Transformerを用いた化合物名から化学構造への変換

表 悠太郎 [1,3]　松下 京群 [2,3]　岩倉 友哉 [2,3]　田村 晃裕 [1,3]　二宮 崇 [1,3]

[1] 愛媛大学 大学院理工学研究科 電子情報工学専攻

[2] 株式会社富士通研究所

[3] 理研 AIP-富士通連携センター

[1]{omote@ai., tamura@, ninomiya@}cs.ehime-u.ac.jp

[2]{m.kyoumoto, iwakura.tomoya}@fujitsu.com

1 はじめに

化学物質の知識は，新材料や新薬の開発，材料を用いた製品開発に用いられている．化学物質の知識を活用するために，論文や特許で数分ごとに報告される化学物質の物性値や化学物質間の相互関係といった情報を構造化する作業が行われている．[1] その中で，表記は異なるが同一の化合物を統合するため，化合物名と化学構造を紐づけた情報が活用される．

化合物のデータベースとして PubChem[2]などがあるが，既知の化合物は約 10 億件あると言われているのに対し，PubChem などの大きなデータベースでもせいぜい 1 億件程度しか登録されていない[3]．さらには，化合物のデータベース化の一部は人手で行われており，作成に時間とコストがかかる．そのため，日々刻々と誕生する新規化合物の登録が間に合わないのが現状である．

化合物名称からの構造予測の既存手法として，ルールベースによる変換手法 [1] があるが，ルールベースでは規則に含まれない化合物名を変換することができない．例えば，ルールベースによる変換手法では化合物名の体系的命名法である IUPAC [2] に準拠した化合物名であれば高い精度で変換ができる．しかし，IUPAC の命名規則が非常に膨大で複雑であり，化学文書や特許文書中でも命名規則に違反した名称が散見され，実際の文書中に登場する化合物に対して変換できない場合も少なくない (3 節参照)．

本研究では，系列変換ニューラルネットワークモデルを用いて化合物名から Simplified Molecular Input Line Entry System (SMILES) [3] への変換精度の改善を試みる．系列変換ニューラルネットワークモデルを用いた場合，IUPAC の命名規則に違反した化合物名に対しても何らかの構造表記を予測することができ，ルールベースでは網羅しきれない化合物名に対しても正しく構造予測できると期待される．本研究では，系列変換ニューラルネットワークモデルの中でも様々なタスクで高い精度を達成している，Transformer モデル [4] に着目する．また，Transformer モデルに基づく手法の精度改善のために，構造表記に含まれる元素記号数に対する制約を加えた手法及び SMILES の変換を InChI (IUPAC International Chemical Identifier) [2] への変換と共に行うマルチタスク学習を提案する．

本研究では PubChem からデータセットを作成し，既存手法と提案手法の Synonyms(IUPAC に準拠しない化合物名も含む) に対する構造予測の精度比較を行った．結果としてルールベースの既存手法に対し，Transformer を用いた変換手法の方が高い精度を達成した．特に，SMILES，InChI への変換のマルチタスク学習を行った場合に最も高い精度を達成した．

2 SMILES

SMILES は，化学情報処理向けに設計された化学構造表記法である．分子グラフ理論の原理に基づき，自然言語に似た記号の線形表記で分子構造を表すため，人間から見ても解釈が比較的容易なものとなっている．SMILES は原子およびその結合，分岐，環等を表す記号から構成されており，基本的には分子の 2 次元構造を線形表記したものである．SMILES の例を図 1 に示す．化学構造と SMILES の紐付けを一意に定めるために，本研究では正準化された SMILES である Canonical SMILES を用いる．

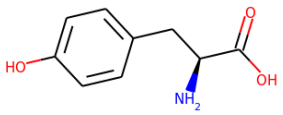
3 従来手法

OPSIN (Open Parser for Systematic IUPAC Nomenclature) [1] は化合物名 (IUPAC 名) を対象に化合物名から SMILES や InChI などを生成するルールベースのパーサである．OPSIN では BNF を用いて化合物名をトーカナイズし，あらかじめ記述しておい

