反復隣接木構造による化学化合物の機能予測について

甲斐 蒼一朗[1,a)]　正代 隆義[1,b)]

概要：反復隣接木構造とは，グラフの頂点の隣接関係の情報をラベルなし木構造として表現するグラフ構造パターンである．本論文では，化学化合物グラフのデータベースを対象として，反復隣接木構造による化学化合物の機能予測実験を行った．データベースに現出するラベルなし反復隣接木構造を辞書化して，その辞書を用いて機能予測実験を行ったので，その結果を報告する．

キーワード：知識表現，機械学習，グラフアルゴリズム，頻出パターン

Functional Prediction of Chemical Compounds by Iterative Adjacent Tree Structures

Soichiro Kai[1,a)]　Takayoshi Shoudai[1,b)]

Abstract: Iterative adjacency tree structure is a graph structure pattern in which the adjacency information of vertices of a given graph is represented as an unlabeled tree structure. The structure is called a *fingerprint*. In this paper, we conducted an experiment on predicting the functions of chemical compounds using iterative adjacency tree structures for a database of chemical compound graphs. We report the results of our experiments on the prediction of functions of chemical compounds by using fingerprints which appear in the database.

Keywords: Knowledge representation, machine learning, graph algorithms, frequent patterns

1. はじめに

薬理学の分野では化学化合物の分子構造を解明するために，原子を頂点，原子間の化学結合を辺としたグラフマイニングが行われている．また，社会学におけるソーシャルネットワークの分析や，Web ページのリンク構造を表す Web グラフを利用した Web コミュニティの抽出など，グラフとして表現したデータを直接解析することを目的としたネットワーク科学が急速に発展している．このようなグラフマイニングにおける課題のひとつとして，頻出部分グラフマイニングがある．この問題は gSpan[4] をはじめとして非常に良く研究されている．

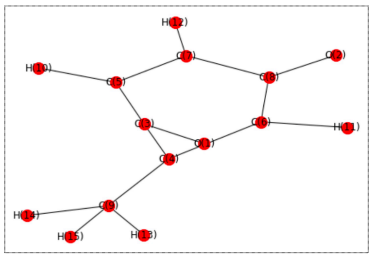

図 1　NCI Chemical Database に登録された化学化合物のグラフ表現の例

図 1 にグラフの例を挙げる．このグラフは NCI（米国国立がん研究所, National Cancer Institute）が公開している薬理学上の化学化合物データベース NCI Chemical Database に登録された 265,242 個の化学化合物の 1 つである．グラフがどれだけ木に近いかを表すグラフの特徴量としてグラフの木幅がある．薬理学の化学化合物をグラ

[1] 福岡工業大学情報工学部, 811-0295 福岡県福岡市東区和白東 3-30-1
Department of Computer Science and Engineering, Fukuoka Institute of Technology, 3-30-1 Wajiro-higashi, Higashi-ku, Fukuoka, 811-0295 Japan.
[a)] s18a1302@bene.fit.ac.jp
[b)] shodai@fit.ac.jp

フとしてみたとき，木幅が小さいグラフが多いことはよく知られている．例えば，NCI Chemical Database に登録されている化学化合物のうち 97.57%が木幅 2 以下である．Yamasaki ら [3] は化学化合物グラフが木に近いことを用いて，化学化合物のグラフ構造パターンを定義し，効率の良い機械学習アルゴリズムを与え，NCI Chemical Database を用いてその表現力を実証した．このような研究背景の中，本論文ではグラフの隣接情報を木構造として表現するラベルなし反復隣接木構造を扱う．反復隣接木構造とは，グラフの頂点の隣接関係の情報をラベルなし木構造として表現するグラフ構造パターンである．

化学化合物に対するラベルあり反復グラフ構造としては Rogers と Hahn[2] が提案した拡張された連結部分構造がある．[2] で定義された部分構造は Extended-Connectivity Fingerprints (ECFPs) と呼ばれる．本論文では，ラベルなし反復隣接木構造を定義し，[2] にならって Fingerprint (FP と略す) と呼ぶ．そして，グラフデータベースを FP に基いて辞書化し，さらにその辞書を用いた化学化合物の機能予測を行う．

