

# 文法変分自己符号化器を用いた自然言語文の連続的な統語構造変換

折口希実 (指導教員: 小林一郎)

## 1 はじめに

深層生成モデルとは、変分自己符号化器 (Variational Autoencoder) [1] や敵対的生成ネットワーク (Generative Adversarial Network) [2] が代表例として挙げられる深層ニューラルネットワークで構成された生成モデルである。深層生成モデルでは入力するデータは画像や音響などの連続なデータが一般的であるが、化学式や数式などの離散なデータを扱い、生成するためには離散な構造を持ったデータを深層学習の枠組みで連続的な形へ変換する必要がある。

本研究では、深層学習の枠組みで、文脈自由文法によって表される自然言語の統語構造を連続的に変換することを試みる。

## 2 文法変分自己符号化器

Kusner ら [3] は、文法変分自己符号化器 (Grammar VAE) を提案した。このモデルは、分子を対象としており、分子の化学構造を文字列化する SMILES 記法を用いている。GrammarVAE の仕組みを図 1 に示す。

GrammarVAE は連続なデータを扱う VAE を応用し、文法の生成規則を用いることで離散なデータを扱えるようにしている。主な仕組みは同じであるが、潜在空間を連続的に滑らかにすることによってそれを可能にしている。

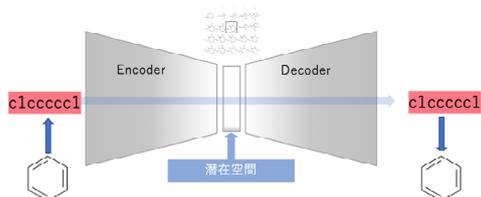


図 1: GrammarVAE の概要

## 3 統語構造を対象とした GrammarVAE

本研究では、自然言語の統語構造を対象としたモデルを生成した。その encoder と decoder の概要を図 2, 図 3 に示す。また、文法に関しては自然言語の構文解析に広く応用されてきた文脈自由文法を用いている。

### 3.1 GrammarVAE の仕組み

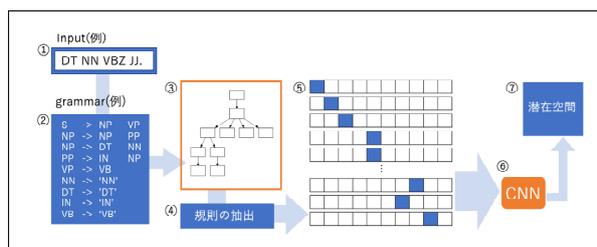


図 2: 自然言語の統語構造を適用した GrammarVAE の Encoder

**Encoding** 図 2 に示している流れが encoder の動きである。①にあるような自然言語の統語構造文を入力とする。また、②にあるような文法の生成規則を定義しておく。まず、入力した統語構造を文法に従って解析を行い、③に示すような木構造へ変換する。そしてその木構造から前順走査し、その解析で用いられた規則だけを抽出する。抽出した規則を⑤のような 1-hot ベクトルに変換する。このとき、抽出した規則の数だけベクトルは存在し、そのベクトルごとに全規則のうち抽出したベクトルが該当するようになっていく。⑥では⑤で生成した複数の 1-hot ベクトルを 1 次元畳み込みニューラルネットワーク (Convolutional Neural Network) を用いて畳み込み、潜在空間へ落とし込む。

**Decoding** 図 3 に示している流れが decoder の動きである。まず潜在空間からベクトルを抽出する。再帰型ニューラルネットワーク (Recurrent Neural Network) を用いて③のように複数正規化されていないベクトル ('logits') を生成し、Last-in First-out (LIFO) スタックを用いて⑤に示すような生成規則をサンプリングする。

サンプリングした生成規則の終端記号から⑥のような統語構造を生成する。

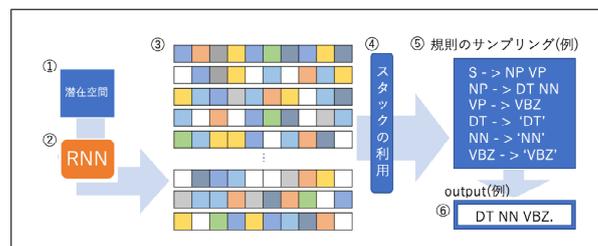


図 3: 自然言語の統語構造を適用した GrammarVAE の Decoder

**Training** Algorithm 1 に training アルゴリズムを示す。encoder での出力を  $X$ 、定義された生成規則の総数を  $K$  とする。decoder では、RNN の時間ステップ  $t$  の最大値を  $T_{max}$  とし生成された logits の集合は行列  $\mathbf{F} \in \mathbb{R}^{T_{max} \times K}$  と表せる。文法における非終端記号を  $\alpha$  とし、logits をマスクするベクトルを  $\mathbf{m}_\alpha \in [0, 1]^K$  としている。次の分布の式 (1) は時間ステップ  $t$  でマスクされた logits から生成規則を抽出するためのものである。また、 $f_{tk}$  は行列  $\mathbf{F}$  の  $(t, k)$  要素を指す。

$$p(\mathbf{x}_t = k | \alpha, \mathbf{z}) = \frac{m_{\alpha,k} \exp(f_{tk})}{\sum_{j=1}^K m_{\alpha,j} \exp(f_{tj})} \quad (1)$$

### 3.2 連続的な統語構造変換

GrammarVAE において、潜在空間はガウス分布によって表されている。潜在空間の潜在変数を徐々に変化させることにより、連続的な統語構造変換することが可能である。

