

モンテカルロ木探索を用いた頑健な文生成

理学専攻 情報科学コース

1540649 熊谷香織

1 はじめに

自然言語処理の分野において、非言語情報を言語で表現する文生成の研究が盛んになってきている。近年の一般的な文生成手法は、ニューラルネットワークを用いた言語モデルによる単語の予測をするもので、統語情報を考慮したものは数少ない。また、人がある事象を表現するときに生成し得る文は一つに定まらないが、統計的手法を用いた多くの文生成手法はそれぞれのアルゴリズムにおいて最尤となる文を出力するものである。

これに対して本研究では統語構造を考慮した非決定的な文生成手法を提案する。提案手法は、文生成のアルゴリズムとしてモンテカルロ木探索 [1] を使用し、適用規則に確率文脈自由文法を用いた構文木構築に基づく文生成を行った。文生成の方針として、生成内容とする SVO に相当する単語を入力情報として与え、探索のシミュレーションで得られた生成文を生成内容との類似度に基づき評価することで、生成内容を踏まえかつ多様な言い回しが可能な文生成手法を開発した。

2 モンテカルロ木探索

本研究では UCB (Upper Confidence Bounds) 1 値 [2] をモンテカルロ木探索に応用した UCT アルゴリズムを文生成に用いた。UCB1 値は、勝率の項、および探索が不十分なノードに対して選択の可能性を考慮した項から構成される (式 1)。

$$v_i + C \sqrt{\frac{\log N}{n}} \quad (1)$$

v_i はそのノードの勝率、 C は調整係数、 N は全試行回数、 n はそのノードを選択した回数を示す。UCB1 値における第 1 項が「知識の適用 (exploitation)」を、第 2 項が「探査 (exploration)」を考慮している。それによりバランスをとった探索が実行される。

また、MCTS はコンピュータ囲碁におけるゲーム AI の手法として注目されたが、ゲームの途中の状態の評価関数を使用せず、終了時点での勝敗と探索回数を元に評価する。そのため、MCTS は試行の途中の段階を評価することが難しいタスクに適している。

3 MCTS を使用した文生成

確率文脈自由文法 (PCFG) を用いた構文木形成において MCTS を使用したアルゴリズムを図 1 に示す。文の生成内容として、文の主要な要素となる SVO の単語情報を与えた (e.g. S:dog, V:eat, O:bread)。このような 3 つ組の単語情報をゴールとして探索を進め、適切な文を生成する。

3.1 探索の進め方と SVO 判定

文中の主要な箇所から構文木を構築することを考えた。V→S→O→SVO 以外の自立語 → ストップワード

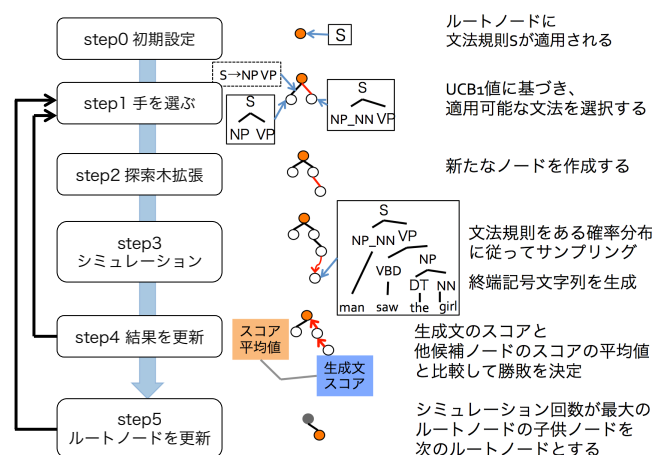


図 1: MCTS を用いた文生成

の順に木を展開する。ここで、PCFG によって構築される構文木から直接 SVO を判定することはできないため、以下に示す順に木を展開し、SVO を決定した。

1. V から始まる品詞を展開。
初めて展開された終端記号を文の述語とする。
2. 述語より左の N から始まる品詞を展開する。
初めて展開された終端記号を文の主語とする。
3. 述語より右の N から始まる品詞を展開する。
初めて展開された終端記号を文の目的語とする。
4. その他の品詞を展開する。

3.2 サンプルング対象となる確率分布の設定

MCTS を用いて文生成をする際、膨大な文法規則と語彙により探索範囲が広大になってしまうことが問題になる。そのため、図 1 中 step3 のシミュレーション中に、適用可能な文法の確率分布を設定し、その分布に従って文生成に用いる文法や語彙などをサンプルングする。その内容を以下に示す。

- 非終端記号
PCFG の確率を使用する。
- SVO
word2vec により与えられた単語との cos 類似度を使用する。cos 類似度の正規化式を式 (2) に示す。

$$\frac{\exp(\beta r)}{\sum(\exp(\beta r))} \quad (2)$$

ここで、 r は cos 類似度であり、 $\beta = 2.0$ とした。

- SVO 以外の自立語
与えられた入力情報を W (e.g. {dog, eat, bread}) とした時の単語分布 $p(v|W)$ を求めることを考えた。 W と共起しやすい単語を求めるため、LDA (Latent Dirichlet Allocation [3]) を用いた。LDA で学習済みのモデルから新規文書 W のトピック分布 θ_{new} を求め、これを $p(\theta_{new}|W)$ とす

