類似度を考慮した言語の同一性判定

呉 靭[†] 松野 浩嗣[†]

[†] 山口大学大学院理工学研究科

Yamamoto-Data と SilGIS-Data は異なる学者によって編成された世界諸言語に関するデータである．言語の別名の存在や表記ゆれなどのため，両データに含まれる言語の同一性を判定する必要がある．本研究の目的は，言語名または言語系統分類のゆれが原因で，同一言語として判定できなかった言語を検出することである．本論文では，まず言語系統木を用いた言語の同一性判定方法について述べる．次に，文字列アラインメントに基づく言語名と言語系統分類の類似度を考慮したゆれのある言語に対する同一性判定手法を提案する．さらに，Yamamoto-Data と SilGIS-Data について処理した結果を提示し，提案した類似度が有用かつ効果的であることを示す．

Identity Judgment of Languages Based on String Similarity

Ren WU[†] Hiroshi MATSUNO[†]

[†] Graduate School of Science and Engineering, Yamaguchi University

Yamamoto-Data and SilGIS-Data are world's languages data individually provided by different language researchers. Because of the existence of alternative names of languages as well as their ambiguities, some same languages are expressed by different writings in Yamamoto-Data and SilGIS-Data. Therefore, it is important to identify if two writings express a same language. Our purpose is to find a way to do such language identification by taking the ambiguities into account. In this paper, firstly we introduce World Language Tree and then propose a method to absorb the ambiguities of language names by applying string alignment technique. Our experimental result for the two language data shows that our proposed method is usefull and effective.

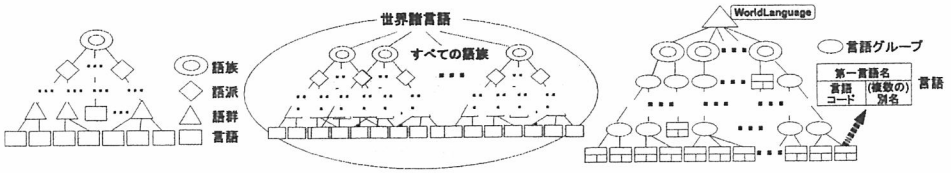
1 はじめに

異なる言語学者によって編成された言語データ（ここでは特に世界諸言語に関する属性情報を集めたデータを指す）を併せて言語情報処理に利用するとき，それらの言語データ間の言語の同一性判定処理が必要になることがある．

我々はこの問題を指摘し [1], [2]，言語名に加えて，言語の系統分類も考慮した言語の同一性判定の方法について考察してきた [2]．我々は，(i) 文献 [3] の「言語別語順データ」と「系統別語順分布表」の両データを合成し，変換したデータ（2,870 言語 [4]，以降 **Yamamoto-Data** とよぶ）[2], [6]，(ii) *Ethnologue* 第 15 版 Web サイト [5] から世界諸言語の属性と系統分類の情報を取得し，変換したデータ（7,229 言語，以降 **SilGIS-Data** とよぶ）[6]，の 2 つの言語データを対象とし，両データのそれぞれに含まれる 2 つの言語の言語名が一致し，かつ系統分類が一致するならば，その 2 つの言語は同一言語であるとして処理した結果，Yamamoto-Data の約 36% の言語が同定できた．しかし，その方法では言語名の表記ゆれに起因する言語名の曖昧性やそれぞれの言語データにおける言語系統分類の違いを考慮していなかったため，言語名または系統分類にゆれのある言語は同定できなかった．

そこで，本研究では文字列の類似性評価に広く用いられている動的計画法による文字列アラインメント [7]〜[10] に基づき，言語名と系統分類の類似度の概念を導入し，言語同一性判定の基準をゆるめて，言語名と言語系統分類がともに一致するという条件を満たさなくても，言語名または言語系統分類が類似していて，同一性を肯定することが相当であるなら，同一言語として判定する手法を提案する．

以下，**2.** では言語の同一性判定の必要性について述べる．**3.** では言語系統木を用いた言語名と言語の系統分類がともに一致する言語の検索法とその問題点について述べる．**4.** では文字列アラインメントおよび文字列の類似性評価の方法について説明した後，言語名の類似度と言語系統分類の類似度についての求め方，またこれらの類似度を用いた言語名類似言語と系統分類類似言語の検出手法を述べる．**5.** では，処理結果について述べ，本研究で提案した手法の有効性や，言語の同一性の肯定または否定を判定する



(A) 語族　(B) 世界諸言語　(C) 言語系統木

図 1: 言語系統分類と言語系統木

基準となる言語名または言語系統分類の類似度のしきい値の妥当性などを考察する．

2　言語同一性判定の必要性

言語の名前は唯一とは限らない．「日本語」を例にとれば，英語読みでは "Japanese"，日本語読みでは/nippon-go/と/nihon-go/，などの名前がある．

