情報処理学会論文誌　数理モデル化と応用　Vol. 3　No. 3　24–35 (Oct. 2010)

木構造と文字列類似度に基づく言語の同一性判定

呉　　勒[†1]　　松野浩嗣[†1]

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．両データに含まれる言語の同一性を判定する際には，言語の別名の存在や表記ゆれなどに対応する必要がある．本研究は Yamamoto-Data と SilGIS-Data に含まれる同一言語を見つけ出すことを目的とし，言語名に加えて言語系統分類も取り入れた手法を提案する．本論文では，まず，系統分類がなす木構造に基づき，言語系統木を定義する．次に，言語名と言語系統分類の曖昧な性質に対し，文字列類似度に基づく言語名と系統分類の類似度を導入し，これらの類似度を用いた言語同一性判定の手法を述べる．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．

Identifying Same Languages Based on Tree Structure and String Similarity

Ren Wu[†1] and Hiroshi Matsuno[†1]

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 identical languages are expressed by different writings in Yamamoto-Data and SilGIS-Data. Therefore, it is important to identify if two writings express the same language. In order to cope with this problem, we introduce a new method to absorb these ambiguities by applying string alignment technique. Our experimental result for the two language data shows that our proposed method is usefull and effective.

†1 山口大学大学院理工学研究科
Graduate School of Science and Engineering, Yamaguchi University

1. はじめに

言語類型論的研究は世界諸言語の通言語的な特徴を考察することにより，言語の普遍性ならびに多様性を見つけ出すことを目的とする．近年，GIS（地理情報システム）の持つ情報の多角的な分析機能が注目され，有効かつ強力な手段として，新展開をもたらすことが期待されている[1),2)]．GIS を用いた言語類型論的研究を行う際，世界諸言語の言語特徴を集めた属性データと言語の話されている地域の地理情報などを提供する空間データが必要であるが，世界諸言語に関する GIS データは，一般的に利用可能なものは少なく，入手困難な状況にある．そこで，我々は，Yamamoto-Data と SilGIS-Data（詳細は後述する）という 2 つの言語属性データを処理することによって，言語研究に必要な GIS データを生成する手法を提案した[3)]．その手法では，Yamamoto-Data と SilGIS-Data のそれぞれに含まれている言語の同一性判定，すなわち両データに含まれている言語の対応づけが必要である．このような研究はいままでなされていない．

SilGIS-Data では言語の一意的識別子である言語コードがあるが，Yamamoto-Data では言語の識別に言語の名前が使われている[3)–5)]．両データに含まれている多くの言語は言語の名前が一致していて，一見して言語の名前の一致によるマッチング処理が可能なように見えるが，実際には，一方のデータに同じ名前の言語が複数含まれていることや，両方のデータに異なる名前を持つ同一言語が存在していることなど，言語名が一致するか否かのみによって言語の同一性を結論づけることができない．よって，言語の名前のほかに，さらなる判定尺度が必要となる．

そこで，我々は，この問題を解決するため，世界諸言語が系統的に分類されていて，系統分類情報が木構造をなしていることに注目し，言語名に加えて系統分類の情報も言語同定に取り入れることを着想した．

言語名も系統分類も文字列で表されており，いずれも曖昧な性質を持つ情報である．文字列の類似性の評価には編集距離[7)–10)] による手法がよく用いられている[11)–16)]．それに基づき，我々は言語名の類似度と系統分類の類似度という概念を導入し，それらの類似度に基づく判定ルールを定め，Yamamoto-Data と SilGIS-Data に含まれている同一言語を見つけ出す手法を提案する．

以下，2 章では Yamamoto-Data と SilGIS-Data の言語同一性判定の必要性について述べる．3 章では系統分類の角度からの言語データ構造である言語系統木について定義を行い，完全一致言語の検出法について述べる．4 章では言語名の類似度と系統分類の類似度に

ついて定義を行い，Yamamoto-Data の言語に対し，SilGIS-Data から同一言語を見つけ出す手法について述べる．5 章では閾値設定に関する実験結果と同一言語の判定結果について述べ，本研究で提案した手法の妥当性および有用性などを考察する．最後の 6 章で本論文をまとめる．

2. 言語同一性判定の必要性

2.1 2 つの言語データの概要

表 1 (A) は文献 5) に掲載されている「言語別語順データ」を指し，2,932 言語の語順に関する言語特徴がまとめられている．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 である．一方，表 1 (B) は *Ethnologue* 第 15 版 Web サイト[4),17)] から世界諸言語の属性情報を取得し，表形式にしたデータを指し，言語数は 7,299 である．表 1 の (A) と (B) をそれぞれ Yamamoto-Data と SilGIS-Data で表す．

表 1 (A) と (B) のいずれも，各行のレコードは 1 言語を表す．表 1 (A) の 3 つのフィールドは表 1 (B) にもあり，共通項目となっている．**第一言語名**[4)] は言語の名前の 1 つで，**属性**は複数フィールドを含む場合があり，語順や話者人口や言語使用状況などの言語に関する属性情報である．*No* は各々のデータのレコード番号である．なお，アルファベット表記は特に大文字と小文字を区別しない．

一方，**言語コード**[4)] と**（複数の）別名**[4)] は表 1 (B) にのみ用いられている．言語コードは国際標準化機構によって定められた ISO639_3 言語コード[4)] で，アルファベット 3 文字から構成され（たとえば，日本語は jpn），言語の一意的識別子となる．一方，（複数の）別名は複数の**別名**[4)] を合成した文字列である．

表 1　Yamamoto-Data と SilGIS-Data
Table 1　Yamamoto-Data and SilGIS-Data.

(A) Yamamoto-Data

No.	第一言語名	属性
212	BAI	…
213	BAI	…
485	CHINANTECO, LALANA	…
1015	JAPANESE	…
1855	NHARON	…
1959	OTOMI, STATE OF MEXICO	…

(B) SilGIS-Data

No.	第一言語名	言語コード	（複数の）別名	属性
733	Bai	bdj	Bari	…
1565	Chinantec, Lalana	cnl	Chinanteco de San Juan Lalana	…
3295	Japanese	jpn		…
5763	Naro	nkr	Nharo, Nharon, Nhauru, …	…
6262	Otomi, Estado de Mexico	ots	Hnatho, Otomi del Estado de Mexico, …, State of Mexico Otomi	…

1 つの言語に複数の名前が付けられていること（たとえば，「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）がよくある．*Ethnologue* 第 15 版[4),17)] では，その研究・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．複数の名前の中の 1 つは第一言語名，その他は別名とされている．以降では，**言語名**は第一言語名または別名を指す．

第一言語名と別名の指定は学者独自に行われている．例として，表 1 (B) *No*=5763 の言語は Naro が第一言語名で，Nharo, Nharon, Nhauru, … などの複数の別名がカンマ（,）でつないで列挙されている．それに対し，(A) *No*=1855 では NHARON が第一言語名となっており，別名に関する情報は書かれていない．この 2 つの言語は第一言語名が異なっているが，実際は同一言語である．

このように，異なる学者によって編成された言語データでは，同じ言語が違う言語名（第一言語名）になっているケースがよくある．我々は Yamamoto-Data と SilGIS-Data のそれぞれに含まれる言語の対応づけが必要であるが，両データのいずれも言語数が千単位にのぼるため，自動処理によって，なるべく多くの同一言語を発見することが狙いである．

次節において，この処理における問題点を述べる．

2.2 言語名による言語の同一性判定の問題点

第一言語名には，(i) アルファベットからなる Japanese のような文字列（**語**とよぶ），(ii) 2 つ以上の語をカンマ（,），空白（Space）またはハイフン（-）（**区切り記号**とよぶ）でつないだ ‘‘Otomi, Estado de Mexico’’ のような語のリスト，という 2 つのケースがある．

（複数の）別名は複数の別名をカンマでつないだ文字列になっている．表 1 (B) *No*=6262 の（複数の）別名 ‘‘Hnatho, Otomi del Estado de Mexico, …, State of Mexico Otomi’’ は，それぞれ下線部分の別名をカンマでつないでいる．したがって，カンマを検出すれば，複数の別名に分割が可能である．

