Vol.2010-NL-196 No.3
Vol.2010-SLP-81 No.3
2010/5/27

木の編集距離を用いた文の類似度計算方式

三上崇志[†]　平野敬[†]　川又武典[†]

業務の効率化や知識獲得を目的として文書の自動分類技術や類似文検索技術の要求が高まっている．従来これらの技術に対して，ベクトル空間モデルによる研究が行われてきたが，文構造を考慮することができない．そこで本稿では，自然文を木構造グラフに変換して解析し，同型構文や意味が類似する文の距離計算方式を提案する．提案方式では木の編集距離を応用して文と文の距離を計算する．木構造内のノード移動を考慮したコスト計算や子ノードのソートを行うなどの改良により，180 文を 20 クラスタに分類する実験において F 尺度 0.738 を得た．

Calculating Similarity between Sentences Using Tree-Edit-Distance

Takashi Mikami[†], Takashi Hirano[†] and Takenori Kawamata[†]

It is required that the technique of auto classification of documents and the technique of retrieving similar sentences to a query sentence are enhanced in order to increase efficiency of business and require more knowledge. In traditional researches, the vector space model has been developed for those techniques, but it cannot take in structures of a sentence. This report proposes a method of calculating similarity between sentences which have similar structures or between sentences which have similar meanings, by using tree structures gotten by syntactic analysis. This method calculates similarity between sentences using developed tree-edit-distance. In the experiment, the F-measure for classification of 180 sentences into 20 categories was 0.738.

1. はじめに

近年，インターネットの発達などにより参照可能なテキスト情報が増加している．これに伴い，業務の効率化や知識獲得を目的として文書の自動分類技術や類似文検索技術の要求が高まっている．従来これらの技術に対して，ベクトル空間モデルによる研究が行われてきた．しかし，ベクトル空間モデルでは文の構造を捉えることが出来ず，言語学的な意味情報を単語の共起情報という部分的な側面でしか反映できない．文の構造を考慮することによって同型文の分類や係り受け構造を指定した検索などができると考えられる．そこで筆者らは，自然文を木構造グラフに変換して解析し，Web サイトの FAQ における同型質問文の分類や報告書の記載内容を係り受け構造を用いて分析する方式を検討している．このような分析を実現するには，次の 3 つを構築する必要があると考えている．

(1)木構造グラフの同型判定処理（距離計算）
(2)グラフパターン抽出処理
(3)クラスタリング処理

今回，上記(1)木構造グラフの同型判定処理（距離計算）に関して，木の編集距離を用いた計算方式を提案し，同型質問文や意味の類似する文を分類する実験を行った．木の編集距離を用いた距離計算では，文や単語の見出しではなく構造の類似性を距離に反映することができ，単語が異なるだけで同じ構文を持つ文の発見や，部分的に同じ構文を持つ文の検索などに適用できる．

以降，2章で関連研究について述べ，3章で提案する距離計算方式について説明する．4章で提案方式による文の分類実験の結果を示し，5章で実験結果の考察を行う．最後に6章でまとめと今後の課題を述べる． a

2. 関連研究

テキスト自動分類の従来法として，ベクトル空間モデルによる分類方式[1][2][3]が提案されている．河合[1]は単語の意味属性を用いることで分類精度を向上させ，湯浅ら[2]は単語の共起関係を利用した分類方式を提案している．また，藤井ら[3]は単語の共起情報から多義性を解消することにより分類精度を向上させている．しかし，これらの方式は文の構造を捉えておらず，言語学的な意味情報を語句の共起情報という部分的な側面でしか分類に反映できないという課題があった．この課題に対して，文の構造をグラフ構造（木構造）として捉えて文書あるいは文の分類を行う先行事例がある．工藤ら[4]は部分木を素性とする Decision Stump とそれを弱学習器とするブーステ

a [†] 三菱電機株式会社　情報技術総合研究所
Information Technology R&D Center Mitsubishi Electric Corporation