実際，1つの言語に複数の名前が付けられていることがよくある．*Ethnologue* 第 15 版 [5], [11] では，その研究および調査の結果がデータベースにまとめられ，公開されている．複数の言語名の中の 1 つは**第一言語名** [11]，その他は**別名** [11] とされている．SilGIS-Data にも言語の第一言語名と別名の情報が含まれている．

第一言語名および別名の指定は学者独自に行われているため，SilGIS-Data と異なる言語データにおいて同一言語が異なる第一言語名になっていたり，またはその逆で，本来異なる言語が異なる言語データにおいて**言語名**（第一言語名または別名を指す，以降も同じ）が同じだったりすることもある．また，別名による原因のほか，言語名の表記ゆれ（例えば，外来語の日本語表記で「バイオリン」と「ヴァイオリン」や，「サーバー」と「サーバ」など）による変化も含まれる（含めて**言語名のゆれ**とよぶ）．そのような言語名のゆれがある複数の言語データを併せて利用するために，言語の同一性判定処理が必要になる．

言語の同一性が問題となるのは，言語を識別するのに言語名が使われているデータを扱う場合である．言語に**言語コード**（ISO639_3 言語コード [11], [12]）が付与されている言語データではこのことは問題とならない．

言語コード体系の標準化が進み，近年は情報科学を用いた言語研究を意識して編成され，発表された言語資料は言語コードが付与されるようになった [13]．しかし，言語同一性判定の問題は依然として消え去っていないように思われる．それは比較的に最近発表された言語資料でも言語コードが付与されていないのが少なくないのが現実である．Routledge 社の *Atlas of the world's languages* [14] は世界諸言語に関する類型論的研究分野では価値の高い資料といわれているが，2007 年に改訂された第 2 版で言語コードは付与されていない．我々は，この資料に掲載されている世界諸言語が話されている地理情報を語順研究に活かせようとして，再び言語同一性の問題に直面させられた．

言語コードが付与されていない，価値ある言語資料は数多く存在する．言語の同一性の問題が障害となり，言語研究に活かせないのならば，それは大変残念なことである．人類最大の文化遺産ともいえる言語に関する資料を研究に活かせるようにすることは重要な意義をもつ．そして，言語同一性問題の解決はまさにその効果をもたらすものである．

3　言語系統木を用いた言語同一性判定

別名の情報を使えば，少なからずの言語に対し同一性が判明できる．しかし，異なる言語がそれぞれの言語データで同じ言語名になっていることもあり，言語名だけでは同一性判定ができないことがある．我々は，言語名に加え，言語系統分類も考慮した言語系統木を用いれば，同じ言語名をもつ言語の識別もできるようになる，と考える．本節では，その方法について述べる．

3.1　言語系統木

世界諸言語は多くの語族に分類され，1つの語族は1つの系統樹を構成する [15]．語族は系統樹の最大の分類で，語派と語群は同じ語族の中での中分類と小分類で，最下位にあるのが言語である．図 1(A) は語族のイメージを示している．

図 1(B) に示すように，語族の森を世界諸言語の下にまとめ，1本の**木**として扱うことにする．さらに，(i) 語派と語群をまとめて言語グループとし，(ii) 語族名，言語グループ名，第一言語名はそれぞれ木構造の節点のラベルとする．また，言語の言語コードと**（複数の）別名**（複数の別名をカンマでつないだ文字列）[11] は葉の節点である言語の属性情報として，それぞれもたせることにする．その構造を**言語**

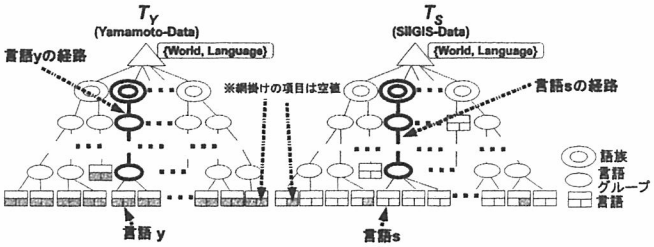


図 2: 2つの言語系統木 T_Y と T_S

系統木とよび，図1(C)に示す．言語系統木では，節点ラベルはアルファベットからなる文字列（**単一文字列**とよぶ）の集合として定義されている．

図2に Yamamoto-Data と SilGIS-Data がそれぞれ構成する言語系統木 T_Y と T_S を示す．

節点ラベルの形式について説明する．例えば，SilGIS-Data に含まれている言語 s は第一言語名 "Otomi, Estado de Mexico" のほかに，"Hnatho","Otomi del Estado de Mexico", "Otomi de San Felipe Santiago", "State of Mexico Otomi" という4つの別名をもち，言語コードは "ots" である．図2の T_S での言語 s は葉節点で，節点ラベルが {Otomi, Estado, de, Mexico} となる．つまり，節点ラベルは第一言語名 "Otomi, Estado de Mexico" の文字列をカンマ (,) または空白 (Space)（言語名中のこのようなアルファベットではない記号を**区切り子**とよぶ）で分割した "Otomi", "Estado", "de", "Mexico" という4つの単一文字列の集合で表されている．