[1] https://www.jaici.or.jp/annai/img/20150709_CAS_PressRelease.pdf

[2] https://pubchem.ncbi.nlm.nih.gov/

[3] https://pubchemdocs.ncbi.nlm.nih.gov/statistics



N[C@@H](Cc1ccc(O)cc1)

図 1: L-tyrosine の化学構造 (上) と SMILES(下)

た部分構造や結合ルールに当たるものを抽出する．その後，それらを用いて再構築することで，化合物名を化学構造へと変換している．

先述したように，論文などに記載される化合物名は IUPAC 名などの体系的命名法に従わないものも多く，ルールベースの手法ではこれらすべてに対応するのは困難である．実際にルールベースが主だと思われる既存手法では本研究で作成したデータセット中の IUPAC 名に対して 86.2% ～93.0%の割合で正しい変換が可能であるが，Synonyms に対しては 65.3% ～71.1%の変換精度に落ちることが確認された．

4 提案手法

4.1 Transformer モデルによる化合物名から SMILES への変換

本節では，本稿で提案手法のベースラインとなる Transformer モデルについて述べる．

Transformer は，系列中のトークン間の関連度を捉える Self Attention 構造を持つエンコーダとデコーダから構成されるモデルである．エンコーダでは，入力系列から中間表現を獲得し，デコーダでは，獲得した中間表現から出力系列を予測し，出力する．

Transformer は，エンコーダレイヤとデコーダレイヤがそれぞれ複数層スタックされたエンコーダ･デコーダ構造を持つ．エンコーダとデコーダでは，まず，埋込み層で入力トークン列（エンコーダ側は入力系列の，デコーダ側は出力系列）を埋込み表現を表す行列に変換する．その後，Positional Encoding により単語の系列位置情報を付与する．具体的には，入力トークン列の埋込み表現行列に対して，各トークンの系列における絶対的な位置情報をエンコードした行列 PE を加える．PE の各成分は異なる周波数の sin，cos 関数を用いて次式により算出したものである．

$$PE(pos, 2i) = \sin(pos/10000^{2i/d_{model}}) \quad (1)$$

$$PE(pos, 2i+1) = \cos(pos/10000^{2i/d_{model}}) \quad (2)$$

ここで，d_{model} は入力トークンの埋込み次元，pos はとクーンの位置，i は各成分の次元を表す．トークン埋込み表現行列に PE を加えたものが，第 1 層目のエンコーダレイヤやデコーダレイヤの入力となる．

エンコーダレイヤは，下位のサブレイヤから順に，入力系列中のトークン間の関連度を捉える Self Attention，位置ごとのフィードフォーワードネットワーク（Feed Forward Nerwork; FFN）の 2 つのサブレイヤで構成されている．デコーダレイヤは，下位のサブレイヤから順に，出力系列のトークン間の関連度を捉えるマスキング付き Self Attention，入力系列のトークンと出力系列のトークン間の関連度を捉える Attention（Source-Target Attention），位置ごとの FFN の 3 つのサブレイヤで構成されている．Self Attention と Source-Target Attention は Multi-Head Attention を用いて実現される．Multi-Head Attention では，まず，入力長 N の 3 つの入力行列 $Q, K, V \in \mathbb{R}^{N \times d_{model}}$ を重み行列 $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d_{model} \times d_z} (i = 1, \ldots, h)$ により，d_{model} 次元から d_z 次元に線形写像した後，h 個の内積 Attention を計算する．ここで，d_{model} は元々の入力ベクトルの埋込み次元であり，$d_z = d_{model}/h$ である．また，それぞれの内積 Attention をヘッド（H_i $(i = 1, \ldots, h)$）と呼ぶ．

$$MultiHead(Q, K, V) = Concat(H_1, H_2, \ldots, H_h)W^O$$

$$H_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

その後，各ヘッドを連結した後，重み行列 $W^o \in \mathbb{R}^{d_{model} \times d_{model}}$ で線形写像する機構が Multi-Head Attention である．