本研究の目的は，NCI Chemical Database に含まれる 265,242 個の化学化合物のうち，AIDS に対する機能の有無がラベル付けされた 42,689 個の化学化合物に対して FP の辞書化を行うことである．そして，各々の化学化合物が有する FP の個数を学習データとして機能予測実験を行う．本実験では，原子番号等のグラフの頂点や辺に関する特徴量やラベル情報を用いない．すなわち，化学化合物が持つグラフ構造のみで有効な特性を持つ化学化合物の分類を行う．その際，サポートベクタマシン (SVM)，ニューラルネットワーク (NN)，ランダムフォレスト (RF) の 3 種の学習モデルで各学習モデルの出す実験結果を比較し，各学習モデルが最良の予測精度を比較する．

本論文の構成は以下の通りである．第 2 章では，本論文の準備として，反復隣接木構造としての FP と機能予測実験で使用した NCI Chemical Database に関して述べる．第 3 章では，反復隣接木構造としての FP に基づき，Fingerprint からなるリスト (これを Fingerprint 辞書と呼ぶ．FPD と略す) を作成し，これを使用して行った機能予測実験について述べる．第 4 章では，まとめと今後の課題について述べる．

2. 準備

本章では，整数 $i \geq 0$ に対して，反復隣接木構造としてレベル i の Fingerprint (FP) を定義する．本論文では，リストの要素のインデックスは 0 から始まるものとする．

2.1 Fingerprint 辞書 (FPD) の構成

この節では，ラベルなしグラフ構造の辞書化を行うための手順を示す．

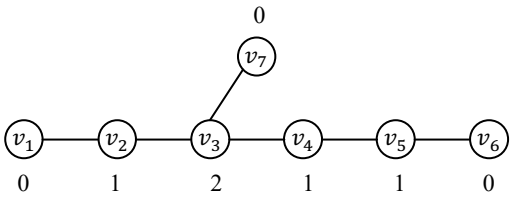

図 2 グラフの例 $G = (V, E)$, ここで，$V = \{v_1, v_2, \ldots, v_7\}$, $E = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_5\}, \{v_5, v_6\}, \{v_3, v_7\}\}$ である．また，各頂点に割り振られた数字は G の次数構造 $D(G) = [[1], [2], [3]]$ のインデックスを表す．

まず，グラフの頂点の次数を基準に次数構造を作る．次数構造は各頂点を昇順に並べて重複を取り除いた際の順位を割り振ったものである．グラフ $G = (V, E)$ の頂点 $v \in V$ の次数を $d_G(v)$ で表す．図 2 のグラフ G においては，$d_G(v_1) = d_G(v_6) = d_G(v_7) = 1$, $d_G(v_2) = d_G(v_4) = d_G(v_5) = 2$, $d_G(v_3) = 3$ である．このグラフで現れる次数 $1, 2, 3$ をその数字ごとにリスト化し，そのリストをリストの要素 (次数) の昇順に並べたリストを，グラフ G の次数構造と呼ぶ．グラフ G の次数構造を $D(G)$ で表す．図 2 のグラフ G では $D(G) = [[1], [2], [3]]$ である．

$G = (V, E)$ をグラフとする．G の頂点 $v \in V$ の隣接頂点の集合を $A_G(v)$ で表す．レベル 0 の Fingerprint (FP) を作る．これは，頂点 v を根とする反復隣接木構造の元となる．グラフ $G = (V, E)$ の各頂点 $v \in V$ の全ての隣接頂点 $u \in A_G(u)$ に対し，次数構造 $D(G)$ において u に対応する次数のインデックスを昇順にリスト化し，先頭 (ヘッド) に v の次数構造 $D(G)$ におけるインデックスを加えたものを v のレベル 0 の FP と定める．図 2 のグラフ G において，頂点 v_3 の隣接頂点 $2, 4, 7$ の次数構造におけるインデックスはそれぞれ $1, 1, 0$ であるので，それを昇順にソートした上でリスト化し，ヘッドに頂点 v_3 のリスト構造におけるインデックスを加えて，頂点 v_3 のレベル 0 の FP を $[2, 0, 1, 1]$ と定める．この操作を全ての頂点に行い，得られた各頂点ごとのレベル 0 の FP を，リストの辞書式順序の昇順で，重複を除いた上で $D(G)$ の末尾に連結したリストを $D_0(G)$ で表す．$D_0(G)$ をグラフ G のレベル 0 の Fingerprint 辞書 (FPD) と呼ぶ．図 2 のグラフ G では $D_0(G) = [[1], [2], [3], [0, 1], [0, 2], [1, 0, 1], [1, 0, 2], [1, 1, 2], [2, 0, 1, 1]]$ となる (図 3)．従って，頂点 v_3 のレベル 0 の FP の $D_0(G)$ におけるインデックスは 8 である．

