１９９４年度

卒業論文

論題：

英語版京都観光経路決定支援システムの作成

指導教員	山崎 勝弘 教授

立命館大学大学理工学部
情報工学科　　　4回生
2170910027−8　奥村　建
2170910068−5　多田　和生

内容梗概

本論文では、京都における観光地の経路作成に対して、実際の観光経路を事例とし、ユーザの希望に類似する事例を検索・修正して観光経路を生成するシステムの英語版作成について述べる。

すでにあるシステムの構成は、実際の観光経路を持つ事例と事例検索部、事例修正部からなる。また、従来のルールベース推論による手法に基づいて経路の作成を行う経路作成部と、観光地間の交通手段を求める交通手段作成部、及びユーザ・インターフェース部がある。

本システムは国際的な観光都市である京都を扱ったものであり、また、本学には外国人のお客様も多いため英語版観光経路決定支援システムを作成した。また、これまでは、コンピュータに不慣れなひとでも簡単に扱えるための手段として、UNIX の X-Window を用いてフルマウスオペレーティングを実現していたが、この考えを発展させ、本研究ではより馴染みやすいシステムにするために、マルチメディア導入の第一歩として観光地の写真を取り入れ、ユーザ・インターフェース部に改良を加えた。

本論文では、英語版観光経路決定支援システムのシステム全体の機能と構成、アルゴリズムとモジュール化及び、英語版地図の作成とユーザ・インターフェース部の説明、改善、システムの評価について述べる。

目 次

1　はじめに

本研究の観光経路決定支援システムは、事例ベース推論（Case-Based reasoning. 以下CBR と略す）を用い観光経路を推論し決定することを軸としたシステムである。

CBR は１９８０年代の半ばから人工知能で注目されるようになった推論技術である。基本的には次のような推論方法である。

- 過去に経験した事例（問題、解答、解答を得る筋道）を事例ベースに蓄えておく。
- 新たな問題の解答を得たい時には、まず、その問題に類似した既知の問題を事例ベースから検索する。
- その既知の問題の事例を利用することによって当該問題の解答を得る。
- 当該問題をその解答や反省材料とともに事例ベースに収納する。

CBR は他の推論技術にくらべ事例と規則の両方を用いて推論をする所に特徴があり

- 専門家からの経験的知識の獲得が容易である。
- 計算負荷を緩和できる。

といった点で注目されており、現実的な問題解決に有効な推論方法であると考えられている。[1][2]

事例ベース推論が観光経路決定に向いている理由を挙げる。京都のように多数の観光地が存在すると、その観光経路は幾通りもでてくる。そこで、過去に利用した経路あるいは有効な経路を事例としておけば、最も類似した事例を検索し修正するだけで容易に観光経路を求めることができる。

インターフェースとは、人間-コンピュータ相互作用の中で最も中心になる部分である。人間とコンピュータとのギャップを橋渡しする介在者としての役割を担う。最近ではコンピュータを扱う上での必要な教育を受けてない人でも自然に使えるという意味で、ヒューマン・インターフェース、ユーザ・インターフェースという言葉が使われるようになった。人間にとって優しく柔らかな技術を実現するために、ヒューマン・インターフェースあるいはユーザ・インターフェースでは必然的に人間のもつ本質との関わりが重要である。言い替えれば、人間にとって分かりやすく使いやすいシステムを作成することが重要である。様々なシステムが様々な社会、例えば企業などに広範囲にわたって使用されており、

今後、計算機システムはますます増えていくだろう。現在、ヒューマン・インターフェースとユーザ・インターフェースに明確な区別がなく、本研究ではユーザ・インターフェースと呼ぶことにする。[4]

本システムでは、ユーザ・インターフェースを取り入れ、全ての操作をマウスでできるようにしている。これは、本システムが観光案内的な面を持ったシステムであり、対象とするユーザがコンピュータに詳しくなくても扱えるシステムを目指しているからである。同時に、ユーザにより興味深く使ってもらうために、写真を表示したりカラーを使うなどの改良を加えた。また、京都は国際的な観光都市であり毎年多くの外国人観光客が訪れる。本学にも外国人のお客様がたくさん来られるので、その人たちにも使ってもらえるように英語版システムの作成をした。

