論 文

Technical Papers

木構造データに対するカーネル関数の設計と解析

Tree Kernels

鹿島 久嗣
Hisashi Kashima
日本アイ・ビー・エム株式会社 東京基礎研究所
Tokyo Research Laboratory, IBM Research
hkashima@jp.ibm.com

坂本 比呂志
Hiroshi Sakamoto
九州工業大学 知能情報工学科
Department of Artificial Intelligence, Kyushu Institute of Technology
hiroshi@donald.ai.kyutech.ac.jp

小柳 光生
Teruo Koyanagi
日本アイ・ビー・エム株式会社 東京基礎研究所
Tokyo Research Laboratory, IBM Research
teruok@jp.ibm.com

keywords: kernel methods, convolution kernels, tree kernels, support vector machines

Summary

We introduce a new kernel function for labeled ordered tree based on the idea of convolution kernels, and several extensions to allow flexible labels and structures in the labeled ordered tree kernel. Also, we show a hardness result in designing tree kernels for more general rooted labeled trees.

1. Introduction

近年，予め特徴ベクトルで表現されたデータのみならず，配列，木，グラフなどの構造をもったデータを扱うような学習の重要性が増している．たとえば，自然言語処理 [Manning 99] の分野において，テキストは配列として表現され，また，構文解析されたテキストは構文解析木として表現される．バイオインフォマティクス [Durbin 98] の分野においては，DNA，RNA，たんぱく質などの配列データや，RNA の木構造データなどがある．時には，たんぱく質の 3 次元構造は，グラフデータとして表現される．Web データの解析においては，ドキュメントは通常 XML や HTML などのいわゆる半構造データと呼ばれる形式で記述され，また，Web サイト内での購買履歴や，行動履歴も配列や木，グラフ構造をもったデータとして表現できる．さらに，データ内の構造だけではなく，データ間の構造も考えられる．たとえば，たんぱく質や遺伝子のネットワーク，WWW のリンク構造なども，たんぱく質や遺伝子，ハイパーテキストなどの関係を表す大きなグラフ構造として捉えることができる．

本論文で我々は，特に木構造をもったデータを対象とする学習問題について考える．一般的な学習問題では，対象となるデータは特徴空間中の 1 点 (ベクトル) として与えられ，例えば 2 値分類の問題であれば，分類器の学習は正例の点と負例の点を分類する超平面などのルールを学習することになる．しかしながら，特徴空間の定義は予め与えられていることを前提として設計されており，構造をもったデータに対する一般的な特徴空間の定義は自明ではない．構造を持ったデータを扱う方法として自然である考えられる方法のひとつは，データに含まれる部分構造を特徴として用いることであろう．この場合，例えば，特徴ベクトルのある次元の定義は，ある部分構造が対象のデータに含まれるかどうかを表す 2 値 (0/1) の値，あるいは，含まれる回数などを用いることが考えられる．木構造データの場合には，各特徴として用いる部分構造は，対象の木に含まれるパスや，部分グラフとするのが自然であろう．しかしながら全ての部分構造を考えて，陽にベクトル表現をしてしまうと，たとえばある木の中に含まれる部分グラフの数は，木のサイズに関して指数的に多くなりうるため，計算コスト的な問題がある．また，もうひとつの問題として，高次元のデータを扱うことは，いわゆる「次元の呪い」と呼ばれる予測性能の低下を引き起こすという問題がある．従来は，この問題に対して，なんらかの方法によって分類に有効な属性のみを選ぶことで，用いる属性の数を絞るということがなされてきた．

一方，カーネル法 [Shawe-Taylor 04] と呼ばれる学習器のクラスが近年注目を集めている．カーネル法の重要な特徴の 1 つとして，カーネル関数を用いたデータアクセスが挙げられる．カーネル法は，データにアクセスする際に，単体ではなく，必ず 2 つのデータの内積の形でアクセスする．これはすなわち，特徴空間の次元がどん

なに高くとも (たとえ，無限次元であろうとも)，なんらかの方法によって特徴ベクトルの内積だけを高速に計算することができるならば，学習器の訓練にも予測にも特徴空間の次元が明示的に現れることがないことを意味する．この内積を与える関数は「カーネル関数」と呼ばれ，カーネル法はカーネル関数を用いることで，高次元の特徴空間においても効率的に働くことができる．さらに，カーネル法の代表的な手法であるサポートベクターマシン [Vapnik 95] においては，次元に依存しない汎化性能が理論的，実験的に示され，次元の呪いを克服できることが確認されている．

カーネル関数の設計方法に関しては絶対的な指針は存在しないが，Haussler は，離散的な構造をカーネル法によって扱うためのカーネル関数の一般的な枠組みとして「畳み込みカーネル」を提案している [Haussler 99]．畳み込みカーネルの基本的な考え方は，対象のデータはそれに含まれる「パーツ」に分解され，カーネル関数はそれらの間のカーネル関数の和によって定義される，というものである．この枠組みに従い，Collins らは自然言語処理において現れる構文解析木に対する畳み込みカーネルを提案している [Collins 02]．彼らは，構文解析木を，その中に含まれる，ある種の部分グラフの出現回数を用いて (非明示的な) ベクトル表現を行い，効率的な内積の計算方法を提案している．しかしながら，彼らの用いた部分グラフは，あるノードの子ノードが順番によってラベル付けされており，しかも，彼らの提案した計算方法はその仮定に強く依存しているためより一般的なクラスの木にはそのままでは適用できない．

そこで，一体どのくらい一般的な木に対して木カーネルを設計可能かという疑問が生ずる．この論文で我々はこの問いに対してある程度答えることを目的とする．次に，我々はラベル付き順序木に対する，任意の部分グラフを属性とするような木カーネルを効率的に計算できるアルゴリズムを示し，その計算量は構文解析木カーネルと同等であることを示す．さらに，我々は提案したラベル付き順序木カーネルを計算量を増やすことなく，ラベルや，構造の曖昧さを許すような柔軟性をもつように拡張する．また，一般化の限界として，一般的なラベルつき根付き木に対して，任意の部分グラフを属性とするような木カーネルを計算することの計算困難性を示す．