また，言語系統木の葉節点の属性としての（複数の）別名は単一文字列集合の集合として定義されている．例えば，言語 s の（複数の）別名は "Hnatho, Otomi del Estado de Mexico, Otomi de San Felipe Santiago, State of Mexico Otomi" のように，4つの別名をカンマでつないだ文字列になっている．別名には区切り子として空白しか使われていないため，(複数の）別名をカンマで分割し，それぞれの別名に復元することが可能である．T_S では，言語 s の（複数の）別名は節点の属性情報の1つとして，{{Hnatho}, {Otomi, del, Estado, de, Mexico}, {Otomi, de, San, Felipe, Santiago}, {State, of, Mexico, Otomi}} のように単一文字列集合の集合として表されている．

SilGIS-Data はすべての言語に言語コードが付与されている．よって，T_S では葉節点の属性としての言語コード（例えば，言語 s は "ots" という文字列である）は必ず存在する．Yamamoto-Data の言語は言語コードが与えられていないため，T_Y のどの葉節点においても，言語コードは空値になっている（図2で網掛けの項目は空値である）．

または，名前が1つしかもたない言語，つまり別名のない言語，または別名の情報が与えられていない言語も存在する．そのような言語は言語系統木において（複数の）別名が空値となる．Yamamoto-Data では別名についての情報はないため，すべての言語につき（複数の）別名が空値である．

3.2 言語系統木を用いた完全一致言語の検索法

言語系統木の言語は葉にあたり，言語系統分類は**経路**（根の子から葉の親までの節点ラベルの列を指す，図 2 を参照）で表せる．例えば，言語 s の系統分類が "Oto-Manguean","Otopamean","Otomian","Otomi" であるなら，T_S での s の経路は {Oto, Manguean}, {Otopamean},{Otomian},{Otomi} という単一文字列集合の列で表される．

異なる2つの言語データのそれぞれに含まれる2つの言語に対し，系統分類が一致し，かつ言語名が一致するならば，この2つの言語は同一言語と判定してよい．このような2つの言語を**完全一致言語**とよぶ．

SilGIS-Data の言語には言語コードが付与されていて，Yamamoto-Data の言語には言語コードが付与されていない．Yamamoto-Data と SilGIS-Data の完全一致言語を見つけ出すために，T_S を基準に，T_Y の任意の言語 y に対し，次の処理（**処理 0**）を行う．

(1) 完全一致経路の検索

T_S から，T_Y の言語 y の経路と完全一致する経路を検索する．経路が完全一致するとは，(i) 経路上の節点の数が等しい，(ii) 経路上のすべての節点ラベルが一致する，という2つの条件をともに満たすことである．もちろん，このような経路は T_S に必ず存在しているわけではない．もし言語 y の経路と完全一致する経路が見つからなければ，処理は終了し，

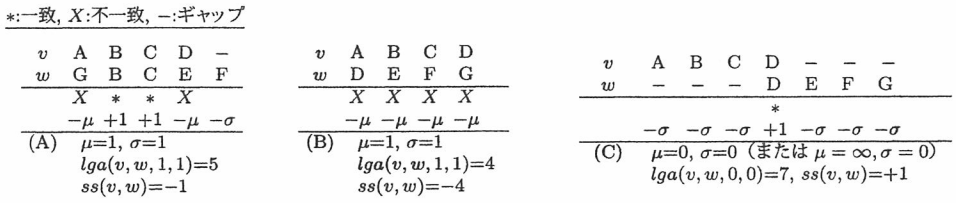


図 3: 2つの文字列のアラインメントの例

SilGIS-Data における言語 y の同一言語の存在は確認できないことになる.

(2) 経路と言語名がともに完全一致する言語の検索

完全一致経路が見つかったとき, T_S でその経路をもつ言語は $s_1, s_2, \cdots$ と複数存在する可能性がある. それらの複数の言語 $s_1, s_2, \cdots$ から, 完全一致言語名をもつ言語 s を検索する. 言語名が完全一致するとは, 次のいずれかの条件を満たすことをいう.

(i) s と y の第一言語名が一致する. つまり, それぞれの節点ラベルの文字列集合が一致すればよい.

