julien Kloetzer, Motoki Sano, István Varga, Jong-Hoon Oh, and Yutaka Kidawara. Toward future scenario generation: Extracting event causality exploiting semantic relation, context, and association features. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pp. 987–997, 2014.

[5]Patrick Pantel and Marco Pennacchiotti. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 113–120, 2006.

[6]Shuya Abe, Kentaro Inui, and Yuji Matsumoto. Acquiring event relation knowledge by learning cooccurrence patterns and fertilizing cooccurrence samples with verbal nouns. In *Proceedings of the Third International Joint Conference on Natural Language Processing*, pp. 497–504, 2008.

[7]工藤拓, 松本裕治. チャンキングの段階適用による日本語係り受け解析. 情報処理学会論文誌, Vol. 43, No. 6, pp. 1834–1842, 2002.

[8]BERT 日本語 pretrained モデル. http://nlp.ist.i.kyoto-u.ac.jp/index.php?ku_bert_japanese. (2020 年 12 月閲覧).