続いてレベル 1 の FP については，レベル 0 で割り振られた FPD $D_0(G)$ のインデックスをもとに構成する．グラフ $G = (V, E)$ の各頂点 $v \in V$ の全ての隣接頂点 $u \in A_G(u)$ に対し，$D_0(G)$ において u に対応する次数のインデックスを昇順にリスト化し，先頭に v の $D_0(G)$ におけるインデックスを加えたものを v のレベル 1 の FP と定める．各頂点ごとのレベル 1 の FP を，リストの辞書式順序の昇順

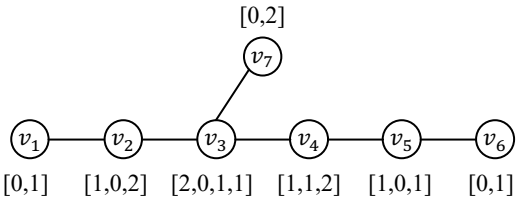

図 3 図 2 のグラフ G のレベル 0 の FP

で，重複を除いた上で $D_0(G)$ の末尾に連結したリストを $D_1(G)$ で表す．図 2 のグラフの各頂点のレベル 1 の FP を図 4 に挙げる．頂点 v_3 のレベル 1 の FP の $D_1(G)$ におけるインデックスは 15 である．

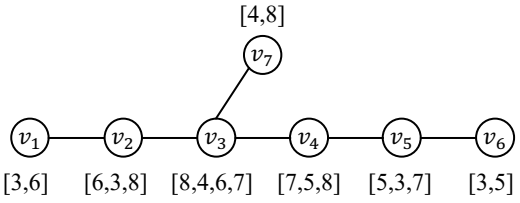

図 4 図 2 のグラフ G のレベル 1 の FP

頂点 v のレベル i　$(i \geq 2)$ の FP については，$D_1(G)$ の構成と同様に，隣接頂点 $u \in A_G(u)$ の $D_{i-1}(G)$ におけるインデックスを昇順にリスト化し，先頭に v の $D_{i-1}(G)$ におけるインデックスを加えたものを v のレベル i の FP と定める．図 2 のグラフでは，頂点 v_3 のレベル 2 の FP は $[15, 11, 13, 14]$ である．$D_2(G)$ におけるそのリストのインデックスは 22 である．

図 2 のグラフ G に対する $D_2(G)$ は次のとおりである．本論文では，グラフ G に対する $D_i(G)$ をレベル i の Fingerprint 辞書 (FPD) と呼ぶ．図 2 のグラフ G に対するレベル 2 の FPD については図 5 のようになる．

2.2 FP によるグラフ構造追跡

本節では，FPD でのグラフ構造追跡について述べる．

図 2 のグラフ G に対するレベル 2 の FPD $D_2(G)$ において，インデックス 13 を持つ頂点 v は次のように隣接構造を追跡できる．$D_2(G)$ においてインデックス 13 の要素は $[6, 3, 8]$ である．すなわち，頂点 v はインデックス 6 を持ち，インデックス 3 と 8 を持つ 2 つの頂点に隣接していることが分かる．さらに，インデックス 6 は $[1, 0, 2]$, インデックス 3 は $[0, 1]$,8 は $[2, 0, 1, 1]$ という FP を持つ．インデックス 0,1,2 は長さ 1 のリストであることから次数を示している．従って，頂点 v はインデックス 1，すなわち次数 2 にあり，隣接頂点はインデックス 0 と 2 の頂点と隣接している．以上のように，FP 辞書を辿ることにより最終的に木構造を得ることができる (図 6).

2.3 NCI Chemical Dataset

本節では，機能予測実験で使用する NCI Chemical Database について説明する．本研究の実験では，AIDS に対しての効果の有無を振り分けた AIDS Antiviral Screen Dataset (https://cactus.nci.nih.gov/download/nci/) に含まれる化学化合物計 42,689 個を使用した．内訳は以下のとおりである．

- Confirmed Active (CA): 423 個（全体の 1.0%）
- Confirmed Moderately active(CM): 1,081 個（全体の 2.5%）
- Confirmed Inactive (CI): 41,185 個（全体の 96.5%）

