

NAIST-IS-MT0251115

修士論文

木構造診療録類似検索手法の開発

松村 憲和

2004年2月6日

奈良先端科学技術大学院大学
情報科学研究科 情報生命科学専攻

本論文は奈良先端科学技術大学院大学情報科学研究科に
修士(工学) 授与の要件として提出した修士論文である。

松村 憲和

審査委員： 湊 小太郎 教授
植村 俊亮 教授
杉浦 忠男 助教授

木構造診療録類似検索手法の開発*

松村 憲和

内容梗概

近年、電子カルテの普及に伴って、蓄積されたデータを2次利用した診療支援が期待されている。本研究では、1患者の診療行為に関する詳細が記述された診療録を用いて、症状・所見が類似した診療録をデータベースから検索できる手法を開発し、システムに実装した。個々の診療録には、カテゴリーデータ、順位データが混在しているが、木構造に配列すれば構造化文書として表現できる。この木構造診療録の特徴を捉えた類似検索手法を開発することが本研究の課題である。木構造診療録の特徴として「下位ノードは上位ノードの詳細所見である」という点があげられる。よって、木構造診療録を類似検索する為には、上位ノードに対する類似性を特に重要視する必要がある。そこで本手法では、最近接類似検索を基に、下位のノードを根とした部分木における類似度(0~1.0)を順次加重平均していくことにより二つの診療録間の類似度(0~1.0)を求める。そして、質問診療録に対する類似検索結果として、過去の診療録を類似度の降順に並べ変えて出力する。本手法を医療現場で実際に使用されている木構造診療録を用いて評価した。この評価実験によって、本システムの出力は、一方で医師の類似判定と近い結果を与えるとともに、他方で医師の経験・価値観にとらわれない結果をも示すことを確認した。

キーワード

XML, 木構造, 事例ベース推論, 電子カルテ, アルゴリズム, 最近接類似検索

* 奈良先端科学技術大学院大学 情報科学研究科 情報生命科学専攻 修士論文, NAIST-IS-MT0251115, 2004年2月6日.

The Development of Similarity Retrieval Method for the Tree Structure of Medical Records*

Norikazu Matsumura

Abstract

In recent years, because of the popularization of Electronic Medical Record, it is expected medical treatment using this database. We developed a method where we can calculate similarity of symptom sign and doctor's impression between medical records. Tree structure of medical records has the feature where the low rank layer node details the high rank layer node. It is necessary to develop a method grasping the feature. The similarity for medical records is calculated first by averaging the weight of similarities for multiple trees found in the low rank layer. This average is determined by multiplying the similarity found between the detailed data of node. This method determines ranking of medical records stored in the database by the similarities. We evaluated this method using tree structure of medical records actually used. By this evaluation experiment, we checked where the output of this system was near to the similarities of a doctor, and was not biased his experience and feeling of value.

Keywords:

XML, tree structure, Case-Based Reasoning, Electronic Medical Record, algorithm, Nearest Neighbor Retrieval

* Master's Thesis, Department of Bioinformatics and Genomics, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-MT0251115, February 6, 2004.

目次

1. 序論	1
1.1 研究背景	1
1.2 関連研究	2
1.3 研究目的	4
1.4 本稿の構成	4
2. 事例ベース推論 (Case-Based Reasoning) について	6
2.1 事例ベース推論 (Case-Based Reasoning: CBR)	6
2.2 CBR システム	7
2.2.1 CHEF	7
2.2.2 HYPO	8
2.3 医療分野における CBR システム	9
2.3.1 CASEY	9
2.3.2 FRORENCE	9
2.3.3 PROTOS	10
2.4 最近接類似検索	11
3. システム概要	12
3.1 木構造診療録	12
3.2 システム構成	13
3.3 木構造診療録類似検索の手順	14
3.4 データベースと関連テーブル	15
3.4.1 リレーショナルデータベースと XML ネイティブデータベース	15
3.4.2 Medical Record Database	16
3.4.3 Relationship Table	16
3.4.4 Weight Table	18
3.5 検索対象木構造	18
3.6 Tree Similarity Retrieval	19

3.7	GUI インタフェース	23
3.7.1	Part Search	23
3.7.2	Ignore	24
3.7.3	Combine	24
4.	評価実験	26
4.1	運動負荷心電図 ST 部分データを用いた実験	26
4.1.1	実験準備	26
4.1.2	実験結果	28
4.1.3	考察	30
4.2	耳診療録シミュレーションデータを用いた実験	32
4.2.1	実験準備	32
4.2.2	実験結果	33
4.2.3	考察	36
4.3	上部消化管造影検査レポートを用いた実験	38
4.3.1	実験準備	38
4.3.2	実験結果	38
4.3.3	考察	47
5.	結論	50
	謝辞	53
	参考文献	54
	付録	57
A.	本研究で扱うデータの構造	57
B.	胃の所見における Relationship Table	60
C.	胃の所見における Weight Table	64

D. 読影医診断結果との比較	67
E. 医師判断とシステム出力結果との差の図	70

目 次

1.1	全国での地裁・簡裁での医療過誤訴訟件数（最高裁事務総局調べ）	1
2.1	Case-Based Reasoning cycle	7
3.1	木構造診療録の例	12
3.2	木構造診療録類似検索システムの構成と類似検索の流れ	13
3.3	Relationship Table への R 入力画面	17
3.4	node N における $V_N(A)$ が null のとき	19
3.5	node N における $V_N(A)$ が複数存在するとき	19
3.6	上位 node が類似していないときの例	20
3.7	GUI インタフェース (Combine 選択画面)	24
3.8	Combine による類似度制御の図	25
4.1	運動負荷心電図 ST 部分データの例	26
4.2	sagging	27
4.3	horizontal	27
4.4	junctional	27
4.5	運動負荷心電図 ST 部分木構造データ	28
4.6	運動負荷心電図 ST 部分データの最適化前, 最適化後の結果	31
4.7	耳診療録 A (ID:030910074514)	34
4.8	耳診療録 B (ID:030910074527)	34
4.9	耳診療録 A の木構造データ	34
4.10	耳診療録 B の木構造データ	34
4.11	右, 左の区別をしないとき ($R(\text{右}, \text{左}) = 1.0$) の出力結果	35
4.12	右, 左の区別をするとき ($R(\text{右}, \text{左}) = 0.0$) の出力結果	35
4.13	全ての W を 1.0 としたときの類似検索結果	37
4.14	W を式 (3.1) を用いて求めたときの類似検索結果	37
4.15	胃の木構造診療録データ	39
4.16	診療録 ID26 の類似検索結果	42
4.17	診療録 ID27 の類似検索結果	42
4.18	診療録 ID34 の類似検索結果	43

4.19 診療録 ID38 の類似検索結果	43
4.20 診療録 ID50 の類似検索結果	43
4.21 診療録 ID63 の類似検索結果	43
4.22 診療録 ID75 の類似検索結果	44
4.23 診療録 ID80 の類似検索結果	44
4.24 診療録 ID86 の類似検索結果	44
4.25 図 4.16 ~ 図 4.24 の凡例	44
4.26 医師判断カテゴリー順位の例	46
4.27 医師判断との差の例	46
D.1 ID26 の読影医診断結果との比較	67
D.2 ID27 の読影医診断結果との比較	67
D.3 ID34 の読影医診断結果との比較	67
D.4 ID38 の読影医診断結果との比較	67
D.5 ID50 の読影医診断結果との比較	68
D.6 ID63 の読影医診断結果との比較	68
D.7 ID75 の読影医診断結果との比較	68
D.8 ID80 の読影医診断結果との比較	68
D.9 ID86 の読影医診断結果との比較	69
D.10 図 D.1 ~ 図 D.9 の凡例	69
E.1 診療録 ID26 における差	70
E.2 診療録 ID27 における差	70
E.3 診療録 ID34 における差	71
E.4 診療録 ID38 における差	71
E.5 診療録 ID50 における差	71
E.6 診療録 ID63 における差	71
E.7 診療録 ID75 における差	72
E.8 診療録 ID80 における差	72
E.9 診療録 ID86 における差	72

表目次

3.1	図 3.6 の Relationship Table	20
4.1	ST 降下の 4 つの分類	27
4.2	ST 部分データに関する Relationship Table	29
4.3	運動負荷心電図 ST 部分データを用いた結果の表	31
4.4	システム出力結果と医師判断との定量的比較	47
A.1	Record	58
A.2	Selected	58
A.3	First Atom	58
A.4	value	58
A.5	Tree Name	58
A.6	Relationship Table	59
A.7	Weight Table	59
B.1	粘膜における Relationship Table	60
B.1	粘膜における Relationship Table	61
B.1	粘膜における Relationship Table	62
B.2	形態における Relationship Table	63
B.3	機能における Relationship Table	63
B.4	部位における Relationship Table	63
C.1	異常の有無における Weight Table	64
C.2	粘膜における Weight Table	64
C.2	粘膜における Weight Table	65
C.3	形態における Weight Table	65
C.4	機能における Weight Table	66

1. 序論

1.1 研究背景

現在の医療にとって電子カルテは必要不可欠なツールとなりつつある。電子カルテは、日々増加する医療情報の保管スペース、それに関わる多大なコスト、診療効率の悪さ等といった従来では解決することが困難であった様々な問題を解決してくれる。実際に政府の e-Japan 重点計画の一つとして、2006 年には 400 床以上の中核病院の 60% に電子カルテを導入するという計画もある [1]。この電子カルテの普及によって、臨床研究で用いるデータウェアハウスとの統合 [2] が促進されて、診療録データがさらに蓄積されてくる。この大量の診療録データを二次的に分析して有用な情報を得るといった診療支援が、近年、大変注目されている。他方、図 1.1 に示す診断名の決め付けや、思い込みが原因となった医療過誤が年々増加傾向にあり [3]、医療の質を向上させるためにも情報技術を活かした診療支援が早急に求められる。

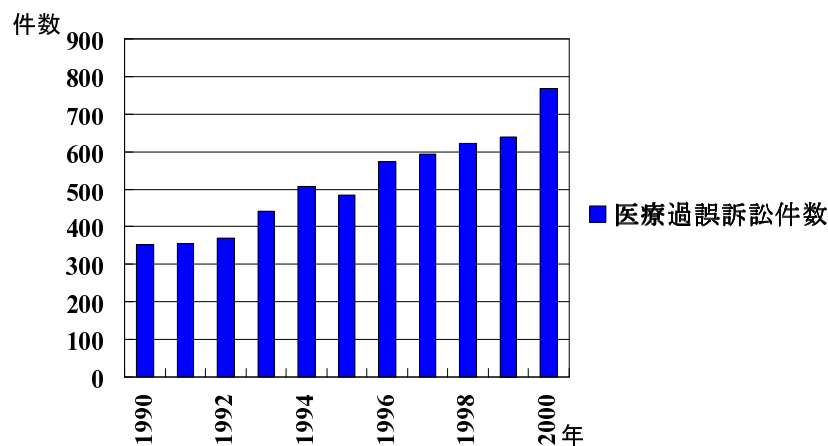


図 1.1 全国での地裁・簡裁での医療過誤訴訟件数（最高裁事務総局調べ）

本研究では、診療支援の中でも診療録の症状・所見による類似検索に注目した。医療に限らずどの問題解決プロセスでも同じであるが、問題に直面したとき、人

間は先人たちが行ったよく似た前例を参考にして、現在の問題を解決してきた。データベースから現在直面している問題とよく類似している前例を見つけ、それを参考にし問題解決に活かしたいというニーズは、どの分野においても急速に高まっている。医療の分野において、データベースに格納されている診療録データは何千、何十万にもわたっているため、人手による類似検索を行うのは不可能である。そこで本稿では、過去の診療録データの中から類似している診療録をすぐにデータベースから取得して参照することのできる類似検索アルゴリズム、そしてそれを実現する環境を構築することを目的とした。

多くの電子カルテでは、所見の記載は自由文で行われる。自由文であれば、患者の症状を自由に表現できる点で勝るものは無いが、グラフ等による視覚化、推論などといったデータ再利用に関しては、文章が構造化されていないので大変困難であり、優れているとはいえない。本研究では、近年増加傾向にあるテンプレートによる入力を行う電子カルテに着目した。小原他の研究 [4] によって、テンプレートによる電子カルテへの入力方法は、紙カルテと比べて、記入情報量の増加や診療記録の電子化による参照・検索の容易性などから多大な有効性を示している。そのような電子カルテは、大阪大学医学部附属病院で使われている Dynamic Template[5]、千葉大学医学部附属病院の system CHIBA[6] 等が挙げられる。これらの電子カルテのデータ形式は、主にカテゴリーデータ、順位データで入力される。さらに、それらの基本要素を木構造に配列することにより、一つのカルテの表現力を高めることができる。下位の要素は、上位の要素を修飾しているというようなデータ間の従属関係を表現することが可能になる。このような、木構造で表現できる診療録を用いた症状・所見の類似検索手法とそのシステムの開発が期待されている。

1.2 関連研究

医療における構造化データを用いて相互比較した研究としては、クリニカルパスの与薬行為を XML 化することによって、薬剤費を相互比較することのできるシステムがある [7]。クリニカルパスとは、それぞれの医療施設ごとに各疾患における標準的な診療パターンが表現されており、記述形式はまちまちであるために

相互比較することが従来では不可能であった．そこでこの研究ではXMLを用いることによって，パスの記述形式の標準化を行った．XMLの相互比較手法としてこの研究では，医薬品名，投与量に関する記述をXML文書から抜き出し，薬価データベースを参照しながらそれぞれの薬剤費を算出し，比較している．

その他の例としては，各大学病院のオーダーエントリーシステムのユーザーインタフェースを5階層の木構造で表現しているHCI(Human Computer Interaction)を用いることによって，定量的に比較した研究 [8] がある．一人の利用者がユーザーインタフェースを用いて行う一つの課題を「ユニットタスク」「ファンクション」「アークギュメント」「キーストローク」という4段階で分類を行い，5階層の木構造HCIを構築する．この4段階の分類は，課題の目標，それを行うコマンド，要点，実際にキーボードを打つ打鍵数のことである．このようにして構築されたHCIの相互比較手法は，1ノードあたりの平均分岐数を各階層ごとに求めてくる．そして，この4つの特徴量を4次元空間に配置し，ユークリッド距離を求めることによって，1課題に対する各大学病院間のオーダーエントリーシステムのユーザーインタフェースの相互比較を行っている．

XML文書検索としての研究では，キーワードを利用したXML文書検索システム [9] がある．このシステムでは，XML文書をそれ自体の文書構造を利用してXML部分文書に分割し，さらにこの部分文書を有益な内容を含んでいる部分文書(CPD:Coherent Partial Document)と不要な部分文書に分ける．そして，不要な部分文書を検索結果から削減することによって，質の良いCPDだけを検索で取得しようと試みている．手法としては，部分文書自身に含まれる単語数や異なり語数などの統計量を利用して明らかに文脈とはなりえないXML部分文書を除去することによって残る文書，つまりCPDが取得できる．

このように構造化データを比較・検索する研究は数多く行われており，全ての研究に共通して言えることは，データの特徴を捉えた比較・検索手法をそれぞれが考えているということである．本研究でも木構造診療録の特徴を捉えた類似検索手法を考案する必要がある．

1.3 研究目的

本研究の目的は、症状・所見が類似している木構造診療録を自動的に取得することのできるシステムを構築することにある。本木構造診療録類似検索手法の目的としては、以下の二つ挙げられる。まず、大病院では、数十万件に及ぶ多量の診療録データが、データベースに格納されているので人間が全てを比較し、類似診療録を検索することは不可能である。診療時において主治医になりかわって、類似診療録を見つけることが、本手法の一つ目の目的である。次に、1.1節の背景でも述べたが、医師は自らの過去の経験をたよりに意思決定を行うため、診断名の決め付け、思い込みなどといった医療過誤が起こる。そこで、そのような過去の経験にとらわれることなく、症状・所見の類似した木構造診療録を取得して提示することが、本手法の二つ目の目的である。これが実現されれば、主治医以外の第2の意見であるセカンドオピニオンとして用いることが可能である。この二つを本手法における目的として定めた。

このシステムの類似検索手法として、全てのデータが完全に一致する場合とそれ以外とに二分する方法がプリミティブであるが、応用上はあまり有用ではない。部分的にそして内容的に一致しているものを類似診療録として出力させるべきである。そこで本研究では、二つの診療録間の類似している指標として0~1.0の範囲で類似度 S を求め、それを降順に並び替え、順位付けを行うことによる木構造診療録類似検索手法を考案した。類似度 S が1.0に近ければ近いほど、二つの診療録は類似していると定義している。1.2節でも述べたように、この類似検索手法は木構造診療録データの特徴を捉えたものでなければならない。さらにその類似度定義には、希な症状・所見をもつ診療録に対しては、優先的に上位にくるようにできることが望ましい。これにより、同じ症状・所見が多い当たり前の診療録ではなく、希な診療録を優先的に取得することが容易になる。

1.4 本稿の構成

2章で、本研究の基盤となった概念である事例ベース推論について述べる。2.2節、2.3節では、事例ベース推論を用いたシステムを紹介する。2.4節では、本研

究で開発した Tree Similarity Retrieval という木構造診療録類似検索アルゴリズムの基となった手法である最近接類似検索について述べる。

3章では、本研究で開発した木構造診療録類似検索手法・システムの概要について述べる。3.1節で本稿で扱う木構造診療録について述べる。refS-structure節でシステム構成、3.3節で類似検索の流れを述べる。3.4節では、類似検索で用いる一つのデータベースと二つの関連テーブルについて述べる。3.5節、3.6節では、本研究で開発した木構造診療録類似検索アルゴリズムについて述べる。3.7節で、類似検索を行う前に操作を行う GUI インタフェースや、より自由度を持たせる為に構築した三つの機能 (Part Search, Ignore, Combine) について述べる。

4章で、本システムを用いた検証実験について述べる。4.1節では、冠動脈病変部位の推定の材料となる運動負荷心電図 ST 部分データを用いた本アルゴリズムの評価実験について述べる。4.2節と4.3節では、大阪大学医学部附属病院の電子カルテシステム MegaOak の診療録入力システムである Dynamic Template で作成された木構造診療録データを用いて、木構造診療録に対しての評価実験について述べる。4.2節では耳のシミュレーションデータ 1000 個の木構造診療録に対して、4.3節では、上部消化管造影検査リポートを木構造診療録に入力した実データに対しての実験を述べる。