本論文では2章においてシステムの機能と構成及びシステムの概要について、3章において英語版観光経路決定支援システムについて述べる。4章ではシステムを使用した評価、5章で考察と今後の課題について述べる。

2　既存の観光経路決定支援システム [3]-[5]

2.1　機能

システムは、ユーザーの希望観光地、観光ブロック、観光テーマなどを入力してスケジューリングされた観光経路を表示する。京都の地図の表示機能としては、京都全体図、各ブロックの拡大図を表示する。また、観光地の見所、入場料、拝観時間、写真などの観光地情報も表示できる。なお、すべての操作はマウスで行なう。

具体的な入出力を以下に示す。

（1）入力

- 観光ブロック

  洛中、洛東、洛北、洛西、洛南の項目の中からどの地域を観光したいかを選択する。

- 観光テーマ

  庭、絵、建築、仏像、伝統工芸、哲学の道、嵐山の項目の中から、どんなテーマで観光したいかを選択する。

- 観光地の指定

  希望の観光地の指定と特になしから選択する。

- 行きたくない観光地の指定

  過去に行った観光地を省くために指定する。

（2）出力

- 観光ルート

  指定観光地のみを含む直接ルートと、その他の観光地も含む紹介ルートを地図と経路で出力する。

- 交通手段

  観光地の最寄りのバス停、駅、および観光地間で使用するバス系統、鉄道、乗り換えるバス停、駅、を表示する。

- 費用、所要時間

  観光ルートにかかる費用と、その所要時間を表示する。

2.2 構成

システムの構成を図１に示す。ユーザからの希望条件の入力が終ると、紹介ルートを求めるために、制御部は事例ベース推論部を起動して類似事例を検索する。類似事例が存在すれば、事例に含まれない観光地を追加して、これを紹介ルートの候補経路として表示する。次に、指定観光地以外のものを削除して、これを直接ルートとする。類似事例が存在しなければ、ルールベース推論部を起動して観光経路が最短になるものを選び、次に、観光地間の交通手段を求めて接続し直接ルートとして表示する。また、各モジュール間の役割について簡単に説明する。

(1) 入力部

ユーザが行きたい観光地、行きたくない観光地、ブロック、指定観光テーマといったデータを入力する。

(2) 事例検索部

入力されたデータからメインテーマ、メインブロックをきめ、これをもとに事例を格納した事例ベースから候補事例を検索する。そして、幾つかの候補事例のなかから事例における指定観光地の含有率が１／２以上で最も高いものを類似事例としてとりだす。含有率が１／２以上のものがない場合は失敗である。

(3) 事例修正部

事例検索部で成功あるいは一部成功した場合に、検索された事例の中に含まれていない観光地を適切な所に追加する。また、行きたくない観光地や事例全体のモジュールの流れに含まれる不要な観光地の削除も行なう。