そして最後に，提案したカーネル関数が構造的な情報を効果的に活用できていることを確認するために，人工データと実際の HTML 文書を用いた簡単な実験を行う．

本論文の構成は以下の通りである．まず 2 章において，畳み込みカーネル [Haussler 99] の考え方を述べ，その枠組みに従って構文解析木カーネル [Collins 02] を紹介する．3 章では，より一般的な構造データであるラベル付き順序木に対する畳み込みカーネルの提案と，そのいくつかの拡張を提案する．4 章では，最も一般的な木カーネルを考えた場合の計算困難性を示す．5 章では提案したカーネル関数を用いた計算機実験の結果を示す．6 章では関連研究を紹介し，7 章は結論とする．

2. 畳み込みカーネルと木カーネル

2·1 畳み込みカーネル

Haussler は，構造をもったデータの特徴は, その構造に含まれる部分構造が担っていると考え, 構造データ同士のカーネル関数を, 部分構造同士のカーネル関数によって再帰的に定義するという考え方に基づき，離散的な構造に対するカーネル関数設計の一般的な枠組みとして**畳み込みカーネル**を提案した [Haussler 99]．畳み込みカーネルは，2 つの構造データ T と T' が与えられたとき

$$K(T,T') = \sum_{s\in S(T)} \sum_{s'\in S(T')} K^S(s,s') \qquad (1)$$

と定義される．ここで $S(x)$ は x から取り出される部分構造の集合を表し, K^S は 2 つの部分構造の間に定義されるカーネル関数であるとする．畳み込みカーネルは, T と T' から部分集合 $S(T)$ と $S(T')$ を取り出し, それらの間のカーネル関数値をすべて足し合わせることで定義される．つまり，2 つのデータの類似度が, データの部分構造の類似度に還元されるのである．そして, 部分構造の類似度は，(1) によって更なる部分構造をもちいて再帰的に定義されうる．K^S がカーネル関数である場合に，(1) もカーネル関数となっていることが保証される．

ここで，T と T' が共に木構造データ，すなわち V と V' を頂点の集合，E と E' を枝の集合として，$T=(V,E)$ と $T'=(V',E')$ であるとき，(1) を**木カーネル**という．木構造データのクラスや，部分構造 S のクラス，K^S の定義を変えることによって，様々な木カーネルを定義することができる．

本論文では部分構造 S の種類を最も一般的に，木に含まれる全ての部分グラフの集合であると定義する．木の部分グラフは，木構造であるため，(1) は以下のように分解することができる．

$$\begin{aligned} K(T,T') &= \sum_{v\in V} \sum_{v'\in V'} \sum_{s\in S_v(T)} \sum_{s'\in S_{v'}(T')} K^S(s,s') \\ &= \sum_{v\in V} \sum_{v'\in V'} K^R(v,v') \qquad (2) \end{aligned}$$

ここで，$S_v(T)$ は頂点 $v\in V$ を根として持つような木構造をもった部分グラフの集合とする．また，$K^R(v,v')$ を $S_v(T)$ と $S_{v'}(T')$ に限定したときのカーネルで，

$$K^R(v,v') = \sum_{s\in S_v(T)} \sum_{s'\in S_{v'}(T')} K^S(s,s') \qquad (3)$$

とする．

2·2 構文解析木カーネル

Collins と Duffy は，自然言語処理で用いられる構文解析木 (図 1 左) の間の畳み込みカーネルを設計した [Collins

02]．構文解析木は，ラベル付き順序木，すなわち，任意の頂点について，アルファベット Σ の内 1 つがラベルとして振られ，子供に全順序関係があるような，根付き木であるとみなせる．$S(T)$ としては，T の部分グラフとして現れる全てのラベル付き順序木を用いる．但し，全ての枝にはその枝が親頂点の何番目の子に繋がる枝かを表す番号でラベル付けされているとする (図 1 右)．また，2 つの部分構造 $s \in S(T)$ と $s' \in S(t')$ の間のカーネル関数を

$$K^S(s,s') = I(s = s') \tag{4}$$

と定義する．ここで $I()$ は括弧内が成立する場合に 1，そうでない場合に 0 となるような関数とする．また，$s = s'$ は，2 つのラベル付き順序木 s と s' が枝のラベルも含めて完全に一致することを意味するとする．

(1) の計算を $S(T)$ と $S(T')$ を明示的に数え上げて計算を行おうとすると，その数は指数的に大きくなってしまうという問題がある．そこで Collins らは，(2) の分解と，(3) が全ての v と v' に対して以下の再帰式によって $O(|V||V'|)$ で再帰的に計算できることを利用して，効率的にカーネル計算を行う方法を示した．

- v あるいは v' が葉のとき，

$$K^R(v,v') = I(\ell(v) = \ell(v')) \tag{5}$$

- v と v' が葉でないとき，

$$K^R(v,v') = I(\ell(v) = \ell(v')) \cdot \prod_{i=1}^{\sharp ch(v)} (K^R(ch(v,i), ch(v',i)) + 1) \tag{6}$$

ここで，再帰式 (6) において $\sharp ch(v)$ は頂点 v の子の数を，$ch(v,i)$ は v の i 番目の子頂点を表すとする．v を根にもつような木は，v の子を根に持つような木を組み合わせて，その上に v を付け加えることで構成できるため，$K^R(v,v')$ は，v の子を根にもつような木と v' の子を根に持つような木のカーネルの全ての組み合わせを列挙し ((6) の右辺 2 項目), その上に v と v' の頂点同士のカーネルを加える ((6) の右辺 1 項目) ことによって構成できる．v と v' のラベルが異なるときには, 両方を根に持つような木構造部分グラフは当然存在しないため, $K^R(v,v') = 0$ となる．また, (6) の右辺 2 項目において, 枝にはその枝が親頂点の何番目の子に繋がる枝かを表すラベルがついているため，v の i 番目の子を根に持つ木と v' の i 番目の子を根に持つ木のカーネルのみが考慮される．