5章で、本研究のまとめとして、課題、応用等を述べる。

2. 事例ベース推論 (Case-Based Reasoning) について

2.1 事例ベース推論 (Case-Based Reasoning: CBR)

本研究のような過去の事例を利用して現在の問題解決を行う手法として事例ベース推論 (Case-Based Reasoning: CBR) という考えがある。CBRの基本となる原理は「類似した問題は類似した解を持つ」という実世界における経験談である。CBRは、与えられた問題に類似する過去の事例を直接利用して解を導く [10]。様々な分野の専門家は、新しい問題が生じた場合、過去に解決された事例の中で類似したものを探し、参考にする。CBRは、まさにこの専門家の問題解決プロセスを素直に反映させた形といえる。さらに CBRの利点としては、エキスパートシステムでよく用いられる if-then のルールなどでは、専門家が持っている知識を取り出しルール化するには非常に困難な問題点であったのに対し、CBRを用いれば事例の形で容易に取得することができる。よって十分に理解されていない問題領域においては非常に有効な推論手法である [11]。

図 2.1 に示しているものが Aamodt によって考えられた CBR の基本的な手順を示した図 CBRcycle [12] である。直面した現在の問題に対して、General Knowledge にある過去の事例の中から類似した事例を Retrieve で検索する。次に問題と類似事例を比較して、問題に対する解決策として再利用できるかどうかを Reuse で判断する。そして、Revise で取得された Solved Case を現在の問題の解決策として適用できる形に変更する。このようにして、得られた Confirmed Solution が現在の問題に対する解決策となる。さらにこの問題に対する解決策を General Knowledge に格納することにより自己組織化も行う。

前述で事例の形で取得するのは容易であると述べたが、「どのような二つのデータを比較したとき、類似していると判断するべきか」という新たな問題が生まれてくる。CBR システムにとって、このような類似性評価がもっとも重要な問題点となる。図 2.1 の CBRcycle でいうなら Retrieve と Reuse の部分にあたる。本研究は、この Retrieve と Reuse を木構造診療録に対して実現しようと試みている。また、今までに報告されている CBR システムもこの問題に対して様々な手法を用いて解決しようと試みている。以下の節で、その手法の説明を交えて現在まで

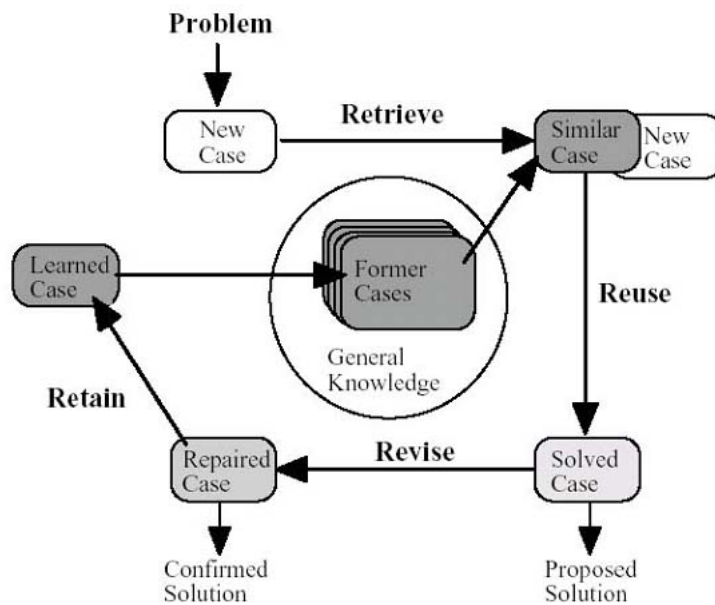


図 2.1 Case-Based Reasoning cycle

に報告されている CBR システムの紹介を行う。

2.2 CBR システム

現在，CBR は大変注目されている推論手法 [13] であり，医療を含め様々な分野において利用されており，CBR を用いて開発されたシステムはこれまでに数百報告例がある．それぞれのシステムが，如何にして類似した事例を探しているのかを述べる．例として，CHEF，HYPO について紹介する．

2.2.1 CHEF

CHEF [14] は，四川料理のレシピの作成を行う CBR システムである．入力はレシピにより作り出されるゴール（例えば，魚を使っている，香りが良いなど）であり，出力はこれらのゴールを作り出すレシピである．事例はゴールによって特

徴付けられている。例えば、「牛肉とブロッコリの炒め物」のレシピは、以下のよ
うなゴールに特徴付けられている。

include beef, include crisp vegetable, use method stir-fly, achieve taste savory.

CHEF は類似する事例を探し、それを修正することによって、新しい事例のた
めのレシピを作る。このシステムにおける類似事例を検索する手法としては、そ
れぞれの食材がどのような役割を果たしているのかを知っておかなければならな
い。「鶏肉とさやえんどうの炒め物」というゴールに対し、「牛肉とブロッコリの
炒め物」という事例を見つけるためには、牛肉と鶏肉は肉、さやえんどうと
ブロッコリは野菜というようにそれぞれの食材がどのような役割を果たしている
のかを知っておかなければならない。そう定義しておくことによって、牛肉は鶏
肉の代わりになる、ブロッコリはさやえんどうの代わりになるとシステムが理解
し、CHEF は類似事例を検索することができる。

2.2.2 HYPO

HYPO[15] は、CBR を用いた法的推論システムである。いくつかの判例を引
用し、現在の問題とどの部分で異なっているのかをシステムは指摘してくれる。
この HYPO における特徴は、論争中に引用する判例をどのように決定するかに
ある。

HYPO では、法的な性質を示している dimension を用いて事例を表現する。di-
mension は、「秘密の公開」「安全対策」といった 13 個の要素に分類されており、
それぞれの dimension を得るためには前提条件がある。その前提条件全てを満足
した事例に対して、これらの dimension がそれぞれの事例に対し与えられる。更
にその dimension には、焦点スロットという dimension 自身の強弱を示す値が用
意されている。HYPO の類似検索手法は、現在の問題が持つ dimension に対して、
同じ dimension を多く共有している事例を探し、最も数多く持つ事例が最も類似
している判例になる。さらに焦点スロットの値も考慮することにより、より最適
な類似事例を取得することが可能である。

2.3 医療分野における CBR システム

医療における知識は、ルールだけで成り立っている教科書的知識だけではなく、経験から得られる知識が混在したものであると言われている [13]。医療の分野において CBR を適用する際、この 2 種類の知識をうまく活用する必要がある。

ここで紹介する CASEY, FRORENCE, PROTOS は、既存の症状とその診断結果を事例ベースに格納しておき、新しい患者の症状が与えられると、既往の診断中の類似した事例をみつけてきて、その情報を用いて与えられた新しい症状を診断するというものである。

2.3.1 CASEY

CASEY[16] は、過去の診断事例に基づいて、与えられた問題に対する心疾患の診断を行うシステムである。CASEY は、心疾患の症状間の因果関係を説明できるモデルを作成し、そのモデル自体を提示することによって診断を支援する。このシステムが持つ前提知識として、既に診断された患者の症例として 45 所見と症状間の因果関係モデルをもつ。そして、一つの症例には、40 個の属性が用意されている。CASEY の類似検索の手順であるが、初めに問題となる患者の症状を入力し、属性ごとに事例と比較し、マッチした数からマッチしなかった数を引いた得点の大きい順に事例を並び替える。そして得点上位から順に患者の症例と事例の症例との差を調べる。その差が重要なものであればその事例は破棄し、重要でなければ事例の因果関係モデルを修正して出力する。

2.3.2 FRORENCE

FRORENCE[17] は、専門領域が特定されていない看護という広い領域での健康管理プランを生成するシステムである。主に FRORENCE は次の三つの質問に答えようとしている。

1. Diagnosis(診断) : 「現在の健康状態はどうか？」
2. Prognosis(予後) : 「将来、健康状態がどのようにかわるか？」

3. Prescription(処方) : 「どのようにしたら，健康状態が改善されるか？」

Diagnosis はルールベース推論，Prognosis は CBR，Prescription ではルールベース推論と CBR の両方を用いて推論を行う．Diagnosis では，健康状態の属性に重み付けのルールを用いて，現在の健康状態のスコア化を行う．Prognosis では，既に診断が確定し病状の進行状態も既知である診療録から，スコア化された患者の健康状態と個々の属性に対する健康尺度により類似している患者を検索する．Prescription では，特定の治療法に対してどのような効果があるかという情報が記されている if-then ルールを用いたルールベース推論と，類似した患者に対して特定の治療法を用いた結果が記されている事例を参照する CBR，この二つの推論によって解を導く．

2.3.3 PROTOS

PROTOS[18] の目的は，難聴の患者の診断結果を推論する．入力は患者の症候であり，出力は診断結果である．PROTOS では，内部に持つ分類木によって事例がカテゴリーに分類される．カテゴリー間には difference link というカテゴリー間の差異を示す症候がラベル付けされている．

PROTOS の推論手法であるが，新しい事例が与えられると，その特徴情報から分類木を用いたヒューリスティック検索によって，最適なカテゴリーを求める．ヒューリスティック探索とは，それぞれの経路で常に最適な方へ探索していき，最適解を見つける手法である [19]．そして，新しい事例とそのカテゴリー内の事例の 1 件ずつを比較し，それぞれの特徴情報の違いから類似しているかどうか判断する．このときに違いが重要なものでなければ，最も類似した事例を見つけられたことになり，新しい事例としてそのカテゴリー内に格納される．もし違いが重要なものであれば，その違いから difference link をたどって，また別のカテゴリーの事例に対して類似しているかどうか判断する．

2.4 最近接類似検索

最近接類似検索 (Nearest Neighbor Retrieval)[20] は，最も一般的に使われる類似検索手法である．これは入力と個別の事例間の類似度を計算し，その類似度の値が大きければ類似していると判断する手法である．類似度計算は，入力と事例のそれぞれに対応する属性の値を比較してその類似度を加重平均する．入力 Q と事例 C の類似度 $S(Q, C)$ の計算式は，以下ようになる．

$$S(Q, C) = \frac{1}{\sum W_i} \sum W_i \times s(Q_i, C_i) \quad (2.1)$$

$s(Q_i, C_i)$ は属性 i における Q と C の類似度である． W_i は属性 i に対する重みである．この手法は，対象となる問題領域が特定されており，問題，事例が属性によって表現されているときには有効な手法である [21]．本研究において，この手法を木構造データに対応できるように改良することによって木構造診療録類似検索手法を考案した．

3. システム概要

3.1 木構造診療録

本稿で扱う木構造診療録について説明する．図 3.1 を用いて，木構造診療録の各名称を説明する．この例は心音の診療録について示している．丸で表記されている箇所が node である．この一つ一つの node が，診療録の項目を表している．その node の中に書かれている下線が引かれた文字が value である．この value は項目に対するカテゴリーデータを表す．さらに node の下の子ノード，これを child node と呼ぶ．node の上の親ノードは parent node と呼ぶ．node と同じ parent node を持つ node を sibling node と呼ぶ．さらにこの木構造診療録の root は，テンプレート自体の名称であり，value は持たないものとする．root に対して，そこからの距離を木構造の階層と定義する．例えばこの例であると，過剰心音，心雑音が第一階層となり，過剰心音部位，心雑音の強度，心雑音のピッチ，心雑音の最強点 が第二階層となる．node 間の edge は，上位階層の node から下位階層の node に対しての詳細データと本木構造診療録では定義する．

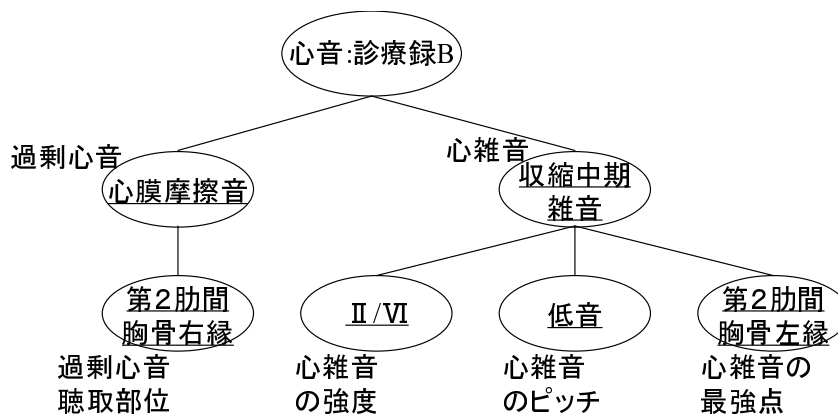


図 3.1 木構造診療録の例

3.2 システム構成

本システムは、大阪大学医学部附属病院で使用されている Dynamic Template[5] という電子カルテ入力システムで生成される XML で記述された木構造診療録データを基にして開発した。Dynamic Template は、症状・疾患ごとに入力項目が与えられており、データは階層構造を持っている。この診療録データは、各科ごとの雛型（心音、耳、上部消化管 etc）が用意されており、その雛形に所見を入力する。

図 3.2 に本システムの構成を示す。本システムは、二つの診療録間の類似度を算出する Tree Similarity Retrieval、一つのデータベース (Medical Record Database)、二つの関連テーブル (Relationship Table, Weight Table) で構成されている。プログラミング言語は JAVATM で構築した。それぞれの詳細については、後の節で説明する。

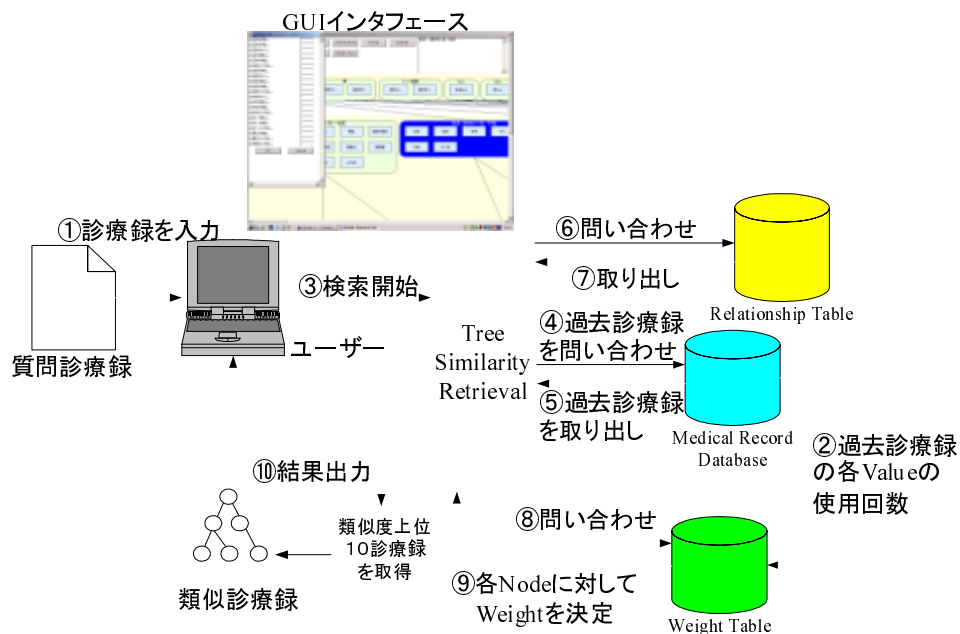


図 3.2 木構造診療録類似検索システムの構成と類似検索の流れ

3.3 木構造診療録類似検索の手順

木構造診療録類似検索を実行する前に、以下の2点を事前に行う必要がある。

1. Medical Record Database に過去に発生した診療録を格納する。
2. value 間の類似度 R を GUI インタフェースを用いて、Relationship Table に入力する。

次に図 3.2 のシステム構成図を用いて、木構造診療録類似検索システムが行う検索手順を説明する。

1. ユーザが診療録をシステムに入力する。
2. Medical Record Database に格納されている質問診療録と同じ型のテンプレートを用いた過去診療録から各 value の使用回数の情報を取得し、Weight Table に格納する。
3. GUI インタフェースを用いて類似検索の前提条件を入力し、検索を開始する。
4. 質問診療録と同じ型の過去診療録を問い合わせる。
5. 過去診療録を一つずつ取得する。
6. 質問診療録と過去診療録の二つの診療録で、同じ node で選択されている value 間の類似度 R を問い合わせる。
7. R を取得する。
8. 各 node(項目) に対する value の使用回数を問い合わせる。
9. 各 value の使用回数から各 node(項目) に対する Tree Similarity Retrieval で用いる W を求め、取得する。
10. Tree Similarity Retrieval によって求めた 2 診療録間の類似度 S が降順になるように過去診療録を並び替え、上位 10 診療録を類似診療録として出力する。

3.4 データベースと関連テーブル

ここでは、システムで用いられているデータベース (Medical Record Database) と二つの関連テーブル (Relationship Table, Weight Table) について述べる。これらは、リレーショナルデータベースの MySQL を用いて Windows 上に実装した。

3.4.1 リレーショナルデータベースと XML ネイティブデータベース

XML 文書を格納するデータベースとしては、一般的にリレーショナルデータベース、そして近年各システムにも導入されつつある XML ネイティブデータベースの 2 種類がある。

リレーショナルデータベースは、関連テーブルという 2 次元の表に格納し、さらにフィールド間にリレーションを設定することができる。XML を格納する際は、XML を分解して格納しなければならない。

それに対し、XML ネイティブデータベースは、XML ツリー構造をそのままの形式で格納することができる。

今回、リレーショナルデータベースを採用した要因としては、

1. パフォーマンス (処理速度, 信頼性)
2. 問い合わせ言語 SQL
3. 部分的な探索が可能
4. 低コスト

にある。しかし、将来、システムの改良に伴うデータベース・スキーマの変更などがある場合は、厳密なスキーマ情報を初めに定義しなければならないリレーショナルデータベースよりは、XML ネイティブデータベースを採用する方が良いと考えている。

