ディープラーニングによる文脈を考慮した論理構造推定手法の提案

平成 30 年 3 月 9 日

1631040 木下涼
kinoshita@ai.is.uec.ac.jp
平成 30 年 3 月 9 日

目 次

表 目 次

図 目 次

1　まえがき

論理構造を自動的に推定することは，自然言語処理において非常に重要なタスクの一つであり，文章中の論理構造を自動推定する論証マイニング (argument mining)[1, 2, 3, 4, 5] と呼ばれる技術が注目を集めている．論証マイニングでは，入力文に対して，どの部分が筆者の主張であり，その主張を支持・反証している部分はどこかを自動的に出力するため，情報検索 [6] や文章要約 [7]，自動採点 [8, 9, 10]，論文執筆支援 [11, 12] など数多くの自然言語処理タスクに応用されている．

論証マイニングにおける論理構造は論理要素 (argument discourse units) と呼ばれる論理に関連する節や文から成り立つ [13]．論理要素は筆者の立場を示す「主張」とそれを支持・反証する「前提」に分けられる．また，多くの論証マイニングの研究ではそれぞれの要素をノードとみなし，ノード間に有向エッジを引くことで親ノードが子ノードを支持・反証しているという因果関係 (argumentative relation) を示すことによって，グラフィカルモデルで論理構造を表している [1, 14]．

論証マイニングには，1) 文章中の論理構造に関係する文や節を抽出する要素抽出 (identifying argument components)，2) その要素が論理構造でどのような役割を果たしているか特定する要素分類 (classifying argument components) ，3) 各要素間の因果関係を特定する関係分類 (identifying argumentative relation)，4) 最終的な出力となる論理構造を決定する構造推定 (deciding argument structure) などのサブタスクが含まれている．これらのサブタスクに関する研究は数多く存在する [14, 15, 16, 17, 18, 19] が，全てのサブタスクを統合し，文章を入力として論理構造を出力する研究は限られている [1, 20]．

サブタスクを統合している代表的な手法として，Peldszus and Stede (2015)[20] が挙げられる．しかし，Peldszus and Stede(2015) の手法は局所的な分類のみを行っており，段落内全体を考慮して論理構造の推定を行っていないことが問題である．

一方，Stab and Gurevych(2017)[1] は，要素抽出，要素分類，関係分類

を機械学習手法を用いて解き，それらの結果を線形計画法を用いて統合することで構造推定を行い，論理構造を決定する手法を提案している．線形計画法により，段落内全体を考慮して，循環が存在せず，一つのノードから引かれる有向エッジは一本以下という制約を踏まえた論理構造の推定が可能になっている．しかし，この手法においても以下のような問題点が存在するため，最終的な論理構造の推定精度は低い．

1) 各要素が「主張」,「前提」のどちらであるかは要素単独で決まるわけではなく，文脈に応じて決定される．しかし，各論理要素が「主張」,「前提」のどちらであるか分類する要素分類において，文脈を考慮した特徴量が限られている．

2) 各要素が「主張」であるかは因果関係の有無と強い相関があると考えられる．しかし，要素分類と因果関係の有無を分類する関係分類を独立に解いている．

3) 最終的な論理構造の推定において，要素分類と関係分類の分類結果を結合する際に，線形計画法を用いているが，その目的関数において各結果に対する重みをチューニングパラメータとして恣意的に決定している．

本研究ではこれらの問題を解決するために，ディープラーニングの一種である Long-Short Term Memory(LSTM)[21] を用いて文脈を考慮した新たな論証マイニング手法を提案する．提案手法は，以下のような特徴を持つ．

・要素分類において，文脈を活用するために，広い文脈を扱うことのできる LSTM を用いた新たな特徴量を導入する．具体的には，要素分類における特徴量として，論理要素の前文も用いて学習した LSTM で文脈情報を圧縮して表現している隠れ層の値を用いる．

・要素分類と関係分類を独立したサブタスクとして扱わず，要素分類の結果を関係分類に用いる．本研究では，各要素が主張である確率推定値が特徴量を所与としたシグモイド関数によって表現されると仮定し，各要素が主張である確率推定値を関係分類の特徴量として採用する．

・構造推定では，因果関係の存在する確率推定値は特徴量と各要素が主

張である確率推定値を所与としたシグモイド関数によって表現されると仮定し，因果関係の存在する確率推定値を，線形計画問題の目的関数の重みとして採用する．

これらの特徴により，提案手法には以下のような利点が期待される．
1)LSTM を用いて文脈の情報を取り込むことにより，要素分類の精度が向上する．
2) 従来は独立に解かれていた要素分類と関係分類を，要素分類における結果を関係分類に用いることで，関係分類の精度が向上する．
3) 構造推定において用いられる線形計画問題の恣意的なチューニングパラメータを排除することができ，論理構造の推定精度の向上が期待できる．

ベンチマークコーパス [1] を用いた評価実験によって，提案手法では最先端の先行研究 [1] と比較して，要素分類における「主張」の分類精度が約７％，関係分類における「因果関係有」の分類精度が約８％向上することが示された．また，提案手法の構造推定では，要素分類の結果を適切に反映できることが示唆された．さらに，提案手法全体では，先行研究と比較して「因果関係有」の分類精度が約７％,「因果関係無」の分類精度が約２％高いことが明らかになった．

本論文は，以下の構成からなる．まず，本研究論理構造について第 2 章にまとめる．その後，第 3 章で論証マイニングの枠組みと，論証マイニングの最先端の研究であり，本研究との比較手法である Stab and Gurevych(2017)[1] について説明する．第４章では提案手法の概要を示し，第５章では比較実験によりその効果を明らかにする．最後に，第６章において本論文のまとめと今後の課題を示す．

2　論理構造の表現

本章では，本研究で用いる論理構造の表現手法であり，論理構造に関する研究 [1, 14] で多く用いられている Argument Diagramming について説明する．これは，論理構造を有向グラフ構造で表現するモデルである．それぞれのノードが論理に関係する文や節からなる論理要素を示し，各エッジは親ノードが子ノードを正当化していることを示している．

局所的な論理構造は大きく分けて図 1 の五つのパターンに分けられる．もっとも単純な形である Single Argument は一つの親ノードから一つの子ノードに有向エッジが引かれている構造である．Serial Argument は三つ以上のノードからなる分岐・合流のない連続した構造であり，Divergent Argument は子ノードが複数ある構造である [22]．Convergent Argument と Link Argument はともに親ノードが複数ある構造だが，Link Argument の親ノードは単体で子ノードを正当化することができない [23] という点で異なる．この二つの論理構造を区別すべきかについては多くの議論があるが [24]，実際の文章において二つの構造を正確に区別するのは困難な場合が多いため，区別しないことが多い．

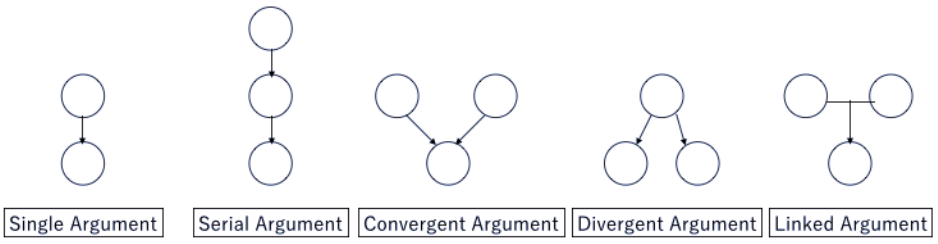

図 1: Argument Diagramming の種類

多くの研究では，論理構造を Divergent Argument が含まれず，循環のない形に制限している [25, 26]．Divergent Argument を含まない理由として，Divergent Argument は他の論理構造で表現できること [24]，あるコーパスでは，Divergent Argument は約 5 %ほどしか含まれていないという報告があることが挙げられる．

また，循環を含まない理由として，論理的な文章を書く際の一般的な過程では，ある主張に対しての根拠を連ねていくことから，論理構造に循環を含むことは不自然であることがあげられる．

これらの理由から，現在のベンチマークとなっているコーパス [1] においても，論理構造中に Divergent Argument や循環構造が存在しないことが仮定されている．本研究ではベンチマークであるコーパスを用いて，学習と評価を行うため，コーパスと同様に，論理構造中には Divergent Argument や循環構造が存在しないものと仮定する．

図 1 の局所的な構造を組み合わせることで，あらゆる論理構造をグラフィカルモデルによって表現でき，隣接していない要素との関係も表すことができるなど他のモデルの欠点を補うことができるモデルのため，現在の論理構造に関する研究で非常に多く用いられているモデルとなっている．そのため，本研究で提案する論証マイニング手法でも，論理構造を Argument Diagramming を用いて表現している．

3　論証マイニングの従来手法

3.1　論証マイニングの枠組み

本節では，本研究で扱う論証マイニングの枠組みについて述べる．

論証マイニングでは，一般に，文章の論理構造を段落ごとに解析する [1, 27]．例として，以下のような段落に対する論証マイニングの流れを示す．

> First of all, <1>*the* zoo gives humans new knowledge. Inevitably, <2>*the* zoo is the important place for innate learning. Besides, <3>*researchers* need the zoo to pick up some examples of animals for their researches because <4>*it* has diverse species of animals.