以下では，表 1 のサンプルデータを例に，言語名による言語の同一性判定の問題点について説明する．

(a) 第一言語名による判定

(A) *No*=1015 と (B) *No*=3295 は，第一言語名がそれぞれ JAPANESE と Japanese で，一致しているため，同一言語と判定できる．

(b) 第一言語名と別名による判定

(A) *No*=1855 と (B) *No*=5763 は，第一言語名がそれぞれ NHARON と Naro で，上記 (a) の方法では判定できないが，(B)（複数の）別名 ‘‘Nharo, Nharon, Nhauru, …’’ の下

線部分の語との一致が認められるため，同一言語と判定できそうである．

(c) 第一言語名と別名による判定その 2

(A) *No*=1959 と (B) *No*=6262 は上記 (a) と (b) の方法では同一性判定ができないが，(A) 第一言語名 ‘‘OTOMI, STATE OF MEXICO’’ と (B)（複数の）別名 ‘‘Hnatho, Otomi del Estado de Mexico, ···, State of Mexico Otomi’’ を比較すると，下線部分の語リストが何らかの方法で一致が認められそうなため，こちらも同一言語と判定できそうである．

上記 (a)〜(c) のケースは少なからず同一言語の判定ができそうである．つまり，第一言語名または別名は言語の同一性を判定する重要な情報であることがいえる．しかし，それだけでは情報不足で，**判定不能**となってしまうケースもある．これを次の (d) と (e) に示す．

(d) 言語名の重複出現

(A) *No*=212 と *No*=213 はともに第一言語名が BAI で，(B) *No*=733 も第一言語名が Bai である．(A) では同一性判定の情報として第一言語名しか含まれていないため，(B) *No*=733 が同じ言語名を持つ (A) の *No*=212 と *No*=213 のどちらに対応しているかが，判定不能になってしまう．あるいはどちらにも対応していないことも否定できない．

(e) 言語名の類似

(A) *No*=485 と (B) *No*=1565 は，第一言語名がそれぞれ ‘‘CHINANTECO, LALANA’’ と ‘‘Chinantec, Lalana’’ で，下線部分の CO と c は直感的に表記上の違いなどによる変化で，本来は同じ言語名なのではないか，との推測がつくと思われるが，第一言語名または別名が一致するかどうかによっては判定できない．

一般的に，言語名には表記ゆれがあることが少なくない．表記ゆれは，たとえば「バイオリン」と「ヴァイオリン」や，「サーバー」と「サーバ」など，多くのケースがあり，外来語の日本語表記で特に多く現れる．世界諸言語データの言語名はデータ編成者にとってはいわばそのような「外来語」ならぬ外国語ばかりであるため，表記ゆれが含まれている可能性は大きい．

また，(d) のケースは第一言語名または別名以外のさらなる情報がなければ判定不能であるが，(a)〜(c)，または (e) のケースについても，2 つの言語が同一である可能性は高いが，これを別の角度からも示すことができれば，その同一性判定はより正確なものになる．

2.3　言語同一性判定処理の意義

表 1 (B) の SilGIS-Data にのみ含まれている言語コードというフィールドがある．言語コードは言語の一意的識別子で，言語の同一性が問題となるのは，言語を識別するのにこの言語コードが付けられていないデータを扱う場合である．

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または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り，文献 18) がその一例である．しかし，比較的に最近発表された言語資料でも言語コードが必ず付与されているわけではない，というのが現実である．Routledge 社の *Atlas of the world's languages* [19] という文献には世界諸言語の地理的分布図が掲載されており，類型論的研究分野では価値の高い資料といわれている．2007 年に第 2 版に改訂されたばかりであるが，言語コードは付与されていない．我々は，この資料にある世界諸言語が話されている地域に関する地理情報を語順研究に用いることを試みたが，再び言語同一性問題の壁にぶつかった．

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，重要な意義を持つ．そして，本研究が狙いとする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といえよう．

3.　言語系統木を用いた完全一致言語の検出

3.1　系統分類を考慮した言語の同一性判定

言語間の系統的関係を探求しようと，サンスクリット語を中心に印欧語の同系性を確認する形で研究が進められ，言語の変化・変遷を表す系統樹モデルが構想された．系統樹モデルは，ヨーロッパの言語をサンスクリット語と関連付け，さらにそれらの源となる祖語という言語を再構し，その祖語を 1 本の木の幹とし，印欧語はそこから枝分かれした言語と位置付けた．系統樹モデルはその後，印欧語以外の言語の系統的関係の研究にも援用されるようになった[20),21)]．

現在用いられている言語系統分類に関するデータの構造は**図 1** (A) に示すようになってい

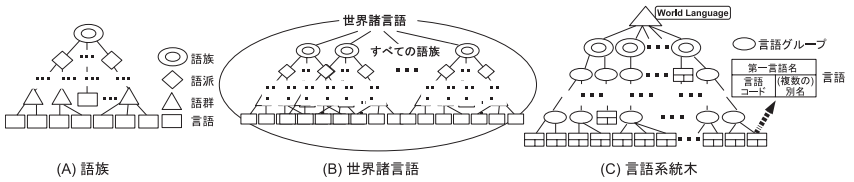

図 1　言語系統分類と言語系統木
Fig. 1　Language classification and world language tree.

る．同系の言語は 1 つの語族を構成する．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あり，最下位にあるのが言語である．SilGIS-Data のデータソースの文献では，世界諸言語を 108 語族に分類している．そして，Yamamoto-Data のデータソースの文献においても，扱われているすべての言語の系統分類（117 語族におよぶ）が掲載されている．このように，世界諸言語は語族系統樹の森である．

言語の系統分類が木構造をなしていることから，言語名に加え，言語系統分類も考慮すれば，2.2 節 (d) のようなケースは，同じ名前を持つ異なる言語の系統分類も同じである場合（そのようなことが発生する可能性は少ない．本論文で扱う両データでは，そのようなことはない）を除き判定可能になる，と考えられる．

また，2.2 節 (e) のような言語名が類似しているケース（系統分類は一致）は，言語名の類似度を導入すれば，同一性判定が可能になる．

一方，言語の系統分類は学者によって異なることがある．ゆえに，本来同じ言語であっても，各々の言語データでの系統分類は必ずしも一致しない．このような異なる学者の異なる知見による相違のほかに，系統分類の表現には言語名（ここでは，語族名なども含む広義の言語名を指す）表記がともなうため，表記ゆれなどによる相違も存在すると思われる．

このようなことをふまえて，我々は言語名の類似度に加えて，言語の系統分類の類似度についても定量化を行う．そして，言語の系統分類は言語名に次ぐ有益な情報として利用する．つまり，言語名の一致または類似を確認したうえで，さらに系統分類も一致または類似しているならば，言語の同一性を肯定する，という 2 つの角度から評価を行う．これによって，同定できる言語の数および正確性を向上させることを狙う．

3.2　言語系統木

本研究では，図 1 (B) に示すように，語族の森を「世界諸言語」というルートの下にまとめ，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) 言語名（語族名，言語グループ名を含む）は木構造のノードのラベルとして，また（複数の）別名と言語コードは最下位にあるリーフノードの言語の属性情報として，それぞれ持たせることにする．その構造を図 1 (C) に示す．以下，木と図 1 (C) の構造を定義する．

[定義 1] T をラベル付き順序木とする．T のルートを $r(T)$，T のノード集合を $V(T)$，辺集合を $E(T)$ で表す．ノード $x{\in}V(T)$ のラベルを $L(x)$ で表す．

(1) $x, z{\in}V(T)$，$(x,z){\in}E(T)$ ならば，x (z) は z (x) の**親**（**子**）という．同じ親を持つノードを**兄弟**とよび，子を持たないノードを**リーフ**とよぶ．T のリーフノード集合を $V_{leaf}(T)$ で表す．

(2) $x_0{=}r(T)$ から x までのパス $x_0x_1\cdots x_{k-1}x_k = x$ を $p(x_0,x)$ で表し，k を x の**レベル**とよぶ．特に，$p(x_0,x)$ の部分パス $x_1x_2\cdots x_{k-1}$ を $p(x)$ で表す．