Transoformer モデルを用いた化合物名から SMILES への変換手法では，入力を化合物名，出力を SMILES として学習を行う．具体的には式 (3) に示した目的関数が最小になるように学習を行う．ここで，$\boldsymbol{\theta}_{enc}$ は化合物名エンコーダのパラメータ，$\boldsymbol{\theta}_{smiles}$ は SMILES デコーダのパラメータ，$X = x_1, x_2, \ldots, x_n$ は化合物名，$T = t_1, t_2, \ldots, t_m$ は正解 SMILES を表す．

$$\mathcal{L}_s = -\log P(T|X; \boldsymbol{\theta}_{enc}, \boldsymbol{\theta}_{smiles}) \quad (3)$$

4.2 SMILES に含まれる元素記号数に対する制約

化合物名から化学構造を正しく予測するためには，少なくとも化学構造に含まれる各元素の原子数を完全に一致させる必要がある．そこで本稿では，Transformer モデルに基づく変換手法に対し，訓練時に予測した SMILES と正解 SMILES の間の元素記号数の差を制約として用いる手法を提案する．SMILES デコーダが出力した SMILES に含まれる元素記号数をそのまま用

いて 2 乗誤差を算出してしまうと，SMILES に含まれるトークンは離散的であり，微分不可能であるために誤差逆伝搬ができない．そのため，本手法ではデコーダが出力した単語確率分布に対して Gumbel Softmax[5] を適応し，サンプリングされた pseudo one-hot ベクトルの和から予測 SMILES に含まれる元素記号数を算出することでこの問題に対処する．つまり，Transformer が出力した SMILES 中の i 番目のトークンの単語確率分布 $\boldsymbol{\pi_i} = \pi_{i1}, \pi_{i2}, \ldots, \pi_{i|\mathcal{V}|}$ に対応する one-hot ベクトル $\boldsymbol{y_i} = y_{i1}, y_{i2}, \ldots, y_{i|\mathcal{V}|}$ の各要素は以下のように算出される．ここで，$\mathcal{V}$ は SMILES の語彙集合を表す．τ は Gumbel Softmax におけるハイパーパラメータである．τ が小さくなるほど分布は one-hot に近づき，大きい τ では一様分布に近づく．本研究では τ の値を 0.1 と設定する．

$$
\begin{aligned}
y_j &= \frac{\exp\left((\log(\pi_j) + g_j)/\tau\right)}{\sum_{k=1}^{|\mathcal{V}|} \exp\left((\log(\pi_k) + g_k)/\tau\right)} \\
g_j &= -\log(-\log(u_j)) \\
u_j &\sim \mathrm{Uniform}(0, 1)
\end{aligned} \tag{4}
$$

上記のことを踏まえ，元素記号数に対する制約は式 (5) のように計算する．ここで，A は SMILES で想定される元素記号集合，λ_a は $\mathcal{L}_a$ を考慮する度合いをコントロールするためのハイパーパラメータ，$N_a(T)$ は SMILES のトークン列 T に含まれる元素記号 a の数を返す関数，$idx(a)$ は語彙集合 $\mathcal{V}$ における元素記号 a のインデックスを返す関数である．

$$
\begin{aligned}
\mathcal{L}_a &= \frac{1}{|A|} \sum_{a \in A} (N_a(T) - \boldsymbol{y}_{idx(a)})^2 \\
\boldsymbol{y} &= \boldsymbol{y_1} + \boldsymbol{y_2} + \cdots + \boldsymbol{y_m}
\end{aligned} \tag{5}
$$