論証マイニングではこのような段落について，以下の四つのサブタスクを解くことによって，論理構造を推定する．

要素抽出　文章中から，論理構造の構成要素となる文または節を論理要素として抽出する．例では，<1>から<4>の 4 箇所を論理要素とみなす．多くの場合，"Beside"や"because"といった論理標識を論理要素に含めない [17, 18]．

要素分類　各論理要素を「主張」あるいは「前提」に分類する．「主張」は，論証における最終的な結論であり，「前提」は，データや例を示すことにより主張や他の前提の説得力を高めている．例では，波線部の<1>が「主張」であり，下線部の<2>から<4>は「前提」である [14, 16]．

関係分類　段落内の全ての論理要素間について因果関係の有無を推定する．ここでの因果関係とは，ある前提が主張や他の前提の内容を支持し，説得力を高めているような関係を示す．各論理要素をノードとみなし，要素 i から要素 j に因果関係があると分類された時，要素 i か

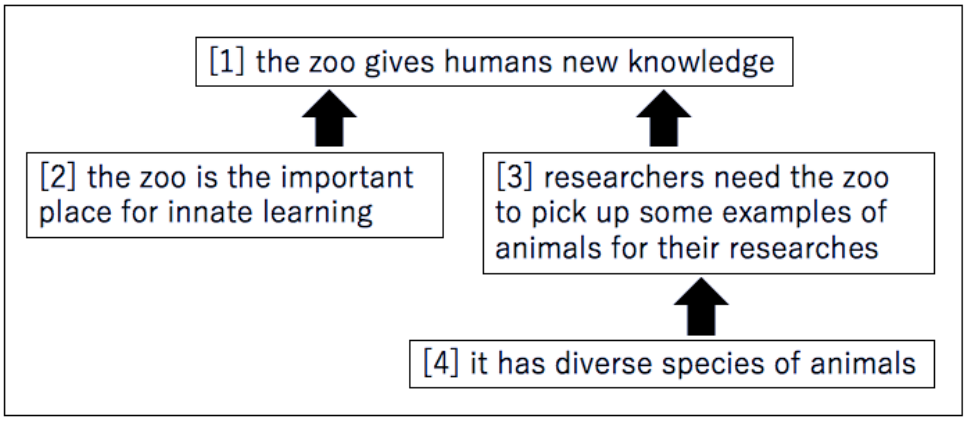

図 2: 論理構造の例

ら要素 j に有向エッジを引き，要素 i を要素 j の原因，要素 j を要素 i の結果と呼ぶ．例では，要素<1>に対して要素<2>と要素<3>からそれぞれエッジが引かれ，要素<3>に対して要素<4>からエッジが引かれる [15, 19]．

構造推定 構造推定では，線形計画法などを用いて 要素分類と関係分類の結果を統合し，制約を満たすような論理構造を推定する．論理構造には様々な種類が存在するが [13, 22, 28]，論証マイニングでは，主に，論理構造中のどの因果関係においても原因となっていない要素のみを「主張」とする one-claim approach[26] を採用している．one-claim approach では，循環が存在せず，各ノードから引かれるエッジは一本以下という制約のもと論理構造を決定する．例では図 2 のような論理構造が得られる．

次節では，論証マイニングの最先端の手法である Stab and Gurevych(2017)[1] について説明する．本研究では，Stab and Gurevych(2017) の作成したコーパスを用いて，学習・評価を行う．また，提案手法の評価実験における比較手法として，複数のサブタスクを統合して構造推定を行う手法の中で，最も高精度な Stab and Gurevych(2017) を用いている．

3.2 Stab and Gurevych(2017) の手法

Stab and Gurevych(2017)[1] は，現在の論証マイニング研究におけるベンチマークとなっているコーパスを作成し，複数のサブタスクを統合して構造推定を行う最先端の手法を提案している．

3.2.1 Argument Annotated Essays

Stab and Gurevych(2017) は，因果関係をタグ付けしたコーパスである"Argument Annotated Essays[1]"を公開している [1, 29]．このコーパスでは，大学生の書いた小論文形式の文章４０２編に対して３名のアノテーターが，1) 論理要素がどの部分か． 2) それぞれの論理要素が「主張」，「前提」のどちらであるか． 3) 各論理要素間について，因果関係の存在するのはどの組み合わせか．の３つをタグ付けしたデータから構成される．以下に，コーパスに含まれる文章を示し，その文章に対するタグの例を表 1 に示す．

原文

> First of all, the zoo gives humans new knowledge. Inevitably, the zoo is the important place for innate learning. Besides, researchers need the zoo to pick up some examples of animals for their researches because it has diverse species of animals.

表 1 の一列目における Ｔ １から Ｔ ４は論理要素タグを示しており，Ｒ １から Ｒ ３は因果関係タグを示している．論理要素タグがつけられた行は，二列目にその論理要素が「主張 (Claim)」あるいは「前提 (Premise)」のどちらであるかが示されており，三列目には論理要素は，文章中の何文字目から何文字目までに存在しているか示されている．そして，四列

[1]https://www.ukp.tu-darmstadt.de/data/argumentation-mining/argument-annotated-essays/

表 1: コーパス例

T1	Claim	16 50	the zoo gives humans new knowledge
T2	Premise	66 116	the zoo is the important place for innate learning
T3	Premise	137 218	researchers need the zoo to pick up some examples of animals for their researches
T4	Premise	236 269	it has diverse species of animals
R1	supports	T2	T1
R2	supports	T3	T1
R3	supports	T4	T3

目には論理要素に相当する原文がそのまま書かれている．一方，因果関係タグがつけられた行は，二列目にその因果関係は「支持 (supports)」関係であることが示されている．そして，三列目に示された要素が原因であり，四列目に示された要素が結果であるような因果関係が存在することが記されている．

コーパス全体では，論理要素 5338 個に対して「主張」が 1506,「前提」が 3832 タグ付けされており，因果関係は 3832 含まれている.. このコーパスは，現在の論証マイニング研究においてベンチマークとして一般に利用されている [19, 30].

3.2.2 論証マイニング手法

Stab and Gurevych(2017)[1] では，コーパスに含まれる文章から学習した分類器を用いて，図 3 のように四つのサブタスクを実行することで論理構造を推定している．まずはじめに，系列ラベリング手法を用いて，与えられた文章から論理に関係する文や節を論理要素として抽出し，抽出された論理要素に対して，独立に二つの分類を行う．一つ目は各論理要素が「主張」であるか,「前提」であるかの二値分類であり，二つ目は，

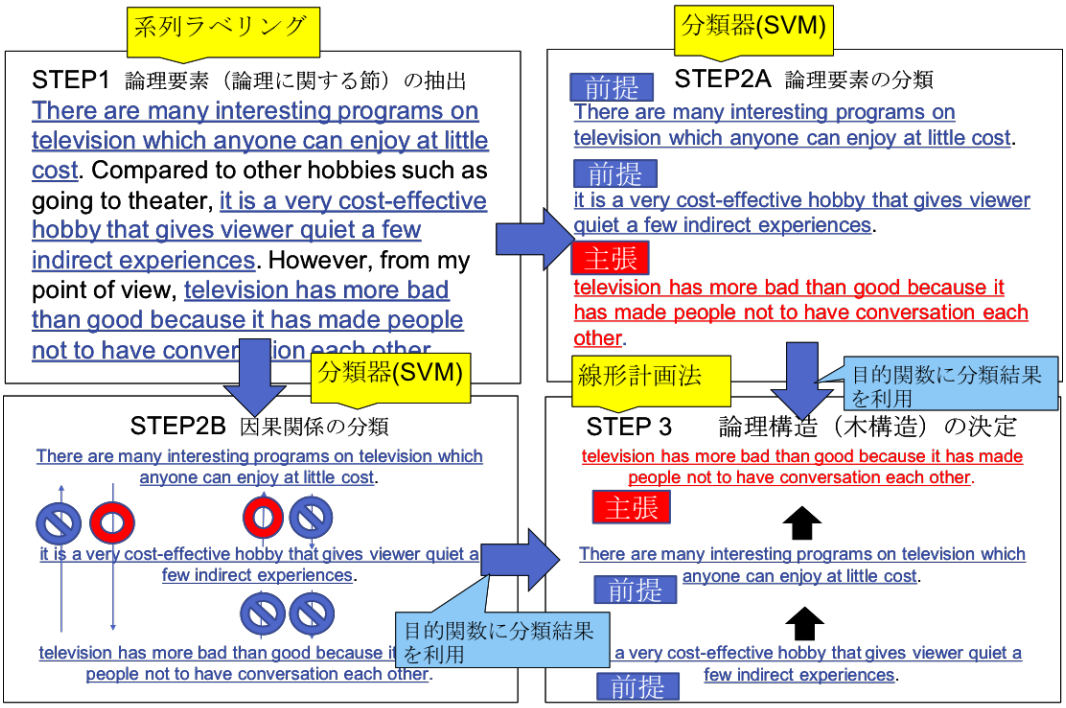

図 3: Stab and Gurevych(2017) の概要

各論理要素間に因果関係が存在するかどうかの二値分類である．これら二つの分類には SVM が用いられている．最後に二つの分類結果を線形計画法を用いて統合し，論理構造を推定する．

