RNN系列変換モデルを用いた高階論理式からの文生成

馬目 華奈 †,1　谷中 瞳 ‡,2　吉川 将司 ⋆,3　峯島 宏次 †,4　戸次 大介 †,5
† お茶の水女子大学　‡ 東京大学　⋆ 奈良先端科学技術大学院大学

g1420542@is.ocha.ac.jp[1], hitomiyanaka@g.ecc.u-tokyo.ac.jp[2],
yoshikawa.masashi.yh8@is.naist.jp[3],
mineshima.koji@ocha.ac.jp[4], bekki@is.ocha.ac.jp[5]

1 はじめに

近年の構文解析と意味解析の技術の発展によって，文の意味を論理式で表して高度な推論を行うシステムの構築が可能となった．このようなシステムは，含意関係認識 [1, 2] や文間類似度計算 [3] のタスクで高精度を達成しており，今後，さらなる自然言語処理タスクへの応用が期待されている．

文からその論理式への変換が高精度に行われる一方で，論理式を自然言語文に戻す方法については自明ではない．しかし，論理式から自然言語文に逆変換することができれば，推論システムの改善や，様々な自然言語処理タスクへの応用が期待できる．推論システムにおいては，実社会への応用を考えると，推論に失敗した場合において，なぜ推論に失敗したのかという解釈性が求められる．そこで，推論において証明不可能と判定された論理式を文に変換することができれば，どのような知識が推論に必要であったかを言語化することができる．

また，論理式から自然言語文に変換する方法は，パラフレーズ抽出 [4]，テキスト平易化 [5] 等への応用も可能である．パラフレーズ抽出については，例えば，二文を論理式に変換した上で，それらの論理式に共通する部分を自然言語に変換することにより，二文の意味の重複を言語化する，といった応用が考えられる．また，テキスト平易化については，統計的機械翻訳を用いた手法 [5] が研究されているが，統語構造の差異による意味の違いを抽象化する論理式の性質を利用すれば，難しい文を論理式に変換し，論理式から同じ意味を持つ平易な文を生成することが考えられる．

そこで本研究では，機械翻訳等の系列変換において高い精度を示しているニューラルネットによる系列変換モデル (Sequence-to-Sequence model)[6] を用いて高階論理式から文を生成する手法を提案する．論理式の埋め込みについては複数の方法を提案・比較した．含意関係認識用データセットを用いて提案手法の評価を行った結果，論理式を先頭の記号から埋め込んだ場合と比較して，論理式を木構造として埋め込むことで精度向上がみられた．

2 背景

2.1 CCG に基づく論理式による文の意味表現

文を高階論理式に変換し，高階論理に基づく自動推論を行うシステムとして，ccg2lambda[2] がある．ccg2lambda では，まず入力文に対して組合せ範疇文法 (Combinatory Categorial Grammar,CCG)[7, 8] に基づく統語解析を行う．CCG は語彙化文法の一種で，統語構造に並行して意味表示の合成をおこなう文法体系として知られている．各語には統語範疇が割り当てられる．CCG では語と語の統語的・意味的な関係を，関数適用や関数合成などの組合せ規則により計算していき，同時に文における各語の貢献の仕方を定めていく．次に，CCG の導出木をラムダ計算に基づいて論理式へと変換する．ラムダ項によって表現されている各語の意味表示から，組合せ規則が指定する計算に沿って，最終的な文の意味表示である高階論理式を得ることができる．高階述語論理による証明には，高階論理・型理論に基づく定理証明支援系である Coq が用いられている．

2.2 意味表現からの文生成

論理式からの文生成システムとして，まずルールベースによる手法がある．これまでに，Event Semantics に基づく一階述語論理の論理式から自然言語文へ変換する手法 [9] や，高階依存型理論の論理式からの変換を行う Grammatical Framework (GF) の手法 [10] が

提案されている．これらは，比較的強い記述力をもつ論理言語から自然言語への変換を試みる点では本研究の手法と共通しているが，テキスト平易化などの応用につながる汎化能力をもつかどうかは自明ではない．

文と意味表現は多対多の関係にあるが，一つの意味表現に対応しうる文の多様さは，一つの文に対応しうる意味表現の曖昧性とは異質である．特定の形の意味表現に対して，ルールベースで特定の形の文を生成する，という手法で，その多様さを捉えきることは容易ではない．より頑健な文生成のためにはデータからの学習に基づく手法が望ましいが，そのためには文とそれに対応する論理式について，大量のリソースが利用できる必要がある．