(3) リーフでない兄弟 x，z に対し，$L(x){\preceq}L(z)$ なら，x を z の左に位置する．　□

定義 1 (3) の $\preceq$ は，$L(x)$ と $L(z)$ の順序関係を示しており，その定義はラベル関数が具体的に与えられたときに，示すことができる．本研究で用いる順序 $\preceq$ は，次の定義 2 で与えている．

[定義 2] 次の条件を満たす T を**言語系統木**とよぶ．

(1) ノード $x{\in}V_{leaf}(T)$ の**ノードラベル** $L(x)$ は $L(x){=}(\mathcal{L}_x, A_x, C_x)$ で表す．ただし，(i) $\mathcal{L}_x$ は集合 $\mathcal{L}_x{=}\{w_1, w_2, \cdots\}$（$w_i{\in}\mathcal{L}_x$ は語）で，第一言語名を表す．(ii) C_x はアルファベット 3 文字からなる文字列で，言語コードを表す．(iii) A_x は集合 $A_x{=}\{\mathcal{A}_1^x, \mathcal{A}_2^x, \cdots\}$ で，（複数の）別名を表す．ここで，$\mathcal{A}_i^x = \{w_1, w_2, \cdots\}$（$w_j{\in}\mathcal{A}_i^x$ は語）は別名を表す．

(2) ノード $x{\notin}V_{leaf}(T)$ の**ノードラベル**は $L(x) = \mathcal{L}_x = \{w_1, w_2, \cdots\}$（$w_i{\in}\mathcal{L}_x$ は (1) (i) と同様）で，言語グループ名を表す．また，ルート $r(T)$ のラベルは $\mathcal{L}_{r(T)}{=}\{\text{World, Language}\}$ である．

(3) ノードラベルで表した $p(x){=}x_1x_2\cdots x_{k-1}$ に対応するパスを $\mathcal{P}(x){=}\mathcal{L}_{x_1}\mathcal{L}_{x_2}\cdots\mathcal{L}_{x_{k-1}}$ で表す．

(4) リーフでない兄弟 x, z のラベルを $\mathcal{L}_x{=}\{w_1^x, w_2^x, \cdots\}$（$w_1^x{\le}w_2^x{\le}\cdots$），$\mathcal{L}_z{=}\{w_1^z, w_2^z, \cdots\}$（$w_1^z{\le}w_2^z{\le}\cdots$）とする．(i) $w_1^x{=}w_1^z$, $w_2^x{=}w_2^z, \cdots$, $w_{i-1}^x{=}w_{i-1}^z$, $w_i^x{<}w_i^z$ となるような i（$\ge$1）が存在する，または (ii) $l{=}|\{w_1^x, w_2^x, \cdots\}|{<}|\{w_1^z, w_2^z, \cdots\}|$ としたときに，$w_1^x{=}w_1^z$, $w_2^x{=}w_2^z, \cdots, w_{l-1}^x{=}w_{l-1}^z, w_l^x{=}w_l^z$ が成立するならば，$L(x){\prec}L(z)$ である．$L(x){\succ}L(z)$ が成立しないとき，$L(x){\preceq}L(z)$ と書く．ここで，$w_i^x{<}w_i^z$ は w_i^x と w_i^z の辞書式順序を表す．　□

なお，本研究の言語系統木を構成する言語データでは，$L(x){=}L(z)$ となるような 2 つの異なる兄弟 x と z は存在しないことを付け加えておく．

3.3　2 つの言語系統木 T_Y と T_S

Yamamoto-Data と SilGIS-Data に関連する言語系統木は次の 2 つである．

(i) Yamamoto-Data のデータソースの文献にある「系統別語順分布表」は Yamamoto-Data の言語を系統分類の観点から整理した語順データである．このデータを言語系統木の定義に従って XML 形式に変換したデータを T_Y とする．言語数は 2,870 で，117 語族で構成されている．

(ii) SilGIS-Data のデータソースである *Ethnologue* 第 15 版 Web サイトには世界諸言語

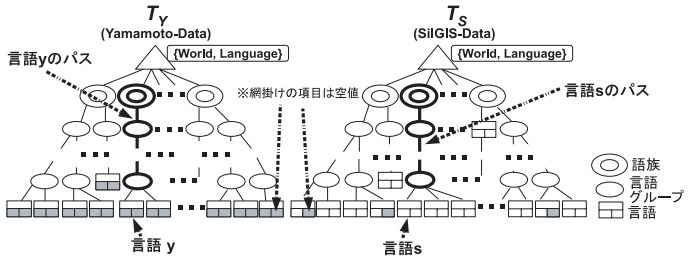

図 2　2 つの言語系統木 T_Y と T_S
Fig. 2　Two world language trees T_Y and T_S.

の系統分類の情報も掲載されている．それを取得し，言語系統木の定義に従って XML 形式に変換したデータを T_S とする．言語数は 7,299 で，108 語族で構成されている．

T_Y と T_S を**図 2** に示す．なお，*Ethnologue* 第 15 版 Web サイトでは，言語名表記に Unicode 文字でないと表現できない文字が使われているところがある．T_Y での言語名表記は ASCII コード体系に従っているため，T_S も ASCII コードに変換した．また，Yamamoto-Data と SilGIS-Data では，言語名（言語グループ名などを含む）にアルファベットと区切り記号以外の'|' や'=' などの記号も使われている．T_Y と T_S では，これらの記号は削除した．この 2 つの言語系統木の生成処理についての詳細は文献 22) を参照されたい．

定義 2 で述べたように，リーフノード $x{\in}(V_{leaf}(T_Y){\cup}V_{leaf}(T_S))$ には言語コード C_x と（複数の）別名 A_x の属性が付与されている．ただし，(i) T_Y のどのリーフノード $y{\in}V_{leaf}(T_Y)$ においても，C_y=Null，A_y=Null（Null は空値を表す．以降も同様）．(ii) T_S のリーフノード $s{\in}V_{leaf}(T_S)$ については $C_s{\neq}$Null，$A_s{\neq}$Null または A_s=Null．つまり，T_S ではリーフノードの属性として，言語コードは必ず存在するが，（複数の）別名は存在しない場合もある．

このように，T_S には言語を一意に識別できる言語コードが付与されており，いわば基準となる言語系統木である．T_Y は何らかの処理で T_S の言語との対応関係を明らかにする必要がある言語系統木であると位置付けられる．

本研究では，2 つの言語系統木 T_Y と T_S を対象に，言語 $y{\in}V_{leaf}(T_Y)$ に対し，T_S から y の同一言語である言語 $s{\in}V_{leaf}(T_S)$ を見つけ出すことを目的とする．

3.4　言語系統木を用いた完全一致言語の検出

T_Y と T_S 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．言語系統木 T の言語はリーフノードにあたり，系統分類はパス，言語名はノードラベルなどで表せる．

［定義 3］ T_Y と T_S は 2 つの異なる言語系統木であり，y と s はそれぞれ T_Y と T_S のリーフ（$y{\in}V_{leaf}(T_Y)$，$s{\in}V_{leaf}(T_S)$）である．

(1)　T_Y のパス $\mathcal{P}(y){=}\mathcal{L}_{y_1}\mathcal{L}_{y_2}\cdots\mathcal{L}_{y_{k-1}}$ と T_S のパス $\mathcal{P}(s){=}\mathcal{L}_{s_1}\mathcal{L}_{s_2}\cdots\mathcal{L}_{s_{m-1}}$ について，(i) $k{=}m$，(ii) $\mathcal{L}_{x_i}{=}\mathcal{L}_{y_i}$（$i{=}1,2,\cdots,$ k）が成立つとき y と s は**系統分類一致**といい，$\mathcal{P}(y){=}\mathcal{P}(s)$ で表す．

(2)　$\mathcal{L}_y{=}\mathcal{L}_s$ または $\mathcal{L}_y{\in}A_s$ が成立つとき，すなわち y と s のノードラベルが一致するか，または y のノードラベルが s の（複数の）別名に含まれるとき，y と s は**言語名一致**という．