本システムのデータベース・スキーマについては、付録 A に記す。

3.4.2 Medical Record Database

Medical Record Database は、過去に発生した診療録を格納するデータベースである。本システムで扱う木構造診療録は、3.2 節でも述べたように XML の書式をもつ。ここで扱う木構造診療録は、node 中の value が複数存在していたり、または省略されたりする半構造データ [22] であるために、リレーショナルデータベースに如何にして格納するかがここで問題となる。

Medical Record Database では、一つの value の情報として

1. 診療録の ID
2. value のある node の ID
3. value の ID
4. parent node の ID
5. parent node で選択されている value の ID

のように parent node を含む情報をリレーショナルデータベースの関連テーブルに格納することによって、選択された value がツリー構造のどの部分を表したものを示すことができる。

これにより、今回の木構造診療録のような半構造データをリレーショナルデータベースに格納することができた。

3.4.3 Relationship Table

Relationship Table は、node 内における value 間の類似度 R を 0 ~ 1.0 まででユーザーが定義し、その類似度を格納する関連テーブルである。類似度の値が大きければ大きいほど、二つの value は類似している、つまり類似度が 1.0 であれば全く同じであるということができる。

Relationship Table への入力は、3.7 節で説明する GUI インタフェースを用いて行う。図 3.3 は、GUI インタフェースを用いて Relationship Table に R を入力する画面である。

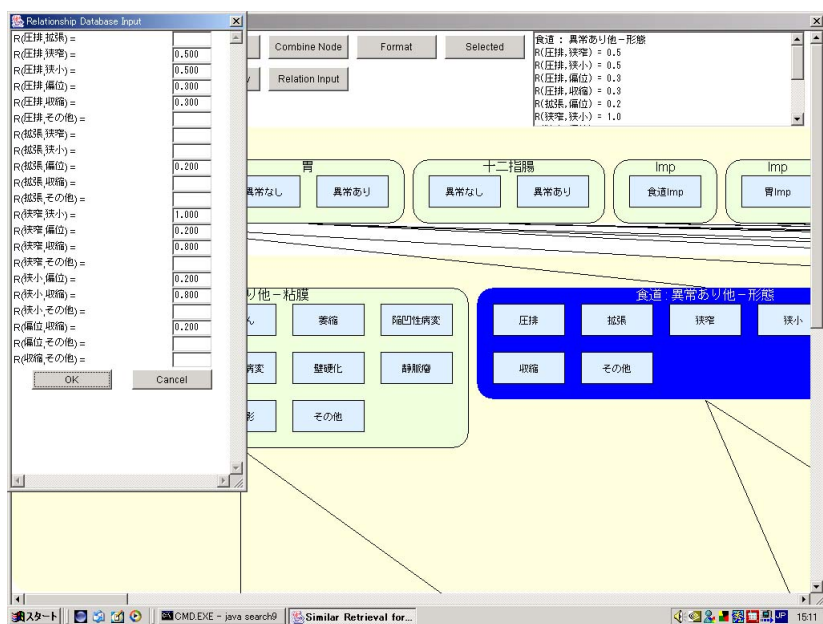


図 3.3 Relationship Table への R 入力画面

図 3.3 は、食道の形態という node の value 間の類似度 R を画面の左上のウィンドウにて入力を行っている図である。

さらに各階層に対して、 R の最小値 R_{min} を設けている。これは、3.6 節で述べる本木構造診療録類似検索手法の特徴である「下位の node よりも上位の node を重視した検索」を行う際に、下位の node における属性が異なっていたとしても上位の node において類似性を持っていれば、全体としてある程度の類似性を持たせるようにする為に設けた。 R_{min} は、定義されている階層で value が異なっているとき、類似性を R_{min} だけ現在の二つの診療録間の類似度に与える。よって本構造診療録の特徴から、下位にいけばいくほど、 R_{min} の値は大きくなる。もちろん R_{min} も範囲は 0~1.0 である。この各階層に対する最小値 R_{min} も 3.7 節で説明する GUI インタフェースを用いて入力を行う。

3.4.4 Weight Table

1.3 節で述べた研究目的の希な症状・所見をもつ診療録に対して、類似度を大きくするために、各 node に対する重み W を変える。類似度の計算方法については、3.6 節で説明する。

各 node に対する重み W は、次の計算式で求める。 N という node に対して、診療録 A,B 間における重み W_N を求めるとする。

$$W_N = \log \frac{2T}{a+b} \quad (3.1)$$

T は Medical Record Database に格納されている診療録の総数、 a,b は各診療録 A,B の N の value を選択している診療録の総数である。数式の中で \log を用いているのは、違いが顕著にならないようにするためである。これは、自然言語処理の技術 idf (inverse document frequency) においても用いられる。

3.5 検索対象木構造

この木構造診療録は、一つの node に対し、value が欠けていたり、複数 value を持っていたりするために、類似度を求める二つの診療録の木構造が必ずしも同じものではない。3.6 節で説明する Tree Similarity Retrieval または最近接類似検索によって求められる類似度の比較をする場合には、木構造を統一する必要がある。しかし、テンプレート自身の木構造情報を利用するだけでは、欠けた value や複数の value に対して、満足な対処ができない。そこで本研究では、質問診療録である診療録 A の木構造情報を基に類似空間を統一した。類似空間を質問診療録に統一させた木構造を検索対象木構造と呼ぶ。次の二つのケースについて説明する。

診療録 A のある node の value が null のとき

診療録 A のある node 内の value が欠けているとき、検索対象木構造は図 3.4 の例に示すように行う。図 3.4 は上部消化管の診療録における食道の木構造であり、食道の形態の value が欠けている。このとき、検索対象木構造は、その node

を消去して，新たな木構造を生成する．そうすることによって，Tree Similarity Retrieval における類似度計算は，その消去された node に対して依存しなくなる．

診療録 A のある node の value が複数存在するとき

図 3.5 は診療録 A のある node 内の value が複数存在する場合を示している．この図も前述と同じ食道の木構造診療録である．この例では，食道の粘膜の部分に 2 個 value が存在している．このような場合には，複数個 value がある node に対して，検索対象木構造でもその数だけ node を増やす．これにより，診療録 A の 2 個の value(壁硬化, 線条影) それぞれに対し，Tree Similarity Retrieval で類似度の計算を行うことができる．

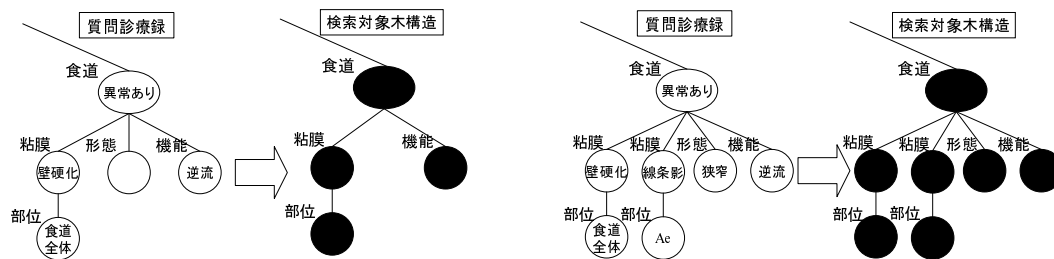


図 3.4 nodeN における $V_N(A)$ が null の 図 3.5 nodeN における $V_N(A)$ が複数存在するとき

このようにして，質問診療録 A に依存した形で検索対象木構造を決定する．

3.6 Tree Similarity Retrieval

Tree Similarity Retrieval は，二つの診療録 A,B 間の類似度 $S(A, B)$ を求めるアルゴリズムである．二つの木構造診療録の類似度を求める上で次の 2 点を重要視した．

- 木構造診療録データの特徴
- Relationship Table の利用

最初の木構造診療録データの特徴であるが，木構造診療録の下位の node は上位の node の詳細な記述であると見なされる [23]．よって，類似性の判断材料としては上位の node に重点をおくべきであると考えた．

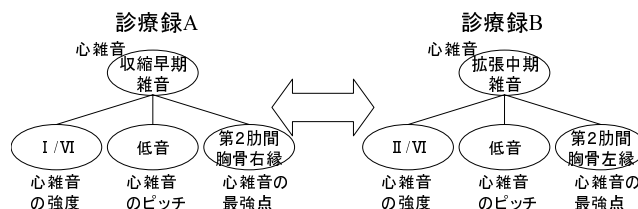


図 3.6 上位 node が類似していないときの例

表 3.1 図 3.6 の Relationship Table

診療録 A の value	診療録 B の value	類似度
収縮早期雑音	拡張中期雑音	0.0
I/VI	II/VI	0.8
低音	低音	1.0
第二肋間胸骨右縁	第二肋間胸骨左縁	0.8

2点目は Relationship Table を利用したことにある．これについては，図 3.6 の例を用いて説明する．二つの診療録 A,B が各 node に対して図 3.6 のような value を持っており，表 3.1 のように Relationship Table が定められているものとする．この例は心音に関する診療録である．このとき，心雑音の強度，ピッチ，最強点に関しては，Relationship Table により，類似度が 0 ではないためにある程度の類似性があるが，上位の node である心雑音は類似度が 0 であるため類似性がない．この場合，上位の node である心雑音が類似していないのであるから，下位の node に対して，たとえ類似性をもっていたとしても類似していないと考えるのがここでは妥当である．

このような 2 点を重要視して定義した類似度計算式を下記に記す．

3.5 節で説明した検索対象木構造に基づいて，類似度計算を Tree Similarity Retrieval で行う．質問診療録を A とし，Medical Record Database に格納されている診療録を B とする．類似度 $S(A, B)$ は，node の少ない部分木における類似度から順に求め，最終的に木全体における類似度，つまり診療録間の類似度を求める． N を root とした部分木における類似度は以下の計算式で求める．この計算式は 2.4 節で述べた最近接類似検索を基に考案した．

node N の子ノードが存在するとき

$$S_N(A, B) = \frac{1}{\sum W_{CN}} \sum W_{CN} \times \phi_{CN}(A, B) \quad (3.2)$$

$$\phi_{CN}(A, B) = \gamma_{CN}(A, B) \times S_{CN}(A, B) \quad (3.3)$$

$$\gamma_{CN}(A, B) = R_{min}(D_{CN}) + R(V_{CN}(A), V_{CN}(B)) \times (1.0 - R_{min}(D_{CN})) \quad (3.4)$$

node N の子ノードが存在しないとき

$$S_N(A, B) = 1.0 \quad (3.5)$$

CN は N の child node である． W_{CN} は，3.4.4 節で説明した CN に対する Weight Table に格納されている頻度を式 (3.1) によって求めてきた重みである．また， $V_{CN}(A)$ は，診療録 A の CN の value をさし， D_{CN} は， CN がある階層の深さを示す． $R(V_{CN}(A), V_{CN}(B))$ は，Relationship Table に格納されている CN 内における $V_{CN}(A)$ ， $V_{CN}(B)$ 間の類似度である． $R_{min}(D_{CN})$ は，3.4.3 節で説明した階層の深さ D_{CN} における R の最小値である． $R(V_{CN}(A), V_{CN}(B))$ の範囲は $0 \leq R(V_{CN}(A), V_{CN}(B)) \leq 1.0$ である．

2.4 節で説明した最近接類似検索の計算式 (2.1) でいうなら式 (3.2) がそれにあたる．属性 i に対する重みが W_{CN} に，属性 i における入力 Q と事例 C の類似度

$s(Q_i, C_i)$ が $\phi_{CN}(A, B)$ である．この式 (3.2) は，検索対象木構造における N の child node の数だけ $\phi_{CN}(A, B)$ の加重平均を行っている．検索対象木構造に同じ child node が複数個存在するときは，診療録 A も同じ個数の value を持っているの
 でそれぞれに対して計算し求める． $\phi_{CN}(A, B)$ は， CN における A,B 間の類似度 $\gamma_{CN}(A, B)$ と CN を root とした部分木における類似度との乗算で求める．leaf から順に式 (3.2),(3.3) を繰り返しながら，上位 node を root とした部分木における類似度を求めていき，最後には診療録 A,B 間の類似度 $S(A, B)$ が求められる． N の CN が存在しない，つまり N が leaf のときは，式 (3.5) のように比較する CN がない為に類似度 $S_N(A, B)$ は 1.0 になる．そして，式 (3.4) のようにして， R に木構造の位置情報も加味した $V_{CN}(A)$ と $V_{CN}(B)$ の類似度 $\gamma_{CN}(A, B)$ を求める．

次に Medical Record Database に格納されている診療録 B について述べる．検索対象木構造は，診療録 A に依存した形で作成されているので，診療録 B については一切情報が入っていない．よって検索対象木構造には存在する node について，診療録 B は value を持っていないか，複数の value を持っていたりする．どのようにしてそれらに対処するかについて述べる．

検索対象木構造には存在する node が診療録 B では存在しないとき

検索対象木構造には存在する node N に対して，診療録 B では value が存在しないとき，次のように定義した．

$$R_N(V_N(A), V_N(B)) = 0.0 \quad (V_N(B) = \text{null}) \quad (3.6)$$

欠けている value に対しては，value 間の類似性を持たせないように定義したことになる．

検索対象木構造に存在する node に対して，診療録 B は複数の value を持っているとき

検索対象木構造に存在する node N に対して，診療録 B が複数個 value を持つときを仮定する．これは式 (3.3) で求めた $\phi_N(A, B)$ の比較により求める．診療録 A の N における value は，検索対象木構造の N の個数だけ存在する．診療録 A の

N の個数が m 個，診療録 B における N の個数が n 個であったとする．そのとき，検索対象木構造には N が m 個あるわけであるから，各診療録 A の N における value m 個にそれぞれに対し $\phi_{N_x}(A_{V_x}, B)$ を求める．診療録 B の value が n 個存在しているので，各 $\phi_{N_x}(A_{V_x}, B)$ の求め方は， n 通りの $\phi_{N_x}(A_{V_x}, B_{V_1}), \phi_{N_x}(A_{V_x}, B_{V_2}) \dots$ と求めていき，値が最大のものが $\phi_{N_x}(A_{V_x}, B)$ となる．

このようにして，過去診療録全てに対して類似度を求め，検索順位をつけ，上位に来るものを診療録 A に対して症状・所見が類似した診療録とする．

3.7 GUI インタフェース

GUI インタフェースは，類似検索を始める前に Relationship Table への R の入力を行うインタフェースである．そして，その他に Tree Similarity Retrieval の問題点を解消する為，次の三つの機能 (Part Search, Ignore, Combine) を設けている．図 3.7 は，上部消化管の診療録を入力した GUI インタフェースに Combine 機能を選択する画面である．このようにこれら三つの機能は node に対して，入力を行う．

3.7.1 Part Search

Tree Similarity Retrieval では，診療録全体間の類似度しか求めることが出来ない仕組みとなっている．もちろん，診療録の中のある一部分の症状のみも検索可能にするべきである．

そこでこの Part Search では，木構造診療録の部分木における類似度を求めることができる．一つの node N に Part Search を選択することによって，その N を root とした部分木における類似度を求めることができる．計算式は 3.6 節で説明した式 (3.2)(3.3)(3.4)(3.5) を用いる．ここでは， N も value が存在する為に式 (3.3) の $\phi_N(A, B)$ が求める類似度となる．

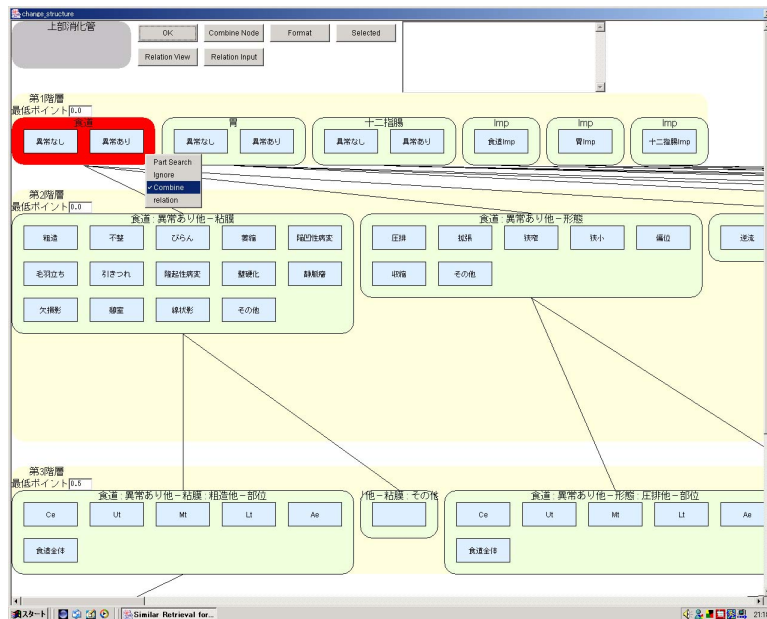


図 3.7 GUI インタフェース (Combine 選択画面)

3.7.2 Ignore

Tree Similarity Retrieval では、診療録全ての症状・所見に対して、value 間の比較を行い類似度を求めていくが、ある node に関しては、検索対象の項目として考慮したくない例もある。例えば、同じ症状が現れている診療録を検索したいのであって、その病変の部位についてはどこに現れていてもよいという場合である。そこで類似度を制御するための機能として Ignore を考えた。

Ignore は、選択した node N に関する全 value 間の組み合わせの R を全て 1.0 に変える。これにより、その N は類似度に対して影響を及ぼさなくなる。

3.7.3 Combine

この Tree Similarity Retrieval の問題点として、木構造の node を越えた類似度制御が難しい点がある。同じ node の value 間に関しては、 R を用いることにより類似度を制御することは可能ではあるが、同じ階層内の各 node に関して、また

違う階層の node に関して，類似度を制御することはできない．木構造診療録の特徴として「下位の node は，上位の node の詳細所見」であり，それが考慮されて構築された木構造であるので，全ての node に関して類似度の制御を与えられるようにする必要はない．ただ，階層が一つ違うもの同士であると，「上位 node に症状，下位 node にその症状の部位」または「上位 node に部位，下位 node にその部位の症状」といったように，明確な定義がないとどちらも可能性としてはありえる．よって，一つ上下の階層の違いに関しては，重要度を同じに制御することのできる機能を考えなければならない．それを実現するのが Combine である．図 3.8 を用いて説明する．