抽象的意味表現 (Abstract Meaning Representation,AMR) [11] は，そのような要請を満たす意味表示体系の一つとして，近年注目を集めている．AMR2.0 では 39,260 文に対して意味表現がアノテートされており，それらを用いて文から AMR を生成する研究だけではなく，AMR から文に変換する研究も進められている．後者の研究としては，フレーズベース機械翻訳 [12] や系列変換モデルによる手法が提案されている．中でも，ノードの探索により AMR のグラフをリスト化し，入力とする系列変換モデルを用いた手法 [13] は高精度を達成している．

AMR は文の意味を有向非巡回グラフで表現する体系であり，高階述語論理と比較して記述力には限界がある．また，現状では AMR を用いた証明体系は提案されておらず，したがって ccg2lambda のような論理的推論を扱うことができない．そのため，1 節で述べたような，推論に失敗した原因を説明できる推論システムを実現する方法も，AMR においては自明ではない．一方，高階述語論理式から文を生成することができれば，そのようなシステムの実現につながる可能性がある．

2.3 系列変換モデル

系列変換モデル [6] とは入出力がシーケンスとなる機構で，意味や構文などには注目せず，入力と出力の対応を学習して覚える，ニューラルネットのモデルである．系列変換モデルは入力列を隠れ状態ベクトルに変換するエンコーダと，隠れ状態ベクトルから出力を行うデコーダからなる．エンコーダでは, 入力の系列を埋め込みベクトルに変換した後，LSTM 等の再帰型ニューラルネットワークによって隠れ状態ベクトルに変換する．デコーダでは，エンコーダで出力された隠れ状態ベクトルを初期値とし，隠れ状態と自身のこれまでの出力結果を基に次のトークンを生成する．

3 提案手法

3.1 学習モデル

本研究では AMR からの文生成においても高精度を実現している系列変換モデルを用いて，ccg2lambda が生成する高階論理式を入力とし，対応する自然言語文を予測することを目的とする．エンコーダ，デコーダには LSTM を用いた系列変換モデルを用いた．学習モデルを図 1 に示す．

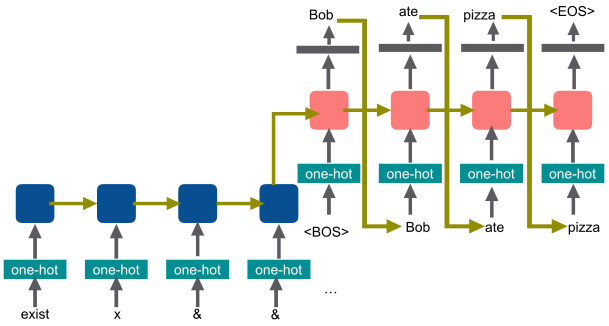

図 1: 学習モデル

3.2 データセット

学習データは，ccg2lambda を用いて文から高階論理式を作成する．データ作成の流れを図 2 に示す．ccg2lambda は複数の CCG パーザによるマルチパージングが可能であり，C&C[14]，depccg[15]，EasyCCG[16]，EasySRL[17] を用いる．実験用テキストには，含意関係認識タスクの評価用データセットである SNLI[18] を用いて，論理式と自然言語文のペアからなる教師データを作成した．長い文に対しては CCG パーザが解析に失敗することがあるため，SNLI データセットに含まれる文例のうち，一文に含まれる単語数が 60 単語以内の文例 20,000 件を対象とした．このうち，構文解析に成功したものが 19,899 件であり，さらに論理式への変換に成功したデータ 19,587 件を使用する．ここで，意味合成後にラムダ抽象構文が残った場合には変換の失敗とみなす．データのうち，18,087 件を教師データ（うち 3,617 件を validation データ），1,500 件をテストデータとする．

3.3 論理式の埋め込み

ccg2lambda で扱う論理式において，$TrueP$ は意味合成の際に必要となる補助的な命題定項であり，トートロジーに対応する．先の “Bob ate pizza.” という例は，論理式に変換すると以下のようになる．