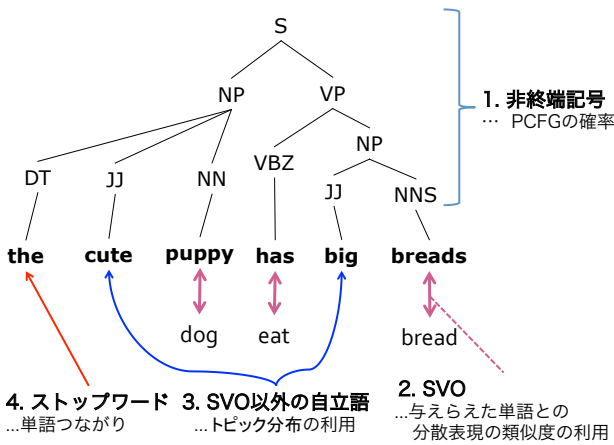


図 2: サンプリング対象となる確率分布

る. $p(v|W)$ は式 (3) より求める. また, $p(v|\theta_{new})$ はトピック毎の単語分布 ϕ を用い, 式 (4) より求める.

$$p(v|W) = \int p(v|\theta_{new})p(\theta_{new}|W)d\theta_{new} \quad (3)$$

$$p(v|\theta_{new}) = \sum_k \phi_{k,v}\theta_{new,k} \quad (4)$$

- ストップワード
単語の前後の繋がりを考慮し, bigram と逆 bigram 確率を求め, それらの平均値を使用した.

上記の場合分けについてのイメージを図 2 に示す. 以上のように, 適切な確率分布に基づいてサンプリングを行うことで, 質の良い文生成が可能となる.

3.3 文の評価

図 1 中 step4 における生成文のスコアの設定方法を説明する. 以下の 2 つの視点から評価を行った.

- 生成内容
ゴールとして与えた単語と生成文の SVO の単語について word2vec による分散表現を求め, それらの cos 類似度を使用した.
- 文としての正しさ
単語の繋がりの良さ
文の容認度を測る指標として適切だとされたスコアである *Acceptability*[4](式 (5)) を用いた.

$$Acceptability(s) = \log \left(\frac{p_{model}(s)}{p_{uni}(s)} \right)^{\frac{1}{|s|}} \quad (5)$$

p_{uni} は, unigram の値である. また, p_{model} として, Kneser-Ney スムージング [5] による 3 グラムを使用した. このスコアを以下, AP と呼ぶ. 構文木の構造の正しさ

構文木の部分木を素性としたロジスティック回帰による識別器を使用した. 正例はコーパス中の構文木とし, 負例は文法をランダム適用して生成した構文木を疑似的な負例として用いた. この識別器により得られる値を, 以下 SP と呼ぶ.

3.4 勝敗判定

図 1 中 step4 における勝敗決定方法を述べる. 3.3 で説明した文の評価値を用いて以下のように決定する.

1. SVO の cos 類似度が 0.5 以下のものがあるとき, 負けとする.
2. SVO の cos 類似度が全て 0.5 以上であり, SVO の cos 類似度と, AP, SP がそれぞれ他候補ノードのそれらの平均値よりも大きいとき, 勝ちとする.

4 実験

4.1 実験設定

データには Microsoft Research Video Description Corpus(MVDC) ¹ を用いた. LDA, AP, word2vec の学習コーパスに MVDC 中の英文全て 85,413 文を用いた. 探索範囲となる PCFG と語彙情報については, MVDC から 30 文を選択し Stanford parser ² を用いて作成した. PCFG の文法数は 39 であり, 語彙数は 97 となった. また, 文生成の内容として, 入力情報に $W = \{dog, eat, bread\}$ を与えた. 制約条件として, 文長は 3 ~ 5 とし, その範囲外のときは負けとした.

4.2 実験結果

生成文例を以下に示す.

```
running puppy eats and bread
puppy panda eats his bread
eating dog eats a bread
dog chef eats bread
dog baby eats his bread
```

以上のように様々な言い回しの文が生成され, SVO に関しては正しく語彙選択されたが, 文として意味的に解釈し難い文も生成された. このことから, 文の評価において不十分な点があるといえる.

5 おわりに

本研究では, 確率文脈自由文法を用いた構文木構築において MCTS を用い, 統語構造を考慮した, 非決定的な文生成手法を提案した. 効果的に MCTS の探索を行うために, 探索の進め方やサンプリング確率分布の設定などを工夫した. 実験では, 与えた生成内容に従った様々な文が生成されることを確認したが, 意味的に解釈し難い文も生成された. 今後は, 文の評価を工夫し, より自然な文の生成を実現したい.

参考文献

- [1] Monte Carlo Tree Search (MCTS) research hub, <http://mcts.ai/>
- [2] P.Auer, N.Cesa-Bianchi, and P.Fischer, Finite-time analysis of the multi-armed bandit problem, *Machine Learning*, 47:235-256, 2002.
- [3] D. M. Blei, Andrew Y. Ng, M. I. Jordan, Latent Dirichlet Allocation, *Journal of Machine Learning Research* 3, pp. 993-1022, 2003.
- [4] J.H.Lau, A.Clark, and S.Lappin, Unsupervised Prediction of Acceptability Judgements, In *Proceedings of the 53rd ACL*, 2015.
- [5] R. Kneser and H.Ney. Improved backing-off for n-gram language modeling. In *Proceedings of ICASSP*, Vol. 1, pp.181-184, 1995.

¹ <https://www.microsoft.com/en-us/download/details.aspx?id=52422>

² <http://nlp.stanford.edu/software/lex-parser.shtml>