

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5103603号
(P5103603)

(45) 発行日 平成24年12月19日(2012.12.19)

(24) 登録日 平成24年10月12日(2012.10.12)

(51) Int.Cl. F I
G06F 17/30 (2006.01)
 G06F 17/30 340Z
 G06F 17/30 110G
 G06F 17/30 350C
 G06F 17/30 380D

請求項の数 6 (全 54 頁)

(21) 出願番号 特願2008-26864 (P2008-26864)
 (22) 出願日 平成20年2月6日(2008.2.6)
 (65) 公開番号 特開2009-187305 (P2009-187305A)
 (43) 公開日 平成21年8月20日(2009.8.20)
 審査請求日 平成22年12月6日(2010.12.6)

(出願人による申告)平成19年度、独立行政法人科学技術振興機構委託研究、「平成19年度地域イノベーション創出総合支援事業 シーズ発掘試験」、産業活力特別措置法第30条の適用を受ける特許出願

(73) 特許権者 504145364
 国立大学法人群馬大学
 群馬県前橋市荒牧町四丁目2番地
 (74) 代理人 100122884
 弁理士 角田 芳末
 (74) 代理人 100133824
 弁理士 伊藤 仁恭
 (72) 発明者 安川 美智子
 群馬県桐生市天神町1丁目5番1号 国立
 大学法人群馬大学内
 (72) 発明者 横尾 英俊
 群馬県桐生市天神町1丁目5番1号 国立
 大学法人群馬大学内

審査官 佐藤 実

最終頁に続く

(54) 【発明の名称】 情報検索システム及び情報検索装置

(57) 【特許請求の範囲】

【請求項1】

クライアントと、前記クライアントに情報を提供する情報検索サーバと、前記情報検索サーバの要求にしたがって所定の情報を出力する検索エンジンとよりなる情報検索システムであって、

前記クライアントは、
 現在位置情報を取得するGPS受信部と、
 所定の情報を表示する表示部と、
 前記表示部を制御する表示制御部と、

前記GPS受信部から得られる前記位置情報を送信すると共に、所定の情報を受信して前記表示制御部に渡す入出力制御部とを備え、

前記検索エンジンは、所定の単語が入力されると前記単語の関連語を出力するものであり、

前記情報検索サーバは、
 前記情報検索エンジンから前記所定の情報を受信して加工するバッチ処理部と、
 前記バッチ処理部で加工された情報を蓄積する不揮発性データ記憶部と、
 前記不揮発性データ記憶部に蓄積された前記加工された情報を前記クライアントに送信するリアルタイム処理部とを備え、

10

20

前記不揮発性データ記憶部は、

前記GPS受信部から前記入出力制御部を通じて得られる前記現在位置情報に対応する地名が格納されているGPS地名マスタと、

前記情報検索エンジンから得られる、前記地名とその関連語群が格納される地名関連語リストと、

前記地名と前記関連語群から作成される前記地名と各々の関連語の組よりなる複数の検索クエリをそれぞれ前記情報検索エンジンにて検索した結果が記録される検索結果ファイル群が格納される検索結果ファイルディレクトリと、

前記検索結果ファイル群に記載されているURLから得られる、前記地名と前記関連語の組が含まれているネットワーク上の文書が格納されるキャッシュファイルディレクトリと、

10

前記検索結果ファイル毎に作成され、前記ネットワーク上の文書群に含まれる前記関連語を中心とする前後の単語が前記ネットワーク上の文書のファイル名と共に記憶されるWAMファイルを格納するWAMファイルディレクトリと、

前記WAMファイル中に出現する前記関連語の関係が記されている関連語出現テーブルと、

前記WAMファイル中に出現する前記関連語の類似度が記されている本ベストスコアテーブルと、

前記本ベストスコアテーブルから生成され、前記クライアントへ送信される本クラスタが格納される本クラスタディレクトリと

20

を備え、

前記バッチ処理部は、

前記GPS地名マスタに含まれている前記地名を前記検索エンジンに与えて前記地名関連語リストを取得し、前記地名関連語リストから前記地名と前記関連語の組を前記検索エンジンに与えて前記検索結果ファイルを前記検索結果ファイルディレクトリに格納すると共に、前記検索結果ファイルに記載されているURLから得られる、前記地名と前記関連語の組が含まれているネットワーク上の文書を前記キャッシュファイルディレクトリに格納するデータ取得処理部と、

前記ネットワーク上の文書に含まれる前記関連語を中心とする前後の単語を取り出して前記ネットワーク上の文書のファイル名と共に前記WAMファイルに書き出し、前記WAMファイルディレクトリに格納するWAMファイル生成処理部と、

30

同一の地名に係る複数の前記WAMファイルに基づいて、前記ネットワーク上の文書群と前記関連語の有無の関係を前記関連語出現テーブルに書き出す関連語出現テーブル生成処理部と、

前記関連語出現テーブルを基に全ての前記各関連語同士の類似度を算出した後、前記類似度にてソートし、基準となる第一関連語に最も類似度が高い第二関連語のレコードと、前記第二関連語の次に類似度が高い第三関連語のレコードとを抜粋して、前記第一関連語、前記第二関連語及び前記第三関連語を夫々フィールドに持つレコードを列挙した本ベストスコアテーブルを生成するベストスコアテーブル生成処理部と、

前記本ベストスコアテーブルを基に、前記クライアントへ送信される本クラスタを作成して前記本クラスタディレクトリに格納する本クラスタ生成処理部と

40

を備え、

前記リアルタイム処理部は、

前記クライアントから得られる前記現在位置情報を受けて対応する地名を前記GPS地名マスタから取得して、前記本クラスタディレクトリから前記地名に対応する前記本クラスタを取得した後、所定の文書形式に変換してクライアントに送信する本クラスタ送信部と

を備えることを特徴とする情報検索システム。

【請求項2】

前記本クラスタ生成処理部は、

50

前記本ベストスコアテーブルの一のレコード中の前記第二関連語が前記第一関連語に存在するレコードと、前記一のレコード中の前記第三関連語が前記第一関連語に存在するレコードとを前記一のレコードと共に同一のクラスタ番号を付与し、

同一の前記クラスタ番号が付与された三レコードの一のレコードの前記第二関連語及び前記第三関連語のいずれも他のレコードの前記第一関連語と一致しない前記一のレコードを削除し、

前記第一関連語が一致する複数のレコードのうち前記類似度が最高値を示すレコード以外のレコードを削除し、

前記第一関連語が所定のストップワードと一致するレコードを削除し、

前記第二関連語と前記第三関連語が上位クラスタの各レコードの前記第一関連語の組と一致するレコードであり、前記レコードが当該クラスタ唯一である場合、或は前記レコードが属する同一クラスタの全てのレコードの前記第一関連語の文字列長が所定長以下である場合、前記レコードに前記上位クラスタのクラスタ番号を付与することによって前記本クラスタを作成することを特徴とする、請求項 1 記載の情報検索システム。

【請求項 3】

予め情報検索エンジンから所定の情報を取得した後に所定の加工を行い、GPS 受信部を内蔵するクライアントから現在位置情報を受信して、前記クライアントに前記現在位置情報に対応する前記加工した情報を提供する情報検索装置であって、

前記情報検索エンジンから受信した前記所定の情報を加工するバッチ処理部と、

前記バッチ処理部で加工された情報を蓄積する不揮発性データ記憶部と、

前記不揮発性データ記憶部に蓄積された前記加工された情報を前記クライアントに送信するリアルタイム処理部と

を備え、

前記不揮発性データ記憶部は、

前記クライアントから得られる前記現在位置情報に対応する地名が格納されている GPS 地名マスタと、

前記情報検索エンジンから得られる、前記地名とその関連語群が格納される地名関連語リストと、

前記地名と前記関連語群から作成される前記地名と各々の関連語の組よりなる複数の検索クエリをそれぞれ前記情報検索エンジンにて検索した結果が記録される検索結果ファイル群が格納される検索結果ファイルディレクトリと、

前記検索結果ファイル群に記載されている URL から得られる、前記地名と前記関連語の組が含まれているネットワーク上の文書が格納されるキャッシュファイルディレクトリと、

前記検索結果ファイル毎に作成され、前記ネットワーク上の文書群に含まれる前記関連語を中心とする前後の単語が前記ネットワーク上の文書のファイル名と共に記憶される WAM ファイルを格納する WAM ファイルディレクトリと、

前記 WAM ファイル中に出現する前記関連語の関係が記されている関連語出現テーブルと、

前記 WAM ファイル中に出現する前記関連語の類似度が記されている本ベストスコアテーブルと、

前記本ベストスコアテーブルから生成され、前記クライアントへ送信される本クラスタが格納される本クラスタディレクトリと

を備え、

前記バッチ処理部は、

前記 GPS 地名マスタに含まれている前記地名を前記情報検索エンジンに与えて前記地名関連語リストを取得し、前記地名関連語リストから前記地名と前記関連語の組を前記情報検索エンジンに与えて前記検索結果ファイルを前記検索結果ファイルディレクトリに格納すると共に、前記検索結果ファイルに記載されている URL から得られる、前記地名と前記関連語の組が含まれているネットワーク上の文書を前記キャッシュファイルディレクトリに格

10

20

30

40

50

納するデータ取得処理部と、

前記ネットワーク上の文書に含まれる前記関連語を中心とする前後の単語を取り出して前記ネットワーク上の文書のファイル名と共に前記W A Mファイルに書き出し、前記W A Mファイルディレクトリに格納するW A Mファイル生成処理部と、

同一の地名に係る複数の前記W A Mファイルに基づいて、前記ネットワーク上の文書群と前記関連語の有無の関係を前記関連語出現テーブルに書き出す関連語出現テーブル生成処理部と、

前記関連語出現テーブルを基に全ての前記各関連語同士の類似度を算出した後、前記類似度にてソートし、基準となる第一関連語に最も類似度が高い第二関連語のレコードと、前記第二関連語の次に類似度が高い第三関連語のレコードとを抜粋して、前記第一関連語、前記第二関連語及び前記第三関連語を夫々フィールドに持つレコードを列挙した本ベストスコアテーブルを生成するベストスコアテーブル生成処理部と、

前記本ベストスコアテーブルを基に、前記クライアントへ送信される本クラスタを作成して前記本クラスタディレクトリに格納する本クラスタ生成処理部とを備え、

前記リアルタイム処理部は、

前記クライアントから得られる前記現在位置情報を受けて対応する地名を前記G P S地名マスタから取得して、前記本クラスタディレクトリから前記地名に対応する前記本クラスタを取得した後、所定の文書形式に変換してクライアントに送信する本クラスタ送信部と

を備えることを特徴とする情報検索装置。

【請求項4】

前記本クラスタ生成処理部は、

前記本ベストスコアテーブルの一のレコード中の前記第二関連語が前記第一関連語に存在するレコードと、前記一のレコード中の前記第三関連語が前記第一関連語に存在するレコードとを前記一のレコードと共に同一のクラスタ番号を付与し、

同一の前記クラスタ番号が付与された三レコードの一のレコードの前記第二関連語及び前記第三関連語のいずれも他のレコードの前記第一関連語と一致しない前記一のレコードを削除し、

前記第一関連語が一致する複数のレコードのうち前記類似度が最高値を示すレコード以外のレコードを削除し、

前記第一関連語が所定のストップワードと一致するレコードを削除し、

前記第二関連語と前記第三関連語が上位クラスタの各レコードの前記第一関連語の組と一致するレコードであり、前記レコードが当該クラスタ唯一である場合、或は前記レコードが属する同一クラスタの全てのレコードの前記第一関連語の文字列長が所定長以下である場合、前記レコードに前記上位クラスタのクラスタ番号を付与することによって前記本クラスタを作成することを特徴とする、請求項3記載の情報検索装置。

【請求項5】

クライアントと、前記クライアントに情報を提供する情報検索サーバと、前記情報検索サーバの要求にしたがって所定の情報を出力する検索エンジンとよりなる情報検索システムであって、

前記クライアントは、

ユーザの操作によって固有名詞である所定の検索クエリの入力を受け付ける入力部と、

所定の情報を表示する表示部と、

前記表示部を制御する表示制御部と、

前記入力部から得られる前記検索クエリを送信すると共に、所定の情報を受信して前記表示制御部に渡す入出力制御部と

を備え、

前記検索エンジンは、所定の単語が入力されると前記単語の関連語を出力するものであり、

10

20

30

40

50

前記情報検索サーバは、
前記情報検索エンジンから前記所定の情報を受信して加工するバッチ処理部と、
前記バッチ処理部で加工された情報を蓄積する不揮発性データ記憶部と、
前記不揮発性データ記憶部に蓄積された前記加工された情報を前記クライアントに送信するリアルタイム処理部と