(ii) s の (複数の) 別名に y の第一言語名が含まれている. (複数の) 別名は単一文字列集合の集合で表されている. y の言語名を表す節点ラベルは単一文字列集合で, (複数の) 別名を構成する集合要素の中, 任意の1つの要素と集合の一致が認めればよい. 例えば, y の節点ラベルが {OTOMI, STATE, OF, MEXICO} で, s の (複数の) 別名が {{Hnatho}, {Otomi, del, Estado, de, Mexico}, {Otomi, de, San, Felipe, Santiago}, {State, of, Mexico, Otomi}} ならば, 下線部分が一致し, この (ii) の条件は満たされている. なお, 言語名表記で大文字と小文字は区別しない (以降も同じ).

この処理0によって検出された T_S の言語 s と T_Y の言語 y は同一言語と判定する.

3.3 問題点

処理0の方法を用いれば, 完全一致言語の検出ができる. しかし, T_S の言語 s が T_Y の言語 y と本来は同一言語で, T_S での系統分類も同じであるが, 表記ゆれで両データでの言語名が少しでも異なっている場合は, 完全一致言語名をもつ言語の検索処理において失敗することになる.

また, 言語系統分類も学者独自の知見が盛り込まれ, 同一言語の系統分類が異なるデータで異なっていることが少なくない (**系統分類のゆれ**とよぶ). そのような場合, 両データでの言語名が一致していても, 完全一致経路の検索において失敗し, 言語名の照合処理対象からはずれることになる.

次節ではこれらの問題を解決する方法について述べる.

4 類似度を用いたゆれのある言語の検出

言語名または言語系統分類のゆれの性質にかんがみ, 我々は文字列アラインメントに基づく類似度を導入し, 言語名と言語系統分類について, それぞれ定量化を行う. T_Y と T_S に含まれる2つの言語につき, 完全一致言語名をもっていなくても, 系統分類が一致し, かつそれらの言語名が一定の条件を満たす類似関係をもっているならば, 同一言語と判定する. 同様に, 系統分類が異なっていても, 完全一致言語名をもち, かつ系統分類が類似しているならば, 同一言語と判定する.

言語名と言語系統分類の比較はいれずも文字列の比較を基本とするため, 本節では, まず動的計画法に基づく文字列類似性の一般的な評価手法について説明する.

4.1 文字列の類似性評価

2つの文字列がどの程度似ているかを計る尺度として**類似性スコア**がよく用いられる [9], [10]. v と w を2つの単一文字列とするならば, v と w の類似性スコア $ss(v, w)$ の求め方は次のようになる.

類似性スコア $ss(v, w)$ は v と w の**アラインメント** [9], [10] のスコアによって評価される. アラインメントとは2つの文字列の同じ部分を揃えた行列 ("−" を挿入した2つの文字列) のことで, 揃え方によっては複数のパターンが存在し得る.

図 3(A) に v="ABCD" と w="GBCEF" のアラインメントの1例を示す. v と w に対応するアラインメントの行はそれぞれ "ABCD−" と "GBCEF" である. "*", "X" はそれぞれ同じ列の2つの文字の一致, 不一致 (置換) を意味する [10]. "*", "X", "−" に対し, それぞれ $+1$, $-\mu$, $-\sigma$ のスコアを与え, またその数をそれぞれ n_m, n_{ms}, n_{id} で表すと, アラインメントのスコアは式 (1) のようにすべての列のスコアの和として評価される.

$$ss(v, w) = 1 \cdot n_m - \mu \cdot n_{ms} - \sigma \cdot n_{id} \quad (1)$$

表 1: 言語名を表す節点ラベル中の単一文字列の組合せとその類似度

		L_1 = {CHINANTECO, LALANA}	
		"CHINANTECO"	"LALANA"
L_2 = {Chinantec,Lalana}	"Chinantec"	***sd_ln*("CHINANTECO","Chinantec")=0.9**	*sd_ln*("LALANA","Chinantec")=0.2
	"Lalana"	*sd_ln*("CHINANTECO","Lalana")=0.2	***sd_ln*("LALANA","Lalana")=1.0**

μ と σ はそれぞれ不一致 ("X") と挿入・削除 ("$-$") に対するペナルティーと考えられている．その値を変更すれば，文字列間のアラインメントも変わり，したがって $ss(v,w)$ も変わるため，μ と σ を調整することで，各々の処理目的に合わせた文字列の類似性に対するスコアづけができる．

また，v と w の 2 つの文字列全体に関わるアラインメントは μ と σ が定められた下では，最適となる**グローバルアラインメント** $ga(v,w,\mu,\sigma)$ が存在する．その求め方は数式化されており，つまり $ga(v,w,\mu,\sigma)$ は μ と σ を係数としての v と w の関数になる [9], [10]．図 3(A) の例では，$ga(v,w,\mu{=}1,\sigma{=}1)$ が "ABCD$-$" と "GBCEF" の 2 つの文字列である．$ga(v,w,\mu,\sigma)$ の長さを $lga(v,w,\mu,\sigma)$ とすれば，$lga(v,w,\mu{=}1,\sigma{=}1){=}5$, $ss(v,w){=}{-}1$ となる．