(4) 経路作成部

観光地座標をもとにして最短のルートを作る。

(5) 交通手段作成部

ある観光地から他の観光地までの交通手段を求める。

(6) 解出力部

推論で得られた結果を表示する。スケジュール表示では、観光ルートの紹介、交通手段、所要時間、費用、観光ルートの表示を行なう。

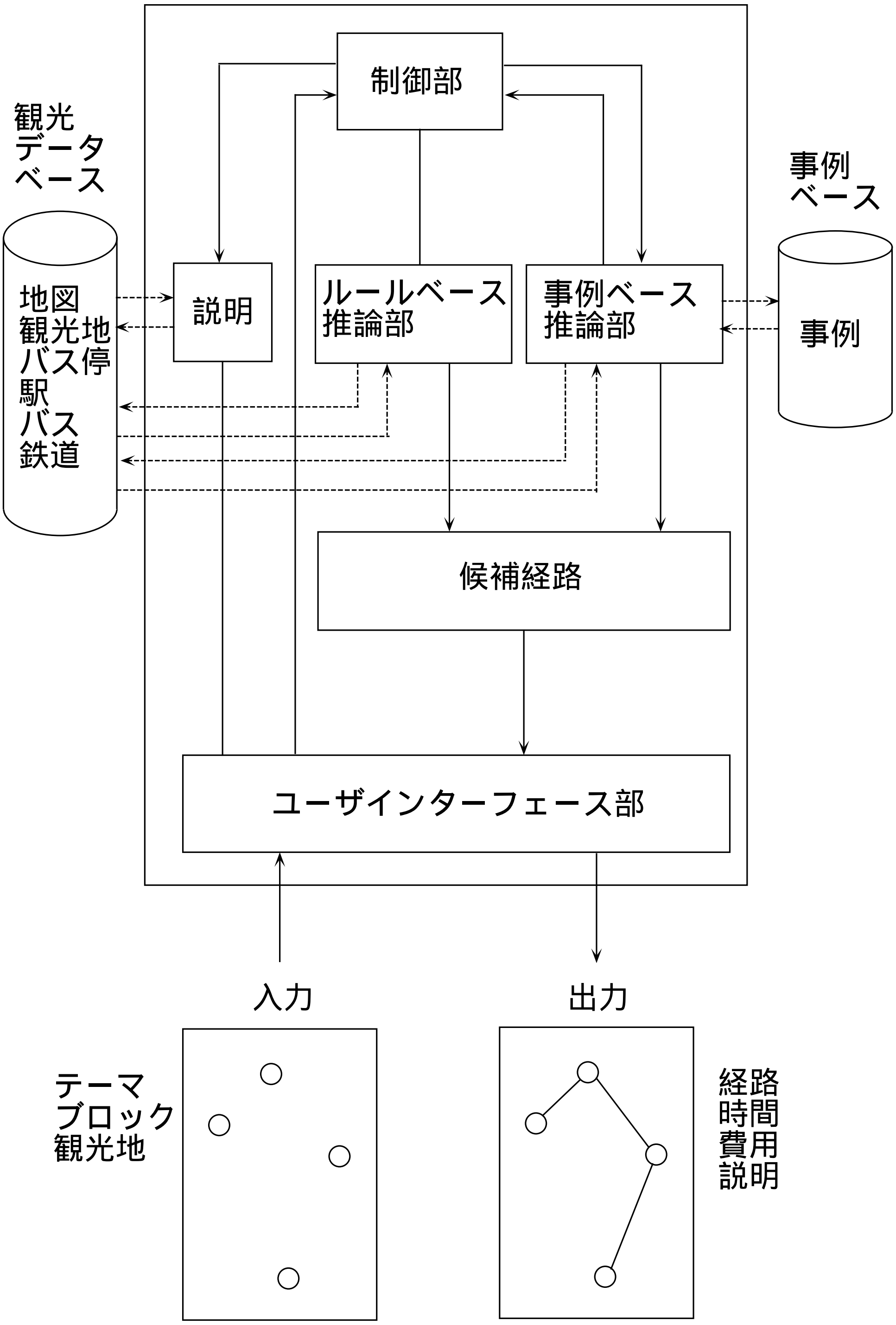
制御部
観光
データ
ベース
事例
ベース
地図
観光地
バス停
駅
バス
鉄道
説明
ルールベース
推論部
事例ベース
推論部
事例
候補経路
ユーザインターフェース部
入力
出力
テーマ
ブロック
観光地
経路
時間
費用
説明

図 1　システム構成

2.3 全体のアルゴリズム

ユーザ・インターフェース部を通して入力部が終了すると、制御部は事例検索部を起動させる。入力項目により次のような場合がある。

(1) テーマあるいはブロック、またはその両方が入力され、観光地が指定されない場合。
(2) 観光地のみが指定され、テーマやブロックが指定されない場合。
(3) 観光地とブロック、観光地とテーマ、観光地とテーマとブロックが入力された場合。