を備え、

前記不揮発性データ記憶部は、

固有名詞が格納されている固有名詞マスタと、

前記情報検索エンジンから得られる、前記入力部から前記入出力制御部を通じて得られる前記検索クエリに対応する関連語群が格納される固有名詞関連語リストと、

10

前記固有名詞と各々の関連語の組よりなる複数の検索クエリをそれぞれ前記情報検索エンジンにて検索した結果を記録する検索結果ファイル群が格納される検索結果ファイルディレクトリと、

前記検索結果ファイル群に記載されているURLから得られる、前記固有名詞と前記関連語の組が含まれているネットワーク上の文書が格納されるキャッシュファイルディレクトリと、

前記検索結果ファイル毎に作成され、前記ネットワーク上の文書群に含まれる前記関連語を中心とする前後の単語が前記ネットワーク上の文書のファイル名と共に記憶されるWAMファイルを格納するWAMファイルディレクトリと、

前記WAMファイル中に出現する前記関連語の関係が記されている関連語出現テーブルと、

20

前記WAMファイル中に出現する前記関連語の類似度が記されている本ベストスコアテーブルと、

前記本ベストスコアテーブルから生成され、前記クライアントへ送信される本クラスタが格納される本クラスタディレクトリと

を備え、

前記バッチ処理部は、

前記固有名詞マスタに含まれている前記固有名詞を前記検索エンジンに与えて前記固有名詞関連語リストを取得し、前記固有名詞関連語リストから前記固有名詞と前記関連語の組を前記検索エンジンに与えて前記検索結果ファイルを前記検索結果ファイルディレクトリに格納すると共に、前記検索結果ファイルに記載されているURLから得られる、前記固有名詞と前記関連語の組が含まれているネットワーク上の文書を前記キャッシュファイルディレクトリに格納するデータ取得処理部と、

30

前記ネットワーク上の文書に含まれる前記関連語を中心とする前後の単語を取り出して前記ネットワーク上の文書のファイル名と共に前記WAMファイルに書き出し、前記WAMファイルディレクトリに格納するWAMファイル生成処理部と、

同一の固有名詞に係る複数の前記WAMファイルに基づいて、前記ネットワーク上の文書群と前記関連語の有無の関係を前記関連語出現テーブルに書き出す関連語出現テーブル生成処理部と、

前記関連語出現テーブルを基に全ての前記各関連語同士の類似度を算出した後、前記類似度にてソートし、基準となる第一関連語に最も類似度が高い第二関連語のレコードと、前記第二関連語の次に類似度が高い第三関連語のレコードとを抜粋して、前記第一関連語、前記第二関連語及び前記第三関連語を夫々フィールドに持つレコードを列挙した本ベストスコアテーブルを生成するベストスコアテーブル生成処理部と、

40

前記本ベストスコアテーブルを基に、前記クライアントへ送信される本クラスタを作成して前記本クラスタディレクトリに格納する本クラスタ生成処理部と

を備え、

前記リアルタイム処理部は、

前記クライアントから得られる前記固有名詞である前記検索クエリに対応する前記本クラスタを前記本クラスタディレクトリから取得した後、所定の文書形式に変換してクライ

50

アントに送信する本クラスタ送信部とを備えることを特徴とする情報検索システム。

【請求項 6】

前記本クラスタ生成処理部は、

前記本ベストスコアテーブルの一のレコード中の前記第二関連語が前記第一関連語に存在するレコードと、前記一のレコード中の前記第三関連語が前記第一関連語に存在するレコードとを前記一のレコードと共に同一のクラスタ番号を付与し、

同一の前記クラスタ番号が付与された三レコードの一のレコードの前記第二関連語及び前記第三関連語のいずれも他のレコードの前記第一関連語と一致しない前記一のレコードを削除し、

前記第一関連語が一致する複数のレコードのうち前記類似度が最高値を示すレコード以外のレコードを削除し、

前記第一関連語が所定のストップワードと一致するレコードを削除し、

前記第二関連語と前記第三関連語が上位クラスタの各レコードの前記第一関連語の組と一致するレコードであり、前記レコードが当該クラスタ唯一である場合、或は前記レコードが属する同一クラスタの全てのレコードの前記第一関連語の文字列長が所定長以下である場合、前記レコードに前記上位クラスタのクラスタ番号を付与することによって前記本クラスタを作成することを特徴とする、請求項 5 記載の情報検索システム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、情報検索システム及び情報検索装置にかかり、特に、検索エンジンによって文書データを検索する情報検索システム及び情報検索装置に関する。

【背景技術】

【0002】

近年、携帯電話の普及により、携帯電話からインターネット上の検索エンジンを用いて、さまざまな検索が行われている。しかし、流行している物や現象、人名、企業、商品、サービス、テレビ番組、地名、駅名等を検索クエリとして情報を検索することは決して易しくはない。これは、ユーザが検索対象についてあまり詳しく知らないために、適切な関連語で検索結果を絞り込むことが困難であるからである。

また、検索クエリで検索される膨大な検索結果を全て閲覧するのではなく、興味のあるウェブページ群だけ概観したいという要求もある。

【0003】

パソコンを用いた検索では、検索対象となる文章集合の中には類似した文章が含まれることが多いため、予め文書集合を類似度に応じて分類しておき、検索時にはこれらのグループと検索クエリとの類似度を計算するクラスタ型の検索が知られている。

また、ある検索クエリで検索される検索結果ウェブページ群は、多数の類似したウェブページが含まれるので、適切な分類を行うことにより、検索結果を容易に絞り込むことができると共に、検索結果を概観することも容易になる。

なお、非特許文献 1 にクラスタ型の検索に関する先行技術文献を記す。

【0004】

【非特許文献 1】徳永健伸、「情報検索と言語処理」、東京大学出版会、(1999)

【発明の開示】

【発明が解決しようとする課題】

【0005】

しかしながら、上記非特許文献 1 に記載の技術では、ウェブページ内のテキストデータ全文を利用して関連語を分類している。このようなテキストデータ全文の中には、ユーザの検索ニーズに合致しない雑多な情報が多数含まれている。そのため、ユーザにとって意味が分からない関連語を含むクラスタや、検索対象を絞り込む上で役に立たないクラスタが生成されてしまうという問題が発生する。すなわち、分類された関連語が、ユーザにと

10

20

30

40

50

って分かりにくく、利便性の低いものとなるという不都合がある。特に、一般名詞とは異なる特徴を持つ固有名詞、例えば地名、駅名及び人名等の関連語の分類を行う際には、上述した問題点が顕著に現れることが、本発明者の過去の研究より明らかとなっている。

【0006】

携帯電話のウェブブラウザでクスタ型検索を実行すると問題はもっと複雑になる。つまり、携帯電話の表示画面は、パソコンの表示画面に比べて小さいため、一行に表示できる文字列の数が限られている。そのため、やみくもに関連語の分類を行い、分類された関連語全てを携帯電話の画面に表示しようとする、見た目が悪いだけでなく、ユーザにとって非常に分かりづらい画面表示となってしまうという問題がある。

【0007】

本発明は斯かる点に鑑みてなされたものであり、固有名詞の関連語を適切に分類することにより、携帯電話のような小さい表示画面に関連語の分類画面を表示する場合でも、ユーザにとって使い勝手が良い情報検索システム及び情報検索装置を提供することを目的とする。

【課題を解決するための手段】

【0008】

上記課題を解決するため、本発明に係る情報検索システムは、クライアントと、クライアントに情報を提供する情報検索サーバと、情報検索サーバの要求にしたがって所定の情報を出力する検索エンジンとよりなる情報検索システムであって、クライアントは、現在位置情報を取得するGPS受信部と、所定の情報を表示する表示部と、表示部を制御する表示制御部と、GPS受信部から得られる位置情報を送信すると共に、所定の情報を受信して表示制御部に渡す入出力制御部とを備え、検索エンジンは、所定の単語が入力されると単語の関連語を出力するものであり、情報検索サーバは、情報検索エンジンから所定の情報を受信して加工するバッチ処理部と、バッチ処理部で加工された情報を蓄積する不揮発性データ記憶部と、不揮発性データ記憶部に蓄積された加工された情報をクライアントに送信するリアルタイム処理部とを備え、不揮発性データ記憶部は、GPS受信部から入出力制御部を通じて得られる現在位置情報に対応する地名が格納されているGPS地名マスタと、情報検索エンジンから得られる、地名とその関連語群が格納される地名関連語リストと、地名と関連語群から作成される地名と各々の関連語の組よりなる複数の検索クエリをそれぞれ情報検索エンジンにて検索した結果が記録される検索結果ファイル群が格納される検索結果ファイルディレクトリと、検索結果ファイル群に記載されているURLから得られる、地名と関連語の組が含まれているネットワーク上の文書が格納されるキャッシュファイルディレクトリと、検索結果ファイル毎に作成され、ネットワーク上の文書群に含まれる関連語を中心とする前後の単語がネットワーク上の文書のファイル名と共に記憶されるWAMファイルを格納するWAMファイルディレクトリと、WAMファイル中に出現する関連語の関係が記されている関連語出現テーブルと、WAMファイル中に出現する関連語の類似度が記されている本ベストスコアテーブルと、本ベストスコアテーブルから生成され、クライアントへ送信される本クスタが格納される本クスタディレクトリとを備え、バッチ処理部は、GPS地名マスタに含まれている地名を検索エンジンに与えて地名関連語リストを取得し、地名関連語リストから地名と関連語の組を検索エンジンに与えて検索結果の文書を検索結果ファイルディレクトリに格納すると共に、検索結果の文書に記載されているURLから得られる、地名と関連語の組が含まれているネットワーク上の文書をキャッシュファイルディレクトリに格納するデータ取得処理部と、ネットワーク上の文書に含まれる関連語を中心とする前後の単語を取り出してネットワーク上の文書のファイル名と共にWAMファイルに書き出し、WAMファイルディレクトリに格納するWAMファイル生成処理部と、同一の地名に係る複数のWAMファイルに基づいて、ネットワーク上の文書群と関連語の有無の関係を関連語出現テーブルに書き出す関連語出現テーブル生成処理部と、関連語出現テーブルを基に全ての各関連語同士の類似度を算出した後、類似度にてソートし、基準となる第一関連語に最も類似度が高い第二関連語のレコードと、第二関連語の次に類似度が高い第三関連語のレコードとを抜粋して、第一関連語、

10

20

30

40

50

第二関連語及び第三関連語を夫々フィールドに持つレコードを列挙した本ベストスコアテーブルを生成するベストスコアテーブル生成処理部と、本ベストスコアテーブルを基に、クライアントへ送信される本クラスタを作成して本クラスタディレクトリに格納する本クラスタ生成処理部とを備え、リアルタイム処理部は、クライアントから得られる現在位置情報を受けて対応する地名をGPS地名マスタから取得して、本クラスタディレクトリから地名に対応する本クラスタを取得した後、所定の文書形式に変換してクライアントに送信する本クラスタ送信部とを備えることを特徴とする。

【0009】

地名や人名等の固有名詞は、事物或は人物そのものを特定する名称であるので、本質的に特有の意味を持たない。しかし、一方で、その土地や人物には、固有に由来する様々な事象がある。つまり、固有名詞は、その人物や事物自体を指し示す観点においては一義的であるものの、その人物や事物に由来する事象において多義的である。

人は固有名詞からそれに由来する様々な事象を連想する。その事象は個々の目的や趣味趣向等によって極めて多彩である。このため、検索システムは固有名詞を検索クエリとして与えられた時に、固有名詞から連想される複数の側面を端的に示すキーワードを用いて、そのような複数の側面を分類する必要がある。

あるキーワードとキーワードとの間の関係は、一つの文書の中に同時に現れるだけでなく、それぞれの出現する場所が極めて近い場合が多い。

発明者はこの点に注目し、関連語同士の出現頻度を算出する対象を、文書中の関連語の前後数ワードに絞った。

更に、関連語同士の類似度を算出し、その上澄みとも言える2レコードだけ取り出し、クラスタリング処理を行った。

【0010】

上記構成によれば、携帯電話の表示画面の大きさにかかわらず、適切に各関連語が分類された本クラスタを作成することができる。

【発明の効果】

【0011】

本発明により、携帯電話のような小さい表示画面に関連語の分類画面を表示する場合でも、ユーザにとって使い勝手が良い情報検索システム及び情報検索装置を提供することができる。