図 3(A) では，$\mu{=}1$, $\sigma{=}1$ とすることで v と w の共通部分列 BC に合わせて揃えられている（**部分列**とは，2 つの文字列中必ずしも連続していないが同じ順序で現れる文字の列のことを指す [9]）．一方，同じスコアづけでも，文字列によっては共通部分列が無視されてしまう結果となることがある．例として，v="ABCD" と w="DEFG" のアラインメントを図 3(B) に示す．ここでのグローバルアラインメントは共通部分列 D の一致は見逃すようになっている．このような場合は，図 3(C) のように $\mu{=}0$, $\sigma{=}0$（または同等のものとして $\mu{=}\infty, \sigma{=}0$）とすることで，**最長共通部分列** [9] D を考慮したグローバルアラインメントを求めることができる．

4.2 言語名のゆれのある言語の検出

上記の類似性スコアの指標を 2 つの言語名の類似性評価に適用するには問題がある．

第一に，例えば，$\mu{=}1$,$\sigma{=}1$ とした場合，v="ABC" に対し，(i) w_1="AB" なら，$ss(v,w_1){=}1$, (ii) w_2="ADEBC" なら，$ss(v,w_2){=}1$. $ss(v,w_1)$ と $ss(v,w_2)$ はともに 1 という結果になるが，これでは "ABC" に対し，"AB" と "ADEBC" のどちらがより類似しているかが判定できない．

第二に，言語名は必ずしも単一文字列とは限らない．例えば，T_Y に第一言語名が "CHINANTECO, LALANA" となる言語が存在している．節点ラベルを L_y で表すなら，L_y={CHINANTECO, LALANA} で，2 つの単一文字列が含まれている．一方，T_S では第一言語名が "Chinantec, Lalana" となる言語が含まれている．節点ラベルを L_s で表すなら，L_s={Chinantec, Lalana}．この 2 つの言語は本来同一言語であるが，下線部分の表記ゆれにより，処理 0 では言語名完全一致の条件を満たさず，検出処理に失敗した．このような単一文字列ではない言語名，つまり節点ラベルが複数の単一文字列を要素とする集合，の間の類似性評価はどうすればよいか，が問題となる．

本研究では，まず単一文字列間の類似度について定義を行い，次に単一文字列間の類似度に基づき，言語名間の類似性を最大に引き出すように，言語名の類似度を計算する．

(1) 単一文字列の類似度

L_1 と L_2 はそれぞれ言語名を表す節点ラベルとする．例として，L_1={CHINANTECO, LALANA}, L_2={Chinantec, Lalana}．v と w をそれぞれ L_1 と L_2 の単一文字列集合の任意の要素であるとするならば，v="CHINANTECO" または v="LALANA", w="Chinantec" または w="Lalana"．v と w の組合せは全部で 4 通りで，表 1 に示す．v と w の類似度 $sd\_ln(v,w)$ を式 (2) のように定義する．

$$sd\_ln(v,w) = n_m / lga(v,w,1,1) \qquad (2)$$

ここで，$lga(v,w,1,1)$ を v と w のグローバルアラインメント $ga(v,w,1,1)$ の長さ，n_m を $ga(v,w,1,1)$ 中の同じ列の 2 つの文字が一致する個数とする．

v と w のすべての組合せについて，類似度 $sd\_ln(v,w)$ を計算した値を表 1 に示している．

(2) 言語名の類似度

言語名の類似度 $sd\_ln(L_1,L_2)$ は，L_1 と L_2 についてのすべての単一文字列ペアの中，類似度が最も大きい単一文字列ペア (v,w) の類似度 $sd\_ln(v,w)$ の和を L_1 と L_2 の単一文字列の含有数の大きい方の値で割った値として定義する．

表 1 において，太字の組合せが類似度が最も大きい単一文字列ペアである．L_1 と L_2 の言語名類似度 $sd\_ln(L_1,L_2)$ は (0.9+1.0)/2=0.95 という結果を

y, s はそれぞれ T_Y, T_S の言語

P_y={AAA}, {BBB}, {CCC}, {DDD}　P_s={kkk}, {aaa}, {jjj}

*:一致, X:不一致, +:挿入, −:削除

line	経路	経路上の節点ラベル				
1	P_y		{AAA}	{BBB}	{CCC}	{DDD}
2	P_s	{kkk}	{aaa}	{jjj}		
3		削除	一致	不一致	挿入	挿入
4		−	*	X	+	+

(A) 経路上の節点ラベルの比較

line	2 経路に含まれるすべての節点ラベルを並び替えた後						
1	{AAA}	{aaa}	{BBB}	{CCC}	{DDD}	{jjj}	{kkk}
2	a	a	b	c	d	e	f
3	P_y={AAA}, {BBB}, {CCC}, {DDD} ⇒"abcd"						
4	P_s={kkk}, {aaa}, {jjj} ⇒"fae"						

(B) 経路の文字列への変換

図 4: 経路の比較

得る.

