言語モデルの統語構造把握能力を測定する より妥当な多言語評価セットの構築

神藤 駿介 [1,2]　能地 宏 [3,2]　宮尾 祐介 [1,2]
[1] 東京大学大学院 情報理工学系研究科
[2] 産業技術総合研究所 人工知能研究センター
[3] LeapMind 株式会社
{kando.s, yusuke}@is.s.u-tokyo.ac.jp　noji@leapmind.io

概要

近年言語モデルの統語構造把握能力を評価する研究が盛んだが，それらは英語テキストを用いたものに偏っている．本研究ではそのような評価セットを生のコーパスから多言語で自動的に抽出する手法を提案する．既存研究は依存構造木から評価セットを自動的に抽出するアルゴリズムを提案しているが，実際には抽出データのほとんどが統語構造の把握を要しない容易なタスクとなっており，妥当な評価セットを構築できていたとは言えないことが分かった．本研究では，同じアルゴリズムを用いつつ評価セットの抽出量を増やすことで，より妥当な評価セット構築を目指す．

1　はじめに

近年 [1] を発端に，言語モデルの統語構造把握能力を評価セットを用いて直接的に測定する研究が盛んに行われている．そのような評価セットは多くの場合は以下のような文法的に正しい文と誤っている文の組（ミニマルペア）から成る:

The farmer near the clerks knows / *know.

評価においては 2 つの文を評価対象の言語モデルに入力し，文法的に正しい文により高い確率が付与されるかどうかを見る．know の正しい活用形はその主語（farmer）を正確に把握することによって判断されるが，この例では farmer が前置詞句で修飾されており，この文法的な構造を把握できていないとより距離的に近い clerks に引きづられて誤った判断をしてしまう恐れがある．評価セット構築およびそれによる言語モデルの評価に関する研究は英語テキストを対象にしたものが数多くあるが [2, 3, 4]，その結論は大枠では一致している．すなわち，LSTM [5] をはじめとする単方向モデルの性能には限界がある一方で，BERT [6] をはじめとする双方向 Transformer 系モデルや RNNG [7] といった階層構造を取り入れたモデルの性能が高いことが知られている．

これらの研究は多言語でも徐々になされており，評価セットの作成方法としてはテンプレートを元に構築する手法 [8] と，依存構造ツリーバンクから自動抽出する手法 [9] がある．前者のアプローチはコストが高く，より多様な言語の評価セットを構築するにあたっては後者のアプローチがより適しているが，[9] においては LSTM が十分な性能を有するという他の研究と食い違う主張がなされている．

本研究では，まず [9] で公開されている評価セットの問題点を指摘し，実際には LSTM の性能が高いとは言えないこと，RNNG のような階層的なモデルが依然として優位にあることを立証した．既存の評価セットの具体的な問題点は，統語構造の正確な把握を要しないケースがほとんどを占めており，実際に構造把握を要するケースの個数は非常に限られていることである．そこで，構文解析を施したコーパスを利用してデータ抽出量を大幅に増やすことでその問題点を解決し，言語モデルの統語構造把握能力をより正当に評価できる妥当な評価セットを構築する手法を提案する．3 つの言語での抽出実験の結果，構造把握を要するケースを最低でも 70 倍程度増やすことに成功した．

研究のモチベーション　我々は RNNG を多言語で学習する手法を提案し [10]，事前の研究でそれが 5 つの言語の評価セットである CLAMS [8] において高い性能を記録することを確認した．この手法の適用範囲を模索していくにあたってはより多様な言語での評価セットが必要であり，本研究はそれに向けての第一段階である．

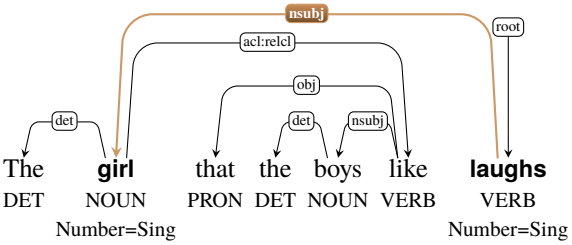

図 1: 評価データを抽出できる依存構造木の例

2 既存評価セットの問題点の指摘

2.1 既存研究 [9] の概要

Gulordava ら [9] は，依存構造木から評価セットを自動的に抽出するアルゴリズムを提案している．抽出手順の概略は以下の通りである．

1. 依存構造ツリーバンクを読み込んで各語彙の情報を集計し，品詞表を得る．
2. ツリーバンクから依存関係の距離が比較的長いパターンを有するデータを取り出し，その依存関係に数の一致現象があるとみなされるものだけを抽出する．
3. 2. で得られたデータについて，1. で得られた品詞表を参照して依存関係の右側の単語を書き換えてミニマルペアを作成する．