3. ラベル付き順序木カーネル

3·1 ラベル付き順序木カーネル

本節では，前の章で紹介した構文解析木カーネルが仮定している制約を取り除き，より一般的なラベル付き順序木カーネルを提案する．

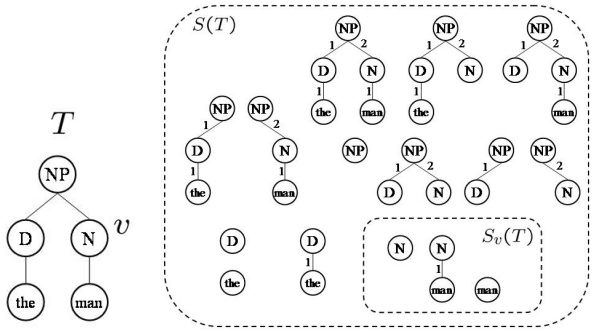

図 1 左: 構文解析木 T の例
右: 構文解析木カーネルにおける部分構造 $S(T)$ と $S_v(T)$

構文解析木カーネルでは，$S(T)$ として，枝に子の位置を示したラベルが振られているため，木の形が一致しても，子の位置を示すラベルが正確に一致しないと同じ部分構造であるとみなされないという問題点がある．従って，例えば図 1 右における部分構造 s は N のラベルを持つ頂点の下に man のラベルを持つ頂点が 2 番目の子として付いているような構文解析木の分類には意味を成さない．この制約は，HTML 文書などのように，ひとつの頂点が多数の子頂点をもちうる一般のラベル付き順序木を考えた際には好ましくない．従って，我々はこの制約を取り除き，$S(T)$ として，T の部分グラフとして現れる全てのラベル付き順序木 (図 1 右で，子の位置を示すラベルを取り除いたもの) を用いる．また，2 つの部分構造同士のカーネル (4) において，$s = s'$ は，2 つのラベル付き順序木が完全に一致することを意味するとする．

辺についている子の位置を示すラベルを取り除いたことにより，(6) の右辺 2 項目において，必ずしも v の i 番目の子を根とする木と v' の i 番目の子を根とする木のカーネルのみだけでなく，すべての i と j の組について考慮する必要がある．従って，v と v' の子頂点を根とするようなカーネルの組み合わせは，子供の順序を保存した，あらゆる組み合わせを考慮する必要がある．しかしながら，v と v' の子供の部分集合の選び方はそれぞれ $2^{\sharp ch(v)}$ 個と $2^{\sharp ch(v')}$ 個，すなわちこれらの間のマッチングの数は，子の数に関して指数個存在するため，これをナイーブに評価するのは困難である．そこで，頂点同士の動的計画法に加え，さらに内側のループとして，子供同士の動的計画法を用いることにより，これを効率的に行う方法を示す．$\bar{K}^R_{v,v'}(i,j)$ を，(6) の右辺 2 項目に対応する値で, v の 1 番目の子から i 番目の子まで，また，v' の 1 番目の子から j 番目の子までに限定した場合のものとする．すなわち，(6) のかわりに，

$$K^R(v,v') = I(\ell(v) = \ell(v')) \cdot \bar{K}^R_{v,v'}(\sharp ch(v), \sharp ch(v')) \tag{7}$$

とする．このとき，以下の再帰式

$$\bar{K}^R_{v,v'}(i,j) = \bar{K}^R_{v,v'}(i-1,j) + \bar{K}^R_{v,v'}(i,j-1) - \bar{K}^R_{v,v'}(i-1,j-1) \tag{8}$$

$$+\bar{K}^R_{v,v'}(i-1,j-1)\cdot K^R(ch(v,i),ch(v',j))$$

が成立する．ここで，境界条件 $\bar{K}^R_{v,v'}(i,0)=\bar{K}^R_{v,v'}(0,j)=1$ とする．この再帰式 (8) は次のように説明できる．まず，1 行目と 2 行目は，v の i 番目の子と，v' の j 番目の子を用いない全てのマッチングを考慮する．2 行目では，1 行目で 2 回考慮している重なり部分を取り除いている．3 行目では v の i 番目の子と，v' の j 番目の子を必ず用いるマッチングを考慮している．これは，既に計算している $1\sim i-1$ 番目の子集合と $1\sim j-1$ 番目の子集合の間のマッチングに，i 番目の子と j 番目の子を加えることで計算できる．この再帰式によって，$K^R(v,v')$ を $O(\sharp ch(v)\sharp ch(v'))$ で計算可能となる．

カーネル関数全体としての計算量は以下のように，$O(|V||V'|)$ となることがわかる．これは，構文解析木カーネルと同じ計算量である．

$$\begin{aligned}&\sum_{v\in V}\sum_{v'\in V'}O(\sharp ch(v)\cdot\sharp ch(v'))\\ &=\sum_{v\in V}O(\sharp ch(v))\cdot\sum_{v'\in V'}O(\sharp ch(v'))\\ &=O(|V|\cdot|V'|)\end{aligned}$$

3·2 ラベル付き順序木カーネルの拡張

前節で提案したラベル付き順序木カーネルでは，2 つの部分グラフ $s\in S(T)$ と $s'\in S(T')$ 同士のカーネル関数は，s と s' がラベルも含めて完全に一致したときのみ値 1 をとるとした．しかしながら，完全には一致しなくても，形やラベルが似ているようなものについては，ある程度一致すると判断したいような場合が考えられる．この節では，ラベルつき順序木カーネルにこのような柔軟性を導入するための 2 つの拡張を考える．

まずは，ラベルについての柔軟性の導入を考える．形がまったく同一である 2 つの部分グラフ s と s' の間のカーネルを，頂点ラベル間のカーネルの積と定義する．言い換えると，s に含まれる頂点を前順で並べた全順序集合を $V_s=(v_1,v_2,\ldots)$ とすると，

$$K^S(s,s')=\prod_{i=1}^{|V_s|}K^\Sigma(\ell(v_i),\ell(v'_i))$$

のように定義する．ここで K^Σ は 2 つの頂点ラベル間のカーネル関数とする．$K^\Sigma(\sigma,\sigma')=I(\sigma=\sigma')$ としたときに，もともとのラベルつき順序木カーネルに一致する．

従って，以上のような変更を加えた場合でも，(7) を

$$K^R(v,v')=K^\Sigma(\ell(v),\ell(v'))\cdot\bar{K}^R_{v,v'}(\sharp ch(v),\sharp ch(v'))$$