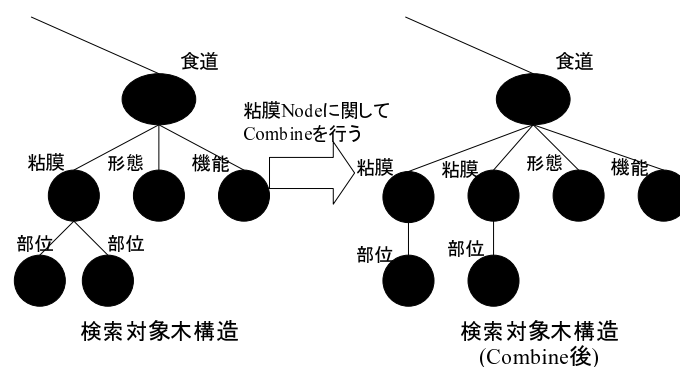


図 3.8 Combine による類似度制御の図

この図 3.8 の例も上部消化管の食道に関する診療録である．質問診療録 A が食道の粘膜の部位に関して二つ value を持っているために，図のように検索対象木構造の粘膜の部位に関する node が二つ生成されている．Tree Similarity Retrieval では，部分木における類似度を求めていくアルゴリズムである．よってこの例であると粘膜の node を root とした部分木における類似度を求めたときに，下位にある粘膜部位に関する情報は半減する．そこで図 3.8 のように，情報の半減をさせないために Combine では，検索対象木構造において，指定した粘膜の node のクローンを child node である部位の数だけ生成する．これにより，Tree Similarity Retrieval では，粘膜の部位 node の数だけ粘膜における部分木が生成され，類似度の減少を食い止めることが可能となる．

4. 評価実験

4.1 運動負荷心電図 ST 部分データを用いた実験

3章で述べたシステムの適用可能性を証明する為に、運動負荷心電図検査による心電図 ST 部分データを用いた冠動脈病変部位の推定実験を本システムを用いて行った。

4.1.1 実験準備

運動負荷心電図検査は、狭心症の診断のために実施される検査であり、狭心症の診断は、最終的に冠動脈造影検査により確定される。冠動脈造影検査は侵襲的検査で患者に負担がかかるため、その前に運動負荷心電図検査を実施し、冠動脈造影検査の必要性が判定される。そこで、図 4.1 の運動負荷心電図 ST 部分データを階層構造に見立てて、その木構造データを本類似検索手法を用いて、類似した別の運動負荷心電図 ST 部分データをデータベースから検出してくる実験を行った。

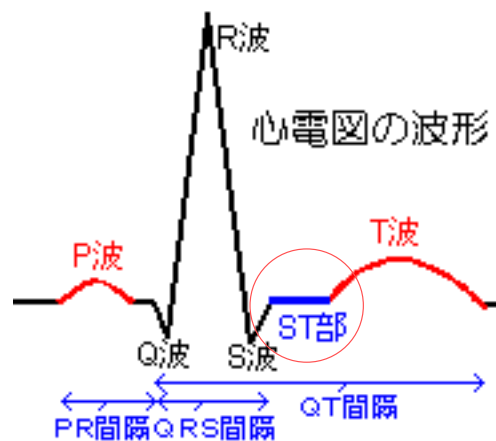


図 4.1 運動負荷心電図 ST 部分データの例

今回の実験では、運動負荷心電図の一つであるエルゴ負荷心電図を用いた。エルゴ負荷心電図とは、自転車をこぐ運動の負荷をかけ計測する心電図である。類

似検索の属性としては、標準 12 誘導部位での ST 降下の種類、低下の大きさの二つで実験を行った。ST 降下のタイプの種類としては、病変の疑いが強いものの順に sagging, horizontal, junctional と図 4.2, 図 4.3, 図 4.4 のように分け、その低下の大きさを表 4.1 のように 4 つの Level に分類した。各 12 誘導には、ST 降下が見られた場合、そのタイプの種類と低下の大きさがデータとして与えられる。この 2 種類のデータを図 4.5 のように木構造データに配列した。Root から 12 個分かれているパスに各 12 誘導に対する ST 降下の内容が記されている。それぞれのパスの左上に書かれている 1,2,3,avf,avr, … が各誘導の名称である。第 1 階層の各 node に ST 降下のタイプの種類を示す value, 第 2 階層の各 node に ST 降下の大きさの value がある。第 1 階層の各 node の value に関して、s が sagging, h が horizontal, j が junctional を示している。

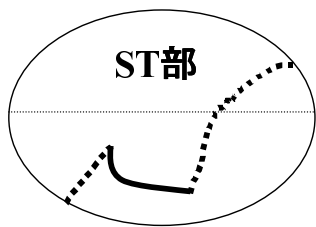


図 4.2 sagging

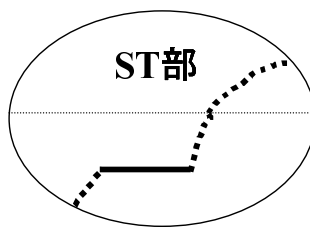


図 4.3 horizontal

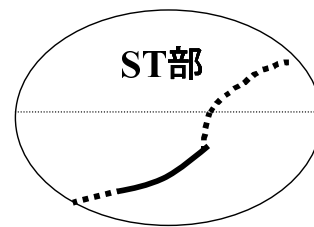


図 4.4 junctional

表 4.1 ST 降下の 4 つの分類

Level	ST 低下の大きさ
1	0.1mV 以上 0.5mV 未満
2	0.5mV 以上 1.0mV 未満
3	1.0mV 以上 2.0mV 未満
4	2.0mV 以上

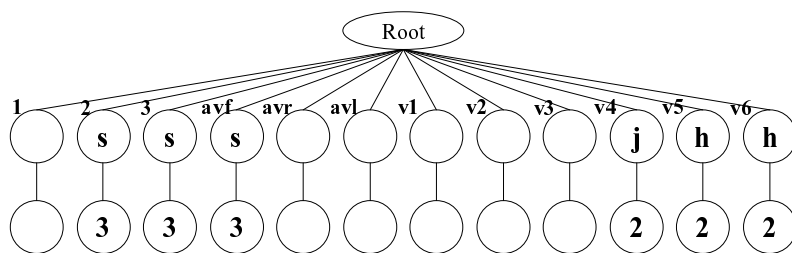


図 4.5 運動負荷心電図 ST 部分木構造データ

4.1.2 実験結果

本実験で扱うエルゴ負荷心電図 ST 部分データ，冠動脈造影検査データは，大阪大学医学部附属病院第一内科で 1996 年 1 月から 1997 年 12 月までの間にエルゴ負荷心電図および冠動脈造影を行われたデータを用いており，その間隔が 3 ヶ月以内の症例を選択している [24]．一つの所見について，それぞれがこの二つのデータを持っている．今回，Medical Record Database に 95 所見格納した．

本実験の方法を説明する．本実験では，この ST 部分データのみを本類似検索を行う対象とし，図 4.5 の木構造データのように配列する．そして，Medical Record Database に格納されている 95 所見の全組み合わせである ${}_{95}C_2 = 4465$ 通りに関して，ST 部分データを属性とした木構造診療録類似検索を行い，それぞれ類似度を求める．システム出力結果の整合性を調べるデータとしては，冠動脈造影検査の結果である冠動脈狭窄病変部位 (RCA,LAD,LCX) のデータを用いる．このデータには，病変の有無のみが記されている．RCA,LAD,LCX それぞれについて，今回は独立に実験を行っている．整合性は以下のように定義した．二つのデータ共に冠動脈病変が存在する場合，共に冠動脈病変が存在しない場合，この二つのケースを類似していると定義し，片方に冠動脈病変が存在しもう片方には存在しない場合，このケースを類似していないと定義した．実験においては，木構造データが図 4.5 のように確定されたものであるので，検索対象木構造は図 4.5 の木構造で固定した．

Relationship Table へ入力する R であるが，木全体では暫定的に表 4.2 のように

定めた．そして，各階層の R の最小値である R_{min} は， $R_{min}(1) = R_{min}(2) = 0.0$ とした．

表 4.2 ST 部分データに関する Relationship Table

value1	value2	類似度
sagging	horizontal	0.7
horizontal	junctional	0.7
sagging	junctional	0.3
1	2	0.8
1	3	0.5
1	4	0.2
2	3	0.8
2	4	0.5
3	4	0.8

Weight Table へ入力する W であるが，心電図を見るとき，ST 部分データに関して頻度の情報は普通考慮しないものである為に，式 (3.1) による W の算出は行わずに，最急降下法・直線探索法・黄金分割法を用いた教師あり学習を行い W を求めた．教師データとしては，前述で定義した類似しているときの教師データを 1 に，類似していないときの教師データを 0 とした．そして，全ての W を $W=1.0$ にして類似度 S を求めた．その後， $W = 1.0$ の類似度 S の値と教師データとの差の二乗和が最小になるように最急降下法を基にした教師あり学習を用いることによって， W の値を変えることによる最適化を行った．

この実験の結果が表 4.3 であり，その表を基にグラフにしたのが図 4.6 である．類似度 S の閾値を 0.3 と 0.7 に定義し，各組み合わせの整合性を次のように定義した． S が 0.3 以下で二つの ST 部分データの組み合わせに対する教師データが 0 の場合と S が 0.7 以上で二つのデータの組み合わせに対する教師データが 1 の場合の 2 ケースが正事例である．反対に S が 0.3 以下で二つのデータの組み合わせに対する教師データが 1 の場合と S が 0.7 以上で二つのデータの組み合わせに対

する教師データが0の場合の2ケースが負事例になる． S が0.3以上0.7以下のときは，どちらにも判断することが出来ないと定義した．最適化前は W を全て等しい1.0とした場合，最適化後は W を最急降下法を基にした教師あり学習により最適化した場合である．

図4.6の結果の縦軸は，95所見のうちの二つのデータの組み合わせの数であり，最大が4465である．図4.6の結果により，RCA，LCXは最適化後は最適化前と比較すると正事例が約2倍弱の増加が分かり，約5割強の正事例を得られたが，LADに関しては最適化後も最適化前とあまり変化することはなく，全体としても約3割しか正事例を得ることは出来なかった．

4.1.3 考察

図4.6に示すように，LADに関して誤りが多い結果となったことについて，次の要因が考えられる．まず，一つ目にこのST部分データだけで冠動脈狭窄病変部位を推定するのは不可能であるという点にある．ST部分データだけでなく，その他にもデータ項目を増やして推定を行う必要があったと考えられる．これが最も大きな原因であった．二つ目にRelationship Tableを正しく定義しなかったことについて問題がある．もっと専門家の意見を取り入れて考えていれば，さらに最適な結果が得られた可能性がある．最後にRCA,LAD,LCXと独立に今回評価実験を行った点である．どの冠動脈狭窄病変部位によってST部分の変化が起きるのか明確にすることができなかつたために，独立で評価を行ったことが問題であったという可能性がある．

このように，本実験で用いたデータはこの類似検索システムの評価に適当なものではなかったが，ともかく臨床データに対する本手法の適用可能性は見出すことができたと考えている．

表 4.3 運動負荷心電図 ST 部分データを用いた結果の表

教師データ	0	1	1	0	0,1
類似度	0.3 以下	0.7 以上	0.3 以下	0.7 以上	0.3 ~ 0.7
RCA 最適化前	34	1450	179	210	2592
RCA 最適化後	55	2732	45	598	1035
LAD 最適化前	51	1223	162	473	2592
LAD 最適化後	99	1235	166	364	2601
LCX 最適化前	34	1420	179	240	2592
LCX 最適化後	64	2393	100	514	1394

類似度0.3以下・教師0
 類似度0.7以上・教師1
 類似度0.3以下・教師1
 類似度0.7以上・教師0
 類似度0.3以上0.7以下

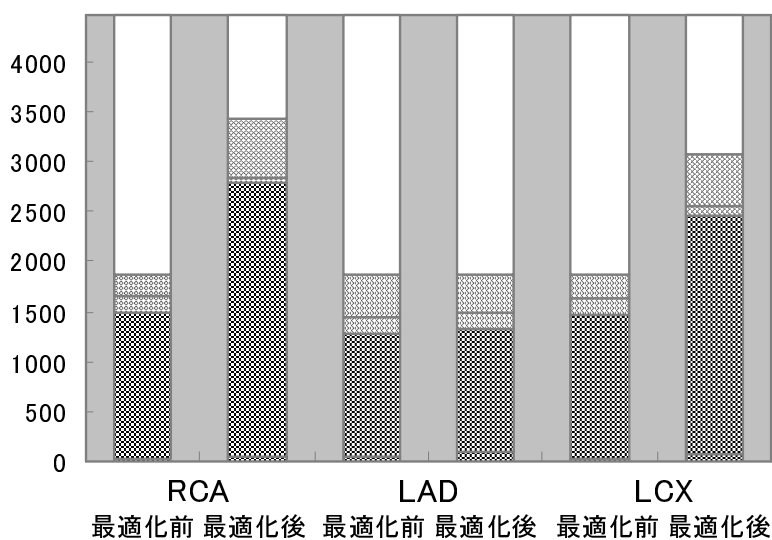


図 4.6 運動負荷心電図 ST 部分データの最適化前，最適化後の結果

4.2 耳診療録シミュレーションデータを用いた実験

Dynamic Template により入力した耳の木構造診療録のシミュレーションデータを用いた実験を行った。本実験により，Relationship Table，Weight Table の効果を示す。

4.2.1 実験準備

コンピュータがランダムに value の選択を行ったシミュレーションデータの 1000 個の診療録と図 4.7 と図 4.8 の Dynamic Template によって生成した二つの診療録 (診療録 A, 診療録 B) を含めた計 1002 個の診療録を Medical Record Database に格納した。このシミュレーションデータの生成方法は，以下の通りである。

1. 耳の診療録に存在する各 node に対して，どのようなデータ形式 (チェックボックス・ラジオボタン等) であるのか調べる。
2. 第 1 階層のある node を N とする
3. node N に対して，選択する value の数をランダムで選ぶ。ラジオボタン等のデータ形式をもつ node に対しては value の数は 1 である。
4. node N に対して，その数だけ value をランダムで選択する。
5. node N の child node が下の階層で存在するようであれば，child node の一つを N として，3 に戻り value の生成を行う。
6. node N の sibling node でまだ value の生成を行っていないければ，その sibling node の一つを N として，3 に戻り value の生成を行う。node N が root のとき，シミュレーションデータの生成を終了する。
7. node N の parent node を N とし，6 に戻る。

図 4.7 と図 4.8 の DynamicTemplate により生成した二つの診療録の木構造データは，図 4.9，図 4.10 のようになる。この二つの診療録の value の内容は，「耳介の炎症部位」「乳様突起」の二つの node の value だけがそれぞれ「右異常」と「左

異常」のように異なっている．他の node については全て同じ value をもつ．本実験では，この二つの診療録を主に用いる．

4.2.2 実験結果

Relationship Table の効果を示す実験

初めに Relationship Table の効果を示すための実験を行う．実験方法は，診療録 A，診療録 B 間の類似度 S を二つの方法により求める．

まず最初に右耳と左耳の区別を行わないとき，つまり Relationship Table に $R(\text{右}, \text{左}) = 1.0$ と定義し類似度 S を求めた．出力結果画面が図 4.11 である．図 4.11 の左上のテキストボックスが診療録 A の木構造データ (XML)，右上のテキストボックスが診療録 B の木構造データ (XML)，そして，下のテキストボックスに表示されているのが診療録 A，診療録 B 間の類似度 $S(A, B)$ である．このときの結果は， $S(A, B) = 1.0$ ，つまり二つの診療録は完全に同じであるという結果が得られた．

次に右耳と左耳の区別をするとき，つまり Relationship Table に $R(\text{右}, \text{左}) = 0.0$ と定義し類似度 S を求めた．このときの出力結果画面が図 4.12 である．このときの類似度は， $S(A, B) = 0.451$ という結果が得られた． $R(\text{右}, \text{左}) = 0.0$ であるのに類似度が 0 ではない理由は，value が同じ node(聴力など) が存在しているために類似性が少し得られたからである．

この二つの結果により，ユーザー (医師) の求めている類似診療録のみを取得するために必要な二つの診療録間の類似度を制御する機能を Relationship Table で可能にしたと言える．

Weight Table の効果を示す実験

次に Weight Table の効果を示す為の実験として，二つの方法による実験を行った．まず，各 node に対する W をすべて 1.0 とし，図 4.7，図 4.9 の診療録 A を質問診療録として Medical Record Database に格納されている他の診療録，計 1001 個についてそれぞれ類似度を求めた．その結果が図 4.13 である．

次に本手法である式 (3.1) を用いた頻度による計算法によって各 node に対する

耳

耳: 異常なし 異常あり

耳介: 異常なし 奇形 変形 炎症 耳性ヘルペス 血腫
 腫瘍 凍傷 湿疹 脂肪囊腫

…炎症
 左右: 右 左 両側

…右
 詳細所見:

乳様突起: 両側異常なし 右異常 左異常

…右異常
 所見: 圧痛 腫脹
 詳細所見:

聴力 右: 左:

耳

耳: 異常なし 異常あり

耳介: 異常なし 奇形 変形 炎症 耳性ヘルペス 血腫
 腫瘍 凍傷 湿疹 脂肪囊腫

…炎症
 左右: 右 左 両側

…左
 詳細所見:

乳様突起: 両側異常なし 右異常 左異常

…左異常
 所見: 圧痛 腫脹
 詳細所見:

聴力 右: 左:

図 4.7 耳診療録 A (ID:030910074514) 図 4.8 耳診療録 B (ID:030910074527)