以下では，それぞれのサブタスクの内容を説明する．

STEP1 要素抽出

文章からの論理要素の抽出には，条件付き確率場 [31] に基づく系列ラベリング手法を用いている．ここでは，学習データとして，独自の IOB タグセットを付与したコーパスを用いる．具体的には，文章中のすべての単語に対して次の 3 種類のいずれかのタグを付与したデータである．

1)Arg-B:要素の最初のトークンを示すタグ

2)Arg-I:要素内のトークンを示すタグ

3)Arg-O:論理要素に関係しないトークンを示すタグ

Arg-B から Arg-I の連続が終わり，Arg-O または Arg-B が現れるまでが一つの論理要素となる．Arg-B の直後に Arg-I 以外のタグが付けられた場合，論理要素とはみなさない．コーパス中の IOB タグ付きデータから学習した条件付き確率場を用いて未知の文章にタグづけすることで，論理要素の抽出が行われる．このとき，条件付き確率場の説明変数として用いる特徴量を表 2 に示した．

表 2: 要素抽出に用いられている特徴量

特徴量グループ	特徴量
構造特徴	トークンが最初または最後の段落か
	文章内の最初，または最後の要素か
	文章・段落・文内の相対位置・絶対位置
	句読点が前後にあるか
	句読点であるか
	トークンの含まれる文の位置
統語論特徴	品詞
	構文木における共通祖先までの最短距離
	共通祖先の constituent types
語彙統語特徴	語彙特徴と統語論特徴の組み合わせ [32]
確率特徴	直前のトークンが与えられた時に要素の
	始まりである条件付き確率

STEP2A 要素分類

SVM を用い，各論理要素を「主張」あるいは「前提」に分類する．具体的には，個々の論理要素に対して表 3 に示した特徴量を算出する．この中で，要素の含まれている文よりも広い文脈を考慮しているものは存在しない．この特徴量ベクトルを説明変数，その要素が「主張」か「前提」かを目的変数とする SVM をコーパスから学習し，未知の文章の要素分類に用いる．

表 3: 要素分類に用いられている特徴量

特徴量グループ	特徴量
語彙特徴	単語ユニグラム
	単語バイグラム
構造特徴	トークン数（要素内，文内，段落内）
	段落内の最初，または最後の要素であるか
	段落内の相対位置
指示語特徴	順接，逆説，仮定，譲歩を表す指示語が
	含まれているか
共通特徴	第一段落，あるいは最終段落と共通する
	名詞・動詞の数
統語論特徴	文に含まれる従属節の数
	構造木の深さ
	主動詞の時制
	法助詞が存在するか
	品詞分布
確率特徴	トークン p が存在する時にそれぞれの
	要素である条件付き確率
論理特徴	PDTB スタイル [33] による特徴量
分散表現	単語分散表現 [34, 35] の和

STEP2B 関係分類

SVM を用いて，段落内の要素全ての組み合わせについて，因果関係の有無を分類する．具体的には，各論理要素ペアについて，表 4 に示した特徴量を算出する．この特徴量ベクトルを説明変数，因果の有無を目的変数とする SVM をコーパスから学習し，未知の文章の関係分類に用いる．関係分類を行う際に，要素分類の結果は考慮していない．

表 4: 関係分類に用いられている特徴量

特徴量グループ	特徴量
語彙特徴	単語ユニグラム
構造特徴	トークン数（要素・段落内） 2 要素間に存在する要素数 2 要素が同じ文内に存在するか どちらの要素が先に出現しているか 段落内の最初，あるいは最後の要素か
統語論特徴	構文木における頻出生成規則 [2] 品詞分布
指示語特徴	順接，逆説，仮定，譲歩を表す指示語が 要素あるいは要素間に含まれているか
共通特徴	第一段落，あるいは最終段落と共通する
論理特徴	PDTB スタイル [33] による特徴量
自己相互情報量	エッジの方向と単語の自己相互情報量
単語共通	2 要素が共有している単語数

STEP3 構造推定

要素分類と関係分類の結果を統合し，論理構造を決定する．論理構造の決定は線形計画問題として以下のように定式化されている．

$$\arg\max_{x} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} x_{ij} \tag{1}$$

s.t.

$$\forall i : \sum_{j=1}^{n} x_{ij} \leq 1 \tag{2}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij} \leq n - 1 \tag{3}$$

$$\forall i : x_{ii} = 0 \tag{4}$$

$$\forall i \forall j : x_{ij} - a_{ij} \leq 1 \tag{5}$$

$$\forall i \forall j \forall k : a_{ik} - a_{ij} - a_{jk} \leq -1 \tag{6}$$

$$\forall i : a_{ii} = 0 \tag{7}$$

ここで，$x_{ij} \in \{0, 1\}$ は要素 i から要素 j に因果関係があるとき 1，そうでなければ 0 をとる変数であり，循環が存在せず，一つのノードから引かれるエッジが一本のみであるような木構造を保証する制約式 (2) から (7) を満たすように，値を決定する．式 (5) から (7) で利用している a_{ij} は循環構造を防ぐために導入されており，要素 i から要素 j に可到達である時 $a_{ij} = 1$，そうでないときに $a_{ij} = 0$ をとる．

ここで目的関数の係数 w_{ij} は要素分類と関係分類の結果を用いた重みであり，式 (8) のように定義されている．

$$w_{ij} = \phi_r r_{ij} + \phi_{cr} cr_{ij} + \phi_c c_{ij}. \tag{8}$$

r_{ij} は関係分類の結果によって式 (9) のように決定される 2 値変数である．

$$r_{ij} = \begin{cases} 1 & (\text{要素iが要素jの原因であると分類された}) \\ 0 & (\text{要素iが要素jの原因でないと分類された}) \end{cases} \tag{9}$$

同様に，cr_{ij} も関係分類の結果を反映しており，その値は式 (10),(11) から算出される．

$$cr_{ij} = cs_j - cs_i \tag{10}$$

$$cs_i = \frac{relin_i - relout_i + n - 1}{rel + n - 1} \tag{11}$$

$relin_i$ は関係分類において因果関係ありと分類された 2 要素間の内，要素 i が結果である因果関係の数であり，$relout_i$ は要素 i が原因である因果関係の数である． また，rel は関係分類で因果関係ありと分類された全ての因果関係の数であり，n は段落内の要素数である．つまり，関係分類によって，要素 i が多くの因果関係の原因であると分類され，要素 j が多くの因果関係の結果であると分類されている場合に cr_{ij} は大きな値となる．

一方，c_{ij} は要素分類の結果を反映しており，式 (12) で決定される 2 値変数である．

$$c_{ij} = \begin{cases} 1 & (\text{要素jが「主張」であると分類された}) \\ 0 & (\text{要素jが「前提」であると分類された}) \end{cases} \tag{12}$$

従って，w_{ij} が大きくなるほど，要素 i から要素 j にエッジが引かれやすくなるように定式化されている．

また， ϕ_r，ϕ_{cr}，ϕ_c はそれぞれの値の重みを決定するチューニングパラメータであり， $\phi_r = 1/2$ ， $\phi_{cr} = \phi_c = 1/4$ の値が恣意的に決められている．

Stab and Gurevych(2017) の手法 [1] を用いても最終的な構造推定の精度はそれほど高くない．その原因として，以下の問題点が考えられる．
1) 各要素が主張か前提かは前後の文脈に強く依存すると考えられるが，この手法では，要素分類に，前後の文脈を活用した特徴量が存在しない．
2) 各要素が「主張」であるかは因果関係の有無と強い相関があると考えられる．しかし，要素分類と関係分類の結果を独立に解いている．
3) 構造推定の (8) 式における ϕ_r，ϕ_{cr}，ϕ_c の三つのチューニングパラメータの値があらかじめ決められている．これらの恣意的に決定されたパラメータが強く影響している．

本研究ではこれらの問題を解決するために，以下のような特徴を持つ新たな論証マイニング手法を提案する．

・要素分類の特徴量として，LSTM により対象要素と前文の情報を統合した隠れ層の値を用いる．

・要素分類と関係分類を独立したサブタスクとして扱わず，要素分類の結果を関係分類に用いる．本研究では，各要素が主張である確率推定値は要素分類において特徴量を入力としたシグモイド関数によって表現されると仮定し，各要素が主張である確率推定値を関係分類の特徴量として採用する．

・構造推定では，各要素間に因果関係の存在する確率推定値は特徴量を入力としたシグモイド関数によって表現されると仮定し，各要素間に因果関係の存在する確率推定値を，線形計画問題の目的関数の重みとして採用する．