のように置き換えればよい．

次に，形に関しての柔軟性を導入する．ここでは，部分構造の集合 $S(T)$ の定義として，任意の頂点集合からなる部分構造を取り出すことができるように拡張する．ただし，取り出される木構造としては，頂点間の相対位置，すなわち先祖・子孫の関係が保存されたような形で取り出されるとする．図 2 に，T から $v_1,\ldots,v_5$ の 5 つの頂点からなる部分構造 s を取り出す例を示す．頂点間の先祖・子孫の関係が保存されていることに注意する．これによって，構造の揺らぎに対して柔軟な特徴定義が実現されることが期待できる．

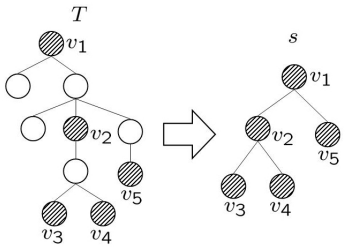

図 2 形に関する柔軟性を導入したときの，ラベル付き順序木 T から取り出される部分構造 s の例．

このような拡張を行った場合のカーネルの効率的な計算は，基本的なアルゴリズムに変更を加えることで実現できる．基本的なアルゴリズムにおける再帰式 (8) や (6) では，ある頂点 v を根にもつような部分構造 $S_v(T)$ の集合は，v の子頂点を根に持つような部分構造を組み合わせて，v を追加することによって再帰的に構成できることを利用していた．しかしながら，柔軟な部分構造取り出しを許すことによって v の子孫となる頂点を根に持つような部分構造までも考慮しなければならない．v および v の子孫の頂点を含むような集合を $D(v)$ とすると，v 以下に含まれる部分構造の集合は

$$S^D_v(T)=\bigcup_{u\in D(v)}S_u(T)$$

と表される．ここで，$S_u(T)$ も「u を根にもつ任意の頂点集合からなる部分構造の集合」というように，定義が拡張されていることに注意する．これを用いて，v と v' 以下の部分構造を用いたカーネル関数を

$$K^D(v,v')=\sum_{s\in S^D_v(T)}\sum_{s'\in S^D_{v'}(T')}K^S(s,s') \quad (9)$$

とおけば，(8) は

$$\begin{aligned}\bar{K}^R_{v,v'}(i,j)&=\bar{K}^R_{v,v'}(i-1,j)+\bar{K}^R_{v,v'}(i,j-1) \quad (10)\\ &-\bar{K}^R_{v,v'}(i-1,j-1)\\ &+\bar{K}^R_{v,v'}(i-1,j-1)\cdot K^D(ch(v,i),ch(v',j))\end{aligned}$$

のように書き換えることができる．ここで $K^D(v,v')$ の計算を (9) の定義どおりに行うと，$O(|S^D_v(T)|\cdot|S^D_{v'}(T')|)$ であるため，全体として $O(|V|^2\cdot|V'|^2)$ になってしまうが，これも以下の再帰式によって，$O(\sharp ch(v)\cdot\sharp ch(v'))$ で計算することができる．

$$K^D(v,v') \quad (11)$$

$$= \sum_{i=1}^{\sharp ch(v)} K^D(ch(v,i),v') + \sum_{j=1}^{\sharp ch(v')} K^D(v,ch(v',j)) - \sum_{i=1}^{\sharp ch(v)} \sum_{j=1}^{\sharp ch(v')} K^D(ch(v,i),ch(v',j)) + K^R(v,v')$$

4. 木カーネル計算の困難性

この章では，前章で提案した木カーネルをさらに一般化することが可能かという問題を考える．ラベル付き順序木よりも一般的な木構造データとしては，子頂点に順序がないようなラベルつき根付き木が考えられる．後に示すように，ラベルつき根付き木に対する木カーネルを定義した場合，ラベル付き順序木と同様に任意の部分グラフを部分構造として使用する場合には#P-完全問題，すなわちこれ以上の一般化は望めないことがわかる．

ラベル付き根付き木に対するカーネル関数を計算する問題として，次の問題 TREE KERNEL を定義する．

問題 1: TREE KERNEL(T,T')
入力: ラベル付き根付き木 T，T'
出力: $K(T,T')$

なお，一般性を失うことなく，常に $|V| \leq |V'|$ であると仮定する．また，部分構造の集合 $S(T)$ は T の全ての部分グラフの集合とし，部分構造同士のカーネル $K^S(s,s') = I(s=s')$ とする．このため，$K(T,T')$ は T と T' の共有する部分グラフの個数を数え上げる問題に等しくなる．ただしこの場合，子頂点に順序がないため，構造の同一性の判定には，子の順序の入れ換えが許されることに注意する．

問題 1 の計算困難性を示すにあたり，まずは次の，限定された部分構造に対するカーネル計算の問題を考える．

問題 2: TREE KERNEL$^{(n)}(T,T')$
入力: ラベル付き根付き木 T，　T'
出力: $K^{(n)}(T,T')$

ただし，ここで $K^{(n)}(T,T')$ とは，T の部分構造を，T の「大きさがちょうど n の」部分グラフに限定した場合の集合 $S^{(n)}(T)$ とした場合の木カーネル

$$K^{(n)}(T,T') = \sum_{s \in S^{(n)}(T)} \sum_{s' \in S^{(n)}(T')} K^S(s,s')$$

と定義されるとする．ここで，特に $n=|V|$ の場合には，$S^{(|V|)}(T) = \{T\}$ となるため，$K^{(|V|)}(T,T')$ は，T が T' 内に出現する（埋め込まれる）回数を数えているのと等価となることに注意する．また，$K^{(n)}(T,T')$ は T と T' の共有する，大きさがちょうど n の部分グラフの個数を数え上げるのに等しくなる．

このとき，TREE KERNEL$^{(|V|)}(T,T')$ の計算困難性を示す次の補題が成立する．

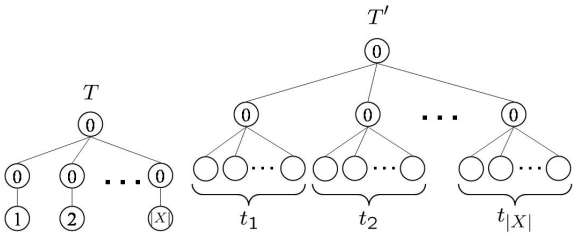

図 3 補題 1 の証明で用いる T と T'