AIDS に対して CA が十分な効果が見込め，CM が適度に効果が見込め，CI が効果が見込めないという特性をもとに分類されているも．AIDS Antiviral Screen Dataset に含まれる 42,689 個の化学化合物は上記で取り上げた通り，特性に対して 3 種の比率に大きく偏りがある．これより，学習データは不均衡データであることが分かる．このままの状態で学習を行うと比率の大部分を占めるデータに予測精度が偏ってしまうため，これらのような不均衡データに対しては何らかのデータ処理を行う必要がある．不均衡データの処理については以下のようなものが挙げられる．

- 重み付け： 少数派のサンプルに対して重みをつけて最重視する．
- アンダーサンプリング： 少数派のデータ群に合わせて多数派のデータ群を削除する．
- オーバーサンプリング： 少数派のデータを多数派に合わせて増やす．

本論文では，サポートベクタマシン（SVM）とランダムフォレスト（RF）に対しては重み付けを，ニューラルネットワーク（NN）に対してはアップサンプリングを行うことで予測精度の偏りを改善を行った．

3. 機能予測実験

本章では，NCI Chemical Dataset の化学化合物グラフの反復隣接木構造をリスト化した FPD について述べる．また，FPD による機能予測実験の結果について述べる．実験でのプログラミング言語は Python 3.9 を用いた．また機械学習アルゴリズムの Python ライブラリとして scikit-learn 1.0.1 を利用した．実験の手順については [1] を参考にした．

3.1 化学化合物グラフの FPD

FPD に含まれる FP が NCI Chemical Database に含まれる各化学化合物グラフに対して，それが現れる数を調べることでそれぞれが持つ特徴構造の抽出を行う．CA,CM,CI, それぞれに多い特徴構造を調べることで分類を行うことにつながる．図 7 に，CA に多く現れる FP $[2, 1, 1, 1]$ を再現した木構造を挙げる．

以降，複数の実験設定に対して行った機能予測実験につ

レベル2のFP辞書

0 [1]	5 [1,0,1]	10 [3,6]	15 [8,4,6,7]	20 [13,10,15]
1 [2]	6 [1,0,2]	11 [4,8]	16 [9,12]	21 [14,12,15]
2 [3]	7 [1,1,2]	12 [5,3,7]	17 [10,13]	22 [15,11,13,14]
3 [0,1]	8 [2,0,1,1]	13 [6,3,8]	18 [11,15]	
4 [0,2]	9 [3,5]	14 [7,5,8]	19 [12,9,14]	

図 5 グラフ G (図 2) のレベル 2 の FPD $D_2(G)$: $D_2(G)$ のインデックスに対応する FP を示す.

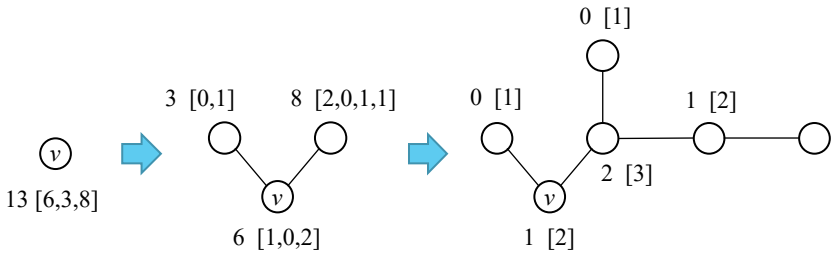

図 6 グラフ G (図 2) のレベル 2 の FPD$D_2(G)$ の追跡: $D_2(G)$ におけるインデックス 13 を持つ頂点の隣接木構造の再現例

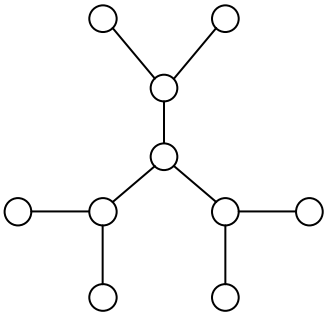

図 7 AIDS Antiviral Screen Dataset の CA に頻出する FP [2, 1, 1, 1]

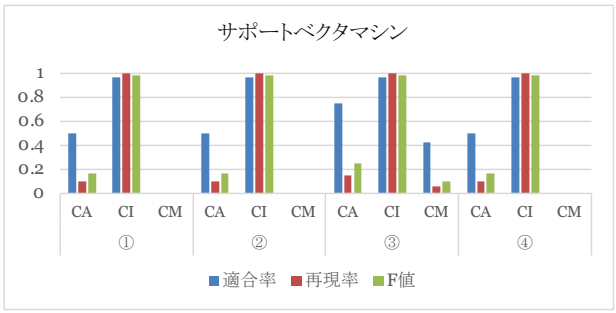

図 8 (a) サポートベクタマシンによる FPD 種別の機能予測結果

いて，その結果を示すとともに考察を行う.