例えば 図 1 の依存構造木において nsubj のラベルに注目すると，エッジで結ばれた二つの単語の数の素性（Number=Sing）が一致している．これと同じタイプの依存関係のデータを収集して常に数の素性の一致が観察される場合（つまり Number=Plur という一致も十分量存在する場合），このデータは一致現象があるとみなされる．ミニマルペアの作成については，品詞表を参照して依存関係の右側の単語 “laughs” を Number=Plur の特徴をもつ “laugh” に入れ替えることでなされる．以下，依存関係の左側と右側の単語をそれぞれ *cue* / *target* と呼ぶ．

Gulordava らはこのアルゴリズムを Universal Dependencies (UD) [11] のイタリア語・英語・ヘブライ語・ロシア語の訓練セットに適用して得られた評価セットを公開しており[1)]，これを元に言語モデルの評価を行って LSTM の性能が十分に高いことを主張している．これは自動化に頼らないより妥当性の高いアプローチで多言語の評価セットを作成している [8] とは異なる結論であり，疑問の余地が残る．

1) https://github.com/facebookresearch/colorlessgreenRNNs/tree/main/data/agreement

colorless 文の作成　ツリーバンクから抽出された評価セットに加え，各文の内容語を同一品詞・同一素性を持つ別の語にランダムに置き換えた文を生成する．以下，このランダムな文を **colorless 文**と呼ぶ．colorless 文は「文法性と意味とは切り離して考えられる」という説 [12] に基づいて生成されている．例えば，我々は “Colorless green ideas sleep furiously.” という文を読んだとき，意味は分からないが文法的には正しいと判断ができる．colorless 文はこのような文を得る目的で考案されており，モデルが語彙情報をもとにタスクを解けなくするようにしている．一方で colorless 文は述語項構造の制約を考慮できないという問題点もあり，本来自動詞である “stay” が “It stays the shuttle” というように誤用されたケースが生成されることがある．

語彙をベースにしたデータ選別　最後に，言語モデルによる評価の妥当性を高めるため，抽出されたデータの cue と target の間の全ての単語が言語モデルで事前に定義された語彙に含まれるケースのみを抽出する．

2.2 Attractor の有無に関する分析

2.2.1 概要

我々は [9] の評価セットのデータを *attractor* の有無によって分類し再度評価を行うことで，評価データが統語構造の把握能力をものとして本当に妥当なものであるかを調査した．*attractor* とは，cue と target の間にある単語で，cue と品詞は同じで数の素性が異なるものを指す．図 1 においては “boys” が attractor である．attractor は cue よりも target の近くにあるため，LSTM のような単方向のモデルだと attractor に引きづられて target の正しい活用形を間違えてしまう可能性が高まる．したがって，attractor のあるデータは解くのが難しいと言える．一方で attractor の存在しないケースは次の二つに分けられる: (1) cue と target の間に cue と同じ品詞の単語がないケース．(2) cue と同じ品詞の単語はあるが，数の素性が cue と同じケース．(1) の場合は cue と target の品詞の情報のみを手がかりにして解けてしまい，(2) の場合は cue ではなく間の単語を手がかりにしても解けてしまうことから，いずれにせよ **attractor が存在しないケースは統語構造の把握能力を測定するデータとしては不適格**であり，解くのが容易なデータであると言える．

表 1: Gulordava らが公開しているデータセットの統計量および言語モデルによる精度．サブキャプションに attractor の無いデータの割合を括弧書きで示した．各データを元に合計で 9 個の colorless 文を生成しているため，元データのサイズは表記されている量の 1/10 である．

(a) 英語（80%）

#(attr)	#（該当データ）	LSTM acc.	RNNG acc.
0	330	0.82	0.84
1	70	0.73	0.84
2	10	0.80	1.00

(b) ヘブライ語（73%）

#(attr)	#（該当データ）	LSTM acc.	RNNG acc.
0	2,710	0.91	0.90
1	950	0.65	0.70
2	40	0.53	0.63
3	30	0.53	0.58

(c) ロシア語（95%）

#(attr)	#（該当データ）	LSTM acc.	RNNG acc.
0	4,220	0.92	0.92
1	180	0.78	0.88
2	10	0.80	0.60
3	10	1.00	0.60

2.2.2 実験設定

実験では，Gulordava らが公開している評価セットのうち英語・ヘブライ語・ロシア語について分析を行う．評価セットを attractor の個数によって切り分け，それぞれ言語モデルで評価して精度を比べる．言語モデルとして LSTM および RNNG を用いて結果を比べる．LSTM の各パラメタは [13] のものを用いる．RNNG は [10] に倣って学習する．ただし，構造は flat なものを用い，非終端記号には dependency label を付与する．RNNG の訓練データは各言語の Wikipedia 記事を 80 M token だけ取り出して UDify [14] で解析して作成し，LSTM の訓練データも同じ文を用いる．