【発明を実施するための最良の形態】

【0012】

以下、本発明の実施の形態の例を、図1～図51を参照して説明する。

【0013】

〔全体構成〕

図1(a)は、本発明の一実施形態例である情報検索システムの概略図である。

また、図1(b)は、本実施形態例に用いられる携帯電話102の機能を示すブロック図である。本実施形態例は、この図1(b)に示す携帯電話102を保有するユーザに対して効果的に情報(コンテンツ)を提供する情報検索システムである。

この情報検索システム101は、携帯電話102、モバイル情報検索サーバ103、検索エンジン104及び地図サーバ105より構成されており、図1(a)に示すように、これらの各要素がインターネット106上で相互に接続されている。

【0014】

携帯電話102は、GPS(Global Positioning System)衛星107からユーザの現在位置情報を取得し、インターネット106を介してその現在位置情報をモバイル情報検索サーバ103に送信する。また、ユーザの操作によって、後述するようなさまざまな指示をモバイル情報検索サーバ103に送信する。

【0015】

モバイル情報検索サーバ103は、携帯電話102から送信されたユーザの指示に基づいて、検索エンジン104及び地図サーバ105よりさまざまな情報を取得する。検索工

10

20

30

40

50

ンジン 104 及び地図サーバ 105 からの情報の取得は、インターネット 106 を介して行われる。モバイル情報検索サーバ 103 は、検索エンジン 104 及び地図サーバ 105 から送信された情報に所定の処理を加え、ユーザの携帯電話 102 に対する操作に応じた h t m l (Hyper Text Markup Language) 文書を周知の H T T P (Hyper Text Transfer Protocol) にて携帯電話 102 へ送信する。

モバイル情報検索サーバ 103 は、情報検索装置ともいえる。

【0016】

検索エンジン 104 は、インターネット 106 に存在する情報(ウェブページ、ウェブサイト、画像ファイル、住所など)を検索する機能を提供する w e bサーバである。

地図サーバ 105 は、入力された位置情報に基づいて、所定の範囲の地図画像をユーザに提供する周知の w e bサーバである。

【0017】

〔携帯電話〕

図 1 (b) は、携帯電話の機能的な構成を示すブロック図である。なお、図 (b) は本実施形態に用いられる機能ブロック図のみを示している。したがって、通話等の、本実施形態と関係のない機能の記載は省略している。

G P S 受信部 108 は、G P S 衛星 107 から受信した電波を受信し、携帯電話 102 のユーザの現在位置情報(緯度・経度等)を取得し、その現在位置情報を入出力制御部 109 へ出力する。

【0018】

G P S 受信部 108 から現在位置情報の入力があると、入出力制御部 109 は、インターネット 106 を介して、モバイル情報検索サーバ 103 へ現在位置情報を送信する。また、モバイル情報検索サーバ 103 から携帯電話 102 へ h t m l 文書が送信された場合、入出力制御部 109 は、その h t m l 文書を受信し、表示制御部 110 を通じて表示部 111 へ送る。そして、表示部 111 において、モバイル検索サーバ 103 から送られた h t m l 文書が表示される。

【0019】

入力部 112 は、操作キー等である。ユーザは、この操作キーを操作することにより、表示部 111 で表示された h t m l 文書の各リンクを選択したり、h t m l 文書の所定の領域に文字列を入力したりする。このようなユーザの入力部 112 に対する操作は、入出力制御部 109 を介して、インターネット 106 上のモバイル情報検索サーバ 103、検索エンジン 104 及び地図サーバ 105 へ送信される。

【0020】

〔モバイル情報検索サーバ〕

図 2 は、モバイル情報検索サーバの機能的な構成を示すブロック図である。

モバイル情報検索サーバ 103 は、大まかには、バッチ処理部 202、リアルタイム処理部 203 及び不揮発性データ記憶部 204 よりなる。

【0021】

バッチ処理部 202 は、例えばシェルスクリプト等により、モバイル情報検索サーバ 103 で行われる処理の一部をサーバのリソースがあまり忙しくない時間帯(例えば、夜間等)に稼動する。ここで処理の一部とは、携帯電話 102 からのユーザの指示を受けなくても行える処理、つまりモバイル情報検索サーバ 103 のみで行える処理のことを指している。具体的には、所定の情報を検索エンジン 104 及び地図サーバ 105 より取得し、これらの情報を携帯電話 102 のユーザに提供するための h t m l 文書を作成するための形式に変換する処理、及び不揮発性データ記憶部 204 に記憶する処理のことである。なお、バッチ処理部 202 の内部構成及びバッチ処理部 202 で行われるより詳細な処理については図 4 ~ 図 33 にて後述する。

【0022】

リアルタイム処理部 203 は、ユーザによる携帯電話 102 の操作に応じた処理をリアルタイムで行う部分である。ここで、リアルタイム処理とは、予めバッチ処理部 202 で

10

20

30

40

50

生成され、不揮発性データ記憶部 204 に記憶された所定の情報に基づいて、携帯電話 102 に適合する html 文書に変換する処理、及び携帯電話 102 へ html 文書を送信する処理のことである。なお、リアルタイム処理部 203 の内部構成及びリアルタイム処理部 203 で行われるより詳細な処理については図 34 ~ 図 46 にて後述する。

【0023】

[バッチ処理部]

図 3 は、バッチ処理部 202 及び不揮発性データ記憶部 204 の一部の機能的な構成を示すブロック図である。

バッチ処理部 202 は、非対話型ウェブクライアント 302 と、データ取得処理部 303 と、本クラスタ作成部 319 とからなる。

データ取得処理部 303 は、非対話型ウェブクライアント 302 を介して所定の情報を取得する。

【0024】

本クラスタ作成部 319 は、データ取得処理部 303 が取得した情報を呼び出し、その情報に基づいて、後述する本クラスタを作成する。なお、本クラスタ作成部 319 は、WAM (Word-Article Matrix) ファイル生成処理部 304 と、関連語出現テーブル生成処理部 305 と、ベストスコアテーブル生成処理部 306 と、本クラスタ生成処理部 307 とからなる。

【0025】

これらバッチ処理部 202 の各ブロックで生成された情報は、不揮発性データ記憶部 204 の所定の領域に適宜記憶される。この不揮発性データ記憶部 204 には、GPS 地名マスタ 308、地名関連語リスト 309、検索結果ファイルディレクトリ 310、キャッシュファイルディレクトリ 311、検索キー・ファイル名対応テーブル 312、WAM ファイルディレクトリ 313、関連語出現テーブル 314、仮ベストスコアテーブル 315 (一時ファイル)、本ベストスコアテーブル 316、仮クラスタ 317 (一時ファイル) 及び本クラスタディレクトリ 318 を記憶する領域が存在する。

不揮発性データ記憶部を構成する上記各要素間の関係とそれぞれの動作については、後述する。

【0026】

以下、バッチ処理部 202 の各ブロックの機能について、図 4 ~ 図 8 を参照して説明する。

【0027】

[データ取得処理部]

図 4 は、データ取得処理部 303 を中心とするブロック図である。

データ取得処理部 303 は、非対話型ウェブクライアント 302 及びインターネット 106 を介して検索エンジン 104 と相互に情報のやり取りを行う。データ取得処理部 303 は、予め不揮発性データ記憶部 204 に記憶してある GPS 地名マスタ 308 にアクセスし、全ての地名の文字列を取得する。この地名の文字列と地図上の経度緯度との関係は後述する図 13 (a) に示されている。

そして、各地名の文字列を順次非対話型ウェブクライアント 302 を介して、検索エンジン 104 に送信する。ここで、検索エンジン 104 から返信される情報を基にして、データ取得処理部 303 は、各地名とその地名に関する複数の単語 (以下、「関連語」という) が対応付けられた、後述する地名関連語リスト 309 を生成し、不揮発性データ記憶部 204 に記憶する。

【0028】

また、データ取得処理部 303 は、不揮発性データ記憶部 204 に地名関連語リスト 309 を生成後、その地名関連語リスト 309 にアクセスし、各関連語毎に「(地名) + (カンマ) + (関連語 1) + (関連語 2) + ... + (関連語 100)」の文字列を取得する。そして、「(地名) + (スペース) + (関連語 1)」、「(地名) + (スペース) + (関連語 2)」、「...」、「(地名) + (スペース) + (関連語 100)」という各文字列 (以下、「地

10

20

30

40

50

名関連語文字列」という)に変換するコマンドを順次実行し、複数の地名関連語文字列を取得する。ただし、地名は複数存在するので、データ取得処理部303が生成する地名関連語文字列は、(全地名の数)×(各地名に対する関連語の数)通りとなる。

【0029】

そして、各地名関連語文字列を非対話型ウェブクライアント302を介して、検索エンジン104に順次送信する。また、データ取得処理部303は、検索エンジン104から返信される検索結果を示す、検索結果htmlファイル及びウェブページのキャッシュファイルにそれぞれ所定の名称を付与し、検索結果htmlファイルディレクトリ及びキャッシュファイルディレクトリ311にそれぞれ記憶する。なお、検索結果htmlファイル及びウェブページのキャッシュファイルについての詳細は、図10及び図11に基づいて後述する。

10

【0030】

また、データ取得処理部303は、地名関連語リスト、検索結果htmlファイルディレクトリに記憶された検索結果htmlファイル名及びキャッシュファイルディレクトリ311に記憶されたキャッシュファイル名をそれぞれ対応付けた、検索キー・ファイル名対応テーブル312を生成する。この検索キー・ファイル名対応テーブル312は、不揮発性データ記憶部204に記憶されるものであるが、その具体的な内容は、後述する図13(b)に示されるようなものである。

ここで、キャッシュファイルに住所文字列が含まれている場合は、データ取得処理部303が、非対話型ウェブクライアント302を介してその住所文字列を検索エンジン104に送信し、その住所に対応する位置情報(緯度・経度)を取得する。そして、これらの住所文字列及び位置情報を検索キー・ファイル名対応テーブル312の所定のフィールドにキャッシュファイル名と対応付けて追記する。

20

【0031】

[WAMファイル生成処理部]

図5は、WAMファイル生成処理部304を中心とするブロック図である。

WAMファイル生成処理部304は、データ取得処理部303で生成され、キャッシュファイルディレクトリ311に記憶されたキャッシュファイル及び検索キー・ファイル名生成テーブル312を参照して、各関連語毎にWAMファイル502を生成する。そして、生成された各WAMファイル502に所定の名称をそれぞれ付与し、WAMファイルディレクトリ313に記憶する。一つの関連語につき一つのWAMファイル502が生成されるので、関連語の数だけWAMファイル502が生成されることになる。

30

【0032】

ここで、WAMファイル502とは、所定の関連語と、その関連語に対応するキャッシュファイル名と、キャッシュファイル内に含まれる関連語の前後にある名詞(以下、「周辺語」という)と、そのキャッシュファイルに含まれる、関連語及び周辺語の出現頻度とが対応付けられて書き込まれたファイルである。なお、WAMファイル502についての詳細な説明及びWAMファイル502の具体的な生成方法については、図14及び図15に基づいて後述する。

【0033】

また、WAMファイル生成処理部304は、データ取得処理部303で生成された検索キー・ファイル名対応テーブル312に、関連語のフィールドの各関連語に対応するようにWAMファイル名を追記し、検索キー・ファイル名対応テーブル312を更新する。

40

【0034】

[関連語出現テーブル生成処理]

図6は、関連語出現テーブル生成処理部305を中心とするブロック図である。

関連語出現テーブル生成処理部305は、WAMファイル生成処理部304で生成され、WAMファイルディレクトリ313に記憶されたWAMファイル502から関連語出現テーブル314を生成する処理を行う。この関連語出現テーブル生成処理部305の処理により、一つの地名に対応する複数の関連語毎の、全てのWAMファイル502から一つ

50

の関連語出現テーブル 314 が生成される。

【0035】

ここで、関連語出現テーブル 314 とは、一つの地名に対する複数の関連語が、各キャッシュファイル内に含まれるか否かの判定が書き込まれたテーブルである。この関連語出現テーブル 314 は、GPS 地名マスタ 308 に含まれる地名の数だけ生成されることになる。なお、関連語出現テーブル 314 についての詳細な説明及び関連語出現テーブル 314 の具体的な生成方法については、図 16 及び図 17 に基づいて後述する。