(3) 言語名類似言語の検索

処理 0 で完全一致経路が見つかり，完全一致言語名をもつ言語は見つからなかった T_Y の言語 y に対し，y と経路が一致する T_S の（複数の）言語 $s_1, s_2, \cdots$ から，言語名の類似度が最大となる言語 s を検索し（そのような言語もまた複数存在するかもしれない），(i) 言語名の類似度が最大となる言語 s が唯一存在する，かつ (ii) その最大となる言語名類似度の値がしきい値 α=0.75 を超える，という 2 つの条件を満たすとき，s と y は同一言語と判定する．この処理を**処理 I** とよぶ.

4.3 系統分類のゆれがある言語の検出

言語の系統分類は言語系統木における経路によって表される．以下では，経路の比較を文字列間の比較に転化させ，文字列の類似度に基づく系統分類の類似度について述べる.

(1) 言語系統分類の比較

T_Y と T_S のそれぞれに含まれる言語 y と言語 s の経路をそれぞれ P_y と P_s とする．経路は言語系統木の根の子から葉である言語の節点の親までの節点ラベルの列として定めている（**3.2** 参照）．P_y={AAA}, {BBB}, {CCC}, {DDD}, P_s={kkk}, {aaa}, {jjj} とした場合，両経路の比較法を図 5 に示す.

図 4(A) の line 1 と line 2 は，両経路 P_y と P_s がラベルが一致の節点 {AAA} と {aaa} に合わせて揃えられていることを示している．T_S を基準にするならば，P_y と P_s のそれぞれの節点ラベル間は line 3 のように，一致，不一致，挿入，削除という対応関係になる．それぞれ $*, X, +, -$ の記号を用いれば，line 4 "$-*X++$" のように，2 つの経路の相違を視覚的に表せる.

さらに，P_y と P_s に含まれる異なる節点ラベルをそれぞれ異なる 1 文字に変換して表せば，P_y と P_s の比較は文字列の比較に転化させることができる.

*:一致

v	A	−	B	C	D	−	−
w	−	G	B	C	−	E	F
			*	*			

図 5: 文字不一致が考慮されていないアラインメント

変換方法としては，例えば，まず図 4(B)line 1 に示すように，すべての節点ラベルを辞書順の 1 列に並べ，次に line 2 のように異なる節点ラベルに順にアルファベット a ～ z を割り当てる．これにより，2 つの節点ラベルは，line 3 と line 4 に示しているように，2 つの文字列となる．ここで，割り当てに使う文字はアルファベットには限定しない．図形文字 [16] なら割り当てが可能なため，多くのパターンの節点ラベルが表現できる.

(2) 経路アラインメントと系統分類の類似度

4.2(1) で述べた単一文字列 v と w の類似度 $sd\_ln(v, w)$ は μ=1, σ=1 を前提条件としている．このようなスコアづけでは図 3(B) のように，"ABC<u>D</u>" と "<u>D</u>EFG" の共通部分列 <u>D</u> が無視され，2 つの文字列の共通する文字列の検出ができなくなる．しかし，この 2 つの文字列が 2 つの経路を表わしているとすれば，両経路に共通する節点ラベルが存在していることを意味し，たとえ短い部分列でも見逃すべきではないと考える．それは，例えば 1 つの言語が一方の言語データでは系統分類が "ABC<u>D</u>" であるのに対し，他方の言語データでは "<u>D</u>EFG" になっている（<u>D</u> が対応する語族が新設され，さらにその下に D → E → F → G と分岐するような分類）可能性もあるためである．そのような場合は，経路を表す文字列のアラインメントは <u>D</u> のような短い文字列も共通部分列として表される必要がある．ゆえに，μ=0, σ=0（最長共通部分列を求めるためのスコアづけ）にすべきである.