補題 1:
TREE KERNEL$^{(|V|)}(T,T')$ は $\#P$-完全である．

（証明）次の 2 部グラフ $G=(X \cup Y, E)$ に関する問題の計算困難性をもとに，補題 1 を示すことにする．

問題 2: #PERFECT MATCHINGS(G)
[Vadhan 01, Valiant 79b]
入力: 2 部グラフ $G=(X \cup Y, E)$，ただし $|X|=|Y|$
出力: G の完全マッチングの総数

ここで X の頂点間，Y の頂点間にはそれぞれ辺は存在しないことに注意する．グラフ G において，辺のある部分集合 $M \subseteq E$ に対して，M のどの 2 つの辺も頂点を共有しないとき，M を G のマッチングといい，M が V の全ての頂点を含むとき，これを完全マッチングという．#PERFECT MATCHINGS(G) は $\#P$-完全であることが知られている．

ラベルの集合を $\Sigma = \{0,1,2,\ldots,|X|\}$ とし，G に対し，図のような T と T' を考えることにする．ここで，頂点集合 t_j は，「$(x_i,y_j) \in E$ であるならば，ラベル i をもつ頂点が t_j に含まれる」という規則によって決まるとする．

G に辺 $(x_i,y_j) \in E$ が存在するときのみ，ラベル i をもつ頂点が t_j に含まれるため，このときに限り，T のラベル i を持つ頂点を t_j に埋め込むことができる．T が T' 内に出現するとき，葉以外の部分は完全に一致するので，T の葉が $t_1,\ldots,t_{|X|}$ にそれぞれ 1 回づつ埋め込まれることになり，これはちょうど G における完全マッチングに対応している．よって，TREE KERNEL$^{(|V|)}(T,T')$ の出力と #PERFECT MATCHINGS(G) の出力は等しくなるため，#PERFECT MATCHINGS(G) から多項式時間で TREE KERNEL$^{(|V|)}(T,T')$ に還元できることが示された．□

以上に示した補題 1 を用いて，TREE KERNEL(T,T') についての計算困難性を示す以下の定理を証明できる．

定理 1:
TREE KERNEL(T,T') は $\#P$-完全である．

（証明）Cook の還元 [Valiant 79a]，すなわち TREE KERNEL$^{(|V|)}(T,T')$ が TREE KERNEL(T,T') を解くオラクルを用いて多項式時間で解けることを示すことで，TREE KERNEL(T,T') は TREE KERNEL$^{(|V|)}(T,T')$

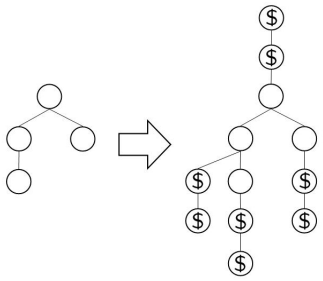

図 4 定理 1 の証明で用いる T の変換（$m=2$ の場合）.

と同様に難しいことを示す.

具体的には, TREE KERNEL$^{(|V|)}(T,T')$ を, [Vadhan 01] のテクニックを援用し, 多項式時間で解ける連立一次方程式に帰着する.

Σ に含まれない記号 \$ を導入し, $\Sigma'=\Sigma\cup\{\$\}$ とする. 与えられた T を, 以下の手順で Σ' 上の木に拡張する.

(1) 任意の $m=0,\ldots,|V|$ に対して, 長さ m の鎖 $v_1-v_2-\cdots-v_m$ をつくり, 鎖のすべての頂点にラベル \$ をつける.

(2) T のすべての頂点に 1 つずつこの鎖を接続する.

(3) このようにしてできた木を T_m とする (図 4).

同様にして T' からも $T_m{}'$ に変換する.

以下では, こうして得られた T_m と $T_m{}'$ に対して, TREE KERNEL$(T_m,T_m{}')$ の解を多項式時間で計算するオラクルが存在すると仮定すると, TREE KERNEL$^{(|V|)}(T,T')$ が多項式時間で計算できることを示す.

さて, 前述したように, $K(T_m,T_m{}')$ は, T_m と $T_m{}'$ の共有する部分グラフの数に等しいが, これらの部分グラフは, 以下の 3 つのクラスに分類することができる.

(a) \$ のみを含むもの,

(b) \$ を含まないもの,

(c) 両方を少なくとも 1 つずつ含むもの.

このうち (a) の形の共通部分グラフの出現回数を $A^{(0)}$ とおく. これは以下のように簡単に計算できる.

$$\begin{aligned}A^{(0)} &= |V||V'|\sum_{k=1}^{m}(m-k+1)^2\\ &= |V||V'|\ m(m+1)(2m+1)/6\end{aligned}$$

また, (b) の形の共通部分グラフのうち, 頂点数が $1\le n\le|V|$ であるものの数を $A^{(n)}$ とする. 同様に, (c) の形の部分グラフのうち, \$ でない頂点数が $1\le n\le|V|$ であるものの数を $\tilde{A}^{(n)}$ とする. このとき, (c) に属する部分グラフから \$ の頂点を取り去ったものと同型なものが (b) の中に存在する. また, \$ は T, T' には存在しないので, $\tilde{A}^{(n)}$ と $A^{(n)}$ の違いは, 付け加えた長さ m の鎖の部分のみに依存する. (b) に属する, 大きさ n のある部分グラフに注目したとき, 各頂点に m 以下の長さの鎖を連結したものが, (c) に属するはずであるので, (c) の中には, \$ の頂点を取り去ることでもとの部分グラフと同型になるものが, m^n 個あるはずである. 従って, $\tilde{A}^{(n)}=m^nA^{(n)}$ の関係が成立する.