Vol.2010-NL-196 No.3
Vol.2010-SLP-81 No.3
2010/5/27

ィングアルゴリズムにより文の分類を行っている．しかし部分木の抽出は最右拡張というアルゴリズムに従っており，必ずしも言語的に意味のある部分木になっているとは限らず，文の意味を分類結果に反映させるのは困難である．また人が意味のある部分木を与えてそれを検索するような用途には使えない．市川ら[5]は部分木単位でインデックスを作成し，部分木とのマッチングにより類似する文を高速に検索する Tree Overlapping を開発している．類似文の検索に部分木のインデックスを利用するのは有効な手段と思われるが，品詞や単語の見出しの微妙な違いを柔軟に照合することはできず，文の分類などには向かないと考えられる．

3. 木の編集距離による文の類似度計算方式

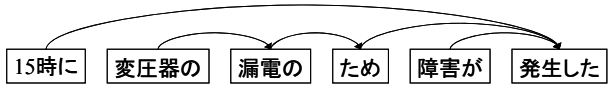
構文解析した結果の木構造グラフに対して，木の編集距離を用いて文と文の距離を計算する方式について説明する．

3.1 構文解析結果

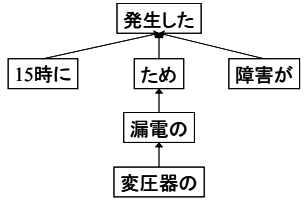
文を構文解析した結果は，木構造で表現できる．例えば次のような文を考える．

15 時に変圧器の漏電のため障害が発生した．

係り受け解析を行うと以下のような結果が得られる．



上の図において，各四角は 1 文節を示し，1 文節から出ている矢印はその文節が矢印の先に係っていることを意味する．この係り受け解析結果を変形すると以下のような木構造になる．



3.2 木の編集距離

木構造間の距離を木の編集距離として定義する．2 つの木構造間の編集距離とは，一方の木構造を編集することでもう一方の木構造に一致させる際の総コストとして定義される．一般に木構造の編集はノードの「挿入」・「追加」・「削除」により行う．

プログラムにより編集距離を求めるには動的計画法などが用いられる[6]. 木構造 T_1，T_2 のノード数をそれぞれ $|T_1|$，$|T_2|$ とすると，動的計画法を用いる場合の計算量は $O(|T_1|^2|T_2|^2)$ となる．計算量に関しては，$O(|T_1||T_2|\min(L_1,D_1)\min(L_2,D_2))$（ただし，$L_i$ は T_i の葉ノードの数，D_i は T_i の深さ）などとなるアルゴリズムが提案されているが，本稿では拡張と実装の容易性のため動的計画法を用いることとし，Bille[6]6)が定式化している以下の式に従って木の編集距離を求める．

$$d(\phi,\phi)=0 \tag{1}$$

$$d(F_1,\phi)=d(F_1-v,\phi)+del(v) \tag{2}$$

$$d(\phi,F_2)=d(\phi,F_2-w)+ins(w) \tag{3}$$

$$d(F_1,F_2)=\min\begin{cases} d(F_1-v,F_2)+del(v) \\ d(F_1,F_2-w)+ins(w) \\ d(F_1(v),F_2(w))+d(F_1-T_1(v),F_2-T_2(w))+rep(v,w) \end{cases} \tag{4}$$

ただし，F_i は順序付けされた木の集合（森，Forest），$d(F_1,F_2)$ は F_1，F_2 間の距離，ϕ は空集合，v は F_1 に属するノードのうち最も右側に位置するルート，w は F_2 に属するノードのうち最も右側に位置するルート，$del(v)$ は v を削除するコスト，$ins(w)$ は w を挿入するコスト，$rep(v,w)$ は v を w に置換するコスト，$F_i(v)$ は F_i のノードまたは木のうち v の子，$T_i(v)$ は v をルートとする木を表し，F_i-v は v を削除した F_i のノードまたは木，$F_i-T_i(v)$ は $T_i(v)$ のノード全てを削除した F_i のノードまたは木とする．上記の式は木の編集距離を森に対して自然に拡張したものであり，木間の編集距離は木を 1 つずつ含む森の編集距離として求めることが出来る．