(3)　y と s が系統分類一致で，かつ言語名一致であるならば，y と s を同一言語と判定し，(y,s) を**完全一致言語**とよぶ．　□

$y{\in}V_{leaf}(T_Y)$ に対し，$\mathcal{P}(y)$ と系統分類一致の $\mathcal{P}(s)$ を持つ $s{\in}V_{leaf}(T_S)$ は複数存在しうる．そのため，完全一致言語の検出は (i) T_S において $\mathcal{P}(y){=}\mathcal{P}(s)$ を満たすパス $\mathcal{P}(s)$ を検索し，(ii) (i) で得られた $\mathcal{P}(s)$ を持つ複数のリーフノードの中から，$\mathcal{L}_y{=}\mathcal{L}_s$ または $\mathcal{L}_y{\in}A_s$ を満たす s を見つければよい．T_Y と T_S が順序木（定義 1 (3)，定義 2 (4)）であるため，(i) の処理を効率良く行うことが可能である．

この完全一致言語の検出処理では，T_Y と T_S のそれぞれにある言語が言語名または系統分類が一致ではなく，類似しているにすぎない場合は，検出されない．次章ではこ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

本章では，言語名または言語系統分類のゆれの性質にかんがみ，言語名の類似度と系統分類の類似度の概念を導入し，それぞれについて定量化を行う．

言語名と系統分類の比較はいずれも文字列の比較を基本とするため，以下では，まず編集距離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し，次に文字列類似度に基づく言語名の類似度と系統分類の類似度の計算方法について述べる．

4.1　文字列の類似性評価

2 つの文字列の類似性または非類似性（距離）を計る尺度として，編集距離（Edit distance）がよく用いられる[7)–10)]．ここでは，w_1=ABCD，長さ n_1=4 と w_2=CBCEF，長さ n_2=5 という 2 つの文字列を例にとり，説明していく．

(1) 編集距離

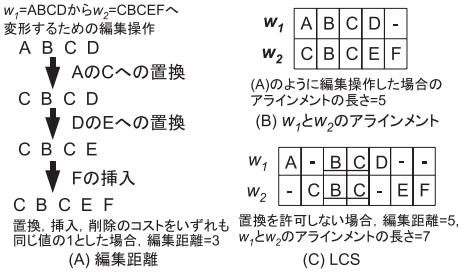

図 3　編集距離と LCS
Fig. 3　Edit distance and LCS.

w_1 と w_2 の編集距離を，1 文字の挿入，1 文字の削除，または 1 文字を別の 1 文字に置換という 3 つの編集操作のコストを定めて，これらの編集操作によって文字列 w_1 から文字列 w_2 に変形するための編集操作のコストの合計の最小値として定義し，これを $ed(w_1, w_2)$ で表す．この例では，挿入，削除，置換の 3 つの編集操作のコストをいずれも同じ 1 とした場合，**図 3** (A) に示すように，編集距離 $ed(w_1, w_2)$ は 3 である．

編集距離の計算は，動的計画法に基づいている．編集距離は編集操作のコストを定めた前提での，編集グラフとよばれるグリッドを通る最短距離となる[7),8)]．

一方，w_1 と w_2 を揃えた様子を図 3 (B) に示す．(ABCD−, CBCEF) のような 2 行の文字列を w_1 と w_2 の**アラインメント**という．アラインメントは編集グラフを走査することで得られる．走査ルートは複数通り可能なため，w_1 と w_2 のアラインメントは複数通り存在することがある．それらの長さ（以降，$l_A(w_1, w_2)$ で表す）はいずれも同じである．

アラインメントの 2 行の文字列について，同じ列の 2 つの文字が一致する個数 $ss(w_1, w_2)$ はアラインメントの長さ $l_A(w_1, w_2)$ から編集距離 $ed(w_1, w_2)$ を引いた値になる．つまり，$ss(w_1, w_2)$ は次の式を満たす．

$$ss(w_1, w_2) = l_A(w_1, w_2) - ed(w_1, w_2) \tag{1}$$

(2) LCS

編集距離を求めるには編集操作のコストを定めることが前提となっているが，置換を排除し，挿入と削除の 2 つの編集操作のみを許し，それらのコストをいずれも 1 とするならば，w_1 と w_2 のアラインメントは図 3 (C) のようになり，編集距離 $ed(w_1, w_2)$ は 5 になる．

このような置換を考慮しない編集操作の場合は，w_1 と w_2 の 2 つの文字列に含まれる最長共通部分列 **LCS**（The Longest Common Subsequence）[7),8)] を求めることができる．ここでは下線部分の BC である．部分列は連続している必要はない．w_1 と w_2 の LCS の長さを $l_{LCS}(w_1, w_2)$ で表すならば，この例では $l_{LCS}(w_1, w_2)$=2 となる．ここで $l_{LCS}(w_1, w_2) = ss(w_1, w_2)$ であることに注意されたい．

4.2　言語名の類似度

言語名は 1 つ以上の語からなる集合として定義されている．例として，$\mathcal{L}_1^{T_Y}$={CHINANTECO, LALANA}，$\mathcal{L}_1^{T_S}$={Chinantec, Lalana} を考える．言語名の類似度の計算は次の 2 つのステップに分けて考える．(1) $\mathcal{L}_1^{T_Y}$ に含まれる語と $\mathcal{L}_1^{T_S}$ に含まれる語との間の類似度を計算し，(2) (1) で計算された語類似度に基づき，言語名の類似度を計算する．

(1) 語類似度

4.1 節で説明した編集距離は，図 3 (A) で示したように，置換，挿入，削除という 3 つの操作のコストをいずれも 1 にしたものである．

言語名に含まれる表記ゆれには (i) $\mathcal{L}_1^{T_Y}$={CHINANTECO, LALANA} と $\mathcal{L}_1^{T_S}$={Chinantec, Lalana}，(ii) $\mathcal{L}_2^{T_Y}$={CHINESE, MEI PEI} と $\mathcal{L}_2^{T_S}$={Chinese, Mei Bei}，のようなケースがある．(i) の違い（下線部分）は文字の挿入または削除によるものであり，(ii) の違いは文字の置換によるものである．

この両者の表記ゆれによる言語名の変化の度合いは同等である．すなわち，CHINANTECO と Chinantec および PEI と Bei の編集距離はどちらも同じ値の 1 と考えるのが妥当，ということである．v と w をそれぞれ 2 つの語とし，v と w の語類似度を次のように定義する．

[定義 4] 次の式を満たす $sd\_w(v, w)$ は 2 つの語 v と w の語類似度である．

$$sd\_w(v, w) = \frac{l_A(v, w) - ed(v, w)}{l_A(v, w)} \tag{2}$$

ここで，それらの $ed(v, w)$ と $l_A(v, w)$ はそれぞれ置換，挿入，削除の 3 つの編集操作を許可し，コストをいずれも 1 としたときの編集距離とアラインメントの長さである．　□

前述の例について計算すると，(i) CHINANTECO と Chinantec については，l_A=9，ed=1，sd_w=8/9 となり．(ii) PEI と Bei について，l_A=3，ed=1，sd_w=2/3 となる．

(2) 言語名の類似度

$\mathcal{L}_1^{T_Y}$={CHINANTECO, LALANA}，$\mathcal{L}_1^{T_S}$={Chinantec, Lalana} を例にとり，説明していく．

$\mathcal{L}_1^{T_Y}$ には 2 つの語，$\mathcal{L}_1^{T_S}$ にも 2 つの語が含まれている．$\mathcal{L}_1^{T_Y}$ の 1 つ目の語 CHINANTECO に対しては，(CHINANTECO, Chinantec)，(CHINANTECO, Lalana) の 2 通り，$\mathcal{L}_1^{T_Y}$ の 2 つ目の語 LALANA に対しては，(LALANA, Chinantec)，(LALANA, Lalana)

の 2 通り，の計 4 通りの組合せがある．式 (2) に従って，前者の 2 通りの組合せの語類似度を計算すると，0.88 と 0.2 が得られる．この中で，(CHINANTECO, Chinantec) の語の組合せの語類似度が最大となる．このような組合せを**語ペア**とよぶことにする．