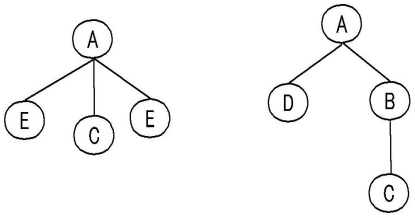

図 5 人工データを用いた分類実験において正例に含まれる 2 つの部分構造

以上により, $K(T_m,T_m{}')$, すなわち T_m と $T_m{}'$ の共有する部分グラフの数は, 以下の式で表現できる.

$$\begin{aligned}K(T_m,T_m{}') &= A^{(0)}+\sum_{n=1}^{|V|}(A^{(n)}+\tilde{A}^{(n)})\\ &= A^{(0)}+\sum_{n=1}^{|V|}(1+m^n)A^{(n)}.\end{aligned}$$

ここで仮定より, $K(T_m,T_m{}')$ は多項式時間で計算可能であるので, この式は, $|V|$ 変数の連立 1 次方程式であり, $m=1,\ldots,|V|$ とすることにより, 異なる $|V|$ 個の式を得ることができる. この連立方程式を解くことで任意の $A^{(n)}$ を決定できる. 一方, $A^{(|V|)}$ は TREE KERNEL$^{(|V|)}(T,T')$ の解であるため, TREE KERNEL$^{(|V|)}(T,T')$ が多項式時間で計算可能になる. 以上により, TREE KERNEL(T,T') は#P-完全である. □

5. 実　　験

この章では, 人工データと実際の HTML データの両方に対する分類問題において, 提案手法が構造情報を有効活用できているかどうかを検証する. カーネル学習器としては, 実装の簡便さから, SVM と比較しても遜色ない性能をもつカーネルパーセプトロン [Freund 99] を用いた. カーネル関数としては, 提案したラベル付き順序木カーネルの, 基本的なバージョン, および, 構造の柔軟性を許したバージョンを用い, ラベルの柔軟性は考慮しなかった. 精度は, 全て leave-one-out のクロスバリデーションによって計測を行った.

5·1 人工データを用いた分類実験

まずは, 人工的に生成したデータを用いた分類タスクにおける実験を行う. ここで, 人工データは部分構造を認識しないと分類できないようなデータを生成し, これを用いた. 具体的には, 図 5 に示すような 2 つの木構造を両方とも部分グラフとして持っている場合に正例, そうでない場合には負例となるようなデータを生成した. 我々は, それぞれ 30 個から 50 個のノードと, 10 種類の

表 1 人工データを用いた分類実験の結果（leave-one-out クロスバリデーションによって計測．5 回の平均.）

d	BoL カーネル	木カーネル
1	**57.8%**	80.5%
2	55.6%	**84.4%**
3	56.7%	80.6%
4	**57.8%**	76.1%
5	55.0%	76.1%

ラベルを持つようなラベル付き順序木を，正例として 30 個，負例として 30 個のデータをランダムに生成した*1．

カーネル関数としては，基本のラベル付き順序木カーネルと，比較対照として，'bag of labels (BoL)' カーネル，すなわち，サイズ 1 の木 (すなわち単一ノード) のみを部分構造とするようなカーネルを用いた．BoL カーネルは木の構造についての情報をまったく用いないため，これを評価のベースラインとして用いた．

また，それぞれのカーネルは以下のように多項式カーネル [Vapnik 95] と組み合わせて用いた．

$$K_d^{\mathrm{poly}}(T,T') = (1 + K(T,T'))^d$$

ここで，d は多項式カーネルの次数である．人工データは図 5 の 2 つの木構造を同時に含むときに正例となるように生成したため，多項式カーネルと組み合わせた場合には $d = 2$ のときに最も精度がよくなると予想される．表 1 に 5 回の実験の平均の分類精度を示す．太字で示したのがそれぞれのカーネルでの最良の精度を表す．明らかに，提案手法の分類精度は，ラベルの数だけを用いた場合の結果をほぼ 20%程度大きく上回っており，ラベル付き順序木カーネルがデータに含まれる部分構造を認識できていることがわかる．また，予想通り，多項式カーネルの次数は 2 であるときが最も精度が高くなっていることが分かる．

5·2 HTML 文書を用いた分類実験

次に，実際の HTML 文書をレイアウトなどの構造情報に基づき分類する実験を行う．従来，テキスト分類のタスクは，含まれる単語の種類を用いた特徴ベクトル表現である'bag-of-words' 表現 [Salton 88, Joachim 98] を用いるのが普通である．しかしながら，HTML 文書などの構造をもった文書は，タグを用いてレイアウトなどの視覚的な情報を木構造として持っている．この実験では，HTML 文書のもつ木構造に基づき，HTML 文書の分類を行うという実験を行う．

我々は，2002 年における日本 IBM の Web サイト *2および米国 IBM の Web サイト *3から，それぞれ 30 の HTML ページを収集した．これらのページはお互いに

図 6 収集した HTML ページの例

非常に似ている (図 6) が，デザインのテンプレートや，デザイナーの違いがタグ構造の微妙な違い，すなわち，HTML の木構造に現われるであろうと考えられる．

この実験では，構造情報にのみ注目しているため，テキスト部分は削除し，タグのみを用いた *4．

得られたラベル付き順序木は，10〜1500 個のノードを持ち (主に 200〜400 個)，90 種のタグが含まれていた．

この実験では，BoL カーネル，基本のラベル付き順序木カーネルに加え，構造の柔軟性を許すラベル付き順序木カーネルも用いた．前の実験と同様，それぞれのカーネルを多項式カーネルと組み合わせて用いた．表 2 に実験結果を示す．基本のラベル付き順序木カーネルが BoL カーネルに 20%の差をつけて，もっとも高い精度を出しており，構造の違いをうまく捉えていることがわかる．

また, 基本のラベル付きカーネルが $d = 4$ のときに, 構造の柔軟性を許す場合には $d = 3$ のときに最高精度であったことから, 分類には 3 個程度の, 多少の曖昧さをもった部分構造が効いている可能性があると考えられる．

構造の柔軟性を許す場合は，基本の場合に比較してかえって精度が落ちているが，これは恐らく過度の柔軟性により過学習を起こしているものと推測される*5．このような問題を解決する方法としては，部分構造の重み付けを行った畳み込みカーネル（あるいは周辺化カーネル [Tsuda 02]）を用いて，

$$K(T,T') = \sum_{s \in S(T)} \sum_{s' \in S(T')} w(s,T) w(s',T') K^S(s,s')$$

のように部分構造の重み $w(s,T)$ を導入して，複雑な部分構造に対して重みを小さくするような拡張を行うことが考えられるであろう．

*1 正例は，図 5 の 2 つの部分構造を含んだうえで，残りの部分をランダムに生成した．
*2 http://www.ibm.com/jp/
*3 http://www.ibm.com/
*4 勿論この場合には，テキスト部分を用いれば文字コードの違いからほぼ 100%の分類精度が得られるはずである．
*5 たとえば [Kashima 02] などで，構造の柔軟性が情報抽出タスクにおいて有効であった結果が示されている．