一般に $^\forall v$，$^\forall w$ に対して $del(v)$，$ins(v)$，$rep(v,w)$ は定数とするが，本稿では分類対象の文集合におけるノードのTF-IDFによる重み付けを行う．また，$rep(v,w)$ はノードの品詞などを用いて柔軟なコスト計算を行う．詳細は3.3.1節で説明する．

3.3 木の編集距離の改良

一般的な木の編集距離に対して，次の 3 方式による改良を行う．

① TF-IDF 重み付け
② ツリー内移動の導入
③ 子ノードのソート

TF-IDF 重み付けでは，木のノードとなる単語の重要度を考慮することで，頻繁に出現する文の構成要素として重要度の低い文節（“「”，“」”，“次に”，“そして”，など）

Vol.2010-NL-196 No.3
Vol.2010-SLP-81 No.3
2010/5/27

の距離計算への影響を軽減する．ツリー内移動の導入は，同じ文節が異なる順序で出現する木構造間では距離が小さくなる効果を狙う．子ノードのソートは，日本語に見られるような語順の曖昧性を吸収する効果を狙う．それぞれ次項以降で説明する．

3.3.1 TF-IDF重み付けと置換コスト

主辞までの見出しによる TF-IDF を編集距離に重みとして与える．TF-IDF によるノードvの重み$TFIDF(v)$を以下の式で定義する．

$$TFIDF(v) = tf(v) \times idf(v) \tag{5}$$

$$tf(v) = \frac{n_v}{\sum_{\forall k \in d, \forall d \in D} n_k} \tag{6}$$

$$idf(v) = \log\left(\frac{|D|}{|\{d : v \in d\}|} \right) \tag{7}$$

ただし，n_vはノードvの出現する頻度，Dは文書集合，dは文書集合Dに含まれる文書である．各ノードの重みは全文書で共通とするため，式(6)のように$tf(v)$を定義した．

上記 TF-IDF 重みを利用して$del(v), ins(v)$を次のように定義する．

$$del(v) = ins(v) = base \times TFIDF(v) \tag{8}$$

$base$は正の定数とする．

$rep(v,w)$は文節の文法的な役割を重視し，次のように定義する．

- vとwの機能語の品詞が等しく，主辞の見出しが等しい
  ⇒ 0.0
- vとwの機能語の品詞が等しく，主辞の見出しが異なり，品詞は等しい
  ⇒ $(del(v) + ins(w))/8.0$
- vとwの機能語の品詞が等しく，主辞の見出し・品詞が異なる
  ⇒ $(del(v) + ins(w))/6.0$
- vとwの機能語の品詞が異なり，主辞の見出しだけが等しい
  ⇒ $(del(v) + ins(w))/4.0$
- vとwの機能語の品詞が異なり，主辞の見出しも異なる
  ⇒ $(del(v) + ins(w))/2.0$

ただし主辞とは，文節の中でメインとなる形態素のこととし，機能語とはその文節の文法的な役割を持つ語とする．例えば「変圧器の」という文節は「変圧」,「器」,「の」という 3 つの形態素から成り，主辞は「器」，機能語は「の」となる．

3.3.2 ツリー内移動の導入

ツリー内でのノードの移動は，単なる削除＋挿入よりもコストが低くなるのが妥当と思われる．そこで以下のように挿入コスト・削除コスト・置換コストを変動させる．

① 挿入するノードvと等しい主辞の見出しを持つノードが自分のツリーF_1内に存在する．
  ⇒ $ins'(v) = ins(v)/2.0$

② 削除するノードvと等しい主辞の見出しを持つノードが相手のツリーF_2内に存在する．
  ⇒ $del'(v) = del(v)/2.0$

③ 置換後のノードwと等しい主辞の見出しを持つノードが自分のツリーF_1内に存在し，置換前のノードvと等しい主辞の見出しを持つノードが相手のツリーF_2内に存在しない．
  ⇒ $rep'(v,w) = rep(v,w)/2.0$

④ 置換後のノードwと等しい主辞の見出しを持つノードが自分のツリーF_1内に存在せず，置換前のノードvと等しい主辞の見出しを持つノードが相手のツリーF_2内に存在する．
  ⇒ $rep'(v,w) = rep(v,w)/2.0$