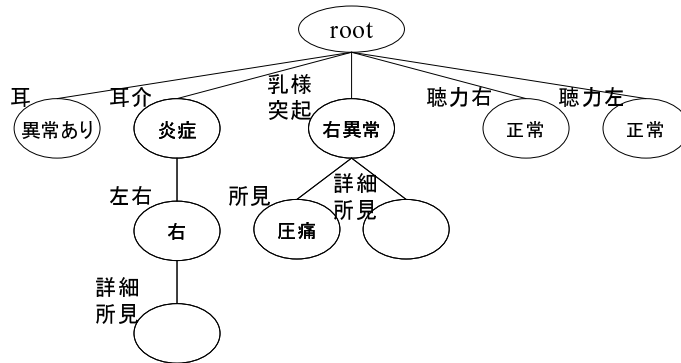


図 4.9 耳診療録 A の木構造データ

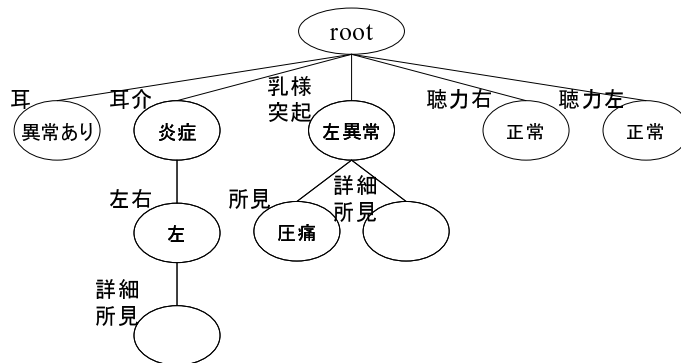


図 4.10 耳診療録 B の木構造データ

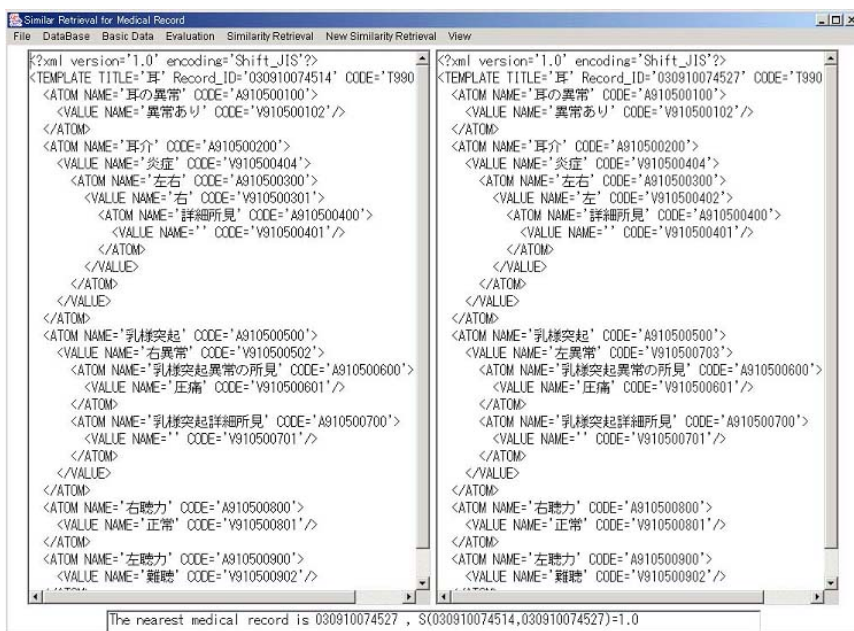


図 4.11 右, 左の区別をしないとき ($R(\text{右}, \text{左}) = 1.0$) の出力結果

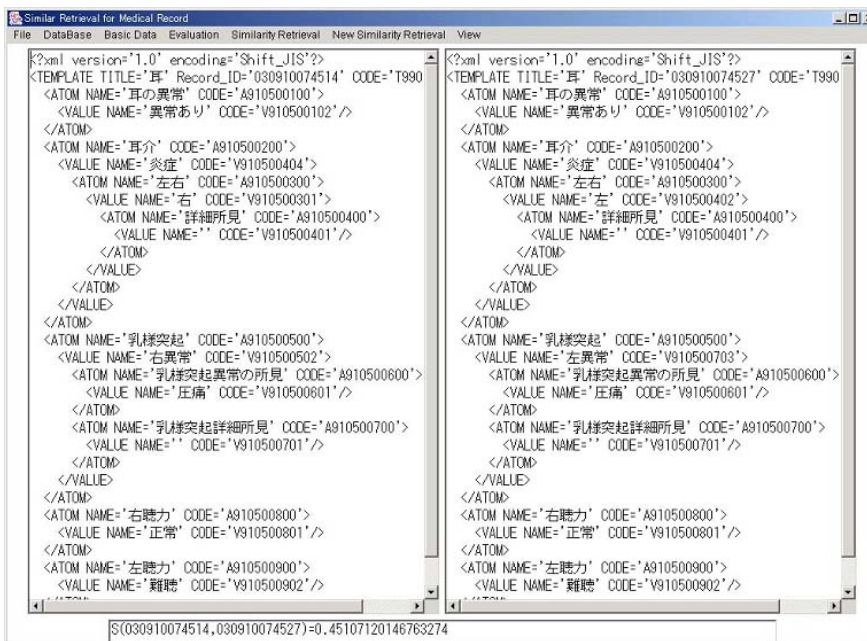


図 4.12 右, 左の区別をするとき ($R(\text{右}, \text{左}) = 0.0$) の出力結果

W を求めて、先の方法と同じように診療録 A を質問診療録として他の診療録について類似度を求めた。その結果が図 4.14 である。

図 4.13 と図 4.14 は、診療録 A を質問診療録としたときの類似検索結果であり、横軸に類似度、縦軸に類似度が同じである類似診療録の数を示しており、一つ一つのプロットが一つの診療録をさしている。図 4.13 は node に対する W を全て 1.0 とした為に、類似度が等しくなる診療録が多数存在し順位付けをすることができなかった。それに対し、図 4.14 は node に対する W を頻度により求めている為に、図 4.13 では類似度が等しくなり順位付けできなかった診療録も順位付けすることが可能であった。さらにその中でも希な診療録の類似度を上げることによって順位が上位にくるようにすることができた。

4.2.3 考察

シミュレーションデータを用いた 2 種類の実験により、Relationship Table、Weight Table それぞれの有効性を示すことができた。最初の Relationship Table における実験によって、ユーザーが求める類似診療録を取得するために、 R を用いた類似度 S の値を制御することにより、実現できることを示した。次の Weight Table における実験では、全ての node に対する $W = 1.0$ のときは類似度が同じ順位であった診療録が多数あったのに対して、頻度による計算式 (3.1) を用いて W を算出したときはそれらの診療録の類似度を分散させ、順位づけが可能になることを示した。この 2 点が本実験によって明らかになった。

Relationship Table と Weight Table を用いると、同じ node 内に対する value 間に対して類似度制御を行うことが可能である。さらに 3.7.3 節で説明した Combine を用いることにより、隣合う階層における node に関して、類似度に対する重要度を同じにすることができる。しかし、異なる node 内の value 間における類似度制御、または下位の node の類似度に対する重要度を上位の node よりも上げることは、本手法では不可能である。

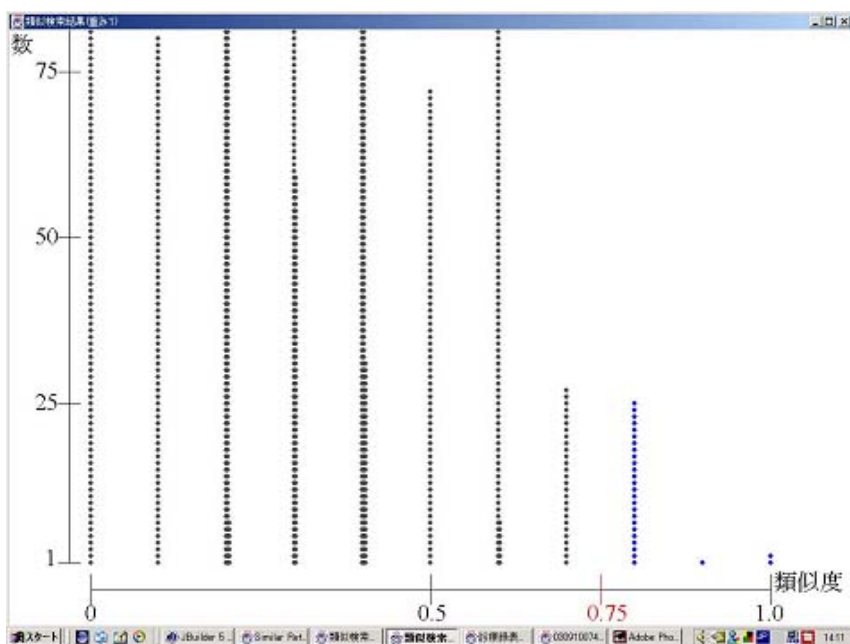


図 4.13 全ての W を 1.0 としたときの類似検索結果

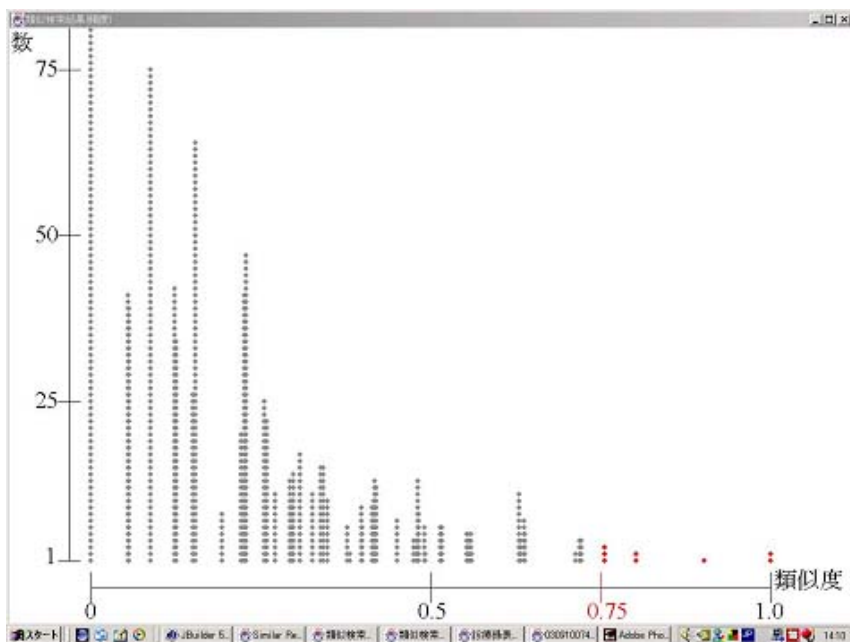


図 4.14 W を式 (3.1) を用いて求めたときの類似検索結果

4.3 上部消化管造影検査レポートを用いた実験

上部消化管造影検査レポートという実データを用いた実験について述べる。この実験により、システムの出力結果と医師の類似判断との近さについて評価する。

4.3.1 実験準備

上部消化管とは、食道、胃、十二指腸のことをさす。上部消化管造影検査とは、発泡剤を飲み上部消化管を膨らませた後に、バリウムを飲んで上部消化管それぞれの粘膜の表面にバリウムを付着させ、X線透視下に消化管の中を映し出すことによって、癌やポリープ、潰瘍、その他の様々な消化管病変の発見のために行う検査である [25]。本実験で扱う放射線レポートであるが、食道、胃、十二指腸それぞれに関して、放射線科の読影医による読影の所見と診断が記載されている。この放射線レポートは自然言語の文章でそれぞれ記載されている。

4.3.2 実験結果

4.3.1 節で述べた放射線レポートを木構造診療録のデータ形式に変換する為に、大阪大学大学院医学系研究科の大村他が開発した上部消化管レポートに関する Dynamic Template の雛形 [26] を用いて入力を行い、木構造診療録を完成させた。本実験では、この上部消化管の中から胃の所見のみを扱う。図 4.15 は、本実験で扱う胃の木構造診療録データである。第 1 階層に「異常あり」「異常なし」といった最も所見において重要な記述があり、第 2 階層には読影においてポイントとなる「粘膜」「形態」「機能」という node をもち、第 3 階層にはその症状が現れている部位、第 4 階層にはさらに詳細な部位といったように、木構造診療録データの特徴を表したものとなっている。

実験のデータは、大阪大学医学部附属病院放射線科で行われた上部消化管検査のレポート 100 枚を用いる。

実験方法であるが、Medical Record Database に格納した 100 枚の放射線レポート (診療録 ID:1 ~ 100) の中の特定の 9 枚のレポート (ID26,27,34,38,50,63,75,80,86) をそれぞれ質問診療録とし、他の 99 枚のレポートと症状・所見が類似しているか

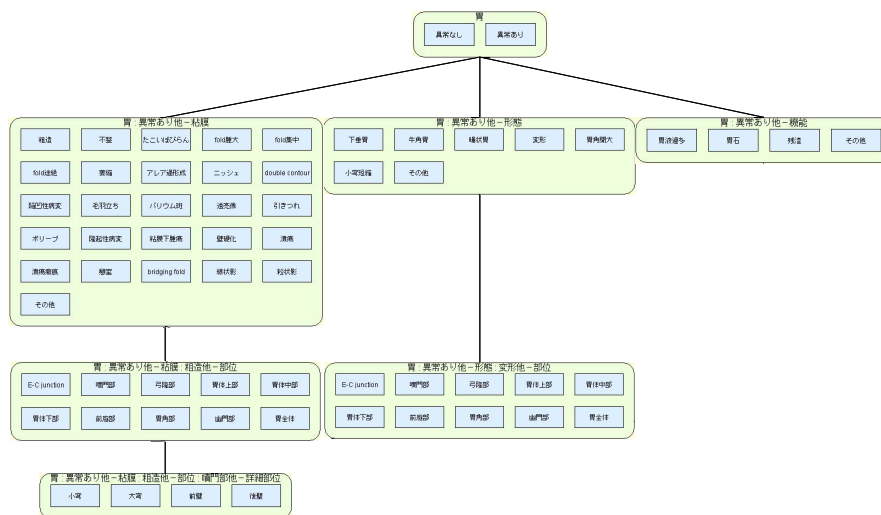


図 4.15 胃の木構造診療録データ

どうかの評価実験を行う。実験には、大阪大学医学部附属病院医療情報部の3人の医師に対して、それぞれ質問診療録三つずつの評価実験を行った。評価実験の方法は、よく類似している場合は4、少し類似している場合は3、あまり類似していない場合は2、全く類似していない場合は1というように4段階の分類(以下、カテゴリー)による評価を依頼した。各カテゴリーの数については、診療録ごとの内容の差、そして評価実験を行う医師の個人差もあるため指定しなかった。

100個の診療録から九つの質問診療録の選択の仕方であるが、読影医診断に胃炎と胃潰瘍の二つのみがあるものを選択した。これらのデータや方法を基に、二つの実験を行った。一つ目の実験では、システム出力結果である検索順位の各上位診療録数を1~99まで変えたときの医師の類似判断を解とした適合率をそれぞれ求めた。二つ目の実験では、出力結果の検索順位と医師の類似判断に分けられた4つのカテゴリーを比較するために類似順位の差を計算することによる定量的評価を行った。それぞれ以下に述べる。

適合率を求める実験

適合率を求める実験について述べる．適合率とは，実際に検索された情報のうち，解と一致している割合である．次の図 4.16 ~ 図 4.25 までが九つの診療録における類似検索結果である．横軸がシステムによる検索結果の上位採用診療録数であり，縦軸が医師の類似判断を解とした適合率である．適合率 P の計算は以下のように行う．

$$P(num) = \frac{\text{検索結果における解の数}}{\text{検索結果上位診療録数}} \times 100 \quad (4.1)$$

num は質問診療録 ID である．

解となる医師の類似性の判断であるが，基本的にカテゴリ 4 と 3 が，医師が症状・所見が類似していると判断した場合であるため，この二つの医師判断をもつ診療録が解となる類似診療録であるが，その二つ足した診療録数が 20 以下と少ない場合，カテゴリ 2 と判断した診療録についても解として加算することを行った．今回の実験では，診療録 ID38,50,86 について，解が 20 個以下と少なかったために，カテゴリ 2 についても解として加算している． R は，3 人のうちの 1 人が入力を行った． R は 98 個の組み合わせについて類似度を定義し，Relationship Table に入力を行っている．今回定義した Relationship Table は，付録 B に記す．定義していない組み合わせに関しては，類似度は 0.0 である． R_{min} は $R_{min}(1) = R_{min}(2) = 0.0$ ， $R_{min}(3) = 0.5$ ， $R_{min}(4) = 0.8$ と定めた．Weight Table に格納する頻度は，第 2 階層までを付録 C に記す．

図 4.16 ~ 図 4.24 は，以下の 4 通りの方法で行った検索結果に対し，適合率 P をそれぞれ求め，グラフにしたものである．

1. 本手法による検索結果
2. 全ての node に対する W を 1.0 としたときの検索結果
3. 全数一致検索結果
4. ランダム検索結果

全数一致検索結果であるが、これはシステム出力結果の上位に医師が類似していると判断した診療録が全て1位から順に並んだ結果を仮定したものである。ランダム検索結果は、システム出力結果をランダムに変えたものに対して、それぞれの適合率 P を求める作業を1000回繰り返し、平均したものである。このランダム検索結果は、やみくもに一つ選んだ場合の適合率となるために、本手法の結果と医師の類似判断結果との近さを知るための目安となる。

全数一致検索結果は、検索結果上位診療録数が、医師が類似していると判断した診療録数になるまで適合率は100%になり、それ以降は適合率が下がる結果になった。ランダム検索結果では、各適合率が、類似判断診療録数を検索対象となる診療録総数99を割った確率の値でほぼ一定となった。そして、本手法による類似検索結果に関しては、検索上位診療録数を10としたとき、九つの質問診療録中、ID26,63を除く七つの診療録でランダム検索結果よりも適合率が上回る結果となった。さらに本手法の検索結果と $W = 1.0$ のときの検索結果との比較であるが、検索上位診療録数を10としたとき、九つの診療録中、ID26,27,63,75,80の五つの診療録で適合率が本手法の方が上回った。ID34,38,86については適合率は同じであり、ID50については $W = 1.0$ の方が上回った。これらの結果に対する考察については、4.3.3節で述べる。

同じ適合率を求める方法を用いて、システム出力結果と読影医診断結果との比較、また読影医診断結果と医師の類似判断結果との比較のために、次の実験を行った。今回の九つの質問診療録は、胃炎、胃潰瘍のみが読影医診断結果となるものを用いたため、解となる読影医診断結果は、「胃炎」、「胃潰瘍」、「胃炎 AND 胃潰瘍」、「胃炎 OR 胃潰瘍」の4通り考えられる。それぞれについて、システム出力結果との比較を行うため、さらに読影医診断結果と医師類似判断結果との比較を行うために、先の方法のようにグラフ化を行った。結果は、付録Dに載せる。読影医診断結果とシステム出力結果との比較であるが、検索結果上位診療録数10としたとき、九つの診療録の平均適合率は「胃炎」に関しては72%、「胃潰瘍」に関しては53%、「胃炎 AND 胃潰瘍」に関しては49%、「胃炎 OR 胃潰瘍」に関しては77%であった。そして、読影医診断結果と医師類似判断結果との比較であるが、付録Dより、ほぼ全ての診療録に対してよく似た適合率変化を示した。これに関

しても 4.3.3 節の考察で述べる .