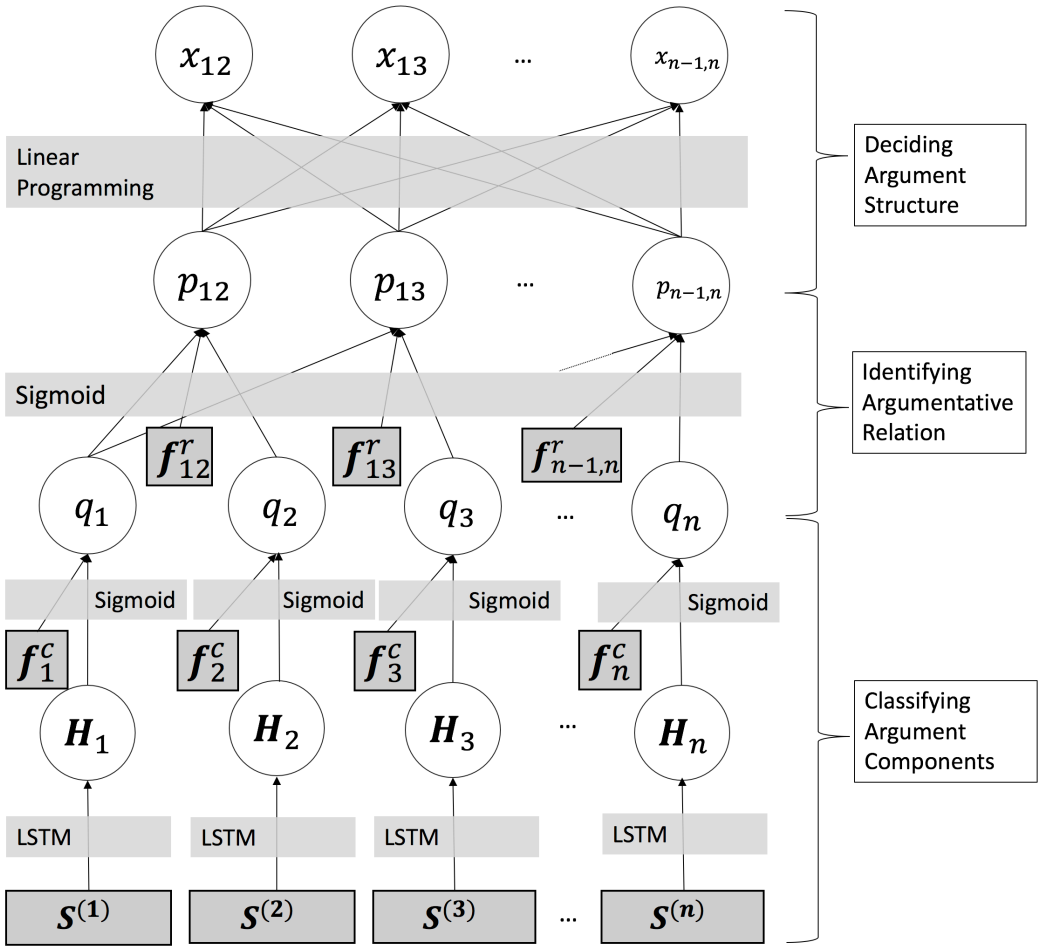

図 4: 提案手法の概要

4　提案手法

4.1　提案手法の流れ

本研究では，LSTM を用いて文脈を考慮した新たな論証マイニング手法を提案する．要素数が n 個の段落に対する提案手法の概要を図 4 に示す．なお，本研究では，比較的高精度に解くことのできる要素抽出に関しては問題としない．

図中の $\boldsymbol{S}^{(i)} = (S_1^{(i)}, S_2^{(i)}, ..., S_l^{(i)})$ は，段落中の i 番目の論理要素とその前後の文の単語ベクトルであり，次元数は論理要素とその前後の文を合

わせた単語数 l となる．

$\boldsymbol{H}_i$ は $\boldsymbol{S}^{(i)}$ を入力とした時の LSTM の隠れ層のベクトル $\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_l$ から計算され，本研究では以下の二種類の値を用いる．なお，$\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_l$，$\boldsymbol{H}_i$ の次元数は，任意の値に決定することができる．

$$\boldsymbol{H}_i^{ave} = \frac{1}{l}\sum_{k=1}^{l} \boldsymbol{h}_k \tag{13}$$

$$\boldsymbol{H}_i^{last} = \boldsymbol{h}_l \tag{14}$$

式 (13) の $\boldsymbol{H}_i^{ave}$ は，$\boldsymbol{S}^{(i)}$ を LSTM の入力とした時に出力される全ての隠れ層 $\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_l$ の平均値のベクトルであり，式 (14) の $\boldsymbol{H}_i^{last}$ は $\boldsymbol{S}^{(i)}$ の最後の要素である $\boldsymbol{S}_l^{(i)}$ を入力とした時に出力される LSTM の隠れ層のベクトル $\boldsymbol{h}_l$ である．

$\boldsymbol{f}_i^c$ は i 番目の要素から抽出される表 3 の特徴量全てを並べたベクトルを示しており，$\boldsymbol{f}_{ij}^r$ は要素 i が要素 j の原因とした時に抽出される表 4 の特徴量全てを並べたベクトルである．ここで，$\boldsymbol{f}_i^c$ の次元数は 86, $\boldsymbol{f}_{ij}^r$ の次元数は 57 となる．

本研究では，要素 i が「主張」である確率推定値 q_i が，$\boldsymbol{H}_i$ と $\boldsymbol{f}_i^c$ を所与としたシグモイド関数 (15) で表現できると仮定する．$\boldsymbol{w}^{(1)} = (w_1^{(1)}, w_2^{(1)}, ..., w_{|\boldsymbol{f}_i^c|}^{(1)})^T$，$\boldsymbol{w}^{(2)} = (w_1^{(2)}, w_2^{(2)}, ...w_{|\boldsymbol{H}_i|}^{(2)})^T$ は，訓練データから学習される重みベクトルである．

$$q_i = \frac{1}{1 + \exp(-\boldsymbol{w^{(1)}}\boldsymbol{f_i^c} - \boldsymbol{w^{(2)}}\boldsymbol{H_i})} \tag{15}$$

同様に，要素 i が要素 j の原因である確率推定値 p_{ij} が，q_i，q_j，$\boldsymbol{f}_{ij}^r$ を所与としたシグモイド関数 (16) で表現できると仮定する．$\boldsymbol{w}^{(3)} = (w_1^{(3)}, w_2^{(3)}, ..., w_{|\boldsymbol{f}_{ij}^r|}^{(3)})^T$ は，訓練データから学習される重みベクトルであり，$w^{(4)}, w^{(5)}$ は訓練データから学習される重みスカラーである．

$$p_{ij} = \frac{1}{1 + \exp(-\boldsymbol{w^{(3)}} \boldsymbol{f_{ij}^r} - w^{(4)} q_i - w^{(5)} q_j)} \tag{16}$$

提案手法では，まず要素 i とその前後の文章を LSTM の入力として得られた $\boldsymbol{H}_i$ を既存の特徴量 $\boldsymbol{f}_i^c$ と組み合わせて要素 i が主張である確率推定値 q_i と要素 j が主張である確率推定値 q_j を求める．次に，確率推定値 q_i, q_j を既存の特徴量 $\boldsymbol{f}_{ij}^r$ と組み合わせて，要素 i が要素 j の原因である確率推定値 p_{ij} を求める．最後に，p_{ij} を目的関数とする線形計画法を用いて，循環が存在せず，一つのノードから引かれるエッジは一本以下という制約のもと，全ての要素間について式 (17) の x_{ij} の値を決定し，構造を推定する．

$$x_{ij} = \begin{cases} 1 & (\text{要素iが要素jの原因である}) \\ 0 & (\text{要素jが要素jの原因でない}) \end{cases} \tag{17}$$

以降では，図 4 の各段階について詳細を説明する．

4.2 要素分類 (Classifying Argument Components)

本研究では，要素分類 (Classifying Argument Components) の特徴量として，既存の特徴量に加えて，文脈を考慮できるディープラーニング手法として知られる LSTM の隠れ層を用いる．LSTM は，リカレントニューラルネットワーク (RNN) の一種であり，時系列データに対するディープラーニング手法として自然言語処理分野を中心に広く活用がなされている．近年，文書分類においてディープラーニングの隠れ層を用いる手法が広く活用されており [36, 37]，分類精度の向上に有効であることが示されてきた．LSTM の隠れ層は，長期の依存関係を反映したものであり，文脈の情報が保持されているとみなせる．

LSTM

LSTM は，RNN の学習時に入力長が長くなると指数関数的に勾配が小さくなってしまう勾配消失問題 [38] を解決するために，長期依存を保存しているセルとゲートベクトルが導入されている．時点 t における LSTM の内部構造を図 5 に示す．具体的には，$\boldsymbol{S}^{(i)}$ の時点 t における入力 $\boldsymbol{S}_t^{(i)}$ と時点 $t-1$ の隠れ層 $\boldsymbol{h}_{t-1}$ を用い，式 (18), (19) で忘却ゲート $\boldsymbol{d}_t$，入力ゲート $\boldsymbol{i}_t$ の値を求め，式 (21) のようにセル $\boldsymbol{c}_t$ を更新する．忘却ゲート $\boldsymbol{d}_t$ は，セルが保存している長期依存をどのくらい維持するかを調節し，入力ゲート $\boldsymbol{i}_t$ は，新たにセルに保存する値を調整している．これらのゲートベクトルの導入により $\boldsymbol{c}_t$ に保存する情報と削除する情報を制御することが可能となり，長期依存を扱うことが可能になっている．