表 2 HTML 文書を用いた分類実験の結果（leave-one-out クロスバリデーションによって計測.）

d	BoL カーネル	木カーネル (基本)	木カーネル (柔軟)
1	41.7%	63.3%	61.7%
2	55.0%	71.7%	60.0%
3	**58.3%**	75.0%	**66.7%**
4	51.7%	**80.0%**	60.0%
5	51.7%	71.7%	63.3%

6. 関 連 研 究

カーネル法を用いるほかにも，構造をもつデータを扱う学習のアプローチがいくつか存在する．

関係学習 [Mitchell 97] は，述語論理で記述されたデータを扱う一般的な手法である．部分構造は，予め定義されたデータの構成要素の関係を組み合わせによって表現され，属性として用いられる．属性は，分類に役立つものが訓練の際に逐次的にに構成される．しかしながら，通常，目的関数を最適化する仮説の探索は NP 困難となり，ヒューリスティックな探索手法が用いられる．

もう 1 つのアプローチとして，データマイニングで用いられる構造データからの頻出パターン発見手法 [Inokuchi 00] に基づく方法が挙げられる．この方法は，データに頻繁に現れる部分構造を列挙しておき，これらを属性として用いる．この方法では，クラス情報のないデータを利用できるという利点があるが，パターン発見の問題は通常 NP 困難な問題である．Kudo らは，マイニング手法とブースティングを組み合わせることで，有効な属性のみを発見する興味深い手法を提案している [Kudo 05].

構造をもったデータに対するカーネル関数の設計方法としては，いくつかの枠組みが提案されている*6．たとえば，構造データの生成モデルが分かっているような場合には，Fisher カーネル [Jaakkola 99] を用いることが可能である．Fisher カーネルでは，特徴ベクトルにおける各属性は，生成モデルの尤度関数のパラメータについての勾配によって定義され，特徴ベクトルの次元はパラメータ数に等しくなる．

本論文のように，生成モデルを仮定しない場合には，畳み込みカーネル [Haussler 99] が一般的である．この枠組みに基づき，様々な構造を持ったデータに対する畳み込みカーネルが提案されている．たとえば，Lodhi らは，配列構造におけるすべての部分列を部分構造として用いるようなカーネルを提案している [Lodhi 02]．また，部分構造を連続した部分列に限った場合には，接尾辞木を用いて線形時間で高速にカーネルを計算する手法 [Leslie 02, Leslie 03] も提案されている．グラフ構造を持ったデータに関しては，グラフ上のランダムウォークによって生成される経路を部分構造とするようなカーネルが提案されている [Kashima 03, Schölkopf 04, Gärtner 02, Suzuki 03].

また，2 つのアプローチの中間的アプローチとして，生成モデルによる部分構造の重み付けを用いながら，畳み込みカーネルを計算するような，周辺化カーネル [Tsuda 02, Kin 02] の枠組みも提案されている．

カーネル法では, 特徴空間での表現が暗黙的であるために, 有効な部分構造を特定するのが困難になるという問題があるが, パターン発見手法を援用することで重要な部分構造を発見するというハイブリッド的アプローチ [Kudo 03, Suzuki 04] も提案されている．

7. 結 論

本論文では，木構造をもったデータに対するカーネル関数の設計について論じた．まず，畳み込みカーネルの枠組みにおいてラベル付き順序木に対して任意の部分グラフを部分構造として用いた場合の，効率の良いカーネル計算のアルゴリズムを提案し，曖昧なラベルや構造を取り込むような拡張を行った．さらに，より一般的な木構造として，順序のないラベル付き根付き木に対するカーネルを考えた場合には，カーネルの計算が#P-完全問題であることを示した．

今後の課題としては，まず 5 章の最後で述べたような，部分構造の重み付けを考慮することが挙げられるであろう．これには 2 つのアプローチが考えられる．1 つは 6 章で述べた, 陽に特徴選択を行うアプローチ [Kudo 03, Suzuki 04] であり, もう 1 つは特徴選択をも暗黙のうちに行うというアプローチであろう．暗黙の特徴空間における特徴選択法としてはカーネル主成分分析 [Schölkopf 88] などが挙げられるが，畳み込みカーネルの部分構造の重み付けパラメータを予め学習しておくことも暗黙の特徴選択の効果があると考えられ，これら各種の特徴選択法の比較を行うことも 1 つの課題であろう．

また，もう 1 つの課題としては，より高速な木カーネルの設計が考えられる．動的計画法に基づくカーネル計算アルゴリズムは基本的に 2 つの構造の大きさの積程度の計算時間が掛かってしまうが，よりスケーラビリティが重要な場合では，線形時間に近い速さで計算できるような木カーネルが望まれる．ラベルの曖昧さを許さず，部分構造を根から葉へ向かうパスの集合などに限定し，接尾辞木 [Leslie 02] を用いるなど，表現力と計算速度のバランスをとるための工夫が必要であろう．

*6 [Gärtner 03] に詳細なサーベイがある．

◇ 参 考 文 献 ◇

[Collins 02] Collins, M. and Duffy, N.: Convolution Kernels for Natural Language, in *Advances in Neural Information Processing Systems 14*, Cambridge, MA (2002), MIT Press

[Durbin 98] Durbin, R., Eddy, S., Krogh, A., and Mitchison, G.: *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press (1998)