【0036】

[ベストスコアテーブル生成処理部]

図 7 は、ベストスコアテーブル生成処理部 306 を中心とするブロック図である。 10

ベストスコアテーブル生成処理部 306 は、WAM ファイル生成処理部 304 で生成され、WAM ファイルディレクトリ 313 に記憶された WAM ファイル 502、及び関連語出現テーブル生成処理部 305 で生成された関連語出現テーブル 314 に基づいて、仮ベストスコアテーブル 315 (一時ファイル) を不揮発性データ記憶部 204 に生成する。そして、仮ベストスコアテーブル 315 に対し所定の処理及び計算等を行い、仮スコアテーブルを順次更新し、本ベストスコアテーブル 316 を生成し、不揮発性データ記憶部 204 に記憶する。本ベストスコアテーブル 316 は、GPS 地名マスタ 308 に含まれる地名の数だけ生成される。ここで、ベストスコアテーブルとは、所定の地名の関連語から、所定の関連語との類似度が最も高くなる他の関連語 2 語を選び、これら二つの関連語とその類似度を、各関連語毎に書き込んだテーブルのことである。なお、ベストスコアテーブルについての詳細な説明及びベストスコアテーブルの具体的な生成方法については、図 18 ~ 図 21 に基づいて詳細に説明する。 20

【0037】

[本クラスタ生成処理部]

図 8 は、本クラスタ生成処理部 307 を中心として機能させた場合の機能ブロック図である。

本クラスタ生成処理部 307 は、ベストスコアテーブル生成処理部 306 で生成されたベストスコアテーブルから、仮クラスタ 317 (一時ファイル) を不揮発性データ記憶部 204 に生成する。そして、仮クラスタ 317 に対し、所定の処理及び計算を行い、仮クラスタ 317 を順次更新し、本クラスタ 802 を生成し、本クラスタディレクトリ 318 30

【0038】

ここで、本クラスタ 802 とは、所定の地名の関連語全てをグループ分けし、そのグループに含まれる関連語について、類似度の合計の高い順に並べたテーブルである。なお、全ての関連語と記載したが、不必要な関連語に関連する項目は本クラスタ 802 には含まれていない。本クラスタ 802 についての詳細な説明及び本クラスタ 802 の具体的な生成方法については、図 22 ~ 図 32 に基づいて行われる。

【0039】

[バッチ処理の動作説明]

次に、本実施形態の情報検索システム 101 におけるモバイル情報検索サーバ 103 のバッチ処理の流れを説明する。 40

図 9 は、モバイル情報検索サーバ 103 を中心とするバッチ処理の動作の流れを示すシーケンス図である。

まず、モバイル情報検索サーバ 103 (クライアント) は、予め GPS 地名テーブルに記憶してある地名の文字列を検索エンジン 104 側へ送信する (ステップ S901)。

【0040】

検索エンジン 104 は、地名の文字列を受けて、その地名の関連語 100 語をモバイル情報検索サーバ 103 に返信する (ステップ S902)。

【0041】

図10(a)は、地名とその関連語の例を示す図である。

まず、関連語の詳細について図10(a)を参照して説明する。

例えば不特定多数の人が、検索エンジン104による検索を行う際、「新宿」プラス「単語X」を検索クエリとして入力したとする。このとき「単語X」が「新宿」の関連語となる。本実施例では、このような「単語X」の中で、「新宿」という単語と最も多い頻度で組み合わせられた「単語X」のうち上位100語を関連語とする。なお、検索クエリとは、ユーザが検索エンジンからウェブページを訪問する際に検索エンジンから入力したキーワードのことである。

【0042】

図10(a)では、「ルミネ(登録商標)」が1番多い頻度で「新宿」と組み合わせられた関連語1である。「イタリアン」及び「レストラン」は、2番目及び3番目多い頻度で「新宿」と組み合わせられた関連語2及び関連語3である。ただし、「新宿」は地名の一例であるので、「新宿」以外の地名の関連語も存在する。

10

【0043】

再び、図9の説明に戻る。

モバイル情報検索サーバ103は、関連語100語を受信すると、前述の地名関連語文字列100通りを検索エンジン104側に送信する(ステップS903)。

【0044】

検索エンジン104は、100通りの地名関連語文字列を受けて、各地名関連語文字列を検索クエリとして検索を実行する。そして、100通りの地名関連語文字列毎の検索結果のhtmlファイルである図10(b)に示す検索結果htmlファイル1003をモバイル情報検索サーバ103(クライアント)側へ送信する(ステップS904)。

20

【0045】

モバイル情報検索サーバ103は、検索結果htmlファイル1003を受信すると、各検索結果htmlファイル1003毎にリンクが付与されている1000件のキャッシュページのリンク先URL1004を検索エンジン104に送信する(S905)。ここでは、クライアントが受信した検索結果htmlファイル1003は100通りなので、クライアントから検索エンジン104に送信されるキャッシュページのリンク先URL1004は100×1000件となる。

【0046】

30

検索エンジン104は、キャッシュページのリンク先URL1004を受信すると、そのURLにアクセスして、図11に示すキャッシュファイル1102を取得する。そして、検索エンジン104は、1つの検索結果htmlファイル1003に付き、キャッシュファイル1102が取得できたものの中から100件のキャッシュファイル1102をクライアント側へ送信する(S906)。ここでは、検索結果htmlファイル1003は、100通りなので、検索エンジン104からモバイル情報検索サーバ103側に送信されるキャッシュファイル1102は100×100件となる。

【0047】

そして、モバイル情報検索サーバ103側のGPS地名マスタ308に書き込まれている地名文字列の数だけステップS901～ステップS906の処理を繰り返す(ステップS907)。

40

【0048】

ステップS907の処理が完了すると、モバイル情報検索サーバ103は、地名及び検索エンジン104から取得した、関連語、検索結果htmlファイル1003及びキャッシュファイル1102に対し所定の内部処理を行い前述の本クラスタ802を生成する。

【0049】

[バッチ処理メインフロー]

図12は、本実施形態の情報検索システム101において、バッチ処理部202の動作を示すフローチャートである。

データ取得処理部303(図3を参照)は、GPS地名マスタ308に含まれる地名の

50

検索を開始する(ステップS1201)。

【0050】

図13(a)は、GPS地名マスタ308を示す図である。GPS地名マスタ308の詳細について図13(a)を参照して説明する。前述したように、GPS地名マスタ308は、予め不揮発性データ記憶部204に蓄積されている。このGPS地名マスタ308の「地名」フィールドには、例えば「新宿」等各地の地名文字列が書き込まれている。また、「緯度範囲」フィールドには、同一のレコードにある地名の緯度範囲が書き込まれている。例えば「新宿」の緯度範囲は、「(x,)」となる。また、「経度範囲」フィールドには、同一のレコードにある地名の経度範囲が書き込まれている。例えば「新宿」の経度範囲は、「(,)」となる。

10

【0051】

再び、図12の説明に戻る。

これ以降はループ処理である。最初に、カウンタを構成する変数*i*を1に初期化する(ステップS1202)。データ取得処理部303は、非対話型ウェブクライアント302を介して、GPS地名マスタ308の*i*番目のレコードの「地名」をインターネット106上の検索エンジン104に送信する。そして、送信した「地名」の関連語100語の文字列(図10(a)を参照)を検索エンジン104から取得する(ステップS1203)。そして、地名及びその100語の関連語がフィールドセパレータで区切られた地名関連語リスト309を不揮発性データ記憶部204に記憶する。

【0052】

次に、データ取得処理部303は、地名関連語リスト309に書き込まれた関連語100語の文字列を「(地名)+(スペース)+(関連語1)」、「(地名)+(スペース)+(関連語2)」、...、「(地名)+(スペース)+(関連語100)」に変換するコマンドを実行し、地名関連語文字列を100件取得する。そして、非対話型ウェブクライアント302を介して、100件の地名関連語文字列をインターネット106上の検索エンジン104に順次送信し、各地名関連語文字列を含むウェブページの検索結果htmlファイル1003(図10(b)を参照)100件を検索エンジン104から取得する(ステップS1204)。その後、各検索結果htmlファイル1003毎に検索結果htmlファイル名を付与し、検索結果ディレクトリに記憶する。

20

【0053】

次に、データ取得処理部303は、検索結果ディレクトリに記憶された各検索結果htmlファイル1003毎に記載された1000件のキャッシュページのリンク先URLに、アクセスする。このアクセスは、非対話型ウェブクライアント302及び検索エンジン104を介して行われる。そして、各検索結果htmlファイル1003につき100件のキャッシュファイル1102(図11を参照)を、キャッシュファイル1102を取得できるキャッシュページ100件から取得する(ステップS1205)。そして、各キャッシュファイル1102毎にキャッシュファイル名を付与し、キャッシュファイルディレクトリ311に記憶する。

30

【0054】

ここで、データ取得処理部303は、キャッシュファイル1102内に住所文字列が含まれているか否かを判定する。住所文字列が含まれているならば、この住所文字列を非対話型ウェブクライアント302を介して、インターネット106上の検索エンジン104へ送信する。そして、住所に応じた位置情報(緯度、経度)を検索エンジン104から受信する。

40

【0055】

そして、データ取得処理部303は、各地名と、各検索結果htmlファイル名と、各キャッシュファイル名と、各住所と、各位置情報(緯度、経度)とが対応付けられた検索キー・ファイル名対応テーブル312(図13(b)を参照)を生成し、不揮発性データ記憶部204に記憶する。ただし、住所文字列を含まないキャッシュファイル1102と同一レコードにある住所フィールド及び位置情報(緯度、経度)フィールドについては、

50

空のままにしておく。また、検索キー・ファイル名対応テーブル312のWAMファイル名のフィールドについては、後述する。

【0056】

次に、WAMファイル生成処理部304は、キャッシュファイルディレクトリ311に記憶されたキャッシュファイル1102及び検索キー・ファイル名対応テーブル312に書き込まれている情報に基づいて各関連語毎にWAMファイル502を生成する(ステップS1206)。そして、各WAMファイル502に名称を付与し、WAMファイルディレクトリ313に記憶する。それと同時に、関連語とWAMファイル502が対応するように、各WAMファイル502の名称を検索キー・ファイル名対応テーブル312のWAMファイル名のフィールドに書き込み、検索キー・ファイル名対応テーブル312を更新する。このステップS1206のWAMファイル生成処理の詳細は、図14に基づいて後述する。

10

【0057】

次に、関連語出現テーブル生成処理部305は、WAMファイルディレクトリ313に記憶されたWAMファイル502に基づいて、関連語出現テーブル314を生成し(ステップS1207)、不揮発性データ記憶部204に記憶する。なお、このステップS1207の関連語出現テーブル生成処理については、図16に後述する。

【0058】

次に、WAMファイルディレクトリ313に記憶されたWAMファイル502及び関連語出現テーブル生成処理部305で生成された関連語出現テーブル314に書き込まれている情報に基づいて、ベストスコアテーブルが生成され(ステップS1208)、不揮発性データ記憶部204に記憶する。

20

ステップS1208のベストスコアテーブル生成処理については、図18に後述する。

【0059】

次に、ベストスコアテーブル生成処理部306で生成され、ベストスコアテーブルに書き込まれている情報に基づいて、本クラスタ802を生成する(ステップS1209)。そして、ステップS1209で生成された本クラスタ802が、本クラスタディレクトリ318に記憶される。このステップS1209の本クラスタ生成処理の詳細については、図22に後述する。

【0060】

以上の処理完了後、カウンタを構成する変数*i*をインクリメントする(ステップS1210)。

30

そして、変数*i*がGPS地名マスタ308の最大レコード数を越えたか否かを確認し(ステップS1211)、超えていなければ再び繰り返す(ステップS1211のN)。超えていれば(ステップS1211のY)、処理を終了する(ステップS1212)。

【0061】

[WAMファイル生成処理フロー]

図14は、図12のステップS1206のWAMファイル生成処理の詳細な処理の流れを示すフローチャートである。

【0062】

図12のステップS1206で、WAMファイル生成処理部304がWAMファイル生成処理を開始すると(ステップS1401)、まず、現在対象としている地名で検索キー・ファイル名対応テーブル312(図13(b)を参照)を絞り込む(ステップS1402)。