3.2 機能予測実験におけるパラメータ設定

本実験では，作成した FPD をもとに NCI Chemical Database の各化学化合物中に現れる FP の数をリスト化したものを学習データとした．そして，サポートベクタマシン (SVM) とニューラルネットワーク (NN), ランダムフォレスト (RF) の 3 つの学習モデルを用いて，すべて訓練データ数 10,000，テストデータ数 2,000 で機能予測実験を行った．実験設定については，ラベルなしグラフ集合から FPD を構成する際の CA,CM,CI の数の違いによるものを用意した．機能予測実験を行う際に実行した実験設定と学習モデルは以下のとおりである．ここで規格化とは学習データにおいて一定数以上 0 が連続で表示されているデータの足切りを行ったことを示す.

実験設定

① 規格化無し, CA 100[個], CM 100[個], CI 100[個]
② 規格化無し, CA 423[個], CM 0[個], CI 0[個]
③ 規格化無し, CA 0[個], CM 423[個], CI 0[個]
④ 規格化無し, CA 0[個], CM 0[個], CI 423[個]
⑤ 規格化無し, CA 0[個], CM 1,000[個], CI 0[個]
⑥ 規格化無し, CA 0[個], CM 0[個], CI 1,000[個]
⑦ 規格化有り, CA 100[個], CM 100[個], CI 100[個]
⑧ 規格化有り, CA 423[個], CM 0[個], CI 0[個]
⑨ 規格化有り, CA 0[個], CM 423[個], CI 0[個]
⑩ 規格化有り, CA 0[個], CM 0[個], CI 423[個]

使用学習モデル

(a) サポートベクタマシン (SVM)
(b) ニューラルネットワーク (NN)
(c) ランダムフォレスト (RF)

3.3 FPD 種別の機能予測実験結果

まず，CA,CM,CI に分類されるグラフ構造を各 100 個使用して作成した①と CA,CM,CI のうち 1 種類を 423 個使用した②,③,④を比較する．適合率，再現率，F 値を CA,CM,CI で分けてグラフ化したものを図 8〜10 に示す.

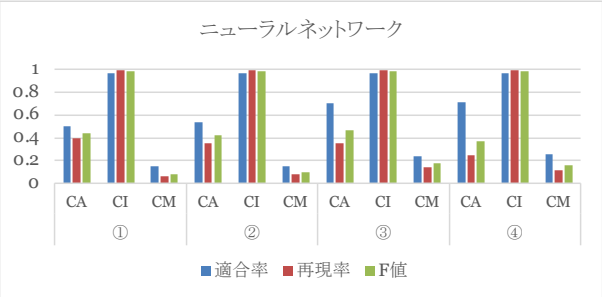

図 9 (b) ニューラルネットワークによる FPD 種別の機能予測結果

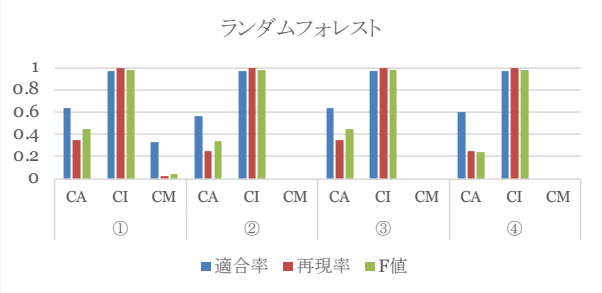

図 10 (c) ランダムフォレストによる FPD 種別の機能予測結果

表 1 ①,②,③,④の各学習モデルに対しての予測精度

予測精度	①	②	③	④
SVM	0.9655	0.9655	0.9655	0.9655
NN	0.9605	0.9575	0.961	0.9625
RF	0.967	0.9655	0.966	0.9655

サポートベクタマシンについては，①，②，④はCA,CM,CI ともに大きな違いは見受けられないが，③についてはCM の値が他に比べ，高いことからCM も中程度，分類可能であることが分かる．また，③はCA の値も他の 3 実験設定よりも高いことも分かる．これより，サポートベクタマシンでCA,CM,CI の 3 種を分類する場合，CMを中心に作成したFPD を学習に用いることでCA,CM,CIの分類で最大限の機能予測ができると考えられる．各学習モデルの予測精度を表 1 に挙げる．①，②，③，④ともに0.9655 という値を示しているため，予測精度はラベルの種類によって変化するものではない．