元素記号数の制約を含め，目的関数は式 (6) のようになる．

$$
\mathcal{L}_s + \lambda_a \mathcal{L}_a \tag{6}
$$

4.3 SMILES,InChI への変換のマルチタスク学習

SMILES と InChI は異なる情報に着目して化学構造を線形表記に起こしているため，それぞれの線形表記を同一の中間表現から予測させることで，より高い表現能力を有した中間表現が得られる可能性がある．

そこで，4.2 節で提案した元素記号数に対する制約の他に，化合物から SMILES への変換と化合物から InChI への変換を同時に学習する手法も提案する．具体的には，式 (7) に示した目的関数が最小になるように学習を行う．ここで，$\boldsymbol{\theta}_{inchi}$ は InChI デコーダのパラメータを表し，λ_{inchi} は $\mathcal{L}_{inchi}$ を考慮する度合いをコントロールするためのハイパーパラメータである．

$$
\begin{aligned}
&\mathcal{L}_s + \lambda_{inchi} \mathcal{L}_{inchi} \\
&\mathcal{L}_{inchi} = -\log P(I|X; \boldsymbol{\theta}_{enc}, \boldsymbol{\theta}_{inchi})
\end{aligned} \tag{7}
$$

表 1: 開発・テストデータ件数

	IUPAC	Synonyms
テスト	182,291	11,194
開発	18,267	1,343

5 実験

5.1 データ

本研究で使用したデータはいずれも (化合物名，正解 SMILES) の組で構成される．それぞれ PubChem のダンプデータ[4]より，化合物名は各 CID に紐づいた IUPAC 名及び Synonyms を，正解 SMILES は Isomeric SMILES を RDKit[5]を用いて Canonical SMILES に変換したものである．PubChem に記載されている Synonyms は IUPAC 名，慣用名のほか化合物データベースの ID など様々なものを含むことに注意されたい．また，Isomeric SMILES を用いたのは SMILES に対応する CID の重複が最も少なかったためである．

開発データおよびテストデータは，ダンプデータ中から CID をランダムに選び，CID ごとに編集距離が最も遠い 2 つの化合物名のみを使用した．またテストデータの Synonyms に関しては，化合物データベースの ID のようなものは正規表現を用いて人手で取り除いた．それぞれのデータ数を表 1 に示す．訓練データはダンプデータのうち，開発データおよびテストデータに用いた (化合物, SMILES) の対と重複しないデータから，OPSIN が内部に保有する Parser でトーカナイズできた化合物名の中から PubChem 上で Synonym と分類されている化合物名 3,000,000 件を用いた．

5.2 トーカナイザ

訓練データ，開発データ，テストデータに含まれる化合物名は，訓練データ中の化合物名を用い，fastBPE[6] でバイトペア符号化 (Byte Pair Encoding; BPE) を学習し，その BPE 辞書を用いてサブワード化を施した．BPE の結合回数は 2,500 とした．また SMILES

[4] ftp://ftp.ncbi.nlm.nih.gov/pubchem/
[5] https://github.com/rdkit/rdkit
[6] https://github.com/glample/fastBPE

表 2: Synonyms に対する各変換器の評価結果

method	recall	precision	fscore	validity	entire	MISS	ERROR	IGNORE
opsin	0.693	0.836	0.758	0.829	11194	1911	1	2175
tool A	0.711	0.797	0.751	0.893	11194	1147	51	2175
tool B	0.653	0.800	0.719	0.816	11194	2055	7	2175
transformer	0.780	0.792	0.786	0.984	11194	0	174	2175
atomnum(0.25)	0.781	0.795	0.788	0.982	11194	0	201	2175
inchigen(0.25)	0.797	0.807	0.802	0.987	11194	0	150	2175