（1）の場合には、テーマやブロックをキーワードに持つ事例をそのまま紹介ルートとしてユーザ・インターフェース部で表示する。ただし、行きたくない観光地が事例に含まれているならば、その観光地を事例から削除する。削除することによって観光地間の交通手段が必要ならば、その前後の観光地の接続を2点間接続で行ない、事例を修正する。（2）の場合には、観光地の持つテーマ所在ブロックをキーワードにし、事例検索部においてもっとも類似した事例を検索する。指定した観光地が全て含まれる場合は紹介ルートとして表示する。指定した観光地の一部が含まれる場合は、類似事例に含まれる観光地と追加する観光地を接続部によって経路付けし、交通手段が必要な所は2点間接続部によって接続したものを表示する。（3）の場合には、（2）と同様にして類似事例を検索し、それらの中で指定されたテーマ、ブロックを満たすものを類似事例とし観光地の追加、削除が必要ならば修正する。

次に、指定観光地のみを含む直接ルートを求めるために、接続部で巡回セールスマン問題を使用して指定観光地の最短距離の経路を求め、2点間接続部でルールベース推論を用いて各観光地間の交通手段を検索し、観光経路を決定する。（2）（3）において、事例検索部で類似事例の検索に失敗したときには、直接ルートの観光経路のみを表示する。

以下に日本語版システムの入出力の例を示す。

図 2　清水寺の観光地紹介文

図 3　ブロック拡大図

図 4　ブロックメニュー

図 5　テーマメニュー

図 6　入力終了

図 7　紹介ルート

3　英語版観光経路決定支援システム

ここでは、図１の観光データベース内の地図・観光地等とユーザ・インターフェース部(とそれを制御する制御部)を変更し英語版システムの作成をおこなった。

3.1　英語版地図の作成

従来の日本語版システムを以下の点を作りかえ英語版にした。ここでは、作りかえた所と、それが図１においてはどのモジュールであるかについてを述べる。

(1) 英語への書き換え

もともと日本語であった観光データベース部、結果出力部、ボタンウインドウ等、すべての文字出力を英語に書き換えた。観光データベース部は全てデータファイルとして作成し、そのファイルから必要なデータを読みだし出力を行なっている。データファイルからの読みだしは、制御部よりscanf命令を用いて行なっている。

(2) 英語出力に伴う最適なフォントの選出

フォント選出は制御部でなされているため、制御部を作りかえている。文字の出力はXライブラリー関数XDrawStringを用いて行なうがこの命令を行なっただけではデフォルトのフォントが返され、そのフォントで文字が出力されてしまうため、文字はXライブラリー関数XSetFontで最適なフォントを設定した上で、出力を行なっている。

(3) 文字表示

X-Windowの場合、文字はラインや図形等と同様にグラフィックスとして扱うため、文字の出力にはXライブラリー関数XDrawStringを用いて行なった。前述のデータファイルに格納されているデータを読みだし、そのデータをXDrawString関数に入れている。

3.2　ユーザ・インターフェースの改善

全体的に見て画面にメリハリが無かったので以下の改良を加えた。

(1) イメージ・スキャナーによる画像取り込み

観光地紹介ウインドウにおいて、紹介文だけでなくその観光地の写真画像を出力することによってさらに観光地に対するイメージを湧かせやすくすることを目的とした。取り込みにはSONYのNWP-5601を使用した。この機種は解像度が400dpi、階調度が２５６階調となっている。読み込んだ画像はgif形式でセーブした。gif形式は他の圧縮方式に比

較して、ファイル容量が小さく読み込み速度も速いため、本システムに使用した。本システムで用いた出力方法は、X-Window のシステム関数を用いて、フリーソフトウエア XV により行なった。当初は、X ライブラリ関数を用いて独自にプログラムを作成し、画像を表示させる予定であったが、UNIX システムにおいてはどの UNIX マシンにおいても共通のソフトが使用できるのを前提として作られているので、個々のマシン性能に依存するグラフィックス機能は X ライブラリ関数ではサポートされていない部分が多く、独自にプログラムを作成してフルカラー画像を表示させることは非常に困難であるので結局この方法を採用した。表示例を図 8 に示す。

(2) 画面の改善

画面を見た時にマウスのポインタがどれであるのかが判断しにくかったので、色づけを行なった。他にも背景などの色づけを行なった。また、マッキントッシュで作成した Rits ロゴや KYOTO CITY,YAMAZAKI LABLATORY といった画像を移植し、画面に出力した。これらのことによって、画面にメリハリをもたせ、興味を引きやすいようなシステムにした。また、マウスの移動範囲が少なくてすむようになるべくボタン類を画面の中心付近に持ってくるように配慮した。