### Algorithm 1 Training the GrammarVAE

Input: Dataset  $\{\mathbf{X}^{(i)}\}_{i=1}^N$

Output: Trained VAE model  $p_\theta(\mathbf{X}|\mathbf{z})$ ,  $q_\phi(\mathbf{z}|\mathbf{X})$

**while** VAE not converged **do**

Select element:  $\mathbf{X} \in \{\mathbf{X}^{(i)}\}_{i=1}^N$  (or minibatch)

Encode:  $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{X})$

Decode: given  $\mathbf{z}$ , compute logits  $\mathbf{F} \in \mathbb{R}^{T_{max} \times K}$

**for**  $t$  in  $[1, \dots, T_{max}]$  **do**

Compute  $p_\theta(\mathbf{x}_t|\mathbf{z})$  via Eq.(1), with mask  $\mathbf{m}_{xt}$   
and logits  $\mathbf{f}_t$

**end for**

Update  $\theta, \phi$  using estimates  $p_\theta(\mathbf{X}|\mathbf{z})$ ,  $q_\phi(\mathbf{z}|\mathbf{X})$ , via  
gradient descent on the ELBO

**end while**

## 4 実験

### 4.1 実験設定

画像とキャプションのデータセットである Microsoft COCO [4] の中からキャプションのみを取り出し、中でも著者の観点から比較的平易と思われる統語構造を持つキャプション 6,668 文を実験データとした。GrammarVAE におけるコードは深層学習用ライブラリである Keras を用いている。実験を行った際の設定を表 1 に示す。

表 1: 実験設定

データセット	Microsoft COCO
データ数	6668 文
train:epoch 数	40
勾配法	Adam
損失関数	二値交差エントロピー (BSE)

**生成規則の抽出** 自然言語に対する Grammar VAE における文法に文脈自由文法 (CFG) を採用したため、その生成規則を抽出する必要がある。そこで CFG 解析と係り受け解析を同時に行う構文解析器である Stanford Parser<sup>1</sup> を用いて構文解析を行い、その結果から本実験で使用する 214 個の生成規則を取り出した。また、モデルの入力が CFG に基づく統語構造であることから、生成規則の終端記号は品詞としている。

### 自然言語文の統語構造化

自然言語の汎用ツールである Stanford CoreNLP を用いてキャプションを構文解析し、統語構造に変換したものを入力とする。

### 4.2 実験結果

GrammarVAE を用いて連続的に統語構造変換した結果を図 6, 7 に示す。潜在空間上での連続的な変換が行われているか視覚的にわかるよう木構造によって示している。入力文となる統語構造を次の図 4, 5 に示す。

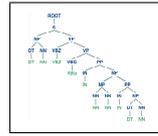


図 4: 入力文 1

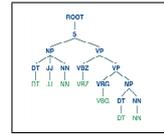


図 5: 入力文 2

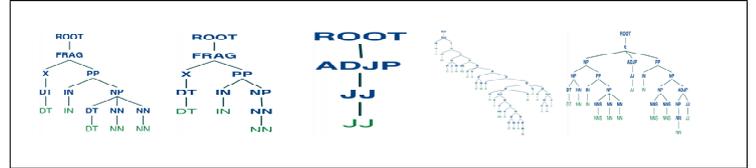


図 6: 1 に対する潜在空間における連続的な統語構造

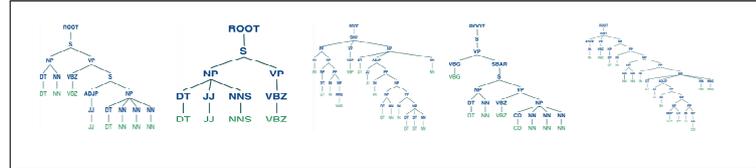


図 7: 2 に対する潜在空間における連続的な統語構造

ガウス分布の分散の変化による統語構造の変化を示している。分散の値については、入力文の構造を  $\sigma = 0.0$  として左から右に約 0.01 ずつずらしたものになっている。

### 4.3 考察

図 6, 7 より、部分的に連続的な統語構造の生成ができてきているものの、大部分ではまだ連続的に統語構造の変換をできてはいないということがわかる。そもそも統語構造の生成自体の精度が低く向上させる必要がある。単純な構造から急に複雑な構造へ変換している箇所も多く見受けられ、データ数の不足によるモデルの学習不足や、潜在変数の変化量などの影響も考えられる。また、統語構造の元の自然言語文は画像に対するキャプションであり、名詞句だけの文も多く存在しているため、余計に GrammarVAE での統語構造の生成を難しくしているという可能性も考えられる。

## 5 おわりに

自然言語文の統語構造を対象にした文法変分自己符号化器を用いて連続的な統語構造変換を試みた。今後の課題として、より連続的で滑らかな潜在空間を生成するためにデータ数や次元を増やしたりより緩やかに潜在変数を変化させるなど、様々な工夫をしながら精度の高い統語構造変換を実現していきたい。

## 参考文献

- [1] Kingma, Diederik P and Welling, Max. Auto-encoding variational Bayes. In Proceedings of the International Conference on Learning Representations (ICLR), 2014.
- [2] Goodfellow, Ian and Pouget-Abadie, Jean and Mirza, Mehdi and Xu, Bing and Warde-Farley, David and Ozair, Sherjil and Courville, Aaron and Bengio, Yoshua. Generative Adversarial Nets (NIPS2014), 2014.
- [3] Kusner, M. J.; Paige, B.; and Hernández-Lobato, J. M. 2017. Grammar variational autoencoder. In Proceedings of the 34th International Conference on Machine Learning- Volume 70, 1945-1954. JMLR. org.
- [4] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In ECCV, 2014.

<sup>1</sup><https://nlp.stanford.edu/software/lex-parser.shtml>