40

【0063】

以降はループ処理である。最初に、カウンタを構成する変数*j*を1に初期化する(ステップS1403)。次に、検索キー・ファイル名対応テーブル312の*j*番目のレコードの検索結果HTMLファイル名を一時変数*fname*に代入する(ステップS1404)。そして、この一時変数*fname*からWAMファイル名を生成する(ステップS1405)。

【0064】

50

続いて、WAMファイル生成処理部304（図3参照）は、地名で絞り込まれた検索キー・ファイル名対応テーブル312のj番目のレコードのキャッシュ名でキャッシュファイルディレクトリ311を検索する。そして、検索したキャッシュファイル1102をプレーンテキストに変換する（ステップS1406）。

【0065】

次に、WAMファイル生成処理部304が、プレーンテキストに対し形態素解析を行い、プレーンテキストの形態素から名詞のみを抽出する。そして、抽出した名詞の形態素のテキスト（以下、「形態素テキスト」という）を生成する（ステップS1407）。

ここで、プレーンテキストに対して行われる形態素解析とは、対象言語の文法の知識（文法のルールの集まり）や辞書（品詞等の情報付きの単語リスト）を情報源として用い、自然言語で書かれた文書を形態素（言語で意味を持つ最小単位）の列に分割し、それぞれの品詞を判別する作業のことである。

【0066】

次に、WAMファイル生成処理部304は、形態素テキストを検索キー・ファイル名対応テーブル312のj番目のレコードの関連語で検索し、関連語及びその周辺語を抽出する。そして、ステップS1404の処理にて生成された名称のWAMファイル502へ出力する（ステップS1408）。

このステップS1408の処理の際には、検索キー・ファイル名対応テーブル312のj番目のレコードのキャッシュファイル名と、形態素テキストにおける関連語と各周辺語の出現頻度とが、同時にステップS1404にて生成された名称のWAMファイル502へ出力される。なお、本実施形態例では、周辺語の定義を、形態素テキスト内で関連語に一致する形態素の前後に2語ずつある、合計4語の形態素としている。

【0067】

以上の処理完了後、カウンタを構成する変数jをインクリメントする（ステップS1409）。

そして、検索キー・ファイル名対応テーブル312に、インクリメントしたj番目のレコードが存在するか否かを確認する（ステップS1410）。そして、インクリメントしたj番目のレコードが存在する場合は（ステップS1410のN）、続いて、fnameと、検索キー・ファイル名対応テーブル312のj番目のレコードの検索結果htmlファイル名が一致するか否かを確認する（ステップS1411）。

そして、両者が一致すると判定された場合は（ステップS1411のY）、ステップS1406の処理に戻り、再びステップS1406以降の処理を繰り返す。検索キー・ファイル名対応テーブル312のj番目のレコードの検索結果htmlファイル名がfnameと一致しない場合は（ステップS1411のN）、ステップS1403の処理に戻り、それ以降の処理を再び繰り返す。

【0068】

一方、ステップS1410において、検索キー・ファイル名対応テーブル312のi番目のレコードが存在しないと判定された場合は（ステップS1410のY）、処理を終了する（ステップS1412）。

【0069】

図15は、WAMファイル生成処理によって生成されるWAMファイル502の例を示す図である。

WAMファイル502は、検索結果htmlファイル名が「shinjyuku_001」、すなわちWAMファイル名が「shinjyuku_001.wam」（図14のステップS1404の処理を参照）で、関連語が「新宿二丁目」（図13（b）を参照）のWAMファイルである。WAMファイル502を例に、WAMファイルの構造について説明する。

【0070】

WAMファイル502の括弧部分1502は、図14のステップS1405の処理～ステップS1407の処理（一回目）で生成される箇所に相当する。この括弧部分1502に注目すると、一行目には、図13（b）に示す検索キー・ファイル名対応テーブル312

10

20

30

40

50

の1番目のレコードのキャッシュファイル名である「shinjyuku_001_001」が書かれている。また、2行目以降には、関連語である「新宿二丁目」と、その周辺語である「東京」、「都」、「写真」、「街」、「開催」、「バー」及び「こんど」と、が記載されている。関連語及び各周辺語の左には、キャッシュファイル1102「shinjyuku_001_001」の形態素テキストに出現する関連語及び各周辺語の出現頻度が書かれている。なお、WAMファイル502において、キャッシュファイル名「shinjyuku_001_002」、「shinjyuku_01_003」、...、「shinjyuku_001_100」に相当する箇所は、ステップS1405～ステップS1407の処理(二回目、三回目、...、百回目)にて書き込まれる。なお、上述したように、WAMファイル502は、一つの地名につき、その関連語の数だけ生成される。

【0071】

[関連語出現テーブル生成処理フロー]

図16は、図12のステップS1207の関連語出現テーブル生成処理の詳細な処理の流れを示すフローチャートである。

【0072】

図12のステップS1207で説明したように、関連語出現テーブル生成処理部305が関連語出現テーブル314生成処理を開始すると(ステップS1601)、まず、現在対象としている地名に対応する全てのWAMファイル502(図13(b)を参照)から図17(a)に示す全フィールドが“0”の関連語出現テーブル314を作成する(ステップS1602)。

ここで、図17(a)に示す関連語出現テーブル314の関連語のレコードの「term(1)」、「term(2)」、「term(3)」、...、「term(100)」は、ステップS1602の処理で対象とした各WAMファイル502に対応する関連語に相当する。また、関連語出現テーブル314のキャッシュファイル名のフィールドの「shinjyuku_001_001」、「shinjyuku_001_002」、...、「shinjyuku_100_100」は、ステップS1602の処理で対象とした全てのWAMファイル502に書かれているキャッシュファイル名に相当する。

【0073】

再び、図16の説明に戻る。

これ以降はループ処理である。最初に、カウンタを構成する変数 k 、 l 、 m を1に初期化する(ステップS1603、ステップS1604及びステップS1605)。そして、1番目のWAMファイル502(図15を参照)の m 行目がキャッシュファイル名であるか否かを確認する(ステップS1606)。キャッシュ名であるならば(S1606のY)、そのキャッシュ名に対応する関連語出現テーブル314のレコードに移動する(ステップS1607)。

【0074】

キャッシュ名でないならば(ステップS1606のN)、1番目のWAMファイル502の m 行目の単語が関連語出現テーブル314の k 番目のフィールドに対応する関連語と一致するか否かを確認する(ステップS1608)。両者が一致すると判定された場合は(ステップS1608のY)、ステップS1607の処理で移動した、関連語出現テーブル314のレコードの k 番目のフィールドに“1”を書き込む(ステップS1609)。

【0075】

以上の処理完了後、カウンタを構成する変数 m をインクリメントする(ステップS1610)。

続いて、1番目のWAMファイル502において、インクリメントした m 行目が存在するか否かを確認する(ステップS1611)。1番目のWAMファイル502の m 行目が、まだ存在すると判定された場合は(ステップS1611のN)、ステップS1606の処理に戻り、それ以降の処理を繰り返す。ステップS1611の処理で1番目のWAMファイル502の m 行目が存在しないと判定された場合は(ステップS1611のY)、カウンタを構成する変数 l をインクリメントする(ステップS1612)。

【0076】

以上の処理完了後、続いて、インクリメントされた1番目のWAMファイル502が存

10

20

30

40

50

在するか否かを確認する（ステップS 1 6 1 3）。1番目のWAMファイル5 0 2が、まだ存在すると判定された場合は（ステップS 1 6 1 3のN）、ステップS 1 6 0 5の処理に戻り、それ以降の処理を繰り返す。1番目のWAMファイル5 0 2が存在しないと判定されると（ステップS 1 6 1 3のY）、カウンタを構成する変数kをインクリメントする（ステップS 1 6 1 4）。

【0077】

次に、関連語出現テーブル3 1 4のk番目のフィールドが存在するか否かを確認する（ステップS 1 6 1 5）。そして、インクリメントされたk番目のフィールドが関連語出現テーブル3 1 4に存在すると判定された場合は（ステップS 1 6 1 5のN）、ステップS 1 6 0 4の処理に戻り、以降の処理を繰り返す。関連語出現テーブル3 1 4にk番目のフィールドが存在しないと判定された場合は（ステップS 1 6 1 5のY）、関連語出現テーブル3 1 4の全フィールドが“0”のレコードを削除し（ステップS 1 6 1 6）、処理を終了する（ステップS 1 6 1 7）。

10

【0078】

図17は、関連語出現テーブルの例を示す図である。

図17(a)は、初期状態の関連語出現テーブルである。この関連語出現テーブルは、図16のステップS 1 6 0 2の処理で生成されるものであり、その全フィールドには“0”が書き込まれている。

【0079】

図17(b)は、不要レコード削除前の関連語出現テーブルを示す図である。この関連語出現テーブルは、図16のステップS 1 6 1 5において、インクリメントされたk番目のレコードがないとき（ステップS 1 6 1 5のY）に生成されるテーブルであり、テーブルの各フィールドには、“0”或いは“1”が書き込まれている。

20

また、図17(b)に示す関連語出現テーブルには、キャッシュファイル名の形態素テキストに関連語が含まれる場合は“1”が、含まれない場合は“0”が書き込まれている。例えば、キャッシュファイル名「shinjyuku_001_001」の形態素テキストには、関連語「tem1(1)」が含まれていない。そのため、関連語出現テーブル3 1 4の「shinjyuku_001_001」のレコードの「temp(1)」のフィールドには、“0”が書き込まれている。また、キャッシュファイル名「shinjyuku_100_001」の形態素テキストには、関連語「tem1(3)」が含まれている。そのため、関連語出現テーブル3 1 4の「shinjyuku_100_001」のレコードの「temp(3)」のフィールドには、“1”が書き込まれている。

30

【0080】

図17(c)は、不要レコード削除後の関連語出現テーブルを示す図である。

この関連語出現テーブル3 1 4は、図16に示すステップS 1 6 1 6の処理により、図17(b)に示す関連語出現テーブル3 1 4から全フィールドが“0”のレコードが削除されたものである。例えば、図17(b)に示す関連語出現テーブル3 1 4の「shinjyuku_001_001」のレコードに注目すると、「term(1)」～「term(100)」までのすべてのフィールドが“0”となっている。このような、レコードがステップS 1 6 1 6の処理にて削除されることにより、図17(c)の関連語出現テーブル3 1 4が生成される。

【0081】

[ベストスコアテーブル生成処理フロー]

図18及び図19は、図12のステップS 1 2 0 8のベストスコアテーブル生成処理の詳細な処理の流れを示すフローチャートである。

40

【0082】

図12のステップS 1 2 0 8で、ベストスコアテーブル生成処理部3 0 6がベストスコアテーブル生成処理を開始すると（ステップS 1 8 0 1）、それ以降はループ処理となる。

最初に、カウンタを構成する変数n, pを1に初期化する（ステップS 1 8 0 2及びステップS 1 8 0 3）。次に、nとpとが等しいか否かを確認する（ステップS 1 8 0 4）。nとpが等しくない場合には（ステップS 1 8 0 4のY）、不揮発性データ記憶部2 0

50

4に記憶された関連語出現テーブル314(図17(c)を参照)のn番目のフィールドに対応する関連語を配列変数term_xに、p番目のフィールドの関連語を配列変数term_yに追記する(ステップS1805)。

【0083】

次に、関連語出現テーブル314のn番目のterm(n)とp番目のterm(p)のいずれかに“1”が立っているレコード数を、配列変数ornumに追記する(ステップS1806、図20参照)。そして、関連語出現テーブル314のn番目のterm(n)とp番目のterm(p)の両方に“1”が立っているレコード数を、配列変数andnumに追記する(ステップS1807、図20参照)。

【0084】

そして、ステップS1807の処理の最後に、図20に示すandnumとornumへ追記された各値に基づいて、term_xとterm_yに最後に追記された関連語間の類似度を計算し、この類似度を、配列変数scoreに追記する(ステップS1808)。ここで、類似度とは、相異なる2つの単語間にある関連性の強さを数値で表したものである。別の言い方をすると、これらの2つの単語の共起関係を数値で示したものである。なお、類似度の計算式を以下に示す。

【0085】

{数1}

score=(andnum/ornum)*log(andnum)