2 つの言語名のすべての語ペアを求めるには，次の操作を行えばよい．(i) すべての組合せの語類似度を計算し，最大語類似度を持つ語の組合せを見つけ，語ペアとする．(ii) 語ペアに含まれる語を含む組合せを削除する．(iii) 残りの組合せの中から，最大語類似度を持つ語の組合せを見つけ，語ペアとする．(iv) 残りの組合せがなくなるまで，(ii) と (iii) を繰り返す．

$\mathcal{L}_1^{T_Y}$ と $\mathcal{L}_1^{T_S}$ の語ペアは全部で 2 つで，(LALANA, Lalana) と (CHINANTECO, Chinantec) が得られ，それぞれの語ペアの類似度が 1 と 0.88 である．

言語名 $\mathcal{L}_1$=$\{v_1, v_2, \cdots, v_m\}$ と $\mathcal{L}_2$=$\{w_1, w_2, \cdots, w_n\}$（$m \geq n$）の類似度 $sd\_ln(\mathcal{L}_1, \mathcal{L}_2)$ は次のように定義する．

[定義 5] $\mathcal{L}_1$=$\{v_1, v_2, \cdots, v_m\}$ と $\mathcal{L}_2$=$\{w_1, w_2, \cdots, w_n\}$（$m \geq n$）は言語名であり，$v_i \in \mathcal{L}_1$ に対応する語ペアは (v_i, w_i') である．ただし，$w_i' \in \mathcal{L}_2$ で，v_i の語ペアが存在しない場合は w_i'=NULL である．次の式を満たす $sd\_ln(\mathcal{L}_1, \mathcal{L}_2)$ は言語名 $\mathcal{L}_1$ と $\mathcal{L}_2$ の類似度である．

$$sd\_ln(\mathcal{L}_1, \mathcal{L}_2) = \frac{\sum_{i=1}^{m} sd\_w(v_i, w_i')}{m} \tag{3}$$

□

すなわち，$\mathcal{L}_1^{T_Y}$=$\{$CHINANTECO, LALANA$\}$ と $\mathcal{L}_1^{T_S}$=$\{$Chinantec, Lalana$\}$ の例では，$sd\_ln(\mathcal{L}_1^{T_Y}, \mathcal{L}_1^{T_S})$=(1+0.88)/2=0.94 となる．

4.3　言語系統分類の類似度

言語の系統分類は言語系統木におけるパスで表すことができる．以下では，パスの比較を文字列の比較に転化させ，文字列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

言語 $y \in V_{leaf}(T_Y)$ と $s \in V_{leaf}(T_S)$ のパスをそれぞれ $\mathcal{P}(y)$ と $\mathcal{P}(s)$ とする．パスは $\mathcal{P}(y)$=$\mathcal{L}_{y_1}\mathcal{L}_{y_2}\cdots\mathcal{L}_{y_{k-1}}$ のように，言語系統木のルートの子からリーフノードである言語の親までのノードラベルのリストとして定めている（3.2 節参照）．

言語 y は “Algic” という語族（レベル 1）の言語で，レベル 2，レベル 3，レベル 4 の言語グループ名はそれぞれ “Algonquian”，“Algonquian Proper”，“Arapaho” である．**図 4** に示すように，$\mathcal{P}(y)$=$\{$Algic$\}\{$Algonquian$\}\{$Algonquian, Proper$\}$ $\{$Arapaho$\}$ となる．言語 s のパスは $\mathcal{P}(s)$=$\{$Algic$\}\{$Algonquian$\}\{$Plains$\}\{$Arapaho$\}$ である．

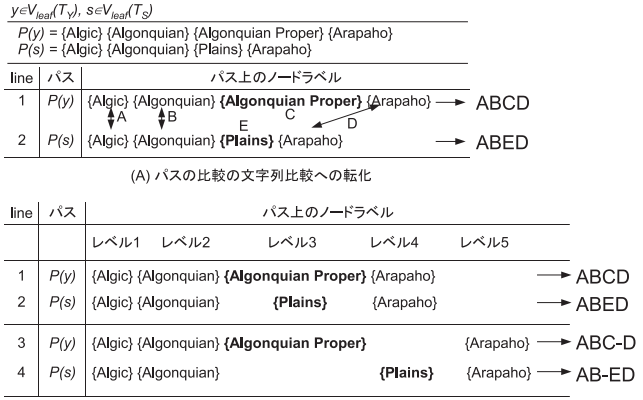

図 4　系統分類の比較
Fig. 4　Comparison of language classification.

$\mathcal{P}(y)$ と $\mathcal{P}(s)$ に含まれる異なるノードラベルをそれぞれ異なる 1 文字に変換して表せば，$\mathcal{P}(y)$ と $\mathcal{P}(s)$ の比較を文字列の比較に転化させることができる．ノードラベルの文字への変換方法としては，図 4 (A) に示すように，2 つのパスに含まれる同じノードラベルに同じ文字を割り当てればよい．たとえば，$\mathcal{P}(y)$ の任意の 1 つのノードラベルに対して，$\mathcal{P}(s)$ のノードラベルとの間の類似度を計算し，その類似度が閾値 α を超えるノードラベルが見つかったならば，それらのノードラベルには同じ文字を割り当てる．ノードラベル間の類似度の計算は 4.2 節 (2) で説明した言語名の類似度の計算法を用いる．図 4 (A) の例では，ABCD と ABED という 2 つの文字列が得られる．この変換処理を $\mathcal{F}(\mathcal{P}(y), \mathcal{P}(s))$→(ABCD, ABED) で表す．次に，変換後得られた 2 つの文字列の類似度を計算するが，これは 4.2 節 (1) で説明した語類似度の計算法とは異なる．

図 4 (B) の上半分の line 1，2 は，$\mathcal{P}(y)$ と $\mathcal{P}(s)$ がノードラベルが一致するノード {Algic}，{Algonquian}，{Arapaho} に合わせて，揃えられている．すなわち，レベル 1，レベル 2，レベル 4 のノードラベルは一致しているが，レベル 3 の {Algonquian, Proper} と {Plains} は不一致である．つまり，これは $\mathcal{P}(y)$ の {Algonquian, Proper} から $\mathcal{P}(s)$ の {Plains} への置換があった，としている．それに対し，図 4 (B) の下半分の line 3，4 では別の見方をしており，レベル 3 は {Algonquian, Proper} と {Plains} の不一致（置換）ではなく，$\mathcal{P}(y)$ は，{Algonquian, Proper} の削除および {Plains} の挿入という 2 つの編集操作で $\mathcal{P}(s)$ に

なった，としている．我々は，系統分類の比較は図 4 (B) の line 1，2 に示している方が妥当であると考える．

一方，2 本のパスを 2 つの文字列に変換した後は，その 2 つの文字列の編集距離を求めるが，その求め方としては，両パス中において 1 つでも同じノードラベルがあれば，言語学的観点からその一致を見逃してはいけないことから，4.1 節 (2) で説明したような LCS を求めるべきである．LCS を求めるためには，置換は考慮しないため，図 4 (B) の line 3，4 に示すようなアラインメントになる．ここで矛盾が生じるが，その解決法を次において述べる．

(2) 系統分類の類似度

言語 y と言語 s のパス $\mathcal{P}(y)$ と $\mathcal{P}(s)$ に対し，$\mathcal{F}(\mathcal{P}(y),\mathcal{P}(s))\rightarrow(v,w)$ の変換処理を行い，それぞれ 2 つの文字列 v と w に変換する．v と w は言語 y と言語 s の系統分類の比較を行うための文字列となっている．上記の例 v=ABCD，w=ABED を用いて，系統分類の類似度の求め方について述べる．

(i) v=ABCD，w=ABED に対し，挿入と削除のみを許すように（コストはいずれも 1 とする），文献 7) の動的計画法による LCS 問題を解決する手法に従って，v から w に変形すると，アラインメント (v'=ABC−D, w'=AB−ED) が得られる．前述のように，アラインメントは複数通り可能である．ここでは v から w への変形過程において，文字の不一致が現れたら，v の文字 (C) の削除と w の文字 (E) の挿入という順に操作するとする．また，後述する系統分類の類似度の定義から分かるように，このような限定を加えても，系統分類の類似度の値に影響をおよぼすことはない．

