1N-5

木構造類似度算出法のための複数観点の抽出法

池田 健人 † 波多野 賢治 ††
† 同志社大学大学院文化情報学研究科 †† 同志社大学文化情報学部

1 はじめに

近年，計算機の性能向上や一般ユーザへの普及により，個人が大量のデータを扱う機会が増加した．その中でも，XML などの木構造データは階層構造を用いることにより多様な表現が出来るため，様々なデータで用いられており，今後もその利用は増え続けると考えられる．そのため，大量にあるデータの中からユーザが望むものを的確に検索するための類似度算出法が求められている．木構造の類似度算出法として Tree Edit Distance [3] が有名である．Edit Distance は，二つの比較対象を最短何回の編集操作 (挿入，削除，置換) によって同一の形に出来るか，という考え方で算出する距離である．元々は文字列を対象とした String Edit Distance [1] であり，後に Tree Edit Distance として木構造へ適用された．しかし，この手法では比較する木のノード数に大きく影響されるという問題点がある．また，木構造データが有する複数の特徴の一面からしか捉えられていない．

そこで，本稿では木構造データに含まれる複数の観点を選択し，それらの特徴を反映させた類似度算出法の提案を行う．

2 提案手法

本節では，類似度算出法に用いる観点の選択を行い，その観点を用いた算出法の提案を行う．

2.1 複数特徴量の観点選択

類似度算出法で用いる観点を選択するために，木構造データから取得可能な観点による特徴量の分析を行う．対象データとして複数のデータを用いることにより一つの DTD のみに適した選択にならないよう，Java 言語のソースコードの構文木情報をマークアップするためのモデルである JX-model [2] の DTD に従った 517 件，SIGMOD Record にある 1999 年版の論文に関する 5 種類の DTD に従った 920 件，2002 年版の論文に関する 4 種類の DTD に従った 48 件の計 1,555 件の XML データを用い，木構造データからノード数，深さ，葉ノード数，ノードの種類数，エッジの種類数を求め，主成分分析を行う．

その結果，いずれのデータに対してもノード数と葉ノード数，ノードの種類数とエッジの種類数に関連があることが判明した．そのため，これらの組のデータにさらに回帰分析を行ったところ，いずれも決定係数が 0.9 以上であり線形関係にあるため，これらの組のそれぞれ一方の値を特徴量の観点とする．よって，一つの木構造を特徴付ける観点として深さ，葉ノード数，ノードの種類数を用いる．

以上の様に，一つの木から表現される特徴を三つ選択するが，二つの木を比較する際にも取得出来る特徴がある．本稿では五つの観点から類似度の算出を行う．二つの木の比較時の特徴として α と β，一つの木の特徴として δ と γ と ζ を以下で定義する．

2.2 編集操作に関する特徴量 α

α は，編集操作においてノードの個数よりもノードの出現の有無を重視する特徴量である．まず，各編集操作に対する重み付き編集コスト α_{E_i} を求める．α_{E_i} は一回目の編集操作のコストが α_{E_1} を示しており，Tree Edit Distance に必要な編集操作の回数だけ i を増やしていく．n，m はノード，ε は空ノード，T は木，$N_t(T,n)$ は T に存在する n の数，$N(T)$ は T のノード数，$E\{\varepsilon \mapsto n\}$ は n の挿入，$E\{n \mapsto \varepsilon\}$ は n の削除，$E\{n \mapsto m\}$ は n から m への置換を意味する．

$$\alpha_{E_i} = \begin{cases} \dfrac{1}{N_t(T_1,n)+N_t(T_2,n)} & (\text{if } E\{\varepsilon \mapsto n\} \text{ or } E\{n \mapsto \varepsilon\}) \\ \dfrac{2}{(N_t(T_1,n)+N_t(T_2,n))+(N_t(T_1,m)+N_t(T_2,m))} & (\text{if } E\{n \mapsto m\}) \end{cases} \tag{1}$$

次に，α_{E_i} を用い，α を式 (2) のように定義する．α_{E_i} の和をさらに比較する二つの木のノード数の平均で割ることにより，木全体のノード数による影響を小さくしている．α の値は，$\alpha \geq 0$ であり，必要な編集操作の回数が増えるにつれて大きな値をとる．

$$\alpha = \frac{2}{N(T_1)+N(T_2)} \sum_i \alpha_{E_i} \tag{2}$$

2.3 兄弟ノードの類似に関する特徴量 β

β は，比較する木の同じ深さにおける兄弟ノードの並びの類似に関する特徴量である．同じ深さの兄弟ノードの並びに対して String Edit Distance で距離を算出し，比較対象のノード数の大きい方で割る．なお，式 (3) の i は比較する二つの木が共通して有する深さまで繰り返す．β は，$\beta \geq 0$ の値をとる．

$$\beta = \sum_i \frac{SED\{Str_{depth=i}(T_1), Str_{depth=i}(T_2)\}}{\max\{N_{depth=i}(T_1), N_{depth=i}(T_2)\}} \tag{3}$$