図 8　清水寺の観光地紹介文と写真

3.3　ユーザの入力

ユーザインターフェース部はX-Windowを用いて、すべての操作をマウスで行なえるようにした。ユーザが観光地を指定する際に、行動範囲、見たいテーマをすぐに入力できるようにメニューを設けた。メニューには「BLOCK」(観光地のブロック)、「THEME」(観光テーマ)、「TO GO」(行きたい観光地)、「NOT TO GO」(行きたくない観光地)がある。ユーザは希望するものをマウスでクリックすればよい。

(1) 観光地の指定

ブロック拡大地図上の観光地をクリックすると、その観光地の情報と写真が表示される。ユーザがそこに行きたい場合は、観光地紹介ウインドウの「TO GO」をクリックする。以前行ったことがあるなどの理由で行きたくない場合は、「NOT TO GO」をクリックする。入力が終ると「TO GO」の場合は緑に、「NOT TO GO」の場合は赤に観光地の色が変わる。どちらでもない時は、「CANCEL」をクリックすると戻る。

(2) ブロックの指定

ユーザが行きたいブロックがあれば、図１０の「BLOCK」をクリックすればその下にプルダウンメニューが開き選択項目(RAKUCHU,RAKUTO,RAKUHOKU,RAKUSAI,RAKUNAN,QUIT)が表示されるので(図９)、その中から選ぶ。プルダウンメニューの項目をクリックすると、ブロック拡大地図に選んだブロックが表示される。なお、図１０の右下の広域地図上のブロックをクリックしても同じである。

(3) 観光テーマの指定

ユーザが観光する上で希望テーマがあれば、図１０の「THEME」をクリックする。その下にプルダウンメニューが開き選択項目(Gardens,Pictures,Architecture,A Buddhist Image,Handcrafts,Tetsugaku-no-michi,Ohara,arasiyama,Quit)が表示されるので(図１１)その中から選び、クリックする。

(4) データ入力の終了

データ入力が終ったら、図１０の右上にある「INPUT END」をクリックする。すると、図１２のように指定観光地が最後の確認のために表示される。ここで入力を終了するならば「YES」をクリックする。もう一度入力し直す場合「NO」をクリックする。「YES」ならば推論部へと進む。

図 9　ブロックメニュー

図 10　ブロック拡大図

図 11　テーマメニュー

図 12　入力終了

3.4　推論結果の出力

ユーザの入力データをもとに、推論部で推論を行なった結果を画面に表示する。

(1) 観光経路の出力

推論結果を、図１３のようにウインドウ上に表示する。表示する項目は観光地、交通手段、乗り降りするバス停（もしくは駅）となっている。また、推論部で入力されたデータに付け加えをした場合には最初に紹介ルートを、そうでない場合は直接ルートが表示される。前者の場合なら、図１３の左上「DIRECT/INTRODUCE」をクリックすれば紹介ルートと直接ルートの切替ができる。直接ルートを図１４に示す。

(2) 地図による出力

（１）において、推論結果をウインドウで表示したが、図１３の「ROOT/MAP」をクリックすることによって表示を地図にすることができる。表示される地図には、ユーザが入力した観光地に加えて、紹介ルートで加わった観光地も緑になって表示される。ここで観光地をクリックすれば、入力部と同様に観光地紹介と写真が表示される（図１５）。この地図上で、ユーザは訪れる観光地の位置が確認でき、紹介ルートで新たに加わった観光地の情報を見ることができる。

(3) 出力部の終了

出力部の終了は図１３で「QUIT」をクリックするか「RETRY」をクリックする。「RETRY」を選べば再び観光地の入力から始まる。「QUIT」を選べばシステムは終了する。

図 13　紹介ルート

図 14　直接ルート

図 15　加わった観光地の紹介文 (銀閣寺)

4　システムの評価

ここでは、実際にシステムを使用した人の評価について述べる。本研究室の院生や４回生、他の研究室の４回生の１０人の声をまとめたものである。

当初の目的である英語版システムについては、ほぼ良い評価が得られた。観光地の写真表示についても賛成の声が聞けた。しかし、写真表示については後述する問題点もでた。以下に具体的な意見を示す。