2.2.3 結果

データ数の統計および言語モデルの精度を表 1 に示す．いずれの言語においても attractor が存在しないケースが大多数を占めており，なおかついずれのモデルもその精度が高いことが見て取れる．一方で attractor が存在するケースでは，LSTM はどの言語でも明らかな精度減少が観察される．RNNG は英語とロシア語では attractor があっても高い精度を維持しており，ヘブライ語においても LSTM より良い精度を記録している．

以上のことをまとめると，**[9] における「LSTM の文法構造把握能力が高い」という結論は実際には統語構造の把握能力を要しないデータが大多数を占めているがために導かれたものであり，モデルの性能を正確に評価するには attractor のあるケースを用いる必要がある．**

3 構文解析を用いたデータ拡張

3.1 概要

前節では attractor のあるデータを用いて評価を行うことの重要性について述べたが，表 1 からも見てとれるように実際には attractor のあるデータは非常に少ない．colorless 文を増やすことも考えられるが，述語項構造が誤った文も生成されてしまうことから望ましくない．Kasai と Frank [15] は colorless 文の構文解析の実験を行い，述語項構造が誤っていることが原因でその性能が悪くなることを指摘しており，これは言語モデルの評価にも影響を及ぼす懸念がある．したがって，より評価セットとしての妥当性を高めるに当たっては元の抽出データ量自体を増やすことが必要となる．そこで我々は UD のデータから抽出する代わりに，生のコーパスを構文解析することで多量のツリーバンクを得てから抽出を行い，どれくらいデータが抽出できるかを調査した．本実験の目標は評価セットを作成することであるため，出来る限り信頼のおける構文解析結果を用いたい．そこで我々は二つの構文解析器の解析結果が一致するとみなせるようなケースだけを取り出すことで信頼のおけるデータの抽出を試みた．

3.2 実験設定

本実験においてはイタリア語・英語・ロシア語の評価セット抽出を行う．まず各言語の Wikipedia 記事を 100 M token 分用意した．この際，短すぎる文は長距離の依存関係が存在しないこと，長すぎる文は解析結果が一致しづらくなることを考慮し，一文あたりの token 数が 9 以上 40 以下のものだけを取り出すこととした．構文解析には UD をベース

表 2: 評価セットの抽出結果．評価セット数およびそのうち attractor を含むものの数（4，5 列目）については，Gulordava らが公開している評価セットのデータ数を括弧書きで示した．attractor の数ごとの統計は付録の 表 3 に示す．

言語	#（生コーパスの文）	#（解析結果が一致した文）	#（評価セット）	#（attr 有）
イタリア語	4,263,026	782,964	20,699 (119)	2,510 (34)
英語	4,489,030	630,090	3,572 (41)	590 (8)
ロシア語	4,997,436	891,044	26,177 (442)	3,352 (20)

にした構文解析器である Trankit [16] と Stanza [17] を用いた．得られた 2 つの依存構造木において，CoNLL-U format[2] の 10 のフィールドのうち主要な 4 つ（UPOS, FEATS, HEAD, DEPREL）が全て一致するものを取り出した．なお，本手法ではサブワードを用いた言語モデルによる評価を見越して，語彙による選別は行っていない．

3.3 結果

表 2 に抽出できた評価セットの統計量を示す．いずれの言語においても，attractor のあるケースを評価セットとして十分な数だけ抽出できたと言える．なお，英語の評価セットが極端に少ないのは，英語が形態的に貧弱であるために数の一致現象自体が少ないことに起因する．

4 今後の課題

抽出データを増やすことは出来たが，まだいくつか課題が残されている．

第一に，抽出した評価セットの質を評価する必要がある．解析結果の一致をとることで出来るだけ正確な木構造を得るよう工夫はしているものの，今回の評価セットはあくまで予測された木構造から得られたものであり，信頼のおけるデータであるとは限らない．最も直接的に質を評価する方法としてはクラウドソーシングを用いた人手による評価があり得る．間接的な手法としては，n-gram をはじめとするナイーブな言語モデルが解けないような評価セットとなっていることを確かめることも考えられる．加えて，今回は二つの解析結果を元により正確な木構造を取り出したが，単一の構文解析器の出力を用いて抽出した評価セットの質が十分に高いこともあり得る．仮にそうであれば，構文解析の一致をとるステップは時間的コストが大きい上にデータの抽出量も減るため不要とみなされることになる．したがって，単一の構文解析器による実験も行ってここの是非を確認する必要があるだろう．