2.4 木の特徴の差に関する特徴量 γ，δ，ζ

γ，δ，ζ は，比較を行う木 T_1，T_2 が持つ特徴の違いを表す観点であり，それぞれ木の深さ，葉ノード数，ノードの種類数を用いる．用いる値は違うが，算出方法は共通であり式 (4) で算出する．$f(T)$ は，γ 算出時には木 T の深さ，δ 算出時には T の葉ノード数，ζ 算出時には T のノード種類数を代入する．

$$\gamma \text{ or } \delta \text{ or } \zeta = \begin{cases} 0 & (\text{if } f(T_1)=0 \cap f(T_2)=0) \\ 1-\dfrac{\min\{f(T_1),f(T_2)\}}{\max\{f(T_1),f(T_2)\}} & (\text{else}) \end{cases} \tag{4}$$

これらの値は，各観点から求めた値が近い場合に，その近さに応じて値を小さくする働きがあり，$0 \leq \gamma \leq 1$, $0 \leq \delta \leq 1$, $0 \leq \zeta \leq 1$ の値をとる．

2.5 非類似度 $disSim(T_1, T_2)$

α，β，γ，δ，ζ の各観点に対し，パラメータを付与した非類似度である式 (5) を提案する．この値は，値が小さいほど類似したデータであると判断出来る．パラメータの値は $w_1 + w_2 + w_3 + w_4 + w_5 = 1$ である．

$$disSim(T_1,T_2) = w_1\alpha + w_2\beta + w_3\gamma + w_4\delta + w_5\zeta \tag{5}$$

3 評価実験

本提案手法の評価を行うため，クラスタリングを用いた評価実験を行う．複数の DTD による XML を用意し，提案手法によってそれらの混合したデータを DTD ごとにクラスタリングが可能か確かめる．同じ DTD に従うという事は木の構造においていくつかの観点で同じ規則に従っていると判断出来るため，クラスタリング結果から提案手法の考察を行う．

Selection of Several Values for the Calculating Similarities of Tree Structures

† Kento IKEDA (ikeda@ilab.doshisha.ac.jp)
†† Kenji HATANO (khatano@mail.doshisha.ac.jp)
Graduate School of Culture and Information Science, Doshisha University (†)
Faculty of Culture and Information Science, Doshisha University (††)

Copyright ©2011 Information Processing Society of Japan.
All Rights Reserved.

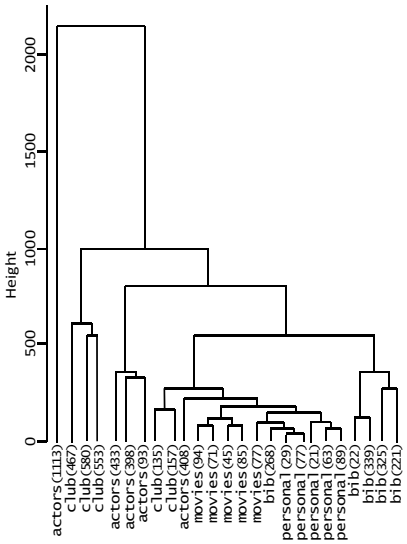

図 1: Tree Edit Distance による異種データのクラスタリング結果

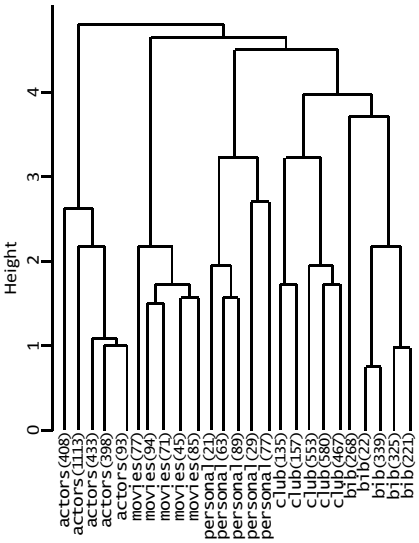

図 2: 提案手法による異種データのクラスタリング結果

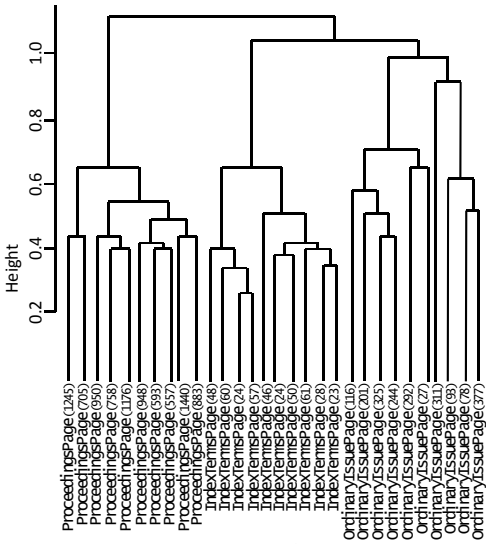

図 3: 提案手法による同種データのクラスタリング結果

3.1 実験手順

複数の x 種類の DTD に従った XML が混在したデータに対し，提案手法の値を用いてクラスタリングを行い，クラスタ数 x の時点で XML を DTD ごとに分類出来るかを確認する．クラスタリングにはデータがいくつかの集団にかたまっている場合も，一つの方向に並んでいる場合にも使用することが可能な群平均法を用いる．実験は，それぞれの DTD に従う XML 群よりランダムに抽出した XML を用いクラスタリングを行う試行を異種データ群と同種データ群の二種類の対象データに対し，それぞれ 10 回ずつ行う．