さらに，式 (20) で計算される出力ゲート $\boldsymbol{o}_t$ とセル $\boldsymbol{c}_t$ を用いて，式 (22) のように時点 t における出力となる隠れ層 $\boldsymbol{h}_t$ を計算する．

$$\boldsymbol{d}_t = \sigma_g(\boldsymbol{W}_f \boldsymbol{S}_t^{(i)} + \boldsymbol{U}_f \boldsymbol{h}_{t-1} + \boldsymbol{b}_f) \tag{18}$$

$$\boldsymbol{i}_t = \sigma_g(\boldsymbol{W}_i \boldsymbol{S}_t^{(i)} + \boldsymbol{U}_i \boldsymbol{h}_{t-1} + \boldsymbol{b}_i) \tag{19}$$

$$\boldsymbol{o}_t = \sigma_g(\boldsymbol{W}_o \boldsymbol{S}_t^{(i)} + \boldsymbol{U}_o \boldsymbol{h}_{t-1} + \boldsymbol{b}_o) \tag{20}$$

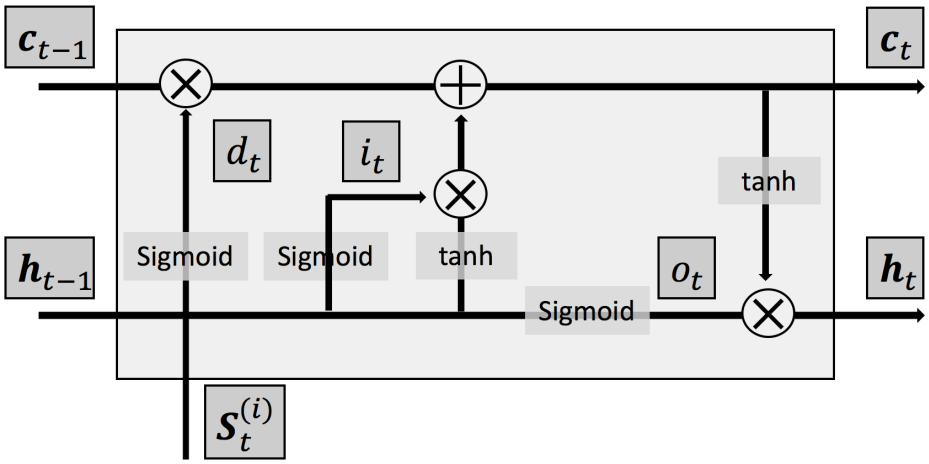

図 5: LSTM の内部構造

$$\boldsymbol{c}_t = \boldsymbol{d}_t \circ \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \circ \tanh(\boldsymbol{W}_c \boldsymbol{S}_t^{(i)} + \boldsymbol{U}_c \boldsymbol{h}_{t-1} + \boldsymbol{b}_c) \tag{21}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \circ \tanh(\boldsymbol{c}_t) \tag{22}$$

ここで，$\boldsymbol{W}_f, \boldsymbol{W}_i, \boldsymbol{W}_o, \boldsymbol{W}_c$ は入力 $\boldsymbol{S}_t^{(i)}$ に対する重みベクトル，$\boldsymbol{U}_f, \boldsymbol{U}_i, \boldsymbol{U}_o, \boldsymbol{U}_c$ は隠れ層 $\boldsymbol{h}_{t-1}$ に対する重みベクトル，$\boldsymbol{b}_f, \boldsymbol{b}_i, \boldsymbol{b}_o, \boldsymbol{b}_c$ はそれぞれのバイアスベクトルである．これらの次元数は，任意に設定する隠れ層の次元数と等しく，全て同時に学習される．σ_g は式 (23) に示すシグモイド関数であり，tanh は式 (24) に示すハイパボリックタンジェントである．ここで，$\circ$ はアダマール積を表している．

$$\sigma_g(x) = \frac{1}{1 + \exp(-x)} \tag{23}$$

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \tag{24}$$

本研究では，LSTM の実装にニューラルネットワークのフレームワークの一つである Chainer[2]を用いた．LSTM の入力 $S_t^{(i)}$ は論理要素 i 及びその文脈（前後の文章）における t 番目の単語である．学習時のエポック数は 70 に固定し，パラメータの最適化アルゴリズムには adaptive moment estimation(Adam)[39] を用いた．

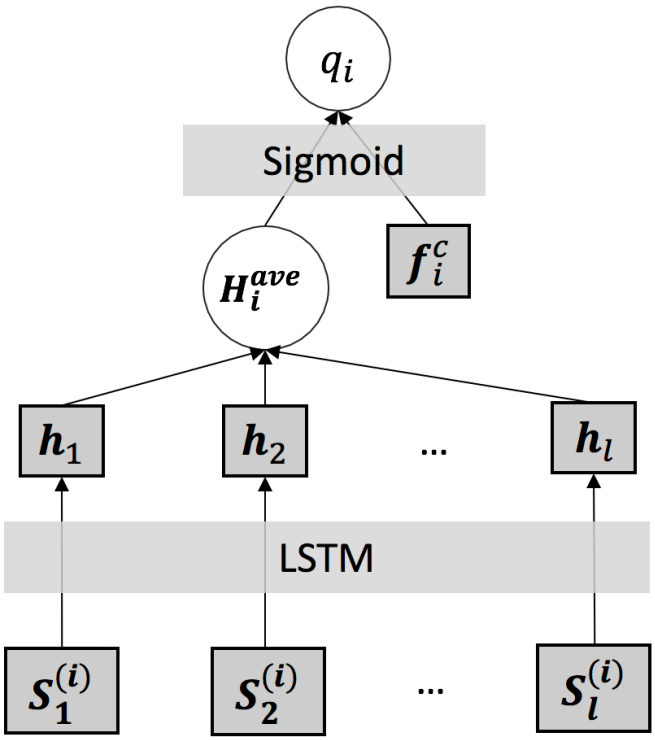

図 6: Average Pooling を用いた要素分類

LSTM の隠れ層を活用した分類問題では，全ての隠れ層の値の平均である Average Pooling や最後の隠れ層の値が用いられている [36]．そのため，提案手法においても，LSTM の最後の隠れ層ベクトルを特徴量として用いる手法と Average Pooling を特徴量として用いる手法の二つを比較する．また，LSTM を用いた特徴量以外の特徴量は，表 3 に示した先行研究と同じ $\boldsymbol{f}_i^c$ を使用する．

提案手法における要素 i に対する要素分類の概要を図 6，図 7 に示す．

図 6 は特徴量に式 (13) の $\boldsymbol{H}_i^{ave}$ の値を用いた場合の要素分類の概要である．入力 $\boldsymbol{S}^{(i)}$ に対して LSTM が出力する全ての隠れ層の値を平均した Average Pooling を $\boldsymbol{H}_i^{ave}$ とし既存の特徴量 $\boldsymbol{f}_i^c$ と組み合わせ，式 (15) のよ

[2]https://chainer.org/

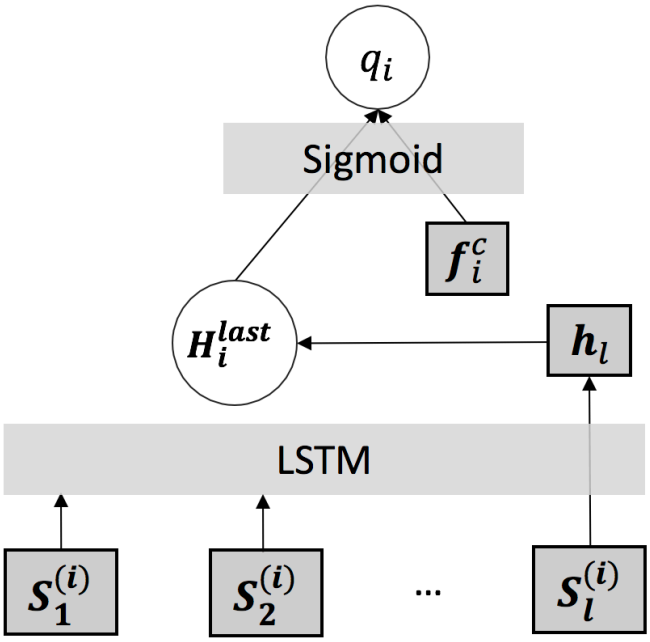

図 7: 最後の隠れ層を用いた要素分類

うに要素 i が「主張」である確率推定値 q_i を出力している．

一方，図 7 は特徴量に式 (14) の $\boldsymbol{H}_i^{last}$ の値を用いた場合の要素分類の概要である．$\boldsymbol{S}^{(i)}$ の最後の要素である $S_l^{(i)}$ を入力とした時に出力される隠れ層 $\boldsymbol{h}_l$ をそのまま $\boldsymbol{H}_i^{last}$ とし，既存の特徴量 $\boldsymbol{f}_i^c$ と組み合わせ，式 (15) のように要素 i が主張である確率推定値 q_i を出力している．

4.3 関係分類 (Identifying Argumentative Relation)