ニューラルネットワークについては，③と④が①と②に比べ，CA,CM が高いことが分かる．これより，ニューラルネットワークで分類を行う場合，CM,CI を中心に作成したFPD を用いることで学習において最大限の効果を発揮することができると考えられる．また，表 1 より，予測精度については，①が 0.9605，②が 0.9575，③が 0.961，④が 0.9625 という値を示すため，大きな差はないが④が最良の予測精度を記録した．

ランダムフォレストについては，①が他の 3 実験設定に比べて CM の値がはっきりと記録されている．これより，ランダムフォレストでCA,CM,CI の 3 種を分類する場合，

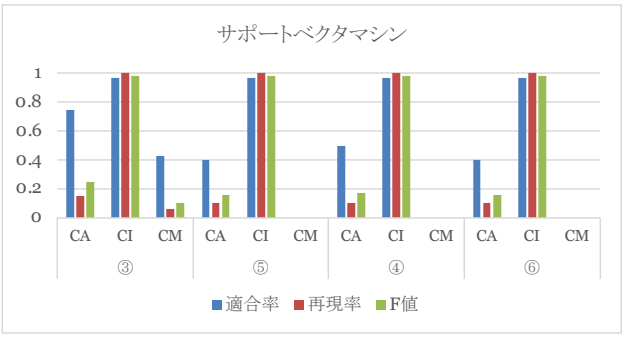

図 11 (a) サポートベクタマシンによる FPD 種別の機能予測結果

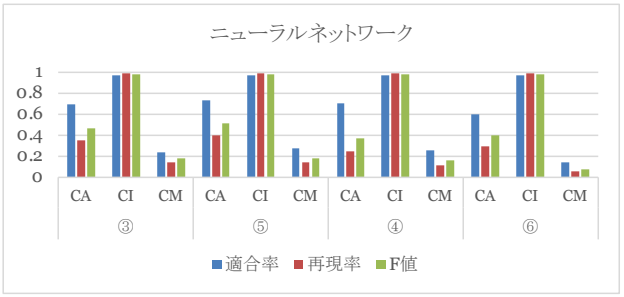

図 12 (b) ニューラルネットワークによる FPD 種別の機能予測結果

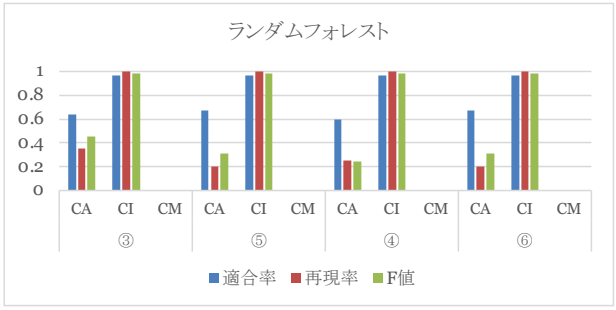

図 13 (c) ランダムフォレストによる FPD 種別の機能予測結果

CA,CM,CI を均等に使用して作成したFPD を学習に用いることで最大限の効果を発揮すると考えられる．また，予測精度については，①が 0.967，②が 0.9655，③が 0.966，④が 0,9655 という値を示すため，大きな差はないが①が最良の予測精度を記録した．

3.4 FPD 作成時のグラフ数種別機能予測実験結果

次に，FPD に使用するグラフ数に着目して考える．CMと CI については，423 個で作成した実験設定と 1,000 個で作成した実験設定がある．よって，CM のグラフ数が違う③，⑤と CI のグラフ数が違う④，⑥を使用して比較したものを図 11〜13 に示す．

サポートベクタマシンについては，グラフ数 423 個で作成したFPD が CM と CI どちらにおいても値が高いため，必ずしもグラフ数が多い方が良いとは言えない．予測精度を表 2 に挙げる．サポートベクタマシンでは，③が

表 2 ③,⑤,④,⑥の各学習モデルに対しての予測精度

予測精度	③	⑤	④	⑥
SVM	0.9655	0.965	0.9655	0.965
NN	0.961	0.9635	0.9625	0.96
RF	0.966	0.966	0.9655	0.966