(ii) v' と w' の 2 行の文字列を 1 行の文字列に変換する．アラインメント (ABC−D, AB−ED) の同じ列の 2 つの文字につき，文字が一致している場合は $*$，v' の文字がギャップ（−）である場合は $-$，w' の文字がギャップ（−）である場合は $+$，の記号にそれぞれ置き換える．この例では，$**+-*$ となる．

(iii) (ii) で得られた文字列 $**+-*$ では置換は考慮されていない．$**\underline{+-}*$ に対し，置換を許すように，下線部分の $\underline{+-}$ を X に変換し（X はアラインメントの 2 つの文字の不一致を意味する），新たな文字列 $**X*$ を得る．この再構成後の文字列 $**X*$ を**言語系統分類の類似特性記号列**（Similarity feature string of language classification）とよび，$SFSLC(v,w)$ で表す．また，$SFSLC(v,w)$ の長さを $l_{SFSLC}(v,w)$ で表す．

$SFSLC$ は，$*$，$+$，$-$，X という 4 つの記号を使って，任意の $y\in_{leaf}(T_Y)$ に対し，T_S での系統分類を基準にした T_Y での系統分類の変化を表現するための記号列とみることができる．また，$SFSLC$ という 1 つの文字列で表すことによって，T_Y と T_S の 2 つの言語系統木での系統分類の相違のとりうる様相を推定し，視覚的にとらえることもできる．なお，$SFSLC$ の求め方は上記提案した手続きによるほか，たとえば，置換コストを 2 として，アラインメントを求めるトレースバック時には，置換を挿入・削除より優先してたどる方法によっても得ることができる．

言語 $y\in V_{leaf}(T_Y)$ と $s\in V_{leaf}(T_S)$ の系統分類の類似度 $sd\_lc(y,s)$ を次に定義する．

[定義 6] 次の式を満たす $sd\_lc(y,s)$ は言語 $y\in V_{leaf}(T_Y)$ と $s\in V_{leaf}(T_S)$ の系統分類の類似度である．

$$sd\_lc(y,s)=\frac{l_{LCS}(v,w)}{l_{SFSLC}(v,w)} \tag{4}$$

ただし，v と w は $\mathcal{F}(\mathcal{P}(y),\mathcal{P}(s))\rightarrow(v,w)$ によって，言語 y と言語 s のそれぞれの系統分類を表すパス $\mathcal{P}(y)$ と $\mathcal{P}(s)$ から変換された文字列である．□

図 4 の例では，v=ABCD，w=ABED，$l_{LCS}(v,w)$=3，$SFSLC(v,w)$=$**X*$，$l_{SFSLC}(v,w)$=4，$sd\_lc(y,s)$=3/4 となる．

4.4　ゆれのある同一言語の検出

3.4 節の完全一致言語の検出処理では，言語名のゆれと系統分類のゆれには対応できない．つまり，この方法では，$y\in V_{leaf}(T_Y)$ と $s\in V_{leaf}(T_S)$ の 2 つの言語が本来同一言語であっても，言語名または系統分類にゆれがあり，あるいはこれらの両方にゆれがある場合は，(y,s) を同一言語ペアとして検出することはできない．

ここでは，y と s の言語名と系統分類がともに一致していなくても，一定の条件を満たす類似関係を持っているならば，y と s を同一言語と判定する方法について述べる．4.2 節と 4.3 節の定義に従って言語名の類似度 $sd\_ln(\mathcal{L}_y,\mathcal{L}_s)$（または $sd\_ln(\mathcal{L}_y,\mathcal{A}_i^s)$，$\mathcal{A}_i^s$ は s の（複数の）別名の中の任意の 1 つの別名である）の最大値，また系統分類の類似度 $sd\_lc(y,s)$ の最大値を計算し，それらの最大値がそれぞれ閾値 α と閾値 β を超えるならば，(y,s) を同一言語ペアとして検出する．閾値 α と β はあらかじめ設定しておく．この α と β の決め方については 5 章で議論する．

任意の $y\in V_{leaf}(T_Y)$ に対して，T_S での同一言語の検出処理は次のように行う．

(1) まず，任意の $y\in V_{leaf}(T_Y)$ に対し，T_S のすべての言語から，言語名の類似度が最大となる言語を検索する．ここで得られる言語名の類似度の最大値を $sd\_ln_{max}(\mathcal{L}_y)$ で表す．なお，y と $s\in V_{leaf}(T_S)$ の言語名の類似度の比較処理は，y の第一言語名 $\mathcal{L}_y$ と s の第一言語名 $\mathcal{L}_s$ に対し，また y の第一言語名 $\mathcal{L}_y$ と s の（複数の）別名 A_s=$\{\mathcal{A}_1^s,\mathcal{A}_2^s,\cdots\}$ の中の各々の別名に対しても行う．$sd\_ln_{max}(\mathcal{L}_y)\leq\alpha$ ならば，y と同一の言語は T_S には存在

アルゴリズム：FSLV

入力：$y{\in}V_{leaf}(T_Y)$, T_S, α, β
出力：y の同一言語ペア SLP
手法：
0°　$SLP{\leftarrow}\phi$
1°　$S^y{\leftarrow}\phi$, 次の (i)〜(iv) を行う．
(i) すべての $s{\in}V_{leaf}(T_S)$ に対し，${\mathcal{L}'}_s{\leftarrow}\mathcal{L}_s{\cup}\mathcal{A}_1^s{\cup}\mathcal{A}_2^s{\cup}\cdots$ とする．
(ii) $L'{\leftarrow}\{{\mathcal{L}'}_s|s{\in}V_{leaf}(T_S)\}$ とする．
(iii) $sd\_ln_{max}(\mathcal{L}_y){=}\max\{sd\_ln(\mathcal{L}_y,\mathcal{L})|\mathcal{L}{\in}L'\}$ を計算する
(iv) $S_y{\leftarrow}\{s|sd\_ln(\mathcal{L}_y,{\mathcal{L}'}_s)=sd\_ln_{max}(\mathcal{L}_y),sd\_ln_{max}(\mathcal{L}_y)>\alpha\}$ とする．
2°　次の (i)〜(iii) を行う
(i) $sd\_lc_{max}{=}\max\{sd\_lc(y,s)|s{\in}S_y\}$ を計算する
(ii) $SLP{\leftarrow}\{(y,s)|sd\_lc(y,s)=sd\_lc_{max},sd\_lc_{max}>\beta\}$ とする．
(iii) $|SLP|>1$ ならば，$SLP{\leftarrow}\phi$ とする．
3°　SLP の要素を出力し，停止する．

図 5　ゆれのある同一言語の検出
Fig. 5　Finding same languages with variations.

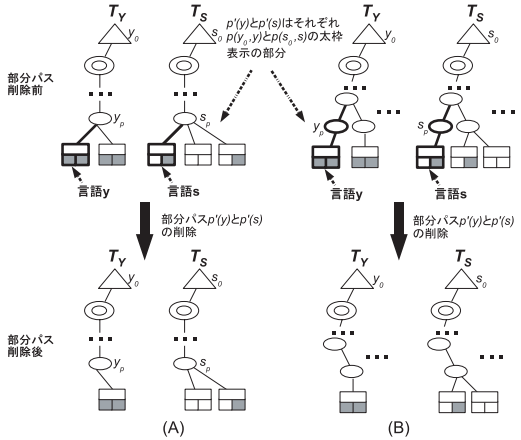

図 6　部分パス削除の例
Fig. 6　Examples of deletion of sub-path.