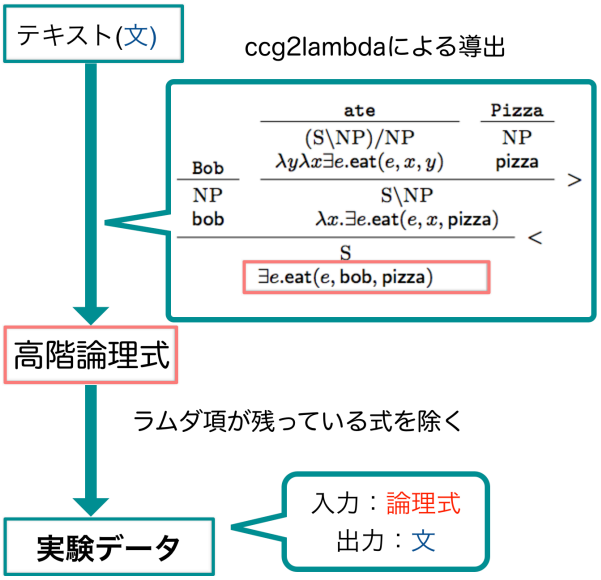

図 2: データの作成過程

$exists\ x.((x = Bob)\&TrueP\&exists\ z1.(Pizza(z1)\&TrueP\&exists\ e.(eat(e)\ \&\ (Subj(e) = x)\ \&\ (Obj(e) = z1)\ \&\ TrueP)))$

論理式を系列変換モデルの入力として扱うためには論理式をベクトル化する必要がある．論理式をベクトル化するうえで，本研究では文字単位，およびトークン単位で数値化する手法を検討する．文字単位の場合は，論理式を先頭の記号から one-hot ベクトルに変換して埋め込みを行う．また，トークン単位で埋め込む場合は，論理式の構成要素を埋め込む順序も問題となる．本研究では先頭から順に埋め込む手法と，先行研究の AMR から文生成の手法 [13] に基づき，論理式を木構造とみなし，前順 (pre-order) による深さ優先探索 (depth-first search, DFS) でリスト化して埋め込む手法を採用した．この 3 種類の埋め込み手法の例を下記に示す（ここで ␣ はスペース文字を表す）．

1. 文字単位で先頭から埋め込む手法
   例：$[e, x, i, s, t, s, ␣, x, ␣, ., ␣, (, (, x, ␣, =, ...]$
2. トークン単位で先頭から埋め込む手法
   例：$[exists, x, (, x, =, \_Bob,), \&, TrueP, ...]$
3. pre-order の DFS で埋め込む手法
   例：$[exists, x, \&, \&, =, \_Bob, x, TrueP, exists, ...]$

4 実験

4.1 実験設定

各埋め込み手法における入出力の語彙，文の最大の長さを表 1 に示す．埋め込みベクトルの次元数として，トークン・DFS ベースではそれぞれ 256・512・1024 次元で実験を行い，validation loss が一番低かった 256 次元を採用した．文字ベースでは文字の種類が少ないため，埋め込みの必要性は高くないが，トークン単位との比較のため，エンコーダは 70 次元，デコーダは 78 次元の埋め込みを行った．

表 1: 各埋込み手法における語彙数・系列長

	文字	単語	DFS
入力語彙数	70	5,142	5,134
出力語彙数	78	7,214	7,214
入力列最長	2,097	728	480
出力列最長	270	55	55

4.2 評価方法

評価は BLEU[19] によって行う．BLEU とは，文と文の表層的な類似度 (n-gram マッチ率) を表す指標である．本研究では，以下の式に基づいて評価を行う．

$$score = BP \exp\left(\sum_{i=1}^{N} \frac{1}{N} \log P_n\right)$$

$$BP = \begin{cases} 1 & (c \geq r) \\ \exp\left(1 - \frac{r}{c}\right) & (\mathrm{c} < \mathrm{r}) \end{cases}$$

$$P_n = \frac{\sum_{i=0} \text{出力文 } i \text{ 中と解答文 } i \text{ 中で一致した } n\text{-}gram \text{ 数}}{\sum_{i=0} \text{出力文 } i \text{ 中の全 } n\text{-}gram \text{ 数}}$$

4.3 実験結果・考察

実験の結果を表 3 に示す．文字ベースでは，“A woman is standing.” という文が “A woman is sleeping.” に変換されてしまうなど，英文の主語，主語＋ be 動詞という単純なパターン以外の予測ができていなかった．また，文字ベースでは BLEU のスコアも他の 2 つに対し劣っているため，以下ではトークンベースと DFS による手法の比較を行う．この 2 つの埋め込み手法で文生成がうまくいかなかった例は (1) 正解の文が長くてデコーダの出力文が途中で途切れてしまう，(2) 学習の際に学習に現れてこなかった単語が再現できていない，というものであった．(1)(2) の例を下記に示し，デコードされた結果を表 2 にまとめる．また，(2) のように，入力の系列には存在しない単語を出力してしまうケースがあり，このような問題に対しては今後系列変換モデルのコピーメカニズム [20] を用いて対処することが考えられる．

(1) Three Oklahoma Sooners playing football against another team, one of the sooners with the ball in their possession.