[Freund 99] Freund, Y. and Shapire, R.: Large Margin Classification using the Perceptron Algorithm, *Machine Learning*, Vol. 37, No. 3, pp. 277–296 (1999)
[Gärtner 02] Gärtner, T.: Exponential and Geometric Kernels for Graphs, in *NIPS*02 Workshop on Unreal Data: Principles of Modeling Nonvectorial Data* (2002), Available from http://mlg.anu.edu.au/unrealdata/
[Gärtner 03] Gärtner, T.: A Survey of Kernels for Structured Data, *SIGKDD Explorations*, Vol. 5, No. 1, pp. S268–S275 (2003)
[Haussler 99] Haussler, D.: Convolution Kernels on Discrete Structures, Technical Report UCSC-CRL-99-10, University of California in Santa Cruz (1999)
[Inokuchi 00] Inokuchi, A., Washio, T., and Motoda, H.: An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data, in *The Fourth European Conference on Principles and Practice of Knowledge Discovery in Databases*, pp. 13–23 (2000)
[Jaakkola 99] Jaakkola, T. S. and Haussler, D.: Exploiting generative models in discriminative classifiers, in Kearns, M. S., Solla, S. A., and Cohn, D. A. eds., *Advances in Neural Information Processing Systems 11*, Cambridge, MA (1999), MIT Press
[Joachim 98] Joachim, T.: Text categorization with support vector machines, in *Proceedings of the tenth European Conference on Machine Learning* (1998)
[Kashima 02] Kashima, H. and Koyanagi, T.: Kernels for Semi-Structured Date, in *Proceedings of the Nineteenth International Conference on Machine Learning*, pp. 291–298, San Francisco, CA (2002), Morgan Kaufmann
[Kashima 03] Kashima, H., Tsuda, K., and Inokuchi, A.: Marginalized Kernels between Labeled Graphs, in *Proceedings of the Twentieth International Conference on Machine Learning*, San Francisco, CA (2003), Morgan Kaufmann
[Kin 02] Kin, T., Tsuda, K., and Asai, K.: Marginalized Kernels for RNA Sequence Data Analysis, in *Genome Informatics 13*, pp. 112–122 (2002)
[Kudo 03] Kudo, T. and Matsumoto, Y.: Fast Methods for Kernel-based Text Analysis, in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* (2003)
[Kudo 05] Kudo, T., Maeda, E., and Matsumoto, Y.: An Application of Boosting to Graph Classifications, in *Advances in Neural Information Processing Systems* (2005)
[Leslie 02] Leslie, C., Eskin, E., and Noble, W. S.: The spectrum kernel: A string kernel for SVM protein classification, in Altman, R. B., Dunker, A. K., Hunter, L., Lauerdale, K., and Klein, T. E. eds., *Proceedings of the Pacific Symposium on Biocomputing*, pp. 566–575, World Scientific (2002)
[Leslie 03] Leslie, C., Eskin, E., Weston, J., and Noble, W.: Mismatch String Kernels for SVM Protein Classification, in Becker, S., Thrun, S., and Obermayer, K. eds., *Advances in Neural Information Processing Systems 15*, Cambridge, MA (2003), MIT Press
[Lodhi 02] Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., and Watkins, C.: Text classification using String Kernels, *Journal of Machine Learning Research*, Vol. 2, pp. 419–444 (2002)
[Manning 99] Manning, C. D. and Schütze, H.: *Foundations of Statistical Natural Language Processing*, The MIT Press, Cambridge, MA (1999)
[Mitchell 97] Mitchell, T.: *Machine Learning*, McGraw-Hill (1997)
[Salton 88] Salton, G. and Buckley, C.: Term weighting approaches in automatic text retrieval, *Information Processing and Management*, Vol. 24, No. 5, pp. 513–523 (1988)
[Schölkopf 88] Schölkopf, B., Smola, A., and Müller, K.-R.: Nonlinear component analysis as a kernel eigenvalue problem, *Neural Computation*, Vol. 10, pp. 1299–1319 (1988)
[Schölkopf 04] Schölkopf, B., Tsuda, K., and Vert, J.-P. eds.: *Kernel Methods in Bioinformatics*, MIT Press, Cambridge, MA (2004)
[Shawe-Taylor 04] Shawe-Taylor, J. and Cristianini, N.: *Kernel Methods for Pattern Analysis*, Cambridge University Press (2004)
[Suzuki 03] Suzuki, J., Hirao, T., Sasaki, Y., and Maeda, E.: Hierarchical Directed Acyclic Graph Kernel: Methods for Structured Natural Language Data, in *Proceedings of the Fouty-first Annual Meeting of Association for Computational Linguistics* (2003)
[Suzuki 04] Suzuki, J., Isozaki, H., and Maeda, E.: Convolution Kernels with Feature Selection for Natural Language Processing Tasks, in *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics* (2004)
[Tsuda 02] Tsuda, K., Kin, T., and Asai, K.: Marginalized Kernels for Biological Sequences, *Bioinformatics*, Vol. 18, No. Suppl. 1, pp. S268–S275 (2002)
[Vadhan 01] Vadhan, S. P.: The Complexity of Counting in Sparse, Regular, and Planar Graphs, *SIAM Journal on Computing*, Vol. 31, No. 2, pp. 398–427 (2001)
[Valiant 79a] Valiant, L. G.: The Complexity of Computing the Permanent, *Theoretical Computer Science*, Vol. 8, pp. 189–201 (1979)
[Valiant 79b] Valiant, L. G.: The Complexity of Enumeration and Reliability Problems, *SIAM Journal on Computing*, Vol. 8, No. 3, pp. 410–421 (1979)
[Vapnik 95] Vapnik, V.: *The Nature of Statistical Learning Theory*, Springer Verlag (1995)

〔担当委員：伊藤 公人〕

2005 年 6 月 30 日　受理

著　者　紹　介

鹿島　久嗣(正会員)

1999 年 3 月京都大学工学研究科応用システム科学専攻修士課程修了．同年 4 月日本アイ・ビー・エム（株）入社．東京基礎研究所に所属．機械学習，データマイニングの研究に従事．

坂本　比呂志(正会員)

1996 年 3 月九州大学大学院システム情報科学研究科情報理学専攻修士課程修了．同年 4 月日本学術振興会特別研究員 (DC1)．1998 年 12 月同研究科博士課程修了．1999 年 1 月九州大学大学院システム情報科学研究科情報理学専攻助手．2003 年 8 月から九州工業大学情報工学部助教授．現在に至る．機械学習と計算量理論，Web 上のテキストデータからの知識獲得，データ圧縮および効率的なデータ構造の研究に従事．博士（理学）．

小柳　光生

1999 年 3 月筑波大学理工学研究科修士課程修了．同年 4 月日本アイ・ビー・エム（株）入社．東京基礎研究所に所属．XML，サーバサイドアプリケーション基盤技術，業務プロセスアプリケーションの研究に従事．