【0086】

以上の処理完了後、カウンタを構成する変数pをインクリメントする(ステップS1809)。

そして、関連語出現テーブル314のp番目のフィールドが存在するか否かを確認する(ステップS1810)。p番目のフィールドが存在する場合は(ステップS1810のN)、ステップS1804の処理に戻り、それ以降の処理を繰り返す。p番目のフィールドが存在しないと判定された場合は(ステップS1810のY)、続いて、カウンタを構成する変数nをインクリメントする(S1811)。

【0087】

次に、関連語出現テーブル314のn番目のフィールドが存在するか否かを確認する(ステップS1812)。n番目のフィールドが存在する場合は(ステップS1812のN)、ステップS1803の処理に戻り、以降の処理を繰り返す。

【0088】

以上の処理完了後、ステップS1812で、関連語出現テーブル314のn番目のフィールドが存在しないと判定された場合は(ステップS1812のY)、図19の処理に移行する(囲みA)。そして、上述したscore、andnum、ornum、term_x及びterm_yの各配列から、図20(a)に示すソート前仮ベストスコアテーブル2002を生成する(ステップS1901)。

【0089】

そして、図20(a)に示したソート前仮ベストスコアテーブル2002のterm_xが一致するレコードの中で、score(類似度)の大きさが上位2に含まれるレコードのみ残して他のレコードは削除し、図20(b)に示すソート済仮ベストスコアテーブル2003を生成する(ステップS1902)。このソート済仮ベストスコアテーブル2003が、図3で示した仮ベストスコアテーブル315に相当する。以下、このソート済仮ベストスコアテーブル2003を、仮ベストスコアテーブル315と呼ぶ。

【0090】

これ以降はループ処理である。最初、カウンタを構成する変数qを初期化する(ステップS1903)。ここで、仮ベストスコアテーブル315のq番目と(q+1)番目の2レコード2102(図21(a)を参照)より、ソート前本ベストスコアテーブル316の(q+1)/2番目の1レコード2103(図21(b)を参照)を生成する(ステップS1904)。すなわち、2レコードを1レコードに変換する処理を行う。具体的には

10

20

30

40

50

、仮ベストスコアテーブル 3 1 5 の q 番目のレコードの score 及び term_y を、score₁ 及び term₁ にそれぞれ書き込み、(q + 1) 番目のレコードの score 及び term_y は score₂ 及び term₂ にそれぞれ書き込む。

【 0 0 9 1 】

また、仮ベストスコアテーブル 3 1 5 の q 番目及び (q + 1) 番目のレコードの term_x は等しいので、term_x を term に書き込む。なお、ソート前本ベストスコアテーブル 3 1 6 の total のレコードに書き込まれる値については、後述する。

【 0 0 9 2 】

ステップ S 1 9 0 4 の処理完了後、ソート前本ベストスコアテーブル 3 1 6 の (q + 1) / 2 番目のレコードの score₁ と score₂ の和を当該レコードの total に書き込む。

10

【 0 0 9 3 】

以上の処理完了後、カウンタを構成する変数 q を 2 インクリメントする (ステップ S 1 9 0 6) 。そして、仮ベストスコアテーブル 3 1 5 の q 番目のレコードが存在するか否かを確認する (ステップ S 1 9 0 7) 。

【 0 0 9 4 】

仮ベストスコアテーブル 3 1 5 にインクリメントされた q 番目のレコードが存在する場合は (ステップ S 1 9 0 7 の N) 、ステップ 1 9 0 4 の処理に戻り、それ以降の処理を繰り返す。一方、仮ベストスコアテーブル 3 1 5 に q 番目のレコードが存在しないと判定された場合は (ステップ S 1 9 0 7 の Y) 、図 2 1 (c) に示すように、ソート前本ベストスコアテーブル 3 1 6 の各レコードの total の値の大きい順に、当該各レコードを並び替

20

【 0 0 9 5 】

[本クラスタ生成処理フロー]

図 2 2 は、図 1 2 のステップ S 1 2 0 6 の処理である、本クラスタ生成処理の詳細な処理の流れを示すフローチャートである。

【 0 0 9 6 】

まず、本クラスタ生成処理部 3 0 7 が本クラスタ 8 0 2 生成処理を開始する (ステップ S 2 2 0 1) 。これ以降はループ処理である。

30

最初に、カウンタを構成する変数 r を 1 に初期化する (ステップ S 2 2 0 2) 。そして、不揮発性データ記憶部 2 0 4 に記憶された本ベストスコアテーブル 3 1 6 (図 2 1 (c) を参照) の r 番目のレコードから仮クラスタ 3 1 7 の cid=r , tid=1 のレコード (図 2 3 (a) を参照) を生成する (ステップ S 2 2 0 3) 。具体的には、本ベストスコアテーブル 3 1 6 の r 番目のレコードの「total」、「term」、「score₁」、「term₁」、「score₂」及び「term₂」を仮クラスタ 3 1 7 の cid=r , tid=1 のレコードの「total」、「term」、「score₁」、「term₁」、「score₂」及び「term₂」にそれぞれ書き込む。

【 0 0 9 7 】

次に、ステップ S 2 2 0 3 の処理完了後、本ベストスコアテーブル 3 1 6 から仮クラスタ 3 1 7 の cid=r , tid=2 のレコード (図 2 3 (b) を参照) を生成する (ステップ S 2 2 0 4) 。具体的には、まず本ベストスコアテーブル 3 1 6 の r 番目のレコードの「term₁」を仮クラスタ 3 1 7 の cid=r , tid=2 のレコードの「term」に書き込む。そして、その「term」で本ベストスコアテーブル 3 1 6 の term フィールドを検索し、該当するレコードの「total」、「score₁」、「term₁」、「score₂」及び「term₂」を仮クラスタ 3 1 7 の cid=r , tid=2 のレコードの「total」、「score₁」、「term₁」、「score₂」にそれぞれ書き込む。

40

【 0 0 9 8 】

ステップ S 2 2 0 4 の処理完了後、本ベストスコアテーブル 3 1 6 から仮クラスタ 3 1 7 の cid=r , tid=3 のレコードを生成する (ステップ S 2 2 0 5) 。具体的には、まず本ベストスコアテーブル 3 1 6 の r 番目のレコードの「term₂」を仮クラスタ 3 1 7 の cid=r

50

,tid=3のレコードの「term」に書き込む。そして、その「term」で本ベストスコアテーブル316のtermフィールドを検索し、該当するレコードの「total」、「score_1」、「term_1」、「score_2」及び「term_2」を仮クラスタ317のcid=r,tid=3のレコードの「total」、「score_1」、「term_1」、「score_2」にそれぞれ書き込む。

【0099】

そして、ステップS2205の処理完了後、仮クラスタ317のcid=r,tid=1,2,3のレコードの各「term」の文字列長を当該レコードの「len」にそれぞれ書き込む(ステップS2206)。なお、以上の処理1回により生成される仮クラスタ317を図23(c)に示す。

【0100】

ここで、図23の仮クラスタ317の「cid」及び「tid」のフィールドについて説明する。

「cid」とは、クラスタ番号であり、「term」、つまり関連語を所定のクラスタに分類するための番号のことである。このクラスタとは、類似した特性を持つ複数の関連語の集合のことを言う。つまり、同一のクラスタ番号が付されている各関連語は、類似した特性を有している。また、クラスタ番号が小さいほど、分類された関連語と地名の関連性(共起関係)が強いということを意味する。また、「tid」とは、ターム番号であり、同一クラスタ番号の各レコード内での「term」の順位付けを行うための番号である。すなわち、仮クラスタとは、「cid」、「tid」の優先順位で各レコードが並んでいるテーブルである。

【0101】

なお、仮クラスタ317の「cnt」、「valid」、「alt」及び「stop」の各フィールドについての詳細は、図26,図28,図30及び図31にて後述する。

【0102】

再び、図22の説明に戻り、本クラスタ生成処理について説明する。

以上の処理完了後、カウンタを構成する変数rをインクリメントする(ステップS2207)。続いて、このインクリメントされたr番目のレコードが本ベストスコアテーブル316に存在するか否かを確認する(ステップS2208)。そして、r番目のレコードが存在する場合は(ステップS2208のN)、ステップS2203の処理に戻る。そして、本ベストスコアテーブル316のレコードの数に応じた回数の処理を繰り返し、図24に示す仮クラスタ317を生成する。

【0103】

インクリメントされたr番目のレコードが本ベストスコアテーブル316に存在しない場合、つまり、仮クラスタ317の生成が完了した場合は(ステップS2208のY)、仮クラスタ317の「cnt」のフィールドに所定の値を書き込むcnt付与処理を行う(ステップS2209)。

【0104】

[cnt付与処理フロー]

図25は、図22のステップS2209のcnt付与処理の詳細な処理の流れを示すフローチャートである。

【0105】

本クラスタ生成処理部307がcnt付与処理を開始すると(ステップS2501)、図25に示すループ処理が行われる。

最初に、カウンタを構成する変数s1に初期化し、更に仮クラスタ317の全レコードの「cnt」を0に初期化する(ステップS2502)。次に、カウンタを構成する変数tを1に初期化する(ステップS2503)。次に、図24に示す仮クラスタ317のcid=s,tid=tのレコードの「term_1」が、cid=s,tid=1,2,3の各レコードの「term」にあるか否かを確認する(ステップS2504)。「term_1」がある場合には(ステップS2504のY)、仮クラスタ317のcid=s,tid=tのレコードの「cnt」に1を加算して書き込む(ステップS2505)。

10

20

30

40

50

【0106】

そして、仮クラスタ317のcid=s,tid=tのレコードの「term_2」が、cid=d,tid=1,2,3の各レコードの「term」にあるか否かを確認する(ステップS2506)。「term_2」がある場合には(ステップS2504のY)、仮クラスタ317のcid=s,tid=tのレコードの「cnt」に1を加算して書き込む(ステップS2507)。

【0107】

以上の処理完了後、カウンタを構成する変数tをインクリメントする。そして、変数tが3より大きいか否かを確認する(ステップS2509)。インクリメントされた変数tが3以下であれば(ステップS2509のN)、ステップS2504の処理に戻り、処理を繰り返す。変数tが3より大きい場合には(ステップS2509のY)、続いて、カウンタを構成する変数sをインクリメントする。

10

【0108】

そして、本ベストスコアテーブル316にインクリメントされたs番目のレコードが存在するか否かを確認する(ステップS2511)。s番目のレコードが存在する場合は(ステップS2511のN)、ステップS2503の処理に戻り、処理を繰り返す。s番目のレコードが存在しないならば(ステップS2511のY)、処理を終了する(ステップS2512)。

【0109】

図26は、cnt付与処理によりcnt値が書き込まれた状態の仮クラスタ317の例を示す図である。このcnt値について図26を参照して説明する。仮クラスタ317の「cnt」のフィールドには、cnt値として0,1又は2のいずれかの値が書き込まれている。

20

【0110】

仮クラスタ317のcid=1,tid=1のレコードに注目する。このレコードの「南口(term_1)」及び「西口(term_2)」の両方が、仮クラスタ317のcid=1,tid=1,2,3のいずれかのレコードの「term」に含まれている。このような場合、cnt値は2となる。

【0111】

また、仮クラスタ317のcid=2,tid=2のレコードに注目する。このレコードの「西口(term_1)」及び「東口(term_2)」の片方、すなわち「西口(term_1)」しか仮クラスタ317のcid=2,tid=1,2,3の各レコードの「term」に含まれていない。このような場合、cnt値は1となる。

30

【0112】

また、仮クラスタ317のcid=6,tid=3のレコードに注目する。このレコードの「野村ビル(term_1)」及び「住友ビル(term_2)」のいずれも、仮クラスタ317のcid=6,tid=1,2,3の各レコードの「term」に含まれていない。このような場合、cnt値は0となる。

【0113】

すなわち、cnt値とは、所定の関連語(term)が、同一クラスタ番号に属する他の2つの関連語(term)との関連性を示す値である。cnt値“2”は関連性がすごくあるということを、cnt値“1”は関連性があるということを、cnt値“0”は関連がないということを、それぞれ示している。

40

【0114】

再び、図22に戻り、本クラスタ生成処理について説明する。

以上、図22のステップS2209のcnt付与処理について図25、図26に基づいて説明したが、このcnt付与処理が完了すると、図26に示す仮クラスタ317の「valid」のフィールドに所定の値を書き込むvalid付与処理が行われる(ステップS2210)。次に、このvalid付与処理について、その詳細を説明する。

【0115】

[valid付与処理フロー]