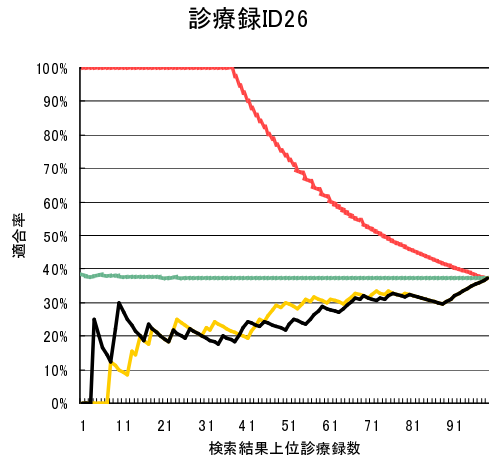


図 4.16 診療録 ID26 の類似検索結果

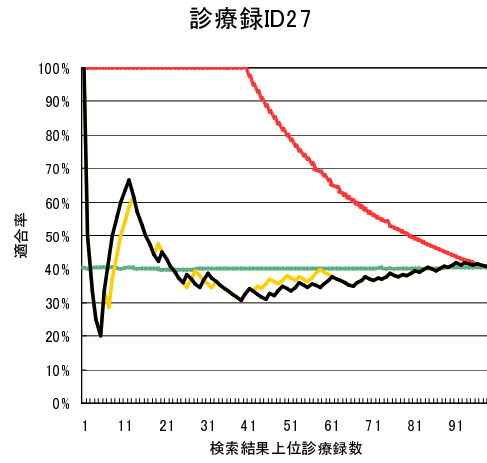


図 4.17 診療録 ID27 の類似検索結果

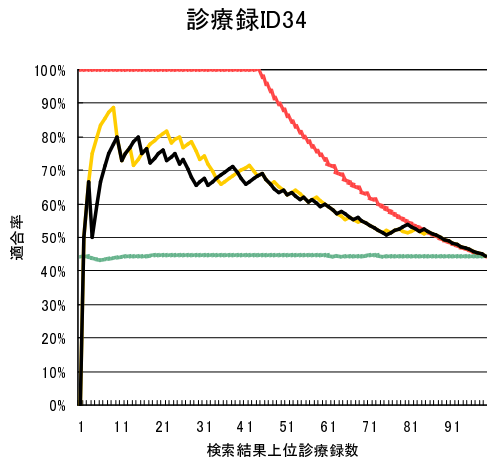


図 4.18 診療録 ID34 の類似検索結果

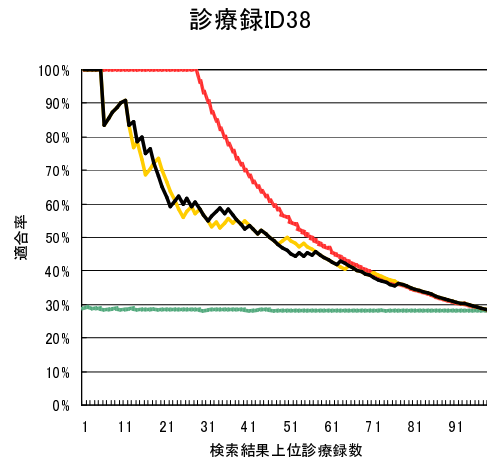


図 4.19 診療録 ID38 の類似検索結果

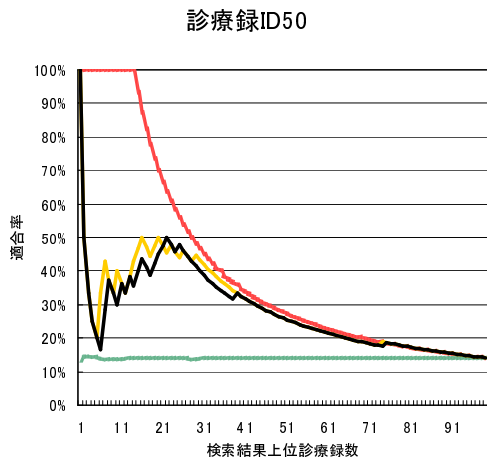


図 4.20 診療録 ID50 の類似検索結果

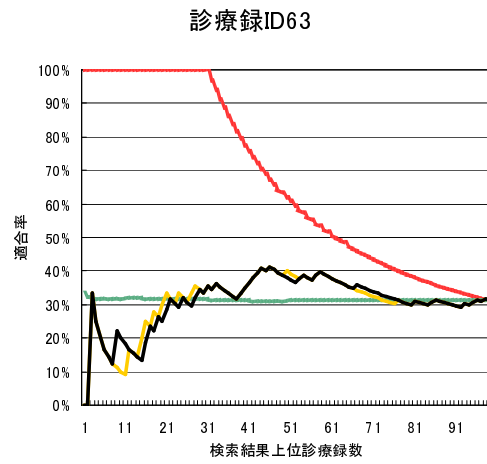


図 4.21 診療録 ID63 の類似検索結果

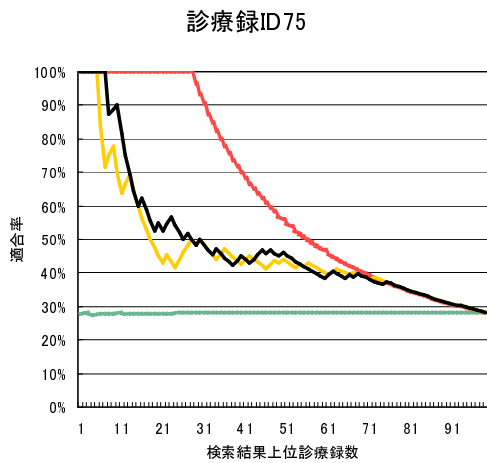


図 4.22 診療録 ID75 の類似検索結果

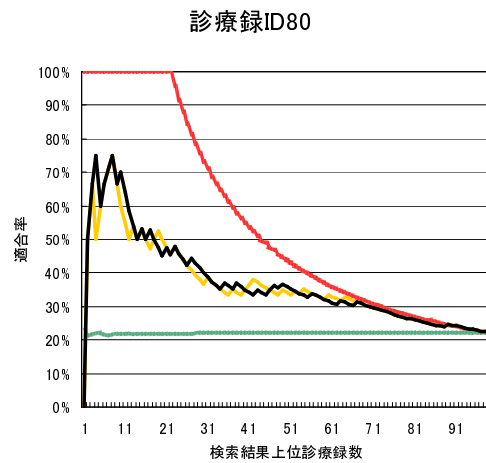


図 4.23 診療録 ID80 の類似検索結果

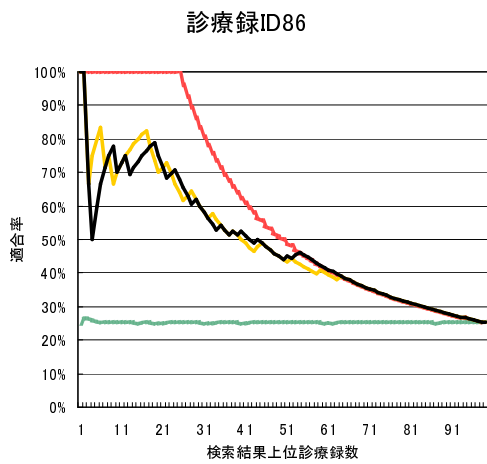


図 4.24 診療録 ID86 の類似検索結果

- 木構造類似検索
- 木構造類似検索(W=1.0)
- 全数一致検索結果
- ランダム検索結果

図 4.25 図 4.16 ~ 図 4.24 の凡例

類似順位の差を求める実験

次にシステム出力結果である各診療録の検索順位と医師の類似判断との比較を行う。検索順位と医師の類似判断の4つのカテゴリーとの比較を行う為には、医師の類似判断結果を順位づけする必要がある。そこでカテゴリーを類似しているものから順に診療録を並び替えたもの(4,3,2,1)が医師の類似判断の順位であるとする。この二つの順位の差を比較するために、次の手法によって、システム出力結果と医師の類似判断との差を定量的に求めた。

1. 各診療録を4つの医師判断のカテゴリーに分類する
2. それぞれのカテゴリー内の各診療録に対して、システム出力結果である検索順位を求め、カテゴリーごとに平均することによってシステムカテゴリー順位を求める。
3. 各カテゴリーにおける医師類似判断順位を求める。医師類似判断順位は、類似性の高いカテゴリー4から順に並べたときに対応する範囲の順位の平均である。
4. 各カテゴリーごとに医師類似判断順位とシステムカテゴリー順位の差の絶対値とカテゴリー内の診療録数を乗算する。この値が、一つのカテゴリーに対するシステム出力結果と医師の類似判断との差となる。
5. この各カテゴリーの差の総和が、求める差となる。

図4.26, 4.27は、システム出力結果と医師の類似判断との差を示した1例である。これらの図は質問診療録ID26に対して行った結果の図である。そのほかの結果については、付録Eに載せる。図4.26は、各カテゴリーごとにシステムカテゴリー順位とカテゴリー内の診療録数を示した棒グラフである。左から類似性のあると判断されたカテゴリー4, 3, 2, 1と順に並んでいる。棒の幅は、カテゴリー内の診療録の総数である。そして、棒の高さは各カテゴリーのシステムカテゴリー順位を表している。図4.27は、図4.26に医師類似判断順位の結果をのせ、その結果と医師の類似判断との差を示している。斜線の箇所がその差を示してい

る．図に示している $y = x$ の直線が各カテゴリーの中央に来るときが，そのカテゴリーに対する医師類似判断順位である．斜線で示された長方形の面積の総和が求める差となる．

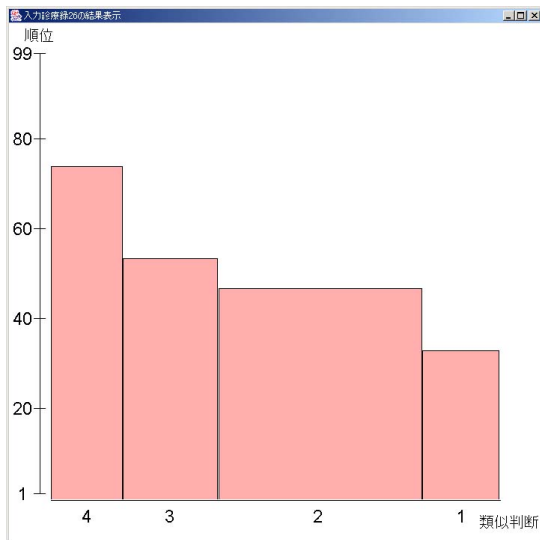


図 4.26 医師判断カテゴリー順位の例

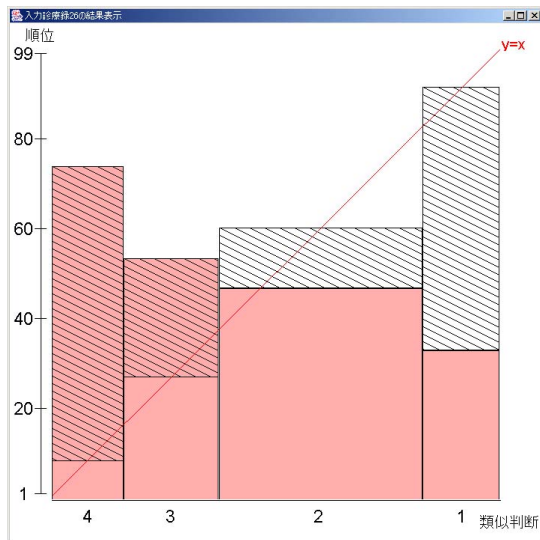


図 4.27 医師判断との差の例

この方法により，先の実験で用いた質問診療録九つそれぞれに対して差を求めた．ランダムに診療録の順位付けを行うシステムがあったと仮定したとき，それらの各カテゴリーのシステムカテゴリー順位は 50 となる．この結果を目安として，本手法の結果と医師の類似判断結果との近さを評価することができる．このシステム出力結果とランダム検索結果，それぞれの医師の類似判断との差を示した結果が，表 4.4 である．

表 4.4 より，質問診療録九つのうち，ID26，27 を除く七つの診療録においてシステム出力結果のほうが，ランダム検索結果より差が少ないという結果が得られた．考察は，4.3.3 節で述べる．

表 4.4 システム出力結果と医師判断との定量的比較

質問診療録 ID	システム出力結果	ランダム検索結果
26	3178	2294
27	2592	2378
34	1184	2420
38	650	1988
50	374	1190
63	1950	2144
75	902	1988
80	846	1944
86	452	1850

4.3.3 考察

先の三つの実験結果についての考察を述べる。

適合率を求める実験，差を定量的に求める実験，どちらも九つの質問診療録において，七つの診療録でシステム出力結果がランダム検索結果よりも，適合率は上回り，医師判断との差は小さかったことから，本手法は医師の症状・所見の類似検索結果と近い類似診療録を検索することができたと考えられる。しかし，どちらの実験においても2例の診療録についてはランダム検索結果よりも適合率は低く，差は大きかった。このように特定の診療録について，ランダム検索結果のほうがシステム出力結果よりも，医師の類似判断結果と近くなった原因としては，以下のことが考えられる。

1. 放射線科医以外の Medical Record Database , Relationship Table の入力
2. 類似判断を行った医師の問題
3. 診療録自体の特徴がない点
4. 木構造診療録では表現できない点

5. 本手法の問題点

各データベース・関連テーブルへの入力であるが、Medical Record Database, Relationship Table はそれぞれ放射線科医ではなく著者または専門外の医師が入力した。したがって、専門医からみて不適当な診療録になっていた可能性は若干存在する。この点が原因である可能性がある。

次に医師自身の問題であるが、三つの質問診療録に対して 99 個の診療録の類似判断を行うのに今回は約 1 時間半かかった。よって疲労による判断ミスもあったと考えられる。また、医師の個人差も考えられる。今回は、3 人の医師に評価実験を依頼したが、それぞれの医師によってシステム出力結果と類似していたり、類似していなかったりしたことから、3 人の医師の考える各項目に対する類似性評価の重みが異なっていたことが考えられる。さらに、実際に時間を誰よりも多くかけて、実験に協力してくれた医師が行った診療録 (ID38, 50, 86) に関する適合率は他の医師が行った診療録に比べ大きく、類似順位の差は小さかったことから、医師個人の問題であると考えられる。

さらに診療録自体の特徴のなさであるが、それぞれの実験においてランダム検索結果を下回った ID26,27,63 に関して、一般的によく見られる症状のみしかもっておらず、数も少ない。よって、医師がほかの診療録と見比べて類似判断をするのも大変困難であったと考えられる。それによりシステム側との差が生じたことが原因であると考えられる。

また、木構造診療録では表現できない点であるが、今回用いた Dynamic Template の雛形にはそれぞれの特徴の程度の表現（少し、大きいなど）が入力できなかった。これが原因で自然言語のレポートとに差が生じたのではないかと考えられる。

最後に本手法の問題点が挙げられる。本手法では、検索対象木構造を質問診療録のみで構成したために、被検索データにたくさん特徴があると一致する確率が高くなる。医師が類似性を判断するとき、余分な特徴が多いと類似性評価は下がる。その差がこの実験によって明らかになったと考えている。この原因をよく示しているものが、付録 D の読影医診断結果と医師類似判断結果の適合率変化が、ほぼ同じになったという事実である。適合率変化が等しいということは、同じ診

療録に対して適合率が増減していると考えられる。そこで、読影医診断も医師類似判断も適合率が減少している診療録を調べると、読影医診断結果が胃炎や胃潰瘍ではなく、更に状態の悪化した胃癌の診療録が多くみられた。よって読影医診断結果の適合率が減少したと考えられる。そして、このような胃癌の診療録では、読影における症状の数が多く見られる。このことから、医師は余分な特徴の多い診療録に対して、類似性評価を下げたことが推測される。これらの医師が類似していないと判断したにもかかわらず、システム出力結果として上位に順位付けられた診療録に対しては、自らの経験・価値観などにとらわれずに出力されたものであることが言える。具体的にそれらの診療録は、症状の数が多いため重度の病気であることが多く、診療録 A の類似度計算対象となる症状からだけみると、客観的に類似している。これらの診療録も医師にとってみれば、必要な情報源として活用することができるのである。

次に行った実験の結果である全ての node に対する W を 1.0 にした検索結果よりも、本手法を用いた方が九つの診療録中、五つの診療録で適合率が高かった点について述べる。このような結果が得られた原因としては、頻度の高い症状は、軽度の症状であることが多い点に尽きる。実際に、その五つの診療録には、胃の粘膜の症状でよく見られる「粗造」が現れている。この「粗造」は、100 個の診療録中 44 個の診療録で現れており、あまり重要視されない軽度の症状である。このように頻度の高い軽度の症状に対して、類似性の重要度を下げる仕組みを W を用いれば可能であることが示された。

この上部消化管造影検査レポートを用いた評価実験によって明らかになった本手法の特徴は以下の 4 点である。

1. ランダム検索結果より医師類似判断に近い結果を与える
2. 質問診療録の特徴が少なければ、医師類似判断に近い結果を得るのは困難である
3. 被検索対象診療録の特徴が多ければ、検索される確率が高くなる
4. 頻度の高い軽度の症状は、頻度の低い重度の症状に比べ、類似検索に影響を及ぼさない