0.9655，⑤が 0.965，④が 0.9655，⑥が 0.965 であるため僅差ではあるが，グラフ数 423 個の③と④の方がグラフ数 1000 個の⑤と⑥と比べて予測精度が高いと言える．これより，サポートベクタマシンで学習を行う場合は，最低限のグラフ数が必要だが，グラフ数を増やし過ぎると学習に不利になる．

ニューラルネットワークについては，大きな変化は見られないため，グラフ数を増やしても一定数以上は変化しないものであると考えられる．表 2 から予測精度については，③が 0.961，⑤が 0.9635，④が 0.9625，⑥が 0.96 であるため大きな差はない．これより，ニューラルネットワークで学習を行う場合は，最低限のグラフ数が必要だが，それ以上増やしても学習に有利になるとは言えない．

ランダムフォレストについては，ニューラルネットワークと同様に大きな変化は見られないため，グラフ数を増やしても一定数以上は変化しないものであると考えられる．予測精度については，表 2 より④が 0.9655 で他の 3 つの実験設定による評価が 0.966 であるため大きな差はない．これより，ニューラルネットワークと同様に，ランダムフォレストで学習を行う場合は，最低限のグラフ数が必要だが，それ以上増やしても学習に有利になるとは言えない．

3.5 規格化有無別の機能予測実験結果

次に，規格化の有無によるの機能予測の変化について考える．規格化無しの①，②，③，④それぞれに対応した規格化有りが⑦，⑧，⑨，⑩である．本節ではこれらを使用した予測実験の結果を比較する．各学習モデルの予測精度を表 3 に挙げる．

サポートベクタマシンについては，規格化の有無による予測精度にはおおよそ変化がない．

ニューラルネットワークについては，規格化を行っているものが行っていいないものと比べて，大きな変化はみられない．予測精度に関しても，変化はみられるが，大きな変化ではない．

ランダムフォレストについては，規格化を行っているものが行っていいないものと比べて，CM の値が高くなっている傾向にある．予測精度に関しては大きな変化はない．

いずれの学習モデルでも規格化無しより規格化有りの方が学習時間は短縮された．

3.6 不均衡データ処理別の有無による機能予測結果

続いて，不均衡データに対して処理を行った場合の処理の有無でどのような変化がみられるのかについて考える．規格化で大きな変化は見られなかったため，①，②，③，④の実験設定でデータ処理の有無による変化を考える．データ処理については，サポートベクタマシンとランダムフォレストは重み付け，ニューラルネットワークはオーバーサンプリングで処理を行った．各学習モデルの予測精度を表 4 に挙げる．

サポートベクタマシンについては，重み付けを行うことで CM の値が大きくなること，CA の適合率が下がり，再現率が上がることの二つの結果が得られた．本論文では具体的な数値を略す．予測精度に関しては，重み付けを行うことで低下することが分かった．

ニューラルネットワークについては，オーバーサンプリングを行うことで CA,CM の値が共に大幅に変化する結果を得た．予測精度に関しては低下しているが，サポートベクタマシンほど大きなものではないことが分かる．機能予測結果に関してはオーバーサンプリングの特性上，精度が向上しているように見えるが分類があいまいになっているものであると考えられる．

ランダムフォレストについては，重み付けを行うことで CA の値が減り，CM の値が上がるというものになった．予測精度に関しては，データ処理を行うことで多少，予測精度が落ちる結果となった．これらの結果に関して，機能予測の変化と予測精度の低下から不均衡データを学習データとして扱っていたため，偏りが生じていたことが分かる．

3.7 最良の学習モデルについて

最後に最良の学習モデルに関して考える．規格化無し，データ処理なしで学習モデル別に比較を行った後，最良学習モデル比較を行う．各学習モデルの予測精度を表 5 に挙げる．

まず，サポートベクタマシンから考える．図 8 から実験設定③が CA,CM の値が他の実験設定に比べ高いことが分かる．表からは実験設定⑤が 0.965，それ以外が 0.9655 と大きなさがないため，サポートベクタマシンの最良実験設定は③である．

次にニューラルネットワークで最良実験設定を考える．グラフから実験設定⑤が CA,CM の適合率，再現率ともに他の実験設定に比べ高いことが分かる．予測精度では，大きな差はないがこちらも実験設定⑤が最良の予測精度を記録している．従って，ニューラルネットワークの最良実験設定は⑤であると考えられる．

続いてランダムフォレストで最良の実験設定を考える．グラフより，CA の値が最も高く得られているのは実験設定⑤だが，CM の値が得られているのは実験設定①だけであることが分かる．予測精度からは実験設定⑤が 0.966 で実験設定①が 0.967 である．従って，ランダムフォレストの最良実験設定は①である．