しかし，単に μ=0, σ=0 と設定を変更するだけでは不都合が生じる．図 4(A) に示す v="A<u>BC</u>D" と

w="GBCEF" を例に，μ=0, σ=0（または同等のものとして μ=∞, σ=0）とした場合のアラインメントを図5に示す．文字の不一致（置換）がまったく考慮されないアラインメントになっている．図3(A)に比べると，経路間の相違比較としては，図3(A)の方がより適切であろう．また，図3(A)と同様，図4(A)でも P_y 上の {BBB} と P_s 上の {jjj} は不一致と判定している．つまり，経路を表す文字列間のアラインメントは単なる2つの単一文字列間のそれとは異なる方法で求めるべきで，最長共通部分列を求めるためのスコアづけをする必要がある一方，不一致も考慮しなくてはならない．

そこで，P_y と P_s が与えられたとき，(i) μ=0, σ=0 とし，経路を表す文字列 v と w のアラインメントを求める．例として，図5に示しているように，v="ABCD", w="GBCEF" のとき，v と w のアラインメントは ("A−BCD−−", "−GBC−EF") になる．(ii) そのアラインメントの2つの文字列を $*$, X, $+$, $-$ から構成される1つの文字列 ("+− ∗ ∗ +− −") に変換する．(iii) (ii) で得られた文字列に対し，下線部分の "+−" または "−+" を "X" に置き換え，文字列の再構成を行い，新たな文字列 "$X**X-$" を求める．

再構成後の文字列 "$X**X-$" を P_y と P_s の**経路アラインメント**とよび，$pa(v,w)$ で表す．また，$pa(v,w)$ の長さを $lpa(v,w)$，$pa(v,w)$ に含まれる記号 $*$ の個数 ("$X**X-$" では2個) を n_m で表す．言語 y と言語 s の**系統分類の類似度**は，式(3)を満たす $sd\_lc(P_y,P_s)$ とする．

$$sd\_lc(P_y,P_s) = n_m/lpa(v,w) \qquad (3)$$

(3) 系統分類類似言語の検索

処理0で完全一致経路が見つからず処理が終了した T_Y の言語 y に対し，T_S のすべての言語から (i) y と言語名の類似度がしきい値 α=0.75 を超える言語を検索し（そのような言語は複数存在するかもしれない），(ii) その（複数の）言語から y と系統分類の類似度が最大で，かつしきい値 β=0 を超える言語 s が唯一存在するとき，s と y は同一言語と判定する．この処理を**処理 II** とよぶ．

4.4 処理全体の流れ

(1) 完全一致言語の検索（処理0）

T_Y のすべての言語に対し，処理0を行う．T_Y の言語を，T_S から (i) 完全一致経路が見つかり，かつ言語名が完全一致となる言語も見つかった場合は V_s，(ii) 完全一致経路は見つかったが，その完全一致経路にぶら下がっている葉から完全一致言語は見つからなかった場合は V_{f1}，(iii) 完全一致経路が見つからず，処理がそこで終了する場合は V_{f2}，という3つの結果セット（T_Y の言語から構成される言語集合）にそれぞれ出力する．

V_s に含まれる任意の言語 y に対し，T_Y から y の節点と y の親までの枝を削除する．y の親節点が兄弟節点をもたない場合は，その親節点も削除する．このように順に言語系統木の根に向かって，y の経路上の節点が兄弟節点をもたない場合はその部分経路を削除し，T_Y を更新する．この処理は V_s に含まれるすべての言語について行う．また，T_S に含まれている同一言語に関しても，同様に行う．含めて，**同定言語削除処理**とよぶ．

(2) 言語名類似言語の検索（処理 I）

V_{f1} に含まれるすべての言語に対し，処理 I を行う．V_{f1} に含まれる言語を，T_S から (i) 言語名類似言語が見つかった場合は V_{f1_s}，(ii) 言語名類似言語が見つからなかった場合は V_{f1_f}，という2つの結果セットにそれぞれ出力する．

V_{f1_s} に関し，同定言語削除処理を行う．

(3) 言語系統分類類似言語の検索（処理 II）

V_{f2} に含まれるすべての言語に対し，処理 II を行う．$V_{f2}(T_Y)$ に含まれる言語を，T_S から (i) 言語系統分類類似言語が見つかった場合は V_{f2_s}，(ii) 言語系統分類類似言語が見つからなかった場合は V_{f2_f}，という2つの結果セットにそれぞれ出力する．

以上の処理で得られた V_s，V_{f1_s}，V_{f2_s} に含まれる言語は言語同一性が判明した言語となる．

表 2: 処理結果

処理	同定できた言語数	比率
処理 0	1,034	36%
処理 I	80	3%
処理 II	1,334	46%
合計	2,448	85%

5 処理結果および考察

4.4 のように処理し，得られた V_s，V_{f1_s}，V_{f2_s} の結果を表2に示す．表2から分かるように，(i) 同定言語の内訳は，完全一致言語が 1,034，系統分類一致で言語名類似の言語が 80，言語名一致で系統分類類似の言語が 1,334 である．(ii) 処理0で判定できた比率がわずか 36% にとどまったのに対し，処理 I と処理 II は合わせて 49%，に達した．そして，3つの処理を合わせると 85% の言語が同定できた．