各要素が「主張」であるかは，因果関係の有無と強い相関があると考えられる．特に，ある要素が「主張」であれば，その要素は因果関係の結果になる可能性が高いことが予想される．そこで，提案手法では，要素分類の出力である，要素 i が「主張」である確率推定値 q_i と要素 j が「主張」である確率推定値 q_j を関係分類の特徴量に加える．

提案手法における，関係分類 (Identifying Argumentative Relation) の概要を図８に示す．$\boldsymbol{f}_{ij}^{r}$ は要素 i が要素 j の原因とした時に抽出される表４の特徴量全てを並べたベクトルである．

提案手法の関係分類では，q_i,q_j と $\boldsymbol{f}_{ij}^{r}$ から，式 (16) により，要素 i が要素 j の原因である確率推定値 p_{ij} を求め，出力とする．

要素分類の結果を関係分類の特徴量として用いているため，ある要素が「主張」であるかを考慮した上で，因果関係の有無を分類することができる．したがって,「主張」であれば因果関係の結果になりやすいといった相関関係を考慮することができると考えられる．

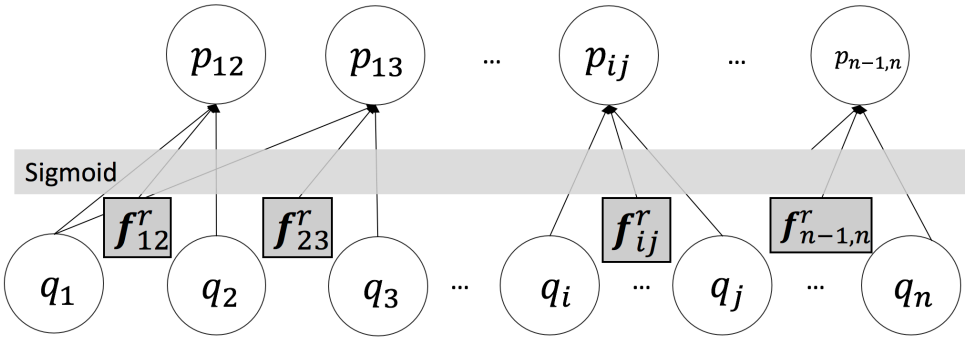

図 8: 関係分類の概要

4.4 構造推定 (Deciding Argument Structure)

提案手法では，構造推定を以下のような線形計画問題として定義する．

$$\arg\max_{x} \sum_{i=1}^{n}\sum_{j=1}^{n} p_{ij}x_{ij} \tag{25}$$

$s.t.$

$$\forall i : \sum_{j=1}^{n} x_{ij} \leq 1 \tag{26}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} \leq n-1 \tag{27}$$

$$\forall i : x_{ii} = 0 \tag{28}$$

$$\forall i \forall j : x_{ij} - a_{ij} \leq 1 \tag{29}$$

$$\forall i \forall j \forall k : a_{ik} - a_{ij} - a_{jk} \leq -1 \tag{30}$$

$$\forall i : a_{ii} = 0 \tag{31}$$

提案手法では式 (25) のように要素 i が要素 j の原因である確率推定値 p_{ij} を重みとする目的関数を設定し，線形計画法を用いて，制約を満たした上で，式 (17) で表される x_{ij} の値を決定することで最終的な論理構造を決定する．この問題は，図 9 に示したように論理要素 $T_1, T_2, ..., T_i, ..., T_n$ 間の全てに確率推定値 p_{ij} を重みとする有向エッジが引かれたグラフを入力として，制約のもとエッジの重み和が最大になるグラフを求める問題と解釈できる．ここでは，恣意的なチューニングパラメータを用いずに構造推定を行うことができる．

制約である (26) から (31) 式は先行研究 [1] と同様であり，循環がなく，一つのノードから引かれるエッジは一本以下である木構造を保証している．式 (29) から (31) で利用している a_{ij} は循環構造を防ぐために導入されており，式 (32) のように決定される．

$$a_{ij} = \begin{cases} 1 & (\text{要素iから要素jに可到達である}) \\ 0 & (\text{要素iから要素jに可到達でない}) \end{cases} \tag{32}$$

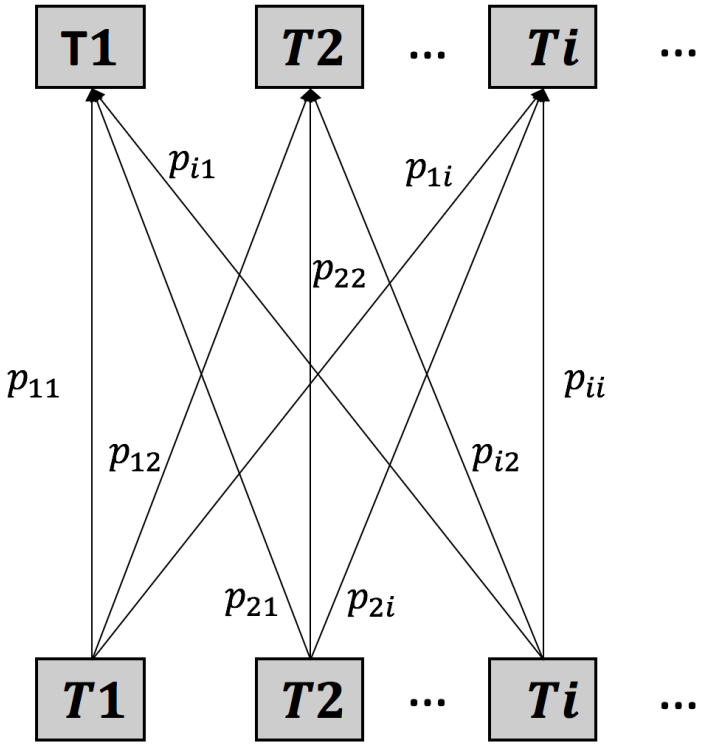

図 9: 構造推定の概念図

5　評価実験

本章では，提案手法が各サブタスクにおいて，既存手法より高い分類精度であることを評価実験によって示す．また，各サブタスクを統合した提案手法が既存手法よりも高精度に論理構造の推定を行うことを明らかにする．評価実験においては，Argument Annotated Essays のうち，データのタグ付けに誤りが含まれる 2 文書を除いた４００文書を用い，10 交差検証を行って精度 (precision) と再現率 (recall) の調和平均である F 値とそのマクロ平均 [40] を求めた．それぞれのサブタスクにおける比較では比較対象である Stab and Gurevych(2017)[1](以降では Stab2017 と表記する) と統一し SVM を用いて分類している．

5.1　要素分類の評価実験

まずはじめに，要素分類に LSTM による特徴量を用いる効果を評価する．LSTM への入力は，対象となる要素のみ（S）に加え，文脈を利用するために，対象となる要素とその前文 1 文,（Pf1)，その前文 2 文 (Pf2），要素以前に存在する段落内の文全て (PfAll)，対象となる要素とその次文 1 文（Pb1)，その次文 2 文 (Pb2)，要素以降に存在する段落内の文全て (PbAll) の 6 パターンを用いた．次文の文脈を用いる際は，より文脈の情報を考慮できるように単語列の順序を逆転した上で入力とした．隠れ層の次元数はそれぞれ，50・100・200 次元で比較している．Stab2017 を再現したものと LSTM による特徴量を用いた提案手法に関して，要素分類の結果を比較した結果を表 5 に示す．次元は隠れ層の次元数を示しており，F-Claim は「主張」の分類についての F 値，F-Premise は「前提」の分類についての F 値である．

表 5 より，分類対象となる要素を LSTM の入力とするだけでは精度はほぼ変わらず，要素とその前文 1 文を含めて LSTM の入力とすることで「主張」,「前提」のどちらの分類に対しても既存の精度を上回ることがわかる．さらに，隠れ層を比較的高次元に設定した LSTM の最後の層にお

表 5: LSTM を用いた比較実験結果

	入力	次元	F-Claim	F-Premise
Stab2017			0.627	0.832
提案：Average	Average_S	50	0.623	0.836
		100	0.623	0.838
		200	0.622	0.833
	Average_Pf1	50	0.663	0.853
		100	0.671	0.858
		200	0.679	0.860
	Average_Pf2	50	0.610	0.823
		100	0.611	0.824
		200	0.605	0.825
	Average_PfAll	50	0.617	0.824
		100	0.616	0.826
		200	0.613	0.817
	Average_Pb1	50	0.613	0.835
		100	0.612	0.836
		200	0.599	0.831
	Average_Pb2	50	0.609	0.830
		100	0.606	0.829
		200	0606	0.827
	Average_PbAll	50	0.623	0.833
		100	0.615	0.825
		200	0.608	0.828

	入力	次元	F-Claim	F-Premise
提案：Last	Last_S	50	0.622	0.838
		100	0.631	0.849
		200	0.648	0.855
	Last_Pf1	50	0.608	0.832
		100	0.642	0.849
		200	**0.698**	**0.875**
	Last_Pf2	50	0.618	0.823
		100	0.601	0.831
		200	0.601	0.834
	Last_PfAll	50	0.614	0.831
		100	0.617	0.836
		200	0.606	0.833
	Last_Pb1	50	0.615	0.834
		100	0.612	0.836
		200	0.598	0.834
	Last_Pb2	50	0.613	0.835
		200	0.601	0.832
		200	0.596	0.833
	Last_PbAll	50	0.613	0.835
		100	0.612	0.836
		200	0.599	0.831