表 2: デコード例

文	トークン	DFS
(1)	Three Oklahoma Sooners playing football against another.	Three Oklahoma Sooners playing football against another team, one of.
(2)	The child is about to get a tennis player in black.	The children are about to get something in the woods.

表 3: 評価結果

指標	文字	トークン	DFS
BLEU	36.6	75.5	81.2

(2) A child is about to go swimming in the lake.

5 おわりに

本研究では，系列変換モデルを用いて高階論理式から文を生成する手法を提案した．含意関係認識用データセットを用いて提案手法の評価を行った結果，論理式をシーケンス化して先頭から埋め込んだ場合と比較して，論理式の順番を考慮して埋め込むことで精度向上がみられた．今後の課題として，他の意味表現からの文生成との比較や他のデータセットによる評価を行う．また，アテンション付き系列変換モデルやコピーメカニズムを用いるなど，モデルの改良に取り組む．更に論理式における変数や論理記号の扱い，スコープなどを考慮するよう，埋め込み方法を改良したい．

謝辞 この研究は，JST CREST「ビッグデータ統合利活用のための次世代基盤技術の創出・体系化」領域「知識に基づく構造的言語処理の確立と知識インフラの構築」プロジェクトの支援を受けたものである．

参考文献

[1] Lasha Abzianidze. A Tableau Prover for Natural Logic and Language. In *Proc. of EMNLP*, 2015.

[2] Pascual Martínez-Gómez, Koji Mineshima, Yusuke Miyao, and Daisuke Bekki. ccg2lambda: A Compositional Semantics System. In *Proc. of ACL System Demonstrations*, 2016.

[3] Hitomi Yanaka, Koji Mineshima, Pascual Martínez-Gómez, and Daisuke Bekki. Determining Semantic Textual Similarity using Natural Deduction Proofs. In *Proc. of EMNLP*, 2017.

[4] Omer Levy, Ido Dagan, Gabriel Stanovsky, Judith Eckle-Kohler, and Iryna Gurevych. Modeling extractive sentence intersection via subtree entailment. In *Proc of Coling*.

[5] Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. A Monolingual Tree-based Translation Model for Sentence Simplification. In *Proc. of CL*, 2010.

[6] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to Sequence Learning with Neural Networks. In *Proc. of NIPS*, 2014.

[7] Mark Steedman. Surface Structure and Interpretation. In *The MIT Press*, 1996.

[8] Daisuke Bekki. *A Formal Theory of Japanese Grammar: The Conjugation System, Syntactic Structures, and Semantic Composition.* Kuroshio, 2010. (In Japanese).

[9] Alastair Butler. Deterministic natural language generation from meaning representations for machine translation. In *Proc. of SedMT*, 2016.

[10] Aarne Ranta. *Grammatical framework: Programming with Multilingual Grammars.* CSLI Publications, 2011.

[11] Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. Abstract Meaning Representation for Sembanking. In *Proc. ACL LAW*, 2013.

[12] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical Phrase-Based Translation. In *Proc. of NAACL*, 2003.

[13] Ioannis Konstas, Srinivasan Iyer, Mark Yatskar, Yejin Choi, and Luke Zettlemoyer. Neural AMR: Sequence-to-Sequence Models for Parsing and Generation. 2017.

[14] Stephen Clark and James R. Curran. Wide-coverage Efficient Statistical Parsing with CCG and Log-linear Models. *Comput. Linguist.*, 2007.

[15] Masashi Yoshikawa, Hiroshi Noji, and Yuji Matsumoto. A* CCG Parsing with a Supertag and Dependency Factored Model. In *Proc. of ACL*, 2017.

[16] Mike Lewis and Mark Steedman. A* CCG Parsing with a Supertag-factored Model. In *Proc. of EMNLP*, 2014.

[17] Mike Lewis, Luheng He, and Luke Zettlemoyer. Joint A* CCG Parsing and Semantic Role Labelling. In *Proc. of EMNLP*, 2015.

[18] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proc. of EMNLP*, 2015.

[19] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proc. of ACL*, 2002.

[20] Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. Incorporating Copying Mechanism in Sequence-to-Sequence Learning. In *Proc. of ACL*, 2016.