第二に，一つ目の課題と少し被るが，抽出データを用いて実際に言語モデルを評価する実験が必要である．LSTM や RNNG の評価結果が Gulordava らが公開している gold な木構造から作成されたデータセットのそれと大きく異なることがあれば，やはりデータセット自体の質を疑わなくてはならない．また，扱う言語の種類によって言語モデルの性能が変わってくるのかなどと言った多言語モデリングの観点からの研究にも大きな意義がある．

第三に，本手法を実際に他の言語に適用することが必要である．実は抽出アルゴリズムを動かすにあたっては，ツリーバンクによっては何らかの前処理を施す必要がある．例えば英語の UD のツリーバンクには動詞に Number=Plur という特徴を明示的にアノテーションしないという決まりがあり，この情報を前処理で付与しないと評価セットを抽出できなくなってしまう．また，本研究では [9] では取り扱われているヘブライ語の抽出を行っていないが，これは既存コードの前処理用のスクリプトが何らかの理由で動作しなかったためである．このように，各言語ごとに自明でない前処理が発生し得るため，各言語ないしツリーバンクのアノテーション規約はどうなっているのか，より一般にはどのような前処理を施せば滞りなく抽出できるのかなどを丁寧に調査する必要がある．

5 おわりに

本研究では，言語モデルの統語構造把握能力を測定するより妥当な多言語評価セットの構築を行った．評価セットの質の評価をはじめ今後解決すべき課題はあるが，この研究を進めていくことで多言語モデリングにおける新たな評価指標を確立し，分野に貢献できることを期待する．

2） https://universaldependencies.org/format.html

謝辞

この成果は，国立研究開発法人新エネルギー・産業技術総合開発機構 (NEDO) の助成事業 (JPNP20006) の結果得られたものである．産総研の AI 橋渡しクラウド (ABCI) を利用し実験を行った．

参考文献

[1] Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. **Transactions of the Association for Computational Linguistics**, Vol. 4, , 2016.

[2] Adhiguna Kuncoro, Chris Dyer, John Hale, Dani Yogatama, Stephen Clark, and Phil Blunsom. LSTMs can learn syntax-sensitive dependencies well, but modeling structure makes them better. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, 2018.

[3] Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. BLiMP: A benchmark of linguistic minimal pairs for English. In **Proceedings of the Society for Computation in Linguistics 2020**, 2020.

[4] Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger Levy. A systematic assessment of syntactic generalization in neural language models. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, 2020.

[5] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. **Neural Comput.**, Vol. 9, No. 8, 1997.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, 2019.

[7] Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A. Smith. Recurrent neural network grammars. In **Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, 2016.

[8] Aaron Mueller, Garrett Nicolai, Panayiota Petrou-Zeniou, Natalia Talmina, and Tal Linzen. Cross-linguistic syntactic evaluation of word prediction models. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, 2020.

[9] Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. Colorless green recurrent networks dream hierarchically. In **Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)**, 2018.

[10] 神藤駿介, 能地宏, 宮尾祐介. 依存構造から句構造への変換による多言語モデリングに向けて. 言語処理学会 第 27 回年次大会, 2021.

[11] Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. Universal Dependencies v2: An evergrowing multilingual treebank collection. In **Proceedings of the 12th Language Resources and Evaluation Conference**, Marseille, France, 2020.

[12] Noam Chomsky. **Syntactic Structures**. Mouton and Co., 1957.

[13] Hiroshi Noji and Hiroya Takamura. An analysis of the utility of explicit negative examples to improve the syntactic abilities of neural language models. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, 2020.

[14] Dan Kondratyuk and Milan Straka. 75 languages, 1 model: Parsing Universal Dependencies universally. 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)**, 2019.

[15] Jungo Kasai and Robert Frank. Jabberwocky parsing: Dependency parsing with lexical noise. In **Proceedings of the Society for Computation in Linguistics (SCiL) 2019**, 2019.

[16] Minh Van Nguyen, Viet Dac Lai, Amir Pouran Ben Veyseh, and Thien Huu Nguyen. Trankit: A light-weight transformer-based toolkit for multilingual natural language processing. In **Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations**, 2021.

[17] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations**, 2020.

A　評価セットの attractor 数ごとのデータ数の統計

表 3: 本研究で作成した評価セットおよび Gulordava ら [9] が公開している評価セットの attractor 数ごとのデータ数の統計.

(a) イタリア語

#(attr)	抽出セット	公開セット
0	18,189	85
1	2,147	23
2	307	10
3	51	1
4	5	0
5	0	0

(b) 英語

#(attr)	抽出セット	公開セット
0	2,982	33
1	466	7
2	102	1
3	15	0
4	6	0
5	1	0

(c) ロシア語

#(attr)	抽出セット	公開セット
0	22,825	422
1	2,648	18
2	566	1
3	109	1
4	24	0
5	5	0