⑤ 置換後のノードwと等しい主辞の見出しを持つノードが自分のツリーF_1内に存在し，置換前のノードvと等しい主辞の見出しを持つノードが相手のツリーF_2内に存在する．
  ⇒ $rep'(v,w) = rep(v,w)/4.0$

⑥ ①～⑤のいずれにも当てはまらない場合は，通常のコストとする．

上記方式ではF_1にもF_2にも含まれるノードvをF_1から一旦削除して他の場所に挿入

Vol.2010-NL-196 No.3
Vol.2010-SLP-81 No.3
2010/5/27

するという単純なケースの場合，コストが軽減される．図 1に移動コストが軽減される例を示す．簡単のためTF-IDFを考慮せず，$TFIDF(v)$を 1.0，$base$を 2.0 とした．

しかし，上記方式ではF_1にはvが 1 個だけあり，F_2には 3 個ある場合でも，F_1にvを 3 回挿入するコストは等しく小さくなるため，ノードの移動を忠実に再現しているとは言えず，複製コストが減少しているに過ぎない．本稿では簡単のためこのように実装するが，今後実装方式は検討したい．

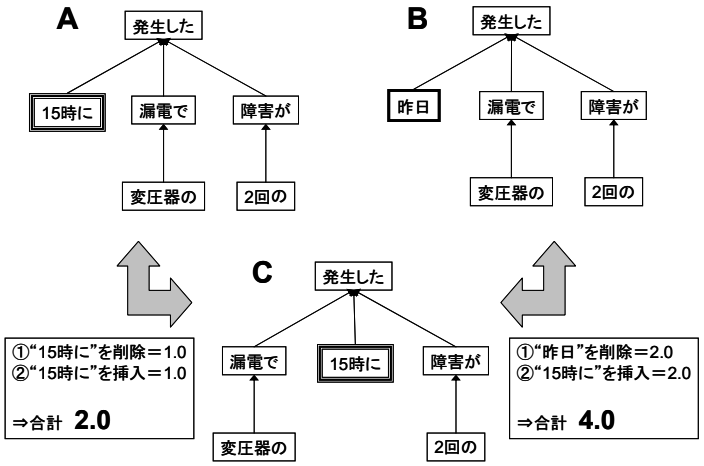


図 1　移動コストが軽減される例

3.3.3 子ノードのソート

日本語では修飾語などの語順が変わっても意味としては変わらない場合が多い．例えば3.1節の例では以下の 3 文はどれも同じ意味と考えられる．

・15 時に障害が変圧器の漏電のため発生した．
・変圧器の漏電のため 15 時に障害が発生した．
・変圧器の漏電のため障害が 15 時に発生した．

これらはいずれも「発生した」に係っている文節（「15 時に」，「ため」，「障害が」）の順番を入れ替えたものであり，木構造で言えば「発生した」というノードの子ノードの順番を入れ替えたものである．木の編集距離を求める前に子ノードのソートを行うことでこのような語順の曖昧性を吸収することが出来る．ソートの方法は一意に求まれば良いと考えられるが，本稿では3.3.2節で定義した置換コストに合わせて，機能語の品詞，主辞の見出し，主辞の品詞の順に優先順位をつけてソートする．

4. 実験

木の編集距離を用いた距離計算では，文や単語の見出しではなく構造の類似性を距離に反映することができる．そのため，単語が異なるだけで同じ構文を持つ文の検索や分類，部分的に同じ構文を持つ文の検索などに適用できる．本実験では *k-medoids* 法を用いて与えられた文集合の文を決められた数のカテゴリに分類し，その分類精度を評価することで本方式の有効性を示す．

まず予備実験として使用されている単語は異なるが構文が類似する文の集合を分類し，3章で説明した方式の効果を確認する．次に電気機器のQAデータに対して質問文の分類実験を行い，意味的に類似する文書（複数の文）に対する分類精度の評価する．