5. 結論

本研究では、木構造診療録に対して、症状・所見による木構造診療録類似検索手法を開発し、その手法を用いて木構造診療録類似検索システムを実装し、三つのデータ(運動負荷心電図ST部分データ、耳診療録シミュレーションデータ、上部消化管造影検査リポート)を用いた実験によりシステムを評価した。このような類似性を評価する手法を開発する為には、対象となるデータの特徴を正確に捉える必要がある。本手法では、木構造診療録の「下位 node は上位 node の詳細所見」という特徴を生かした手法を開発した。木構造診療録の類似性を判断する場合は、詳細所見よりもその根本の所見に対する類似性を重要視しなければならない。それは、Tree Similarity Retrieval によって下位の node を root とした部分木における類似度計算を行い、それらを加重平均し、最後に診療録全体の類似度を求めることにより実現できる。さらに木構造診療録の特徴であるカテゴリーデータ、順位データに対しては、Relationship Table を用いることにより各 value 間の類似度を定義することによって、類似度を制御することができる。これについては、耳診療録シミュレーションデータを用いた実験により証明することができた。

医療に限らず様々な分野のデータの特徴でもある「希な症状が見られる診療録を重要視する」点に関しては、Weight Table を用いて、それを Tree Similarity Retrieval の計算式に組み込むことによって実現した。医療の分野において、この頻度の多い症状をもつ診療録は、医師の類似判断を左右するような大きな特徴をもつデータではない可能性が高く、重要視しない本手法は効果的であったと考えられる。これは上部消化管リポートを用いた実験によって証明した。

また、Tree Similarity Retrieval における問題点であった「部分的な探索」「検索対象に入れない node」「従属関係に対する制御」にはそれぞれ Part Search, Ignore, Combine を設け、それらが簡単に入力できるように GUI インタフェースも開発した。

評価としては、第一に、上部消化管造影検査リポートを用いた実験によって、ランダム検索結果よりも適合率は高く、医師の類似判断とは差の少ない結果を得られたことから、医師の判断と近い結果を得られたことが分かる。

そして、第二に、医師の類似判断とシステム出力結果の上位が異なるケースに

対して、質問診療録の症状だけを考えると類似していることから、医師の経験・価値観にとらわれずに出力されたために結果として生じたといえる。このことから、このシステムはセカンドオピニオンとしても臨床的に利用できると考える。現システムの問題点として、以下の点が挙げられる。

1. Relationship Table への入力の手間
2. 症状が多くみられる診療録が検索されやすい点

Relationship Table へ入力する R は、一つの node がもつ value の全組み合わせに対して定義しなければ、最適な結果を得ることはできない。これは入力するユーザーにとっては非常に手間のかかる作業である。これを改善するためには、入力を簡単にする手法が必要であると考えられる。例えば、各 value を表現できるような特徴を軸とした 1 次元、または 2 次元の空間内に各 value をプロットするだけで R を入力することができるようなユーザーの負担を軽減することのできる仕組みが必要である。

次に 4.3.3 節でも述べた症状の多い診療録が検索されやすいという問題である。検索としては普通のことではあるが、問題となるようであれば解決していかなければならない。例えば、類似診療録側からみて、非適合率のような適合していない比率等を調べることにより、対処していかなければならないと考えられる。

将来的に医療現場にてこのシステムが用いられるために、質問診療録と類似診療録の違いが重要なものであるかどうかを判断し、重要であればその類似診療録を破棄し、重要なものでなければ現在の問題に適合する形に類似診療録を修正するような機能が必要であると考えられる。

実現に向けての課題は、医療の現場にもある。電子カルテの中の診療録データを格納するデータウェアハウスは、まだ全国の病院で普及しているわけではない。この問題点に関しては、この医療情報に携わっている研究者が世間に有効性をアピールしていく必要がある。それらの環境が構築されたとしても問題は残る。それぞれの電子カルテは、データ形式、表現が異なっているため、診療録データを用いた診療支援はやはりシステムに依存する面がどうしても出てくる。MML, HL7 などデータ交換のための標準化言語は、近年、普及してきてはいるが、あくまで

も交換のための言語であり，診療録データの再利用のための言語として用いるのは難しい．データベースに格納されているこれら貴重な情報という資源を最大限に利用する為には，より効果的な検索手法と統一した言語の開発が必要である．

謝辞

本研究は、本当に多くの方々のお力添えにより実現することができました。研究に対して助言、激励を著者にくださった皆様、学生生活を充実したものにしてくださいました皆様、心から厚く御礼申し上げます。

奈良先端科学技術大学院大学 情報科学研究科 湊 小太郎教授には、研究活動の環境を整えていただき、適切なお助言、ご指導、時には叱咤激励を頂きましたことに心より深く感謝いたします。さらに、ゼミナール発表時において、懇切丁寧なお教示、ご助言をいただきました、植村 俊亮教授、杉浦 忠男助教授に心から感謝いたします。また、ミーティングを通し、数々のご助言、的確なお指摘をしていただいた、菅 幹生助手、研究のみでなく学内の生活においても著者に大変気を配っていただいた、佐藤 哲大助手に心から感謝いたします。

本研究の機会ばかりか、細部にわたる熱心なお指導、懇切丁寧なお助言をいただきました、大阪大学医学部附属病院 医療情報部 松村 泰志助教授に心より深く感謝いたします。評価実験に関して、ご協力していただけただけでなく、視点の違った方向からみて、有意義なお意見をいただいた、大阪大学医学部附属病院 医療情報部 寺谷 禎真先生、峯野 隆広先生に心から感謝いたします。その他の大阪大学医学部附属病院 医療情報部の皆様、著者の発表に耳を傾けていただき、本当にありがとうございました。実験の材料となる Dynamic Template の資料を快く提供していただいた、大阪大学大学院医学系研究科の大村 久美子様には、心より感謝いたします。ありがとうございました。

そして、生命機能計測学分野の辰巳 亮子様、島田 秀輝氏、阿賀 俊幸氏、石田 実里様、春名 かおり様をはじめとする研究室の皆様方、学内での生活を充実したものにできたのは、ありとあらゆる面で著者を支えてくださった皆様のおかげです。本当にありがとうございました。

最後に著者のわがママを温かく見守ってくれた家族に深く感謝いたします。

参考文献

- [1] 矢崎義雄, 武田 裕他. ITvision (特集電子カルテは医療をどう変えるか). No. 1. インナービジョン, 2003.
- [2] 小林祥泰. 電子カルテの現状 - 電子カルテと臨床研究用データベースの統合について. 標準的内科診療録 <http://www.naika.or.jp/sinryoroku/>, 1999.
- [3] 日本病院会・医療事故対策委員会, 2001.
- [4] 小原清弘, 高林克日己, 神山卓也, 松尾仁司. 外来診療におけるテンプレート入力電子カルテと紙カルテとの比較評価. 医療情報学, Vol. 22, No. 1, pp. 95-102, 2002.
- [5] Yasushi Matsumura and Hiroshi Takeda. Devices for Structured Data Entry in Electronic Patient Record. *Medinfo98*, pp. 85-88, 1998.
- [6] 山崎俊司, 鈴木隆弘, 本多正幸, 里村洋一. テンプレートと構造データ入力機構を備えた電子カルテ. 電子カルテシンポジウム論文集, pp. 17-20, 1996.
- [7] 岡田修, 大星直樹. クリニカルパスのXML化による薬剤費の医療施設間定量的相互比較評価の試み. 医療情報学, Vol. 23, No. 2, pp. 145-151, 2003.
- [8] 遠藤晃, 湊小太郎, 大草知裕, 小森優, 井上裕二, 永田啓, 高橋隆. 処方入力ヒューマン・インターフェースの定量的相互比較. 医療情報学, Vol. 14, No. 2, pp. 45-55, 1994.
- [9] 波多野賢治, 絹谷弘子, 吉川正俊, 植村俊亮. XML文書検索のための検索結果粒度決定. 第14回データ工学ワークショップ (DEWS2003) 論文集, 2003.
- [10] 小林重信. 「特集事例ベース推論」事例ベース推論の現状と展望. 人工知能学会誌, Vol. 7, No. 4, pp. 559-565, 1992.
- [11] T. Kobayashi. CBR world. Creative Mind in Digital Life <http://homepage2.nifty.com/digitallife/>, 2002.

- [12] A. Aamodt. Case-based reasoning Foundation issues. *AICOM 7*, pp. 39–59, 1994.
- [13] R. Schmidt and L. Gierl. Case-based Reasoning for Medical Knowledge-based Systems. *International Journal of Medical Informatics*, No. 64, pp. 355–367, 2001.
- [14] K.J. Hammond. CHEF: A Model of Case Based Planning. *AAAI-86*, pp. 267–271, 1986.
- [15] K.D. Ashley. Reasoning with cases and hypotheticals in HYPO. *Int.J.Man-Machine Studies 34*, pp. 753–796, 1991.
- [16] P. Koton. Reasoning about Evidence in Causal Explanations. *Proceeding of the DARPA Case-Based Reasoning Workshop*, pp. 273–297, 1988.
- [17] C. Bradburn and J. Zeleznikow. The application of case-based reasoning to the tasksof health care planning. *Proc of European Workshop on CBR*, pp. 365–378, 1993.
- [18] R. Bareiss B.W. Porter and R.C. Holte. Concept Learning and Heuristic Classification in Weak Theory Domains. *Artificial Intelligence 45*, pp. 229–263, 1990.
- [19] 藤村滋, 松村真宏, 岡崎直観, 石塚満. 電子掲示板上の評判情報に基づく意思決定支援. 第17回人工知能学会全国大会論文集, 2003.
- [20] A. Brooder. Strategies for efficient incremental nearest neighbor search. *Pattern Recognition 23*, pp. 171–178, 1990.
- [21] 島津秀雄. 農林水産業の高度情報システム - 事例ベース. 増殖情報ナビゲータ <http://agrinfo.narc.affrc.go.jp/>, 1996.
- [22] 浅井達哉. 半構造データマイニングに関する研究動向. 第14回データ工学ワークショップ (DEWS2003) ミニサーベイ, 2003.

- [23] 松村泰志. 電子カルテの現状 - 構造化データの登録と必要性と実現法. 標準的内科診療録 <http://www.naika.or.jp/sinryoroku/>, 1999.
- [24] 高橋大曜. 診療データベースからの診断ルールの自動生成アルゴリズムの開発, 1999.
- [25] 白壁彦夫. 腹部 X 線読影テキスト 1. 文光堂, 1984.
- [26] 大村久美子. 集団検診のための所見入力ツールの開発-上部消化管造影検査について-, 2004.

付録

A. 本研究で扱うデータの構造

ここでは、木構造診療録データと Medical Record Database, Relationship Table, Weight Table の三つのスキーマについて説明する。

大阪大学医学部附属病院で用いられている電子カルテ入力システム Dynamic Template で作られる診療録は、XML の書式で出力される。しかし、その XML 文書には、本システムにとって不必要なデータが多々存在している。データ量が多いとシステム処理の遅延の原因となるため、必要なデータのみを抽出する対策が必要である。そこで本システムでは、その XML 文書から必要な情報だけを抽出したシンプルな XML 文書を新たに作成した。その DTD は、次の通りである。

```
<!ELEMENT TEMPLATE(atom+)>
<!ELEMENT atom(value+)>
<!ELEMENT value(atom+)>
```

atom が本論文中での node を示す。XML を操作する為の XML パーサとしては、Electric XML+TM(The Mind Electric 社) を用いた。本システムでは、このシンプルな XML 文書を分解し、リレーショナルデータベースに格納している。

次に Medical Record Database, Relationship Table, Weight Table それぞれのスキーマについて説明する。診療録, Dynamic Template のテンプレート, node, value には ID が与えられており、それを利用することによってデータの格納, 取得を行う。木構造診療録は、XML 文書であるために一つ一つの value には、位置情報を持っている。そのために、節 3.4.2 でも述べたが、parent node を格納しなければならない。これは、Medical Record Database の Selected と value の関連テーブル, Weight Table で行っている。

Medical Record Database

Medical Record Database は , record , selected , first atom , value , tree name の五つの関連テーブルで構成されている . record と selected は , 診療録の症状・所見を格納する関連テーブルであり , first atom , value , tree name は , テンプレートの情報が格納されている . record には , 診療録の ID とテンプレートの ID が格納されている . selected には , 診療録の ID と , その診療録がどの value を選択したかの情報を示す関連テーブルである . 一つの value の情報は , atom ID , value ID , parent atom ID , parent value ID の 4 つの項目で示している . first atom は , 各テンプレートの第 1 階層の node を示している . value は , 各テンプレートの全ての value の情報が格納されている . tree name は , ID と名前が対で格納されている . スキーマは以下の通りである .

表 A.1 Record

Field	record ID	template ID
Type	char(12)	char(10)

表 A.2 Selected

Field	record ID	atom ID	value ID	parent atom ID	parent value ID
Type	char(12)	char(10)	char(10)	char(10)	char(10)

表 A.3 First Atom

Field	template ID	first atom ID
Type	char(12)	char(10)

表 A.4 value

Field	template ID	atom ID	value ID	parent atom ID	parent value ID
Type	char(12)	char(10)	char(10)	char(10)	char(10)

表 A.5 Tree Name

Field	template ID	ID	name
Type	char(12)	char(10)	char(20)

Relationship Table

Relationship Table には，node 内の value の全組み合わせに対して，一つ格納される．value1，value2 に，組み合わせの value ID をそれぞれ格納し，similarity に類似度を格納する．何も定義されていない場合は，similarity は 0.0 となる．

表 A.6 Relationship Table

Field	value1 ID	value2 ID	similarity
Type	char(10)	char(10)	int(4)

Weight Table

Weight Table は，テンプレートの全ての value に対して，頻度の情報が格納されている．テンプレートの情報，value の情報，そしてその value の使用回数が格納されている．表 A.7 では，frequency に使用回数を格納する．

表 A.7 Weight Table

Field	template ID	atom ID	value ID	parent atom ID	parent value ID	frequency
Type	char(12)	char(10)	char(10)	char(10)	char(10)	int(5)

B. 胃の所見における Relationship Table

表 B.1: 粘膜における Relationship Table

value1	value2	similarity
粗造	不整	0.8
粗造	たこいぼびらん	0.8
粗造	萎縮	0.5
粗造	毛羽立ち	0.2
粗造	バリウム斑	0.5
粗造	引きつけ	0.1
粗造	壁硬化	0.2
粗造	粒状影	0.2
不整	たこいぼびらん	0.7
不整	萎縮	0.7
不整	ニツシエ	0.4
不整	バリウム斑	0.4
不整	引きつけ	0.2
不整	壁硬化	0.3
不整	潰瘍	0.2
不整	潰瘍瘢痕	0.3
たこいぼびらん	萎縮	0.3
たこいぼびらん	陥凹性病変	0.3
fold 腫大	fold 集中	0.7
fold 腫大	fold 途絶	0.7
fold 腫大	アレア過形成	0.5
fold 腫大	ニツシエ	0.2
fold 腫大	透亮像	0.2
fold 腫大	ポリープ	0.2

表 B.1: 粘膜における Relationship Table

value1	value2	similarity
fold 腫大	隆起性病変	0.8
fold 腫大	粘膜下腫瘍	0.5
fold 腫大	潰瘍瘢痕	0.2
fold 集中	fold 途絶	0.7
fold 集中	アレア過形成	0.5
fold 集中	ニツシエ	0.3
fold 集中	透亮像	0.3
fold 集中	ポリープ	0.3
fold 集中	隆起性病変	0.5
fold 集中	粘膜下腫瘍	0.5
fold 集中	潰瘍	0.7
fold 集中	潰瘍瘢痕	0.7
fold 集中	憩室	0.2
fold 集中	線状影	0.2
fold 途絶	アレア過形成	0.5
fold 途絶	ニツシエ	0.3
萎縮	陥凹性病変	0.5
萎縮	毛羽立ち	0.5
萎縮	バリウム斑	0.5
萎縮	壁硬化	0.2
萎縮	粒状影	0.2
アレア過形成	陥凹性病変	0.3
アレア過形成	透亮像	0.5
アレア過形成	ポリープ	0.5
アレア過形成	隆起性病変	0.6

表 B.1: 粘膜における Relationship Table

value1	value2	similarity
アレア過形成	粘膜下腫瘍	0.5
ニッシェ	陥凹性病変	0.8
ニッシェ	引きつけ	0.7
ニッシェ	潰瘍	0.8
ニッシェ	潰瘍瘢痕	0.9
ニッシェ	線状影	0.7
陥凹性病変	潰瘍	0.9
陥凹性病変	潰瘍瘢痕	0.8
陥凹性病変	憩室	0.9
透亮像	ポリープ	0.8
透亮像	隆起性病変	0.8
透亮像	粘膜下腫瘍	0.8
引きつけ	潰瘍瘢痕	0.9
引きつけ	線状影	0.3
ポリープ	隆起性病変	0.9
ポリープ	粘膜下腫瘍	0.3
隆起性病変	粘膜下腫瘍	0.9
潰瘍	潰瘍瘢痕	0.9
潰瘍	憩室	0.5
潰瘍	線状影	0.8
潰瘍瘢痕	憩室	0.2
潰瘍瘢痕	線状影	0.6

表 B.2: 形態における Relationship Table

value1	value2	similarity
下垂胃	変形	0.6
曝状胃	変形	0.6

表 B.3: 機能における Relationship Table

value1	value2	similarity
胃石	残渣	0.8

表 B.4: 部位における Relationship Table

value1	value2	similarity
E-C junction	噴門部	0.9
E-C junction	弓隆部	0.6
噴門部	弓隆部	0.5
弓隆部	胃体上部	0.7
胃体上部	胃体中部	0.8
胃体中部	胃体下部	0.8
胃体下部	前庭部	0.8
胃体下部	胃角部	0.5
前庭部	胃角部	0.5
前庭部	幽門部	0.6
胃角部	幽門部	0.5

C. 胃の所見における Weight Table

表 C.1: 異常の有無における Weight Table

value	frequency
異常なし	9
異常あり	91

表 C.2: 粘膜における Weight Table

value	frequency
粗造	44
不整	40
たこいぼびらん	0
fold 腫大	17
fold 集中	26
fold 途絶	3
萎縮	0
アレア過形成	2
ニッシェ	29
double contour	7
陥凹性病変	13
毛羽立ち	0
バリウム斑	4
透亮像	2