していないことになる．そうでないならば，次は系統分類の類似度によって判定を行う．

(2) (1) で得られた，y との言語名の類似度の最大値 $sd\_ln_{max}(\mathcal{L}_y)$ を持つ T_S の言語は複数得られる可能性がある．次に，これらの複数の言語 $s_1,s_2,\cdots$ の各々に対し，y との系統分類の類似度 $sd\_lc(y,s_1)$, $sd\_lc(y,s_2)$, $\cdots$ を計算し，その中から系統分類の類似度が最大となる言語を検索する．ここで得られる系統分類の類似度の最大値を $sd\_lc_{max}(y)$ で表す．なお，系統分類の類似度の算出結果は閾値 α の値に関連していることに注意されたい（4.3 節 (1) 参照）．

(3) 最後の判定として，(i) $sd\_lc_{max}(y){>}\beta$，(ii) $sd\_lc_{max}(y){>}\beta$ を満たす言語 $s{\in}V_{leaf}(T_S)$ が唯一であること，という 2 つの条件を満たすならば，(y,s) は同一言語と判定する．

以上の処理のアルゴリズムを**図 5** に示す．

4.5　処理全体の流れ

Yamamoto-Data と SilGIS-Data のそれぞれに含まれる同一言語の検出処理は，T_Y と T_S の 2 つの言語系統木を処理データとし，次の処理 I と処理 II の 2 つの手順に分けて行う．

［処理 I］（完全一致言語の検出） すべての $y{\in}V_{leaf}(T_Y)$ に対し，次の処理を行う．

Step1 T_Y と T_S に対し根から左優先の深さ優先探索を行い，$\mathcal{P}(y){=}\mathcal{P}(s)$ を満たすパス対 $(\mathcal{P}(y),\mathcal{P}(s))$ を見つける．

Step2 1 つのパス対 $(\mathcal{P}(y),\mathcal{P}(s))$ に対し，複数の言語対 $\{(y,s)\}$ が存在しうる（$\mathcal{P}(y)$ と $\mathcal{P}(s)$ に複数の言語がぶら下がっている）．これらの (y,s) に対し，$\mathcal{L}_y{=}\mathcal{L}_s$ または $\mathcal{L}_y{\in}A_s$ を満たした言語 y と s を完全一致言語として出力する．

Step3 完全一致言語として検出された同一言語ペア (y,s) について，パス $p(y_0,y)$ $(y_0{=}r(T_Y))$ とパス $p(s_0,s)(s_0{=}r(T_S))$ のそれぞれの部分パスである $p'(y)$ と $p'(s)$ を削除し，T_Y と T_S を更新する．y の部分パス $p'(y)$ は次の (i) と (ii) を満たす $p(y_0,y)$ の最も長い部分パスである．(i) 部分パス $p'(y)$ は $(y_p,y){\in}E(T_Y)$ を含む．(ii) 部分パス $p'(y)$ 上のノードの子の数は 1 つである（ただし，ノード y の場合は子の数は 0）．s の部分パス $p'(s)$ についても同様に定められる．**図 6** に (A) と (B) の 2 つの例を示す．

図 6 からも分かるように，このような操作によって削除される部分パス上のノードには 2 つ以上の子を持つノードは含まれていないことから，削除後に他の言語の系統分類の類似度が変化してしまうことはない．

［処理 II］（ゆれのある同一言語の検出）

更新された T_Y と T_S において，すべての $y{\in}V_{leaf}(T_Y)$ に対し，次の処理を行う．

Step1 任意の $y{\in}V_{leaf}(T_Y)$ に対し，アルゴリズム FSLV（図 5）を実行し，得られた SLP を同一言語ペアとして出力する．

Step2 同一言語と判定した $y{\in}V_{leaf}(T_Y)$ と $s{\in}V_{leaf}(T_S)$ について，処理 I Step3 と同様にして，T_Y と T_S を更新する．

5. 処理結果および考察

本章では，Yamamoto-Data と SilGIS-Data のそれぞれに含まれている言語の同一性を判定するため，3 章と 4 章で述べた手法に基づき，T_Y と T_S に対し処理を行い，得られた結果を**表 2** に示す．

処理 I と処理 II を合わせると，Yamamoto-Data の総言語数 2,870 中の 2,526 言語（約 88%）について，SilGIS-Data の言語との対応づけが判明した．また，検出された 2,526 言語のうち，処理 I で得られた完全一致言語が 1,034（約 36%），処理 II で得られたゆれのある言語が 1,492（約 52%）であった．さらに，完全一致言語の 1,034 言語のうち，Yamamoto-Data の第一言語名と SilGIS-Data の別名の一致による結果が 156 言語であった．ゆれのある言語の 1,492 言語のうち，言語名一致・系統分類類似，言語名類似・系統分類一致，言語名類似・系統分類類似，の言語がそれぞれ 1,367，81，44 個検出された結果となった．

木構造を取り入れたことにより，2.2 節 (d) のケースの問題は解決できるようになり，表 1 (B) の Bai 言語は表 1 (A) の No=212 の BAI と No=213 の BAI のうち，No=212 の BAI に対応していることが判明した．また，言語名と言語系統分類がともに類似する例として，T_Y の第一言語名が ‘‘YI, GUICHOU’’，系統分類が ‘‘Sino-Tibetan’’/‘‘Tibeto-Burman’’/‘‘Burmese-Lolo’’/‘‘Lolo’’/‘‘Northern’’ の言語と T_S の第一言語名が ‘‘Yi, Guizhou’’，系統分類が ‘‘Sino-Tibetan’’/‘‘Tibeto-Burman’’/‘‘Lolo-Burmese’’/‘‘Loloish’’/‘‘Northern’’/‘‘Yi’’ の言語が，言語名の類似度が 0.93，系統分類の類似度が 0.67，$SFSLC$ が $***X*-$ という結果が得られ，同一言語として検出された．

表 2 の処理 II で検出された同一言語ペアの数は，言語名の類似度の閾値を α=0.75，系統分類の類似度の閾値を β=0 と設定したときの結果である．閾値をこのように設定した理由は次のとおりである．

α，β とも閾値の設定値は小さいほど検出される同一言語ペアの数が増える一方，誤判定（異なる言語を同一言語と判定すること）の言語ペアの数も増える．つまり，処理結果の信頼性が低下する．閾値の最適な値は，検出される同一言語ペアの数とその中に含まれる誤判定の言語ペアの数を総合評価したうえで，設定すべきと考えられる．また，本研究の成果を Yamamoto-Data と SilGIS-Data の言語の対応をとったうえで，SilGIS-Data を媒介とし，Yamamoto-Data の言語の地理的位置情報を取得することに用いることを想定していることをあわせて考慮すると，検出される同一言語ペアの数を多くすることよりも，誤判定の言語ペアの数をなるべく少なくすることに重点をおくべきである．このような考えの下で，適切な閾値を見つけるための実験を行った．

言語名の類似度の閾値 α についての実験結果を**表 3** (A) に示す．実験は次のような手順で行った．まず，β=0，α=0.65 に設定して実験を行った．系統分類の類似度の閾値を β=0 という値から出発したのは，次のような理由からである．系統分類を比較するときに，パス上 1 つでも同じノードラベルがあれば，系統分類の類似性を否定しない（4.3 節を参照）．そのようなケースでは，系統分類の類似度を計算するとき，長いパス上の 1 つのノードラベルが一致している場合は，系統分類の類似度の値は 0 に近くなることがあるためである．β=0，α=0.65 のときの結果として，検出同一言語ペア数は 1,541，誤判定の言語ペア数は 23 となった．次に，α の値を大きくし，β=0，α=0.7，0.75 に設定して実験したところ，α=0.75 のとき，誤判定の言語ペア数が 0 になった．このとき，検出同一言語ペア数は 1,492 となった．さらに，β=0，α=0.74 に設定して実験したところ，検出同一言語ペア数は 1,503 に増えた一方，誤判定の言語数も 0 から 3 に増えた．誤判定をなるべく避けるという方針に従い，言語名の類似度の閾値は α=0.75 を最適な値として採用することにした．

さらに，系統分類の類似度の閾値を β=0 に設定することの効果を確認するため，異なる β の値，すなわち β=0.1，0.15（α=0.75 で一定）の場合の実験も行った．その結果を表 3 (B) に示す．表 3 (B) に示すように，β=0.1，0.15 では，β=0 に設定したときと同じように誤判定のケースは出なかったが，検出同一言語ペアの数が減り，β=0 の場合より悪い結果と

表 2　同一言語の判定結果
Table 2　Experiment results of the language search.