ける値を用いると特に高い精度となることが明らかとなった．

一方で，前後 2 文以上を入力とすると，精度は大きく低下する傾向にあることがわかった．これは言語モデルにおける LSTM の先行研究 [41] で報告されているように，LSTM を用いても 3 文以上の入力に対しては学習が難しくなることが原因と考えられる．

また，今回の結果からは，次文の文脈を考慮することによる分類精度の改善はみられなかった．この原因として，今回のコーパスでは，要素と次文 1 文の単語数の平均が要素と前文 1 文の単語数の平均よりも１０単語以上多く，前後 2 文以上を入力とした時と同様に適切に学習できなかった可能性が考えられる．

今回の実験のうち最も高精度であった Last_Pf1 の次元数２００では，Stab2017 と比較して,「主張」の分類精度を 0.627 から 0.698 まで向上させており,「前提」の分類精度に関しても 0.832 から 0.875 まで精度が向上している．よって，論理要素とその前文 1 文を入力として，隠れ層の次

元を２００に設定したLSTMにおける最後の隠れ層を用いることで文脈の情報を適切に活用できることがわかった．

5.2 関係分類の評価実験

次に，関係分類に要素iが主張である確率推定値q_iを用いる効果を評価する．ここでは，q_iを関係分類の特徴量に加えた場合(Probability_LSTM)の関係分類の精度を求めた．また，q_iを用いずに関係分類を独立して行ったもの(Stab2017)，Stab2017と同じ特徴量$\boldsymbol{f}_i^c$のみを用いて，q_iを求め，特徴量に加えたもの(Probability_Stab)，各要素が「主張」であるか否かにコーパス中の真の値を用いたもの(True)と比較している．これらの結果を表6に示す．F-Relationは「因果関係有」分類のF値，F-No Relationは「因果関係無」分類のF値，F-Allは「因果関係有」と「因果関係無」に関するF値の平均である．

表6より，Stab2017では「因果関係有」の分類精度が0.428に対して，Probability_Stabでは0.460と，関係分類に要素iが主張である確率推定値q_iを用いると精度が大きく向上していることがわかる．また,「因果関係無」の分類精度に関しても0.731から0.791と向上していることが明らかとなった．さらに，Probability_LSTMにおける「因果関係有」の分類精度は0.507,「因果関係無」の分類精度は0.828と最も高いことから，提案手法における要素分類の精度向上が適切に反映されていることが示された．

表 6: 関係分類の評価実験結果

	F-Relation	F-noRellation
Stab2017	0.428	0.731
Probability_Stab	0.460	0.794
Probability_LSTM	**0.507**	**0.828**
True	0.708	0.923

5.3 提案手法全体の評価実験

本節では，提案する構造推定がStab2017よりも適切な重みに基づく構造推定であり，提案手法が既存手法よりも高精度な論証マイニング手法であることを比較実験によって示す．ここではStab2017と同じ特徴量を用いた提案手法 (提案手法-LSTM) とStab2017の構造推定を比較した．また，Stab2017の特徴量にLSTMを加えた手法 (Stab2017 + LSTM) と提案手法についても同様に精度を求めた．

提案手法では今までの結果を踏まえて以下のような値を用いている．
1)LSTMの入力には，論理要素と論理要素の前文1文を用い，隠れ層は200次元に設定し，最後の隠れ層の値のみを要素分類の特徴量として用いる．
2) 要素分類で，要素 i が主張である確率推定値 q_i を求め，その値を関係分類の特徴量として用いる．

ここで，提案手法の精度について，各要素間の因果関係有無のF値を算出した結果を表7に示す．F-Relationは「因果関係有」分類のF値，F-No Relationは「因果関係無」分類のF値，F-Allは「因果関係有」と「因果関係無」に関するF値の平均である．

まず，構造推定の有効性について検討する．表7から，既存の特徴量を用いた場合，提案手法よりもStab2017の精度が高いことがわかる．この原因として，Stab2017の構造推定が要素分類の影響を受けにくい定式化であるため，既存の特徴量では精度の低い「主張」の分類の影響が小さいことが考えられる．その証拠として，Stab2017 + LSTMと提案手法を比較した場合，Stab2017+LSTMでは「因果関係有」の分類精度が0.466なのに対し提案手法では0.539と，提案手法の精度が大きく上回ることがあげられる．従って，提案手法はより要素分類の結果の影響の大きい手法であり，提案手法の特徴量を用いる場合，提案手法の構造推定の方が大きく精度が向上することが明らかとなった．

最後に，提案手法全体の評価を行う．表7より，Stab2017では因果関係ありの分類精度は0.466であるのに対して，提案手法では0.539と大幅

表 7: 提案手法の評価実験結果

	F-Relation	F- No Relation	F-All
Stab2017	0.466	0.881	0.634
提案手法 - LSTM	0.451	0.879	0.627
Stab2017 + LSTM	0.466	0.881	0.638
提案手法	**0.539**	**0.897**	**0.718**

に精度が向上していることがわかる．また既に精度の高かった因果関係なしの分類精度に関しても，Stab2017 の 0.881 から 0.897 と精度が向上している．精度の低さが問題となっていた因果関係ありの分類精度が大きく向上しているだけでなく，既存手法でも精度の高い因果関係なしの分類精度も向上していることから，提案手法によって論理構造をより高精度に推定できることが明らかとなった．

6　むすび

本論文では，文脈を考慮することのできるLSTMを用いた新たな論証マイニング手法を提案した．具体的には，以下のような手順で論証マイニングを行う．

1）論理に関わる論理要素を「主張」か「前提」であるか分類する要素分類において，LSTMの入力を論理要素とその前文1文にした上で得られた最後の隠れ層の値を既存の特徴量と組み合わせて用いる．本研究では，各要素が主張である確率推定値 q_i が特徴量を所与としたシグモイド関数によって表現されると仮定し，推定された値を関係分類の特徴量として採用する．

2）各論理要素間に因果関係があるかどうか分類する関係分類において，要素分類で推定した q_i, q_j を既存の特徴量と組み合わせて用いる．因果関係の存在する確率推定値 p_{ij} が特徴量と q_i, q_j を所与としたシグモイド関数によって表現されると仮定し，推定された値を，線形計画問題の目的関数の重みとして採用する．

3) 最終的な論理構造の推定では，関係分類で推定された p_{ij} を線形計画問題の目的関数の重みとして用いることで，恣意的なチューニングパラメータのない線形計画問題として定式化した．

コーパスを用いた評価実験により，要素分類と関係分類において提案手法を用いることで既存手法と比べて精度が大きく向上したこと，構造推定において提案手法がより強く要素分類の結果を反映したものであることがわかった．また，全てを統合した提案手法の分類結果が「因果関係有」の分類精度で先行研究の精度を大きく上回ることが明らかとなった．ただし，それでもまだ因果関係ありについての分類精度は0.54程度とまだ十分に高いとはいえず，応用のためにはさらなる精度の向上が必要である．そのために考えられる今後の課題を以下に示す．

- 言語処理におけるタスクでは，RNNの中でもLSTMを用いるよりも，より単純な構造を持つGRU(Gated Recurrent Unit)[42]を用

いたほうが精度が高くなるという報告 [43] もあることから，GRU や前後の文脈を考慮できる bi-directional LSTM[44, 45] を用いた特徴量の検討を行う．また，ディープラーニングの手法の一つである CNN(Convolutional Neural Networks) を文書分類に応用することで，大きく精度を改善させている研究もある [46] ため，活用できないか検討する．

- 今回は，新たに提案した特徴量の他には Stab and Gurevych(2017)[1] の特徴量を使用したが，論証マイニングにおける特徴量の検討も進んでおり [47, 48]，適切な特徴量の選択を行う．

- 既に一部で試みられているように [49, 50] に，要素間の関係分類へウェブ上の豊富な資源を活用し因果関係分類の精度を向上させる．

また，現時点で論証マイニングに関する日本語での研究は非常に限られている [51] が，自動採点の必要性などが国内でも強調 [52] されているため，日本語への拡張及び教育分野などへの応用を進めていきたい．

謝辞

本論文を作成するにあたり，指導教員の植野真臣教授から，丁寧かつ熱心なご指導を賜りました．ここに感謝の意を表します．また，日頃から親身になって研究を支えていただいた宇都雅輝助教に深謝いたします．そして，ゼミや日常の議論を通じて多くの示唆や知識を頂いた川野秀一准教授，西山悠助教，研究室の先輩・同期・後輩に感謝いたします．

参考文献

[1] C. Stab and I. Gurevych, “Parsing argumentation structures in persuasive essays,” Computational Linguistics, vol.43, no.3, pp.619–659, 2017.

[2] C. Stab and I. Gurevych, “Identifying argumentative discourse structures in persuasive essays,” Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), eds. by A. Moschitti, B. Pang, and W. Daelemans, pp.46–56, Association for Computational Linguistics, Oct. 2014.