は，元素記号を 1 トークンとし，それ以外の結合を表す記号等はキャラクタ単位で分割した．

5.3 モデルのハイパーパラメータ

Transformer のハイパーパラメータは，エンコーダ及びデコーダレイヤのスタック数は 6，ヘッド数は 8，埋め込み次元は 512，dropout の確率は 0.1 と設定した．また，目的関数の $\mathcal{L}_s$ および $\mathcal{L}_i$ はラベル平滑化交差エントロピーを用い，平滑化のための ϵ は 0.1 とした．また学習率は 4,000 回更新時で 0.0005 となるように線形的に増加させ，以降は更新回数の平方根の逆数に比例して減衰させた．Optimizer は Adam を用い，$\beta_1 = 0.9, \beta_2 = 0.98, \epsilon = 10^{-8}$ とした．また，モデルのパラメータ更新回数は 300,000 回とした．また，制約を考慮する度合いを定めるハイパーパラメータはそれぞれ，$\lambda_a = 0.25, \lambda_{inchi} = 0.25$ と定めた．

5.4 結果

成否判定は変換器が生成した SMILES と正解 SMILES を RDKit を用いて読み込み，Canonical かつ Isomeric な SMILES として出力した文字列の完全一致で行った．また，各変換器の評価指標として以下の 4 つを使用した．

$$\begin{aligned} \text{recall} &= \frac{\text{MATCH}}{\text{entire}} \\ \text{precision} &= \frac{\text{MATCH}}{\text{entire} - \text{MISS} - \text{ERROR}} \\ \text{validity} &= \frac{\text{MATCH} + \text{MISTAKE}}{\text{entire}} \\ \text{fscore} &= \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \end{aligned}$$

ここで，MATCH は正解 SMILES と生成 SMILES が完全一致した件数，entire はテストデータ数，MISTAKE は文法的に正しい SMILES を出力したが，正解 SMIlES と不一致であった件数，MISS は SMILES を出力しなかった (化合物名の解析に失敗した) 件数，ERROR は文法的に間違った (RDKit で Mol オブジェクトへの変換に失敗する)SMILES を出力した件数，IGNORE は取り除いた ID などの文字列の件数である．

表 2 に結果を示す．表 2 中の toolA,B は有償で利用可能な 2 つの商用ツール，atomnum は節 4.2 の元素数制約手法，inchigen は節 4.3 のマルチタスク学習の手法を示す．表 2 より，提案手法は化合物名から化学構造への変換において，既存のルールベースの手法や Transformer よりも高い fscore になることがわかった．

6 おわりに

本研究では，化合物名から化学構造への変換において，Transformer をベースとし SMILES に含まれる元素記号数に対する制約と SMILES，InChI への変換のマルチタスク学習を提案した．そして，既存手法よりも Transformer をベースに用いた提案手法の方が fscore で優位であることを確認した．一方で，precision に関しては既存手法よりも低いケースが見られ (表 2)，Precision の改善が課題として残っている．専門的な知識がなければ複雑な化合物名から化学構造を書き起こすことは難しく，そのような知識のないエンドユーザにとって Precision は重要であり (例えば，化学関連文書の検索システムを構築するエンジニアが必ずしも化学に詳しいとは限らない)，今後の重要な課題と位置づけ改善していきたい．

参考文献

[1] D. M. Lowe, P. T. Corbett, P. Murray-Rust, and R. C. Glen. Chemical name to structure: Opsin, an open source solution. *Journal of Chemical Information and Modeling*, Vol. 51, No. 3, pp. 739–753, 2011.

[2] S. R. Heller, A. McNaught, I. Pletnev, S. Stein, and D. Tchekhovskoi. Inchi, the iupac international chemical identifier. *Journal of cheminformatics*, Vol. 7, No. 1, p. 23, 2015.

[3] D. Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, Vol. 28, No. 1, pp. 31–36, 1988.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In *Proc. of NIPS 2017*, pp. 5998–6008. 2017.

[5] E. Jang, S. Gu, and B. Poole. Categorical reparameterization with gumbel-softmax. *ArXiv*, Vol. abs/1611.01144, , 2016.