図27は、図22のステップS2210のvalid付与処理の詳細な処理の流れを示

50

すフローチャートである。

以下、図 2 6 に示す `cnt` が付与された仮クラスタ 3 1 7 の「`valid`」のフィールドを更新する際の本クラスタ生成部 3 0 7 の動作について説明する。なお、以下では `cnt` が付与された仮クラスタ 3 1 7 のことを、説明の便宜上、単に仮クラスタ 3 1 7 と記載する。

【 0 1 1 6 】

本クラスタ生成処理部 3 0 7 が `valid` 付与処理を開始すると（ステップ S 2 7 0 1）、まず、仮クラスタ 3 1 7 の全レコードの「`valid`」を 0 に初期化する（ステップ S 2 7 0 2）。

これ以降はループ処理となる。最初に、カウンタを構成する変数 `a` を 1 に初期化する（ステップ S 2 7 0 3）。

【 0 1 1 7 】

そして、仮クラスタ 3 1 7 の `a` 番目のレコードの「`valid`」が “ 0 ” であるか否かを確認する（ステップ S 2 7 0 4）。`a` 番目のレコードの「`valid`」が “ 0 ” である場合は（ステップ S 2 7 0 4 の Y）、仮クラスタ 3 1 7 の `a` 番目のレコードの「`term`」を一時変数 `tmpstr` に代入する（ステップ S 2 7 0 5）。

【 0 1 1 8 】

次に、`term=tmpstr` 及び `valid=0` で仮クラスタ 3 1 7 を絞り込む（ステップ S 2 7 0 6）。そして、絞り込んだ仮クラスタ 3 1 7 の各レコードの「`cid`」の小さい順に、当該各レコードを並び替える（ステップ S 2 7 0 7）。そして、この並び替えた各レコードの最上位レコードの「`valid`」に 1 を書き込み、その他のレコードの「`valid`」に - 1 を書き込む（ステップ S 2 7 0 8）。

【 0 1 1 9 】

以上の処理完了後、カウンタを構成する変数 `a` をインクリメントする（ステップ S 2 7 0 9）。一方、ステップ S 2 7 0 4 の判断で、仮クラスタ 3 1 7 の `a` 番目のレコードの「`valid`」が 0 でないと判定された場合は（ステップ S 2 7 0 4）、ステップ S 2 7 0 5 ~ ステップ S 2 7 0 8 の処理をすることなく、ステップ S 2 7 0 9 の処理へ移行する。

【 0 1 2 0 】

次に、インクリメントされた `a` 番目のレコードが仮クラスタ 3 1 7 の中に存在するかを確認する（ステップ S 2 7 1 0）。`a` 番目のレコードが存在する場合には（ステップ S 2 7 1 0 の N）、ステップ S 2 7 0 3 の処理へ戻り、処理を繰り返す。`a` 番目のレコードが存在しないときは（ステップ S 2 7 1 0 の Y）、処理を終了する。

【 0 1 2 1 】

図 2 8 は、図 2 7 で説明した `valid` 付与処理により `valid` 値が書き込まれた状態の仮クラスタ 3 1 7 の例を示す図である。`cnt` 値について図 2 8 を参照して説明する。仮クラスタ 3 1 7 の「`valid`」のフィールドには、`valid` 値として - 1 又は 1 いずれかの値が書き込まれている。

【 0 1 2 2 】

仮クラスタ 3 1 7 の `cid=1, tid=3` のレコードに注目する。このレコードの「西口 (`term`)」は、当該レコードより上位にあるレコードに含まれていない。このような場合、`valid` 値は “ 1 ” となる。

【 0 1 2 3 】

また、仮クラスタ 3 1 7 の `cid=2, tid=1` のレコードに注目する。このレコードの「西口 (`term`)」は、当該レコードより上位にある `cid=1, tid=3` のレコードの「西口 (`term`)」と一致する。このような場合、`valid` 値は “ - 1 ” となる。

【 0 1 2 4 】

ここで、`valid` 値とは、仮クラスタ 3 1 7 において、対象としているレコードの「`term`」が、当該レコードより上位にあるレコードの「`term`」と重複しているか否かを判断するための設定値である。つまり、重複がない場合は、“ 1 ” が記入され、重複が有る場合は “ - 1 ” が記入される。

10

20

30

40

50

【 0 1 2 5 】

再び、図 2 2 の本クラスタ生成処理の説明に戻る。

以上、図 2 2 のステップ S 2 2 1 0 の v a l i d 付与処理について図 2 7、図 2 8 に基づいて説明したが、この v a l i d 付与処理が完了すると、図 2 8 に示す仮クラスタ 3 1 7 の「alt」のフィールドに所定の値を書き込む a l t 付与処理に移行する（ステップ S 2 2 1 1）。

【 0 1 2 6 】

[a l t 付与処理フロー]

図 2 9 は、図 2 2 のステップ S 2 2 1 1 の a l t 付与処理の詳細な処理の流れを示すフローチャートである。

以下、図 2 8 に示す v a l i d が付与された仮クラスタ 3 1 7 の「alt」のフィールドを更新する際の本クラスタ生成処理部 3 0 7（図 3 を参照）の動作について説明する。なお、以下では v a l i d が付与された仮クラスタ 3 1 7 のことを、説明の便宜上、単に仮クラスタ 3 1 7 と呼ぶことにする。

【 0 1 2 7 】

まず、図 3 の本クラスタ生成処理部 3 0 7 が a l t 付与処理を開始する（ステップ S 2 9 0 1）。そして、仮クラスタ 3 1 7 を v a l i d = 1 で絞り込む（ステップ S 2 9 0 2）。

【 0 1 2 8 】

最初に、カウンタを構成する変数 b を 2 に初期化し（ステップ S 2 9 0 3）、図 2 9 に示したループ処理に移行する。そして、絞り込んだ仮クラスタ 3 1 7 の「cid」が b と一致するレコードを全て検索し、それらのレコードを一時ファイルに書き出す（ステップ S 2 9 0 4）。

【 0 1 2 9 】

ここで、一時ファイルに書き出されたレコードの数を確認する（ステップ S 2 9 0 5）。一時ファイルに書き出されたレコードが複数ある場合には（ステップ S 2 9 0 5 の N）、続いて、絞り込んだ仮クラスタ 3 1 7 の「cid」が b の全レコードの「len」の合計が 1 6 以下であるか否かを確認する（ステップ S 2 9 0 6）。そして、「len」の合計が 1 6 以下である場合は（ステップ S 2 9 0 6 の Y）、カウンタを構成する変数 c を 1 に初期化する（ステップ S 2 9 0 7）。また、ステップ S 2 9 0 5 で、一時ファイルに書き出されたレコードの数が一つだけであると判定された場合も（ステップ S 2 9 0 5 の Y）、ステップ S 2 9 0 7 の処理に移行する。

【 0 1 3 0 】

次に、一時ファイルの「cid」が b、レコード番号が c のレコードの「term_1」及び「term_2」で、絞り込んだ仮クラスタ 3 1 7 の当該レコードより上位にあるレコードの「term」を検索する（ステップ S 2 9 0 8）。そして、検索がヒットしたか否かを確認する（ステップ S 2 9 0 9）。なお、レコード番号とは、一時ファイルの各レコードに降順に割り当てられた番号のことである。

【 0 1 3 1 】

また、ステップ S 2 9 0 9 で、仮クラスタ 3 1 7 の当該レコードより上位にあるレコードの「term」検索がヒットした場合は（ステップ S 2 9 0 9 の Y）、ヒットしたレコードの「cid」を、絞り込んだ仮クラスタ 3 1 7 の所定のレコードの「alt」に書き込む（ステップ S 2 9 1 0）。この検索でヒットしたレコードの「cid」がない場合は（ステップ S 2 9 0 9 の N）、絞り込んだ仮クラスタ 3 1 7 の所定のレコードの「alt」に“- 1”を書き込む（ステップ S 2 9 1 1）。ここで、所定のレコードとは、一時ファイルの「cid」が b、レコード番号が c のレコードの「cid」及び「tid」と一致する、仮クラスタ 3 1 7 のレコードのことである。

【 0 1 3 2 】

以上の処理完了後、カウンタを構成するレコード番号の変数 c をインクリメントする（ステップ S 1 9 1 2）。そして、一時ファイルの「cid」が b で、レコード番号がインクリメントされた c のレコードが存在するか否かを確認する（ステップ S 2 9 1 3）。該当

10

20

30

40

50

するレコードが存在する場合には（ステップS 1 9 1 3のN）、ステップS 2 9 0 8の処理に戻り、以降の処理を繰り返す。該当するレコードが存在しない場合は（ステップS 1 9 1 3のY）、カウンタを構成する、一時ファイルの「cid」である変数bをインクリメントする（ステップS 2 9 1 4）。また、ステップS 2 9 0 6で、絞り込んだ仮クラスタ3 1 7の「cid」である変数bの全レコードの「len」の合計が“ 1 6 ”を超えるときも（ステップS 2 9 0 6のN）、ステップS 2 9 1 4の処理へ移行し、変数bをインクリメントする。

【 0 1 3 3 】

次に、絞り込んだ仮クラスタ3 1 7の「cid」の中にインクリメントされた変数bのレコードが存在するか否かを確認する（ステップS 1 9 1 5）。そして、インクリメントされた変数bのレコードが存在する場合には（ステップS 2 9 1 5のN）、ステップS 2 9 0 4の処理へ戻る。インクリメントされた変数bのレコードが存在しない場合は（ステップS 2 9 1 5のY）、処理を終了する（ステップS 2 9 1 6）。

10

【 0 1 3 4 】

なお、仮クラスタ3 1 7の「alt」のフィールドに“ 0 ”以外の値が設定される条件（以下、「alt値設定条件」という）を以下にまとめる。具体的には、以下に示す条件（1）かつ（2）または条件（1）かつ（3）の場合に、仮クラスタ3 1 7の「alt」のフィールドに“ 0 ”以外の値が設定される。

（1）仮クラスタ3 1 7の「valid」が“ 1 ”である。

（2）ステップS 2 9 0 4の処理で書き出された一時ファイルの各レコードの「len」の合計が“ 1 6 ”以下である。

20

（3）ステップS 2 9 0 4の処理で書き出された一時ファイルのレコードが一つである。

【 0 1 3 5 】

図3 0は、図2 9で説明したalt付与処理によりalt値が書き込まれた状態の仮クラスタ3 1 7の例を示す図である。この図3 0を参照して、alt値について説明する。仮クラスタ3 1 7の「alt」のフィールドには、cnt値として- 1、0又はクラスタ番号（cid）のいずれかの値が書き込まれている。

【 0 1 3 6 】

例えば、仮クラスタ3 1 7のcid=2,tid=3のレコードに注目する。このレコードは、前述のalt値設定条件を満たしていないレコードである。そのため当該レコードの「alt」には、0が書き込まれている。

30

【 0 1 3 7 】

また、仮クラスタ3 1 7のcid=9,tid=1のレコードに注目する。このレコードは、前述のalt値設定条件を満たしているレコードである。さらに、当該レコードの「ビジネスホテル（term_1）」及び「ヒルトン（商標登録）（term_2）」の両方とも、これらのレコードより上位にある全てのレコードの各「term」と一致しない場合は、alt値は“- 1”となる。ここで、当該レコードより上位にあるレコードとは、仮クラスタ3 1 7テーブルの中で、cidが1～8の各レコードのことである。

【 0 1 3 8 】

また、仮クラスタ3 1 7のcid=61,tid=1のレコードに注目する。このレコードのクラスタ番号は“ 6 1 ”でありcid=61且つvalid=1の全てのレコードの「len」の合計は1 0であるから、前述のalt値設定条件を満たしているレコードである。さらに、このレコードより上位にあり、同一のcid値を有するレコード（cid=17,tid=1,2）の各「term」が、当該レコードの「映画館（term_1）」及び「映画（term_2）」の両方と一致する。このような場合、alt値は、一致した上位のレコードの「cid」である“ 1 7 ”となる。

40

【 0 1 3 9 】

以上、図2 2のステップS 2 2 1 1のalt付与処理について図2 9、図3 0に基づいて説明した。再び、図2 2の本クラスタ生成処理の説明に戻る。

ステップS 2 2 1 1のalt付与処理が完了すると、図3 0に示す仮クラスタ3 1 7の

50

全レコードを検索して、ストップワードのフラグを立てる（ステップS 2 2 1 2）。具体的には、仮クラスタ3 1 7の各レコードの「term」が予め設計者の設定したストップワードに一致する場合、当該レコードの「stop」に1を書き込む。なお、ストップワードのフラグが立てられた仮クラスタ3 1 7の例を図3 1に示す。