[3] A. Peldszus, “Towards segment-based recognition of argumentation structure in short texts,” Proceedings of the First Workshop on Argumentation Mining, pp.88–97, Association for Computational Linguistics, Baltimore, Maryland, June 2014.

[4] H.V. Nguyen and D.J. Litman, “Context-aware argumentative relation mining,” Proceeding of the 54th Annual Meeting of the Association for Computational Linguistics, pp.1127–1137, Association for Computational Linguistics, Berlin, Germany, 2016.

[5] I. Persing and V. Ng, “End-to-end argumentation mining in student essays,” Proceedings of Human Language Technologies: The 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp.1384–1394, 2016.

[6] L. Carstens and F. Toni, “Toward relation based argumantation mining,” Proceeding of the 2nd Workshop on Argumentation Mining, pp.29–32, Association for Computational Linguistics, Denver, CO, 2015.

[7] E. Barker and R. Gaizauskas, "Summarizing multi-party argumentative conversations in reader comment on news," Proceeding of the 3rd Workshop on Argumentation Mining, pp.12–20, Association for Computational Linguistics, Berlin, Germany, 2016.

[8] Y. Song, M. Heilman, B.B. Klebanov, and P. Deane, "Applying argumentation schemes for essay scoring," Proceeding of the First Workshop on Argumentation Mining, pp.69–78, Association for Computational Linguistics, Baltimore, MA, USA, 2014.

[9] B.B. Klebanov, C. Stab, J. Burstein, and Y. Song, "Argumentation: Content, structure, and relationship with essay quality," Proceeding of the 3rd Workshop on Argumentation Mining, pp.70–75, Association for Computational Linguistics, Berlin, Germany, 2016.

[10] D. Ghosh, A. Khanam, and S. Muresan, "Course-grained argumentation features for scoring persuasive essays," Proceeding of the 54th Annual Meeting of the Association for Computational Linguistics, pp.549–554, Association for Computational Linguistics, Berlin, Germany, 2016.

[11] C. Stab, "Argumentative writing support by means of natural language processing," PhD thesis, Technische Universität Darmstadt, 2017.

[12] C. Stab and I. Gurevych, "Training argumentation skills with argumentative writing support," Proceedings of the 21st Workshop on the Semantics and Pragmatics of Dialogue, pp.174–175, SemDial, Aug. 2017.

[13] T.E. Damer, Attacking Faulty Reasoning: A practical Guide to Fallacy-Free Reasoning, 6th edition, Wadsworth, Cengage Learning, 2009.

[14] M. Lippi and P. Torroni, "Argumentation mining: State of the art and emerging trends," ACM Trans. Internet Technol., vol.16, no.2, pp.10:1–10:25, March 2016.

[15] E. Cabrio and S. Villata, "Natural Language Arguments: A Combined Approach.," 20th European Conference on Artificial Intelligence (ECAI 2012), pp.27–31, Montpellier, France, Aug. 2012.

[16] N. Rooney, H. Wang, and F. Browne, "Applying kernel methods to argumentation mining.," In Proceedings of the 25th FLAIRS Conference, AAAI Press, 2012.

[17] E. Florou, S. Konstantopoulos, A. Koukourikos, and P. Karampiperis, "Argument extraction for supporting public policy formulation," Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities, pp.49–54, Association for Computational Linguistics, Sofia, Bulgaria, Aug. 2013.

[18] T. Goudas, C. Louizos, G. Petasis, and V. Karkaletsis, Argument Extraction from News, Blogs, and Social Media, A. Likas, K. Blekas, and D. Kalles, eds., Springer International Publishing, Cham, 2014.

[19] A. Ferrara, S. Montanelli, and G. Petasis, "Unsupervised detection of argumentative units though topic modeling techniques," 4th Workshop on Argument Mining, pp.97–107, Association for Computational Linguistics, 2017.

[20] A. Peldszus and M. Stede, "Joint prediction in mst-style discourse parsing for argumantation mining," Proceeding of the 2015 Conference on Empirical Methods in Natural Language Processing, pp.938–948, EMNLP'15, Lisbon, Portugal, 2015.

[21] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Comput., vol.9, no.8, pp.1735–1780, Nov. 1997.

[22] M.C. Beardsley, Practical Logic, Prentice-Hall, 1950.

[23] S.N. Thomas, Practical reasoning in natural language, Cambridge University Press, 1973.

[24] J.B. Freeman, Argument Structure: Representation and Theory, Springer, 2011.

[25] C. Reed and G. Rowe, “Araucaria: Software for argument analysis, diagramming and representation,” International Journal of AI Tools, vol.14, pp.961–980, Dec. 2004.

[26] R. Cohen, “Analyzing the structure of argumentative discourse,” Comput. Linguist., vol.13, no.1-2, pp.11–24, Jan. 1987.

[27] P. Potash and A.R. Alexey Romanov, “Here’s my point: Joint pointer architecture for argument mining,” arXiv preprint arXiv:1612.08994, 2017.

[28] T. Govier, A Practical Study of Argument, 7th edition, Wadsworth, Cengage Learning, 2010.

[29] C. Stab and I. Gurevych, “Annotating argument components and relations in persuasive essays,” Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014), eds. by J. Tsujii and J. Hajic, pp.1501–1510, Dublin City University and Association for Computational Linguistics, Aug. 2014.

[30] A.P. Gema, S. Winton, T. David, D. Suhartono, M. Shodiq, and W. Gazali, “It takes two to tango: Modification of siamese long

short term memory network with attention mechanism in recognizing argumentative relations in persuasive essay," Procedia Computer Science, vol.116, no.Supplement C, pp.449–459, 2017. Discovery and innovation of computer science technology in artificial intelligence era: The 2nd International Conference on Computer Science and Computational Intelligence (ICCSCI 2017).

[31] J. Lafferty, A. McCallum, and F.C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," Proceeding of the Eighteenth International Conference on Machine Learning, pp.282–289, ICML '01, San Francisco, CA, USA, 2001.

[32] R. Soricut and D. Marcu, "Sentence level discourse parsing using syntactic and lexical information," Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1, pp.149–156, NAACL '03, Association for Computational Linguistics, Stroudsburg, PA, USA, 2003.

[33] Z. Lin, H.T. Ng, and M.-Y. Kan, "A pdtb-styled end-to-end discourse parser," Natural Language Engineering, pp.1–34, 2014.

[34] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," ICLR Workshop, 2013.

[35] J. Pennington, R. Socher, and C.D. Manning, "Glove: Global vectors for word representation," Empirical Methods in Natural Language Processing (EMNLP), pp.1532–1543, 2014.

[36] Y. Kim, "Convolutional neural networks for sentence classification," arXiv preprint arXiv:1408.5882,2014.

[37] C. Guggilla, T. Miller, and I. Gurevych, “Cnn- and lstm-based claim classification in online user comments,” Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers (COLING 2016), pp.2740–2751, Dec. 2016.

[38] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” Trans. Neur. Netw., vol.5, no.2, pp.157–166, 1994.

[39] D.P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv:1412.6980, 2014.

[40] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” Information Processing & Management, vol.45, no.4, pp.427–437, 2009.

[41] M. Sundermeyer, R. SchlÃijter, and H. Ney, “Lstm neural networks for language modeling,” In INTERSPEECH, 2010.

[42] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” Dec. 2014.

[43] W. Yin, K. Kann, M. Yu, and H. Schütze, “Comparative study of cnn and RNN for natural language processing,” arXiv:1702.01923, 2017.

[44] M. Schuster, K.K. Paliwal, and A. General, “Bidirectional recurrent neural networks,” IEEE Transactions on Signal Processing,1997.

[45] I. Habernal and I. Gurevych, “Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional lstm,” Proceeding of the 54th Annual Meeting of the Association for Computational Linguistics, pp.1589–1599, 01 2016.

[46] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, “Very deep convolutional networks for natural language processing,” In EACL., vol.26, 2016.

[47] Y. Desilia, V.T. Utami, C. Arta, and D. Suhartono, “An attempt to combine features in classifying argument components in persuasive essays,” 2017.

[48] A. Aker, A. Sliwa, Y. Ma, R. Lui, N. Borad, S. Ziyaei, and M. Ghobadi, “What works and what does not: Classifier and feature analysis for argument mining,” Proceedings of the 4th Workshop on Argument Mining, pp.91–96, Association for Computational Linguistics, 2017.

[49] 青野壮志，太田学，“要因検索による因果関係ネットワークの構築,” 研究報告データベースシステム（DBS），vol.2009，no.9，pp.1–8，nov 2009.

[50] 佐藤岳文，堀田昌英，“Ｗｅｂマイニングを用いた因果ネットワークの自動構築手法の開発,” 社会技術研究論文集，vol.4，pp.66–74，2006.

[51] H. Yamada, S. Teufel, and T. Tokunaga, “Annotation of argument structure in japanese legal documents,” Proceedings of the Forth Workshop on Argumentation Mining, pp.22–31, Association for Computational Linguistics, 2017.

[52] 高大接続システム改革会議，“高大接続システム改革会議「最終報告」,” 2016.