3.2 対象データ

対象データは異種データと同種データの二種類で行う．この二つのデータ違いは，木構造を構成するノードの種類の重複度合いである．

異種データでは，DTD だけでなく，表現するデータが違うので出現するノードもそれぞれ異なる傾向にある．そのため，ノードの種類に着目した類似度を用いればクラスタリングが可能だと予測出来る．データは，NIAGARA Query Engine[1] で公開されている文献データ (`bib`) 16 件，部活データ (`club`) 12 件，名簿データ (`personal`) 20 件，俳優データ (`actors`) 483 件，映画データ (`movies`) 490 件を用い，各データからランダムに 5 件ずつ抽出した計 25 件を用いる．

同種データでは，SIGMOD Record[2] で公開されている 1999 年版の会議論文集データ (`ProceedingsPage`) 17 件，会議論文集ではない通常の論文集データ (`OrdinaryIssuePage`) 51 件，各論文に関するインデックスデータ (`IndexTermsPage`) 920 件を用い，各データからランダムに 10 件ずつ抽出した計 30 件を用いる．このデータは，三種類の DTD すべてが論文に関するデータを表現しているため，利用されるノードの種類が異種データの場合と比較して高い割合で存在している．つまり，同種データではノードの種類に着目するだけでは DTD の区別が出来ず，他の観点を用いてクラスタリングを行う必要がある．

3.3 結果

異種データに従来の Tree Edit Distance を用いたクラスタリング結果を図 1 に示す．ラベル名の後ろに追記してる数字はノード数を示している．この結果より，Tree Edit Distance では比較する木のノード数の差による影響を受け，DTD ごとへのクラスタリングは出来ない事が分かる．また，ノード数の差は小さいが，比較対象のノード数が大きい場合も編集操作の回数が自然と増加するため，影響を受けやすい．同種データに関しても同様である．以下で提案手法による結果を示す．

異種データのクラスタリングを行った結果，パラメータ $w_1 = w_3 = w_4 = w_5 = 0, w_2 = 1$ にした場合，つまり β の値を用いた場合に図 2 の様に常に五種類の DTD のデータをそれぞれの DTD ごとに適切なクラスタ分けが出来た．これは，対象データが異種であるため，構成されるノードの種類が異なることが影響していると考えられる．そのため，兄弟ノードの並びに着目する β において，同種のデータでは出現するノードが共通するために，クラスタリングが可能になった．

同種データのクラスタリングを行った結果，パラメータ $w_1 = w_2 = w_4 = w_5 = 0.15, w_3 = 0.40$ にした場合に図 3 の様に三種類の DTD のデータをそれぞれの DTD ごとに常に適切なクラスタ分けが出来た．w_3 つまり γ である木の深さによる影響を強くすれば効果的に働いていると言えるが，γ だけではクラスタリングが適切に行われなかった．三種類の DTD のうち，二つの DTD が同じ深さで定義されているため，γ だけでは二つのクラスタへの分割までしか出来ない．そのため，γ 以外の観点が，不足している情報を補完し，クラスタリングが可能になったと考えられる．

以上の結果より，パラメータを適切に調整することにより，その時々に必要とする観点を重視した類似度を算出し，目的に合ったデータを提示することが可能になると考えられる．

4 おわりに

本稿では，木類似度算出法に必要な観点の選択を行い，その観点を用いた類似度算出法の提案を行った．パラメータの調整を行うことにより，異なる観点を重視した類似度を算出が可能となり，それぞれの目的に適した類似度を算出が出来ることを確認した．しかし，パラメータの目安となる値が判明していないため，今後は適切なパラメータの推定を行う仕組みが必要であると考えられる．

謝辞

本研究の一部は，同志社大学大学院文化情報学研究科研究推進補助金及び財団法人 日揮・実吉奨学会 研究助成金 (研助第 2144 号) によるものである．ここに記して謝意を表す．

参考文献

[1] Vladimir Iosifovich Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, Vol. 10, No. 8, pp. 707–710, 1966.

[2] Katsuhisa Maruyama and Shinichiro Yamamoto. A CASE tool platform using an XML representation of java source code. *4th IEEE International Workshop on Source Code Analysis and Manipulation (SCAM'04)*, pp. 158–167, 2004.

[3] Kuo-Chung Tai. The tree-to-tree correction problem. *J. ACM*, Vol. 26, pp. 422–433, July 1979.

[1] `http://www.cs.wisc.edu/niagara/`

[2] `http://www.sigmod.org/publications/sigmod-record/xml-edition`

Copyright ©2011 Information Processing Society of Japan.
All Rights Reserved.