表 C.2: 粘膜における Weight Table

value	frequency
引きつれ	2
ポリープ	9
隆起性病変	40
粘膜下腫瘍	3
壁硬化	9
潰瘍	2
潰瘍瘢痕	3
憩室	1
bridging fpld	2
線状影	2
粒状影	1

表 C.3: 形態における Weight Table

value	frequency
下垂胃	0
牛角胃	0
曝状胃	2
変形	4
胃角開大	5
小弯短縮	0

表 C.4: 機能における Weight Table

value	frequency
胃液過多	1
胃石	0
残渣	4

D. 読影医診断結果との比較

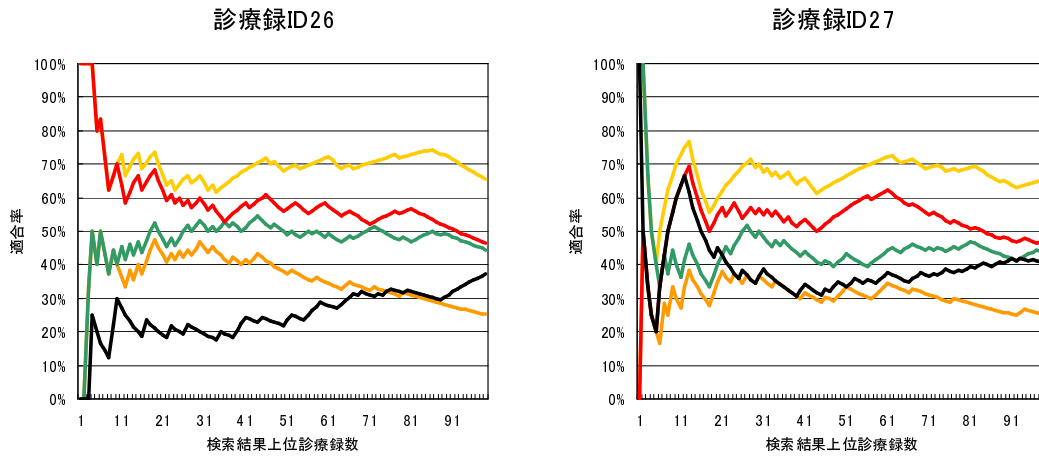


図 D.1 ID26 の読影医診断結果との比較 図 D.2 ID27 の読影医診断結果との比較

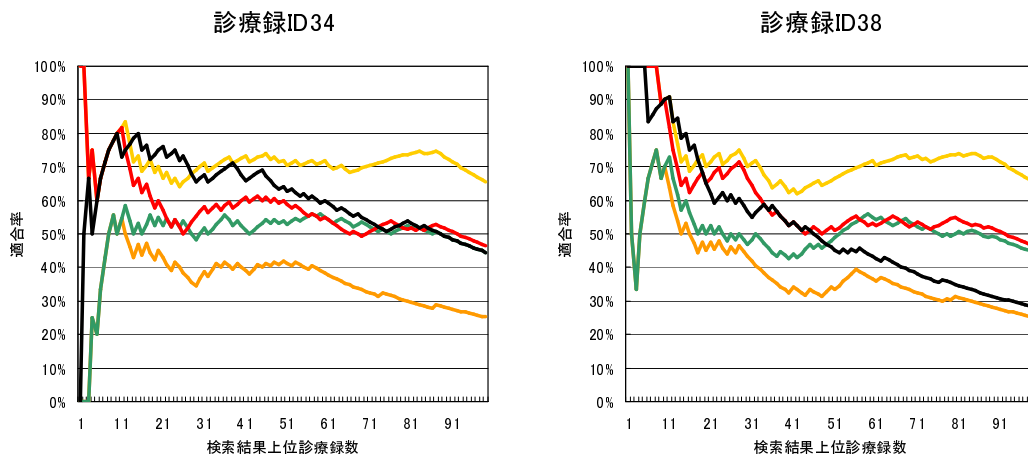


図 D.3 ID34 の読影医診断結果との比較 図 D.4 ID38 の読影医診断結果との比較

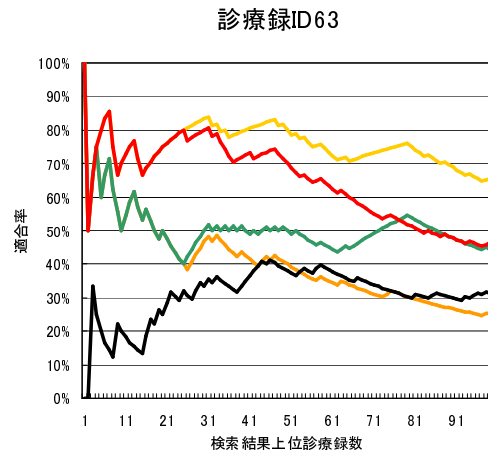
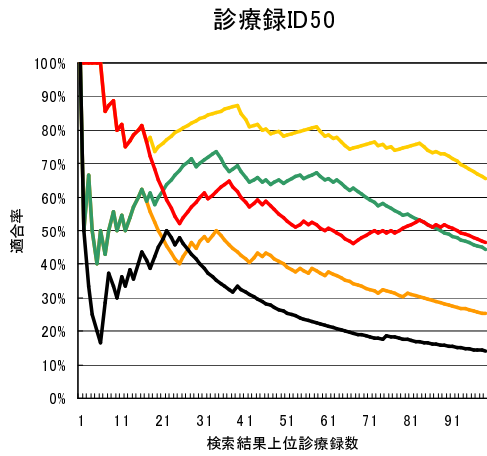


図 D.5 ID50 の読影医診断結果との比較 図 D.6 ID63 の読影医診断結果との比較

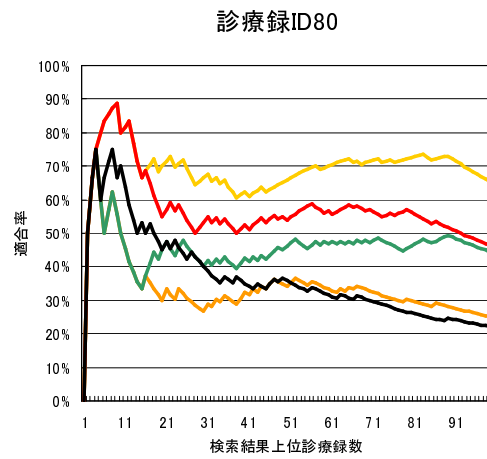
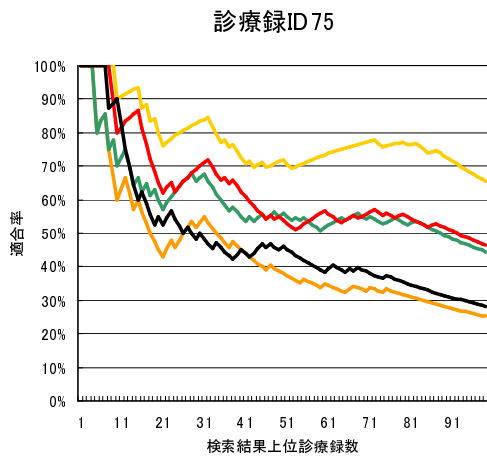


図 D.7 ID75 の読影医診断結果との比較 図 D.8 ID80 の読影医診断結果との比較

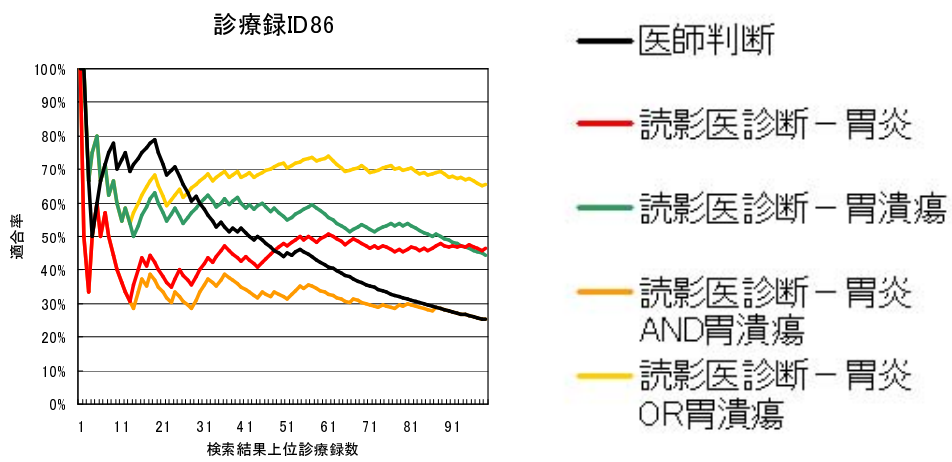


図 D.9 ID86 の読影医診断結果との比較

図 D.10 図 D.1 ~ 図 D.9 の凡例

E. 医師判断とシステム出力結果との差の図

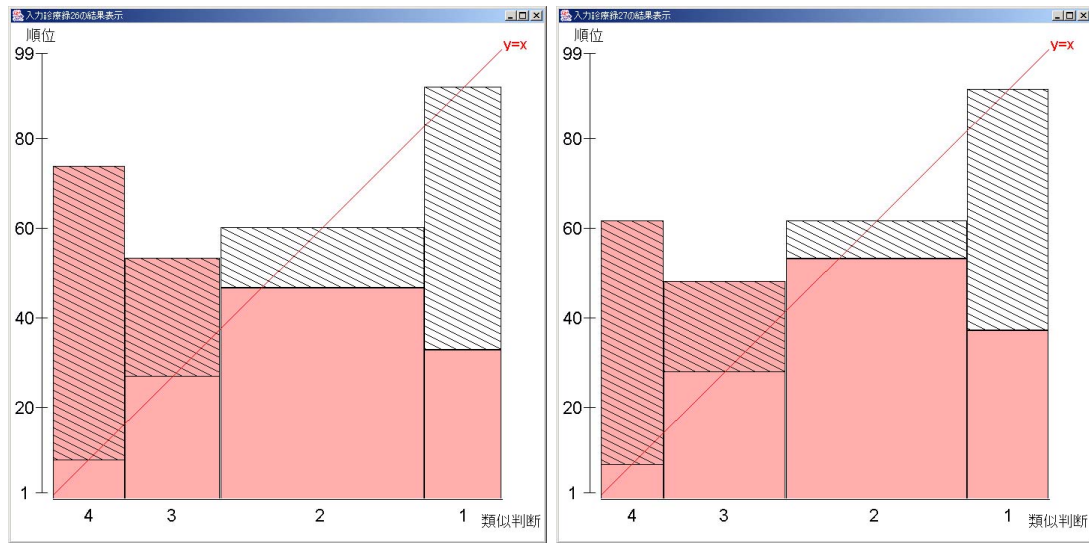


図 E.1 診療録 ID26 における差

図 E.2 診療録 ID27 における差

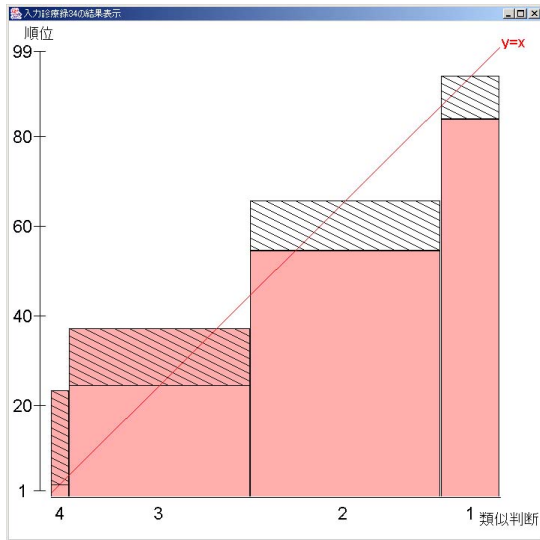


図 E.3 診療録 ID34 における差

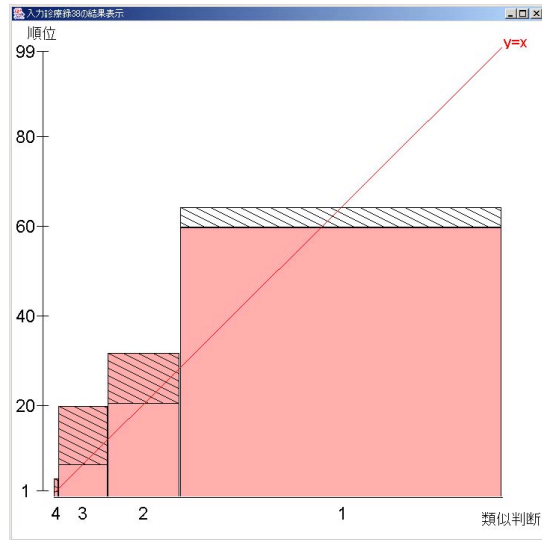


図 E.4 診療録 ID38 における差

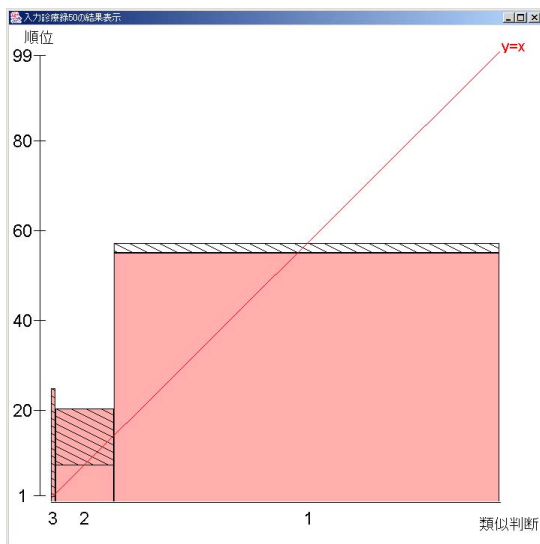


図 E.5 診療録 ID50 における差

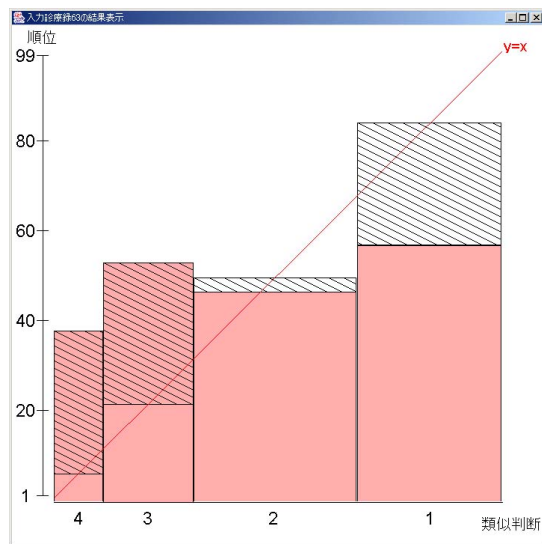


図 E.6 診療録 ID63 における差

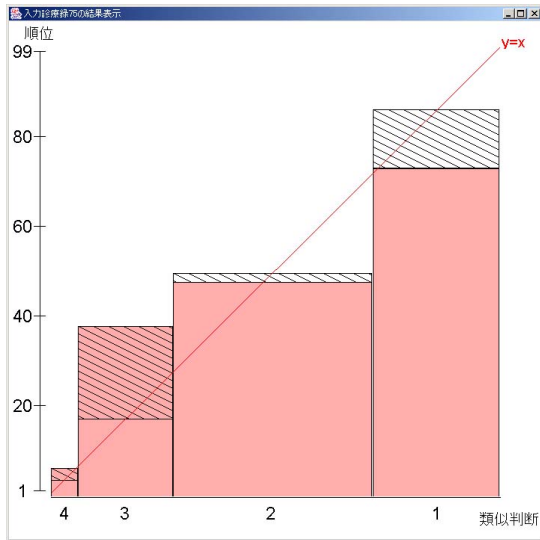


図 E.7 診療録 ID75 における差

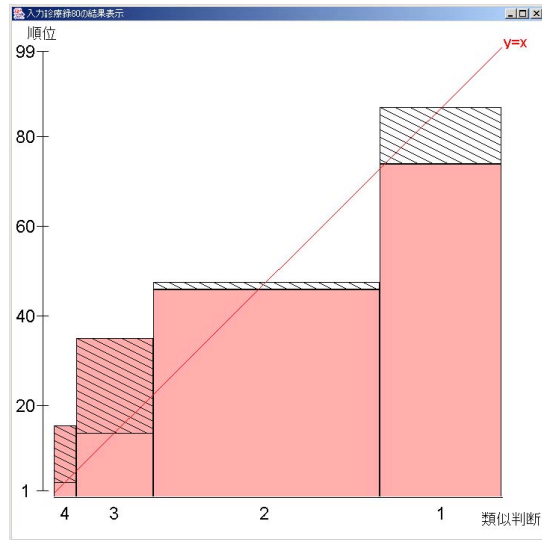


図 E.8 診療録 ID80 における差

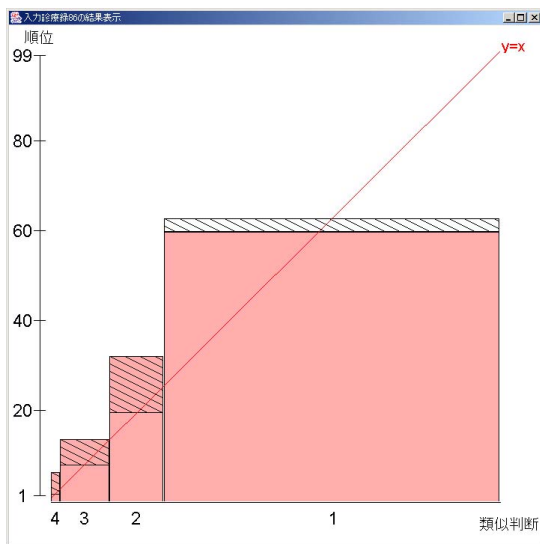


図 E.9 診療録 ID86 における差