表 3 ①,⑦,②,⑧,③,⑨,④,⑩の各学習モデルに対しての予測精度

予測精度	①	⑦	②	⑧	③	⑨	④	⑩
SVM	0.9655	0.9655	0.9655	0.9655	0.9655	0.965	0.9655	0.9655
NN	0.9605	0.9625	0.9575	0.963	0.961	0.9605	0.9625	0.96
RF	0.967	0.966	0.9655	0.965	0.966	0.966	0.9655	0.966

表 4 不均衡データ処理の有無による各学習モデルの予測精度

予測精度	①	①処理あり	②	②処理あり	③	③処理あり	④	④処理あり
SVM	0.9655	0.799	0.9655	0.799	0.9655	0.816	0.9655	0.802
NN	0.9605	0.9365	0.9575	0.9405	0.961	0.9415	0.9625	0.9635
RF	0.967	0.9615	0.9655	0.961	0.966	0.961	0.9655	0.9615

最後に，サポートベクタマシンの実験設定③，ニューラルネットワークの実験設定⑤，ランダムフォレストの実験設定①を比較し，最良実験設定別最良の学習モデルを考える．

図 8〜13 及び表 1〜4 より，CA は適合率がサポートベクタマシン，ニューラルネットワークともに高いが，再現率，F 値まで含めて考えると，ニューラルネットワークが良い値を得ていることが分かる．CI に関しては，3 種とも同等の値を得ており，CM では適合率はサポートベクタマシンが最も良い値を得ているが，ニューラルネットワークが再現率，F 値ともに他の 2 種に比べ高い値を得ている．よって，機能予測の面ではニューラルネットワークが最良実験設定で比較した際の最良の学習モデルである．予測精度の面で考えると，表 5 より，ランダムフォレスト，サポートベクタマシン，ニューラルネットワークの順で高いことから，ランダムフォレストが最良実験設定で比較した際の最良の学習モデルである．

表 5 学習モデル別の最良予測精度

	SVM ③	NN ⑤	RF ①
予測精度	0.9655	0.9635	0.967

4. まとめと今後の課題

本論文では，反復隣接木構造としての Fingerprint (FP) による NCI Chemical Database を用いた辞書化の実装と機能予測実験の結果について述べた．本論文中の実験結果より，機能予測と予測精度の結果を見ると，中程度の分類が可能であるという結果を得ることができた．最良の予測精度に関しては，実験設定が規格化無し, CA 100[個], CM 100[個], CI 100[個] で FPD を作成したもので学習モデルがランダムフォレストであることが分かった．

今後の課題として，不均衡データである NCI Chemical Database を用いた，機能予測実験においてサポートベクタマシン (SVM) とランダムフォレスト (RF) は重み付け，ニューラルネットワーク (NN) においてはアップサンプリングというデータ処理手段をとっていたが，対照実験という観点から考えると実験結果を対比する際に公正であるとは言えないので，ニューラルネットワーク (NN) でのデータ処理手段を他 2 種と同様に重み付けに変え，公正な対照実験を行うことが挙げられる．また，第 2 章で述べたように，頂点自身の FPD におけるインデックスと隣接頂点の FP を合わせて新たなインデックスを割り振るため，レベルを上げ，反復して行うことにより，隣接頂点を辿って先のインデックスを含めた新しいインデックスを持つことができる．これに対し，頂点及び辺に特徴量またはラベルを持たないグラフ構造において，グラフ構造を網羅した FPD を作成することができるグラフ構造と数量の関係性を調べることが課題である．

謝辞 本研究は JSPS 科研費 21K12021 の助成を受けたものです．

参考文献

[1] A. C. Muller, Python ではじめる機械学習 – scikit-learn で学ぶ特徴量エンジニアリングと機械学習の基礎, オライリージャパン, 2017.
[2] D. Rogers and M. Hahn, Extended-Connectivity Fingerprints, *J. Chem. Inf. Model*, 50, 742–754, 2010.
[3] H. Yamasaki, Y. Sasaki, T. Shoudai, T. Uchida, and Y. Suzuki, Learning block-preserving graph patterns and its application to data mining, *Machine Learning*, 76, 1, 137–173, 2009.
[4] X. Yan and J. Han, gSpan: Graph-based substructure pattern mining, Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002), 721–724, 2002.