なお，処理 I および処理 II で用いたしきい値は α=0.75 と β=0 とした．その理由は次の通りである．T_Y と T_S の一部のデータを対象に，α=0.65, 0.70, 0.75, 0.80, 0.85, β=0.10, 0.15, 0.20, 0.25, 0.30 のときのシミュレーションをした結果，(i) α=0.75 のとき，異なる言語を同一言語と判定した例とその逆で同一言語の検出を漏れた例が最も少なかった（それぞれ 3 つと 4 つ）．(ii) β が 0.10 以上になると，(i) の後者の例があった．また，表 2 には載せていないが，言語名一致で系統分類の類似度が 0 となる言語数は 63 もあった．したがって，β=0 のしきい値設定は効果があることが分かる．

言語名と系統分類がともに類似する言語については，今回は検出処理を行っていない．そのため，例えば言語名がそれぞれ "YI, GUICHOU" と "Yi, Guizhou"，言語名の類似度が 0.93 で，経路アラインメントが "$***X*-$"，系統分類の類似度が 0.67 となる言語は同定言語として検出されていない．言語名と系統分類の類似度をともに考慮すれば，同定できる言語の比率のさらなる向上が予想される．これについてはまた稿を改めて議論することにしたい．

6 おわりに

本研究では，我々は言語名と系統分類の類似度を導入し，木構造をなす言語系統木に加え，類似度を考慮した言語の同一判定の手法を提案した．その結果，合わせて 85% の言語の同一性が判定できた．そのうち，49% は言語名と系統分類の類似度の適用による結果であった．このことから，我々が提案した類似度は有用で，ゆれのある言語の検出手法は効果的である，といえる．今後は，(1) 言語名と系統分類がともに類似する言語の検出方法を検討し，(2) 今回の処理で同一性が判明できなかった言語について調査し，ゆれのある言語の検出率の向上を図るなど，本手法をさらに発展させていきたい．

注と参考文献

[1] 呉翰, 乾秀行, 杉井学, 松野浩嗣: "言語研究のための GIS データの生成について－ Ethnologue GIS データを言語特徴の地図化に用いる一手法", 人文科学とコンピュータシンポジュウム論文集, pp.253-258, 2007.

[2] Ren Wu, Hideyuki Inui, Manabu Sugii and Hiroshi Matsuno: "Language Identification for Generating GIS Data Used in Mapping Linguistic Features of the World's Languages", Proceedings of ITC-CSCC2008, pp.153-156, 2008.

[3] 山本秀樹，世界諸言語の地理的・系統的語順分布とその変遷，渓水社，広島，2003.

[4] 文献 [3] の「言語別語順データ」には 2,932 言語についての属性データが掲載されているが，下位の方言を言語として編入しているところがあるため（言語と方言の定義が元々曖昧である），実質言語数は 2,870 となっている．

[5] http://www.ethnologue.com/web.asp

[6] 呉翰，乾秀行，杉井学，松野浩嗣，"Ethnologue15th 言語属性データと言語系統データの生成および言語同定における利用，" コンピュータ&エデュケーション, vol.25, pp.70-73, 2008.

[7] Gonzalo Navarro, "A guided tour to approximate string matching," ACM Computing Surveys (CSUR), vol.33, no.1, pp.31-88, 2001.

[8] P. H. Sellers, "The theory and computation of evolutionary distances: Pattern recognition," Journal of Algorithms, vol.1, no.4, pp.359-373, 1980.

[9] Neil C. Jones, Pavel A. Pevzner 著，渋谷哲朗ほか訳，バイオインフォマティクスのためのアルゴリズム入門，共立出版，東京，2007.

[10] 富田勝監修，斎藤輪太郎著，バイオインフォマティクスの基礎：ゲノム解析プログラミングを中心に，数理科学別冊 SGC ライブラリ 41，サイエンス社，東京，2005.

[11] Gordon, R.G. (ed.), "Ethnologue: Languages of the World, 15th ed.," Dallas, SIL International, Texas, 2005.

[12] http://www.sil.org/iso639-3/default.asp

[13] http://wals.info/index

[14] R.E. Asher and C.J. Moseley, "Atlas of the world's languages," Routledge, New York, 2007.

[15] 呉翰，冨永理恵，乾秀行, 杉井学, 松野浩嗣，" オープンソース可視化ツールを用いた言語系統樹の図式表現，" 人文科学とコンピュータシンポジュウム論文集, pp.333-340, 2008.

[16] 図形文字は印字可能な文字のことで，文字コード体系によってその範囲は変わる．ASCII コードでは 0x21～0x71 であり，8 ビット拡張 ASCII コードでは 0x21～0x71 に加え，0xA0～0xFF が追加されている．