4.1 実験条件

4.1.1 係り受け解析器

文から木構造を作成する際に必要な係り受け解析は，CaboCha[7]を用いる．

4.1.2 実験パラメータ

表 1に示したケースに対応する木の編集距離の計算式を用い，それぞれの場合に分類精度を評価する．

表 1　実験ケース

No.	TF-IDF	移動コスト軽減	子ノードソート
A	なし	なし	なし
B	なし	なし	あり
C	なし	あり	なし
D	なし	あり	あり
E	あり	なし	なし
F	あり	なし	あり
G	あり	あり	なし
H	あり	あり	あり

4.2 予備実験（構造類似文分類）

構造的に類似する文を分類し，分類精度を評価する．インターネットなどから収集した一般文と，それらと構造的に類似する文を分類する実験を行う．

Vol.2010-NL-196 No.3
Vol.2010-SLP-81 No.3
2010/5/27

4.2.1 構造類似文分類：実験データ

実験データはインターネットから収集したニュース記事などの一般文 10 件とそれらに文の構造が類似する文 100 件である．データの作成手順を下記に示す．

1. インターネットのニュース記事などから一般文を 10 件取得する．
2. 取得した文から実験協力者 5 人が人手で文構造が類似し，意味は異なる文を 2 つずつ作成する．

収集した元文と構造類似文の例を表 2に示す．

表 2　収集文と構造類似文の例

元文	構造類似文１	構造類似文２
肝機能は基準値内ですが，パターンから見てやや脂肪肝傾向が認められます．	スケジュールは予定通りですが，これまでの傾向から考えてそろそろ遅延が予想されます．	試験は順調ですが，経験から判断してかなり困難が予想されます．

4.2.2 構造類似文分類：実験手順

実験手順は次の通りである．

1. 元文 10 件と構造類似文 100 件の合計 120 件について係り受け解析を行う．
2. 係り受け解析結果から木構造を作成する．
3. 木の編集距離を用いて *k-medoids* 法で 10 個のクラスタに分類する．初期セントロイドは元文の 10 件を選ぶ．
4. １つの元文とそれに対応して作成された類似構文の集合を 1 つの正解クラスタとし，F 尺度を求める．

本実験においてはTF-IDFはあまり意味を成さないと思われるため，表 1で示したケースのうちA～Dに対応する編集距離の計算式を用い，それぞれの場合に分類精度を評価する．

4.2.3 構造類似文分類：実験結果

表 3に各分類結果に対するF尺度を示す．“No.”は表 1に対応している．F尺度は 0～1 の値を取り，値が大きいほど良い分類であることを示す．表 2のように単語の重なりがほとんどない文に対する分類において，いずれの場合も高いF尺度を示した．BおよびDでF尺度が落ちているが，実験協力者が文の構造を同じにして文を作成したつもりでも，係り受け構造が部分的に異なる場合があり，子ノードをソートすることでかえって構造の違いが顕著になってしまうことがあったためである．

4.3　QA質問文分類実験

本実験では，電気機器の QA と，その QA と意味が類似するように作成した文を実験データとする．

4.3.1 QA質問文分類：実験データ

実験データは電気機器の QA の質問文 20 件とそれに類似する質問文 160 件である．データの作成手順を下記に示す．

1. ある電気機器に対して寄せられた質問文を 20 件選択する．
2. 選択した質問文に対し，実験協力者 4 人が人手で類似する質問文を 2 つずつ作成する．

4.3.2 従来法

比較のため従来法として，単語共起と TF-IDF に基づいたクラスタリングを行う．従来法では文の形態素解析を行い，形態素の組成ベクトルを作成する．文間の距離は組成ベクトルのユークリッド距離として与えるが，TF-IDF により重み付けを行う．従来法におけるクラスタリングでは *k-means* 法を用いる．

4.3.3 QA質問文分類：実験手順

実験手順は次の通りである．

1. QA 質問文 20 件と類似質問文 160 件の合計 180 件について構文解析を行う．
2. 構文解析結果から木構造を作成する．
3. 木の編集距離を用いて *k-medoids* 法で 20 個のクラスタに分類する．初期セントロイドは QA 質問文の 20 件を選ぶ．
4. １つの QA 質問文とそれに対応して作成された質問文の集合を 1 つの正解クラスタとし，F 尺度を求める．