(1) 写真表示

せっかく写真表示をするのだから、見栄えのする大きい写真をいれたらどうかという声が最も多かった。現在、200×200 ドットの写真を表示しているが、これ以上大きい写真を表示すると出てくるまでに時間がかかるので、この大きさにした。この大きさだと３秒かかり、写真を表示するウインドウ（図８の写真が出ている所）いっぱいに表示すると（385×395）５秒かかる。将来もっと高性能なマシンに移植し、画面いっぱいに表示したらどうかという意見があったが、そうすると観光地紹介文ウインドウや京都全体図に重なり見にくくなるので、前述の 385×395 のサイズで表示させたい。また、写真表示には静止画像でなくビデオ画像を取り込んで、その観光地の流し絵や祭りの風景を表示させるという声もあった。

(2) 画面全体

図８や図１０の画面右側と左側で色のギャップがある、という意見が出た。これは、例えば、図８で画面右は観光地の写真など色のついたものが多くにぎやかだが、画面左はモノクロでさびしい、図１３のルート紹介ウインドウにバスや歩くマークなどの簡単な画像を入れて欲しいといったことである。

(3) 改良する点

ブロック拡大図については、通りの名前、主な駅の名前をいれたほうがよいという意見や、拡大図上のマウスのポインタの位置を全体図にも表示して欲しいという意見もあった。また、京都の主要駅（JR 京都駅や三条京阪）での時刻表を表示がほしいという声もあった。

5　おわりに

今回、目標としていた英語版観光経路決定支援システムは一応完成するところまでいたった。また、新たな試みとしてマルチメディア導入の第一歩として観光地の写真を表示させた。これに関してはおおむね良好な評価が得られたが、表示する写真が小さいという声も多かった。

今後の展開としては、以下の点を改善したい。

(1) ブロック拡大地図に関して

道路に色と主要交差点名をつけてカラフルでわかりやすいものにする。ブロックごとの表示ではなく、スクロール機能をつけてスクロールさせる。また、ルート紹介のときに通過するルートを表示する。また、全体図を簡単な絵地図にする。

(2) ルート紹介に関して

ルート紹介画面で徒歩やバス、電車のマークを入れる。紹介ルートをプリントアウトする。主要駅の時刻表を表示する機能をつける。

これらの実現でかなりユーザ・フレンドリーなシステムになるであろう。動画や音声等のマルチメディア導入も含めて、今後の検討課題としたい。

謝辞

本研究の機会を与えて下さり、御指導いただきました山崎勝弘教授に深く謝意を表します。また、英語版観光経路決定支援システムの作成において、貴重な御意見と御指導をいただきました大学院院生の西村高志氏と古川知之氏を初めとする本研究室のみなさまに深く感謝します。

参考文献

[1] 北野宏明 :超並列人工知能，人工知能学会誌，Vol.7，No.2，pp.244-262(1992)．

[2] 松原仁 :推論技術の観点からみた事例に基づく推論，人工知能学会誌，Vol.7，No.4，pp.567-575(1992)．

[3] 西村高志 :事例ベース推論による観光経路決定支援システムの改善と評価，立命館大学理工学部情報工学科卒業論文 (1993)．

[4] 三宅国彦 :観光経路決定支援システムのユーザ・インターフェースの作成，立命館大学理工学部情報工学科卒業論文 (1992)．

[5] 森田真弘 :事例ベース推論推論による観光経路決定支援システム, 立命館大学理工学部情報工学科修士論文 (1993).

[6] 細江亮 :X ウインドウプログラミング (Xlib), HBJ 出版局 (1992).

[7] 井門俊治 :X-Window 実用グラフィックス入門, 日刊工業新聞社 (1992).

[8] Walk 京都３５, 山と渓谷社 (1993).

[9] GUIDE BOOK KYOTO, 京都市観光協会 (1978).

[10] JTB の旅ノート (19) 京都, JTB 日本交通公社出版事業局 (1994).

[11] NEW ブルーガイドブックス (12) 京都, 実業之日本社 (1994).