処理	検出同一言語ペア数	比率（2,870に対する比率）
処理 I	1,034	36%
処理 II	1,492	52%
合計	2,526	88%

α=0.75, β=0

表 3　閾値設定に関する実験結果
Table 3　Experiment results for threshold decision.

(A) α の値設定に関する実験結果（β=0）

実験	1	2	3	4
閾値 α の設定値	0.65	0.7	**0.75**	0.74
検出同一言語ペア数	1,541	1,512	**1,492**	1,503
誤判定の言語ペア数	23	12	**0**	3

(B) β の値設定に関する実験結果（α=0.75）

実験	1	2
閾値 β の設定値	0.1	0.15
検出同一言語ペア数	1,490	1,488
誤判定の言語ペア数	0	0

なった．

また，言語名の類似度の閾値を α=0.75，系統分類の類似度の閾値を β=0 と設定したときの結果について，判定漏れ（本来検出されるべき同一言語ペアが検出されなかったことを指す）の原因について調査したところ，次のような (i) と (ii) のケースがあった．

(i) URARTIAN と Urarina のような言語名が 1 つの語の場合である．類似度が 0.67 で，検出されなかった．

(ii) ‘‘CHONTAL OF OAXACA, HIGHLAND’’ と ‘‘Chontal, Highland Oaxaca’’ のような言語名に助詞が入っている場合で，このときの類似度が 0.75 である．ほかには，類似度が 0.67 の ‘‘CHONTAL OF TABASCO’’ と ‘‘Chontal, Tabasco’’ のような場合も判定漏れになった．下線部分の OF のような助詞が入っている言語名は多数で，検出されたケースも少なくなかった．たとえば，‘‘MAZATECO, SAN JUAN CHIQUIHUITLA’’ と ‘‘Mazateco de San Juan Chiquihuitlan’’ についても，下線部分の de は助詞と思われるが，こちらの場合は言語名を構成する語の数が多いため，類似度が 0.78 になり，判定漏れとはならなかった．この問題に対し，助詞のリストを作成し，あらかじめ言語名から助詞を削除する方法も考えられるが，そもそも両言語データとも多種の言語の文献を参考にし作成されており，助詞をリストアップすること自体が困難であると予想できるため，あえて例外処理を行わないことにした．

また，言語名の類似度は閾値 α=0.75 を超えたが，系統分類の類似度が 0 となったため，検出されなかった言語は 45 もあった．この中では，まったく系統分類が異なる言語が多かったが，次のようなケースもあった．T_Y と T_S での系統分類はそれぞれ ‘‘French-based Creole’’ と ‘‘Creole’’/‘‘French based’’ である．2 本のパスを 2 つの文字列に変換する際のノードラベルの類似度を閾値 α=0.75 にしているため，T_Y のパス上の唯一のノードラベル {French, based, Creole} が T_S のパス上の 2 つのノードラベルの {Creole} と {French, based} のどちらとも同じノードとならず，系統分類の類似度は 0 になり，検出されなかった（4.3 節 (1) 参照）．これは，2 本のパスを 2 つの文字列に変換する際のノードラベルの類似度の閾値の設定を β と関連して考慮する必要性の検討について，示唆を与えてくれたことになる．

6. おわりに

本研究では，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 88%の言語の同一性が判定できた．そのうち，52%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．また，SilGIS-Data の言語との同一性が判定できたことによって，Yamamoto-Data に収集されている世界諸言語に関連する空間データが利用できるようになり，語順特徴に関する地図化など，GIS を用いた語順研究の展開が可能になった．

今後は 5 章の処理結果に対する考察を通して，2 つの言語名がともに 1 つの語からなる場合とそうでない場合の言語名の類似度の閾値 α の値設定の妥当性を検討し，OF などの助詞の問題に対しは，オントロジーアラインメントなどで広く用いられている外部知識源に基づく語の類似度の評価手法を取り入れて解決を図っていきたい．さらに，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して，本手法をさらに発展させていきたい．

謝辞　本研究の遂行にあたり，貴重なご助言と多くのご示唆を与えていただいた弘前大学人文学部山本秀樹教授，山口大学人文学部乾秀行准教授に，深い感謝の意を表する．また，査読者には貴重なご指摘とご示唆をいただいた．ここに敬意ならびに謝意を表す．

参考文献

1) 池田　潤：GIS と言語研究，一般言語学論叢，No.9, pp.1–10 (2006).
2) 呉　靭，山本秀樹，乾　秀行，杉井　学，松野浩嗣：語順地図作成に必要なデータ及び語順地図に現れる語順分布，一般言語学論叢，No.10, pp.31–49 (2007).
3) 呉　靭，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).
4) Gordon, R.G. (ed.): *Ethnologue: Languages of the World, 15th ed.*, Dallas, SIL International, Texas (2005).
5) 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).
6) 浅井達哉，有村博紀：半構造データマイニングにおけるパターン発見技法，電子情報通信学会論文誌，Vol.J87-D1, No.2, pp.79–96 (2004).
7) Jones, N.C., Pevzner, P.A.（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).
8) 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).
9) Navarro, G.: A guided tour to approximate string matching, *ACM Computing Surveys* (*CSUR*), Vol.33, No.1, pp.31–88 (2001).
10) Sellers, P.H.: The theory and computation of evolutionary distances: Pattern recog-

nition, *Journal of Algorithms*, Vol.1, No.4, pp.359–373 (1980).
11) 市瀬龍太郎：情報の意味的な統合とオントロジー写像，人工知能学会誌，Vol.22, No.6, pp.818–825 (2007).
12) 市瀬龍太郎：オントロジーマッピングにおける有効な特徴の抽出，2008 年度人工知能学会全国大会（第 22 回）論文集，2E1-1, pp.1–4 (2008).
13) Stoilos, G., Stamou, G. and Kollias, S.: A string metric for ontology alignment, *Proc. 9th IEEE International Symposium on Wearable Computers*, pp.624–637 (2005).
14) 高橋良平，小山　聡，田中克己：恣意的に名前付けされたオブジェクトの識別手法，日本データベース学会論文誌，Vol.8, No.1, pp.5–10 (2009).
15) 星合　忠，山根康男，津田　宏：カテゴリマッチング技術に基づくオントロジーアラインメント問題への取り組み，人工知能学会論文誌，Vol.20, No.6, pp.437–447 (2005).
16) Pedersen, T., Patwardhan, S. and Michelizzi, J.: WordNet::Similarity — Measuring the relatedness of concepts, *Proc. 19th National Conference on Artificial Intelligence* (*AAAI-2004*), pp.1024–1025 (2004).
17) http://www.ethnologue.com/web.asp
18) Haspelmath, M., Dryer, M.S., Gil, D. and Comrie, B. (eds.): *The world atlas of language structures*, Oxford University Press, Oxford (2005). (http://wals.info/)
19) Asher, R.E. and Moseley, C.J.: *Atlas of the world's languages*, Routledge, New York (2007).
20) ディクソン，R.M.W.（著），大角　翠（訳）：言語の興亡，岩波新書 (2001).
21) 呉　靭，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).
22) 呉　靭，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).

(平成 22 年 2 月 4 日受付)
(平成 22 年 3 月 28 日再受付)
(平成 22 年 4 月 20 日採録)

呉　　靭（学生会員）

昭和 63 年広島大学工学部第一類（機械系）卒業．平成 2 年同大学大学院工学研究科システム工学専攻博士前期課程修了．富士通テン株式会社，株式会社西日本情報システム等を経て，現在山口大学，山口コ・メディカル学院各非常勤講師．山口大学大学院理工学研究科博士後期課程在学中．言語情報処理，マルチメディア教材開発等の研究に従事．コンピュータ利用協議会会員．

松野　浩嗣（正会員）

昭和 57 年山口大学工学部電子工学科卒業．昭和 59 年同大学大学院修士課程修了．昭和 59〜62 年山口短期大学，昭和 62〜平成 6 年大島商船高等専門学校勤務．平成 7 年山口大学理学部助教授．平成 17 年同教授．平成 18 年同大学大学院理工学研究科教授．計算機ネットワーク構築技術と生命のシステム的理解に関する研究に従事．理学博士．IEEE，電子情報通信学会各会員．