4.3.4 QA質問文分類：実験結果

表 4に各分類結果に対するF尺度を示す．従来法が最も高いF尺度を示し，提案方式についてはHのケースが最も良かった．提案方式において，A・Bに対してC・D，E・Fに対してG・HのF尺度は大きく向上しており，移動コスト軽減の効果が大きいことがわかる．TF-IDF，子ノードのソートは必ずしも効果が出ておらず，特に子ノードのソートはHのケースでしか精度の向上に寄与しなかった．

Vol.2010-NL-196 No.3
Vol.2010-SLP-81 No.3
2010/5/27

表 3　構造類似文分類

No.	F 尺度
A	0.889
B	0.814
C	0.888
D	0.843

表 4　QA 質問文分類

No.	F 尺度
従来法	0.972
A	0.489
B	0.384
C	0.693
D	0.671
E	0.480
F	0.427
G	0.673
H	0.738

5. 考察

QA 質問文の分類において，従来法が非常に高い精度を示したが，これは元になった質問文と，それに意味が類似するように作成された質問文が同一の単語を含む場合が多かったからである．逆に文の構造は異なるように作成されたものが多く，単語そのものの見出しだけでなく構造の類似度も考慮する提案方式には不向きなデータとなっていた．このようなデータに対しても H のケースではある程度の精度で分類できており，提案方式の効果は確認できたと考える．

子ノードのソートは H 以外のケースでは良い効果が現れなかったが，これは係り受け関係の微妙な違いが，子ノードをソートすることによって木構造上の大きな違いになってしまうことが原因と思われる．このような係り受け関係への強依存性に対して，H のケースでは移動コストの軽減がうまく働き，木構造の組み換えコストを抑えられたため精度が向上したと考えられる．しかしこれは偶然である可能性が高く，D のケースでは B のケースと比べて精度は落ちている．係り先が遠くはなれている場合はソートによる木構造への影響が大きくなるため，例えば一定上離れている場合はソートを行わないなどの改良が必要である．

同様に TF-IDF も必ずしも有効に働かなかった．今回は主辞だけの見出しで TF-IDF を計算したが，主辞までの見出しを使うなどパラメータを検討する必要がある．

6. おわりに

構文解析結果と木の編集距離を利用した文の分類方式を開発し，評価を行った．構造類似文の分類実験により従来法では分類不可能なデータに対して一定の精度で分類できることを確認した．また QA 質問文の分類実験では従来法より精度は劣るものの，TF-IDF，移動コストの軽減，子ノードのソートなどの改良により分類精度を向上できることを示した．ただし，TF-IDF，子ノードのソートは有効に働かない場合もあり，今後改良やパラメータの調整が必要であることが分かった．

また，今回の実験では文の分類を行いその精度評価を行ったが，今後，係り受け構造による文の検索などに応用していく予定である．検索結果の順位検定などによる評価を行うことで，より提案手法の有効性を示すことが出来ると考えている．

参考文献

1) 河合敦夫: 意味属性の学習結果にもとづく文書自動分類方式,情報処理学会論文誌, Vol.33, No.9, 1992.
2) 湯浅夏樹, 上田徹, 外川文雄: 大量文書データ中の単語間共起を利用した文書分類,情報処理学会論文誌, Vol.36, No.8, 1995.
3) 藤井洋一, 鈴木克志, 今村誠, 高山泰博: 共起情報を利用した文書の自動分類, 情報処理学会研究報告, 自然言語処理研究会報告, Vol.97, No.29, 1997.
4) 工藤拓, 松本裕治: 半構造化テキストの分類のためのブースティングアルゴリズム, 情報処理学会論文誌, 45(9), pp.2146-2156, 2004.
5) 市川宙, 橋本泰一, 徳永健伸, 田中穂積: テキスト構文構造類似度を用いた類似文検索手法, 情報処理学会情報基礎研究会, 79, May 2005.
6) Philip Bille: Tree Edit Distance, Alignment Distance and Inclusion, Technical report TR-2003-23, IT University of Copenhagen, March 2003.
7) 工藤拓, 松本裕治: チャンキングの段階適用による係り受け解析, 情報処理学会論文誌, Vol.2001, No.20, pp.1834-1842.