【0 1 4 0】

そして、図3 1に示す仮クラスタ3 1 7の不要なレコードを削除する（ステップS 2 2 1 3）。このステップS 2 2 1 3で削除される不要なレコードの条件は以下の通りである。

- (1) 「valid」のフィールドが-1のレコード
- (2) 「stop」のフィールドが1のレコード
- (3) 「cnt」のフィールドが0のレコード

10

【0 1 4 1】

ここで、不要なレコードを削除した仮クラスタ3 1 7の「alt」が“1”以上の値が設定されているレコードを検索し、そのレコードの「cid」と「alt」を入れ替える。そして、不要なレコードを削除した仮クラスタ3 1 7の中で、各レコードの「cid」の小さい順に、当該各レコードを並び替え（ステップS 2 2 1 4）、処理を終了する（ステップS 2 2 1 5）。

【0 1 4 2】

ステップS 2 2 1 4の処理によって並び替えられた仮クラスタ3 1 7は、図3で説明した本クラスタ8 0 2に該当する。本クラスタ8 0 2の例を図3 2及び図3 3に示す。

20

以上で、図2 2に基づいたモバイル情報検索サーバ1 0 3のバッチ処理部2 0 2の説明を終了する。

【0 1 4 3】

[リアルタイム処理部]

次に、図2のリアルタイム処理部2 0 3について説明する。

図3 4は、リアルタイム処理部2 0 3及びデータ記憶部の一部の機能的な構成を示すブロック図である。

リアルタイム処理部2 0 3は、概略的にはHTTPを用いて、図1に示す携帯電話1 0 2にhtml文書を送信する、webサーバの機能を有している。この機能は、携帯電話1 0 2のwebブラウザプログラム1 1 3からの要求に応じて行われるものである。

30

【0 1 4 4】

リアルタイム処理部2 0 3は、一般的なネットワークOSが稼働するコンピュータである。ネットワークOSは、一例として、FreeBSD等のBSD系、或いはLinux（登録商標）等のPOSIX（Portable Operating System Interface for UNIX）（UNIXは登録商標）系OS等が挙げられる。

【0 1 4 5】

以下、図3 4に示す機能ブロック図に基づいて、リアルタイム処理部2 0 3の構成について説明する。

リアルタイム処理部2 0 3は、上述したように、webサーバ機能を持つwebサーバプログラム3 4 0 2を備えている。このwebサーバプログラム3 4 0 2は、携帯電話1 0 2から来る要求（コマンド）を受信し、これに応じて、HTTPにてhtml文書をクライアントへ送信する機能を提供する。端的に言えば、webサーバソフトウェアであり、一例としてはApache（http://www.apache.org/）等が、これに該当する。

40

【0 1 4 6】

また、webサーバプログラム3 4 0 2は、携帯電話1 0 2の要求に応じ、cgi（Common Gateway Interface）の実行も行う。cgiの実体は、標準出力にテキストを出力するプログラムである。このcgiが生成するテキストが、webサーバプログラムによってHTTPにて携帯電話1 0 2へ送信される。図3 4では、第1のcgi 3 4 0 3の他、第2のcgi 3 4 0 7～第4のcgi 3 4 0 9まで用意されている。

【0 1 4 7】

50

これらの c g i を構成する要素としては、どのようなプログラミング言語であってもよい。perl (http://www.perl.org/) や PHP (http://php.net/)、python (http://www.python.org/ja/)、ruby (http://www.ruby-lang.org/ja/) 等のインタプリタ言語のみならず、C 言語や Java (登録商標) 等のコンパイルを要する言語であってもよい。簡単な内容であれば、シェルスクリプトであってもよい。

【0148】

各々の c g i は、主に web ブラウザに表示する画面を作るための html 文書を作成する。つまり、html 文書は web ブラウザを通じて GUI を作成する手段である。

なお、図 34 では、上述のように、携帯電話の表示部に表示させる画面ごとに 4 つの c g i が個別に設けられているが、単一の c g i で構成することも可能である。ただし、その場合は、単一の c g i の中に複数の表示画面を作成する機能が含まれることとなる。

【0149】

第一の c g i 3403 は、携帯電話 102 から来るアクセス要求に応じて、web サーバプログラム 3402 によって実行される。

第一の c g i 3403 は、携帯電話 102 から得られたパラメータであるユーザの現在位置情報で不揮発性データ記憶部 204 に記憶されている GPS 地名マスタ 308 を検索する。そして、GPS 地名マスタ 308 から現在位置情報に応じた地名を取得する。また、第一の c g i 3403 は、取得した地名をキーとして、本クラスタディレクトリ 318 を検索し、入力した地名と対応する本クラスタ 802 を取得する。

【0150】

そして、この取得した本クラスタ 802 を表示用本クラスタ 3404 に変換し、不揮発性データ記憶部に表示用一時ファイル 3404 として記憶する。なお、表示用本クラスタ 3404 とは、本クラスタ 802 から所定のレコードを削除したものである。この表示用本クラスタ 3404 の詳細については図 38 及び図 39 にて後述する。

【0151】

また、本クラスタ 802 から表示用本クラスタ 3404 に変換する際には、第一の c g i 3403 は、携帯電話 102 の表示部の大きさ、フォントの大きさ及びフォントの種類を考慮した表示用本クラスタ 3404 を生成するようにする。この表示用本クラスタ 3404 の生成は、予め不揮発性データ記憶部 204 に記憶された携帯端末マスタ 3405 及びリアルタイム処理部 203 に存在しているプロポーショナルフォントファイル 3406 を参照しながら第一の c g i 3403 によって行われる。

【0152】

携帯端末マスタ 3405 には、様々な携帯電話の画面 (表示部) の大きさを示す情報が記憶されている。また、プロポーショナルフォントファイル 3406 には、携帯電話 102 がプロポーショナルフォントでの画面表示を要求してきた際に、そのフォント幅に対応した表示用本クラスタ 3404 を生成するための情報が記憶されている。

【0153】

第一の c g i 3403 は、この表示用本クラスタ 3404 (表示用一時ファイル) に基づいて、携帯電話 102 の表示部に、分類の選択画面を表示する html 文書を作成する。なお、分類の選択画面とは、表示用本クラスタ 3404 を、その「cid」毎に分類して「term」が表示された画面のことである。ただし、分類の選択画面の詳細については、図 36 にて後述する。第一の c g i は、本クラスタ送信部ともいえる。

【0154】

第二の c g i 3407 は、携帯電話 102 から来るアクセス要求に応じて、web サーバプログラム 3402 によって実行される c g i である。

第二の c g i 3407 は、分類の選択画面を表示している携帯電話から、得られたパラメータである地名及び関連語で検索キー・ファイル名対応テーブル 312 (図 13 (b)) を参照) を検索する。そして、検索キー・ファイル名対応テーブル 312 から、地名及び関連語に対応する検索結果 html ファイル名を取得する。

【0155】

第二の c g i 3 4 0 7 は、この検索結果 h t m l ファイル名で検索結果 h t m l ファイルディレクトリ 3 1 0 を検索し、検索結果 h t m l ファイルを取得する。そして、第二の c g i 3 4 0 7 は、取得した検索結果 h t m l ファイルを携帯電話用の h t m l ファイルに変換し、携帯電話 1 0 2 の表示部に検索結果画面を表示するための h t m l 文書を作成する。

なお、検索結果画面とは、地名及び関連語を検索クエリとして、検索エンジン 1 0 4 (図 1 参照) で検索を行った際の画面のことであり、その詳細については図 3 7 (a) にて後述する。ただし、ここでの地名及び関連語は、分類の選択画面を表示している携帯電話 1 0 2 から得られるパラメータのことである。

【 0 1 5 6 】

第三の c g i 3 4 0 8 は、携帯電話 1 0 2 から来るアクセス要求に応じて、w e b サーバプログラム 3 4 0 2 によって実行される。このプログラムによって実行される第三の c g i 3 4 0 8 による処理が後述するリアルタイム処理 (3) を行うブロック (図 3 5 のステップ S 3 5 0 8 、図 4 2) である。

第三の c g i 3 4 0 8 は、検索結果画面を表示している携帯電話 1 0 2 から、得られたパラメータであるキャッシュファイル名で検索キー・ファイル名対応テーブル 3 1 2 を検索する。そして、検索キー・ファイル名対応テーブル 3 1 2 から、キャッシュファイル名を取得する。第三の c g i 3 4 0 8 は、このファイル名でキャッシュファイルディレクトリ 3 1 1 を検索して、キャッシュファイルを取得する。

【 0 1 5 7 】

キャッシュファイルのプレーンテキストに住所文字列が含まれていない場合、第三の c g i 3 4 0 8 は、取得したキャッシュファイルを携帯電話 h t m l に変換する。第三の c g i 3 4 0 8 によって変換された携帯電話 h t m l は、携帯電話 1 0 2 の表示部 1 1 1 にキャッシュファイルのプレーンテキストを表示する h t m l 文書である。

【 0 1 5 8 】

一方、キャッシュファイルのプレーンテキストに住所文字列が含まれている場合、第三の c g i 3 4 0 8 は、検索キー・ファイル名対応テーブル 3 1 2 の緯度・経度のフィールドから位置情報を取得して、地図サーバ 1 0 5 への A タグを生成する。そして、第三の c g i 3 4 0 8 は、A タグ及びキャッシュファイルを基にして、携帯電話 1 0 2 の表示部 1 1 1 に地図付きキャッシュ画面を表示する h t m l 文書を作成する。なお、地図付きキャッシュ画面とは、キャッシュファイルのプレーンテキストと共にプレーンテキストに記載された位置情報を中心とする地図を含む画面のことである。ただし、地図付きキャッシュ画面の詳細については、図 3 7 にて後述する。

【 0 1 5 9 】

第四の c g i 3 4 0 9 は、携帯電話 1 0 2 から来るアクセス要求に応じて、w e b サーバプログラム 3 4 0 2 によって実行される。このプログラムによる第四の c g i 3 4 0 9 の処理が後述する連想語検索処理を行うブロック (図 4 3 のステップ S 4 3 0 3 , 図 4 5) である。

第四の c g i 3 4 0 9 は、分類の選択画面 3 6 0 2 を表示している携帯電話 1 0 2 から得られるパラメータの中で、クラスタ番号で第一の c g i にて生成された表示用本クラスタ 3 4 0 4 クラスタ番号 (c i d) に属する関連語 (t e r m) を全て取得する。そして、この取得した各関連語で W A M ファイルディレクトリ 3 1 3 を検索する。

この検索により、W A M ファイルディレクトリ 3 1 3 から各関連語の W A M ファイル全てを取得し、これら W A M ファイルから連想語テーブル 3 4 1 0 を生成する。連想語テーブル 3 4 1 0 とは、携帯電話 1 0 2 から得られた各関連語の連想語が書かれたテーブルのことである。なお、連想語とは、各関連語から連想される単語のことである。この連想語テーブルについては、図 4 6 にて後述する。

【 0 1 6 0 】

第四の c g i 3 4 0 9 は、連想語テーブル 3 4 1 0 に基づいて、携帯電話 1 0 2 の表示部 1 1 1 に連想語画面を表示する h t m l 文書を作成する。なお、連想語画面とは、連想

10

20

30

40

50

語テーブル3410に書かれている各連想語を羅列した画面のことである。ただし、連想語画面の詳細については、図44にて後述する。

【0161】

[リアルタイム処理部の動作]

次に、リアルタイム処理の流れを説明する。

図35は、携帯電話102(クライアント)、モバイル情報検索サーバ103及び地図サーバ105の間で行われる動作を示すシーケンス図である。

まず、ユーザは、携帯電話102の所定のアプリケーションを起動する。すると、携帯電話102は、GPS受信部108において、GPS衛星107より携帯電話102の現在位置情報(以下、「GPS位置データ」という)を取得し、入出力制御部109を通じてモバイル情報検索サーバ103に送信する(ステップS3501)。ここで、GPS位置データとは、携帯電話102の現在位置の緯度及び経度のことである。以下では、携帯電話102の現在位置が新宿の場合を例として、リアルタイム処理の流れを説明する。また、ステップS3501の処理の際に、携帯電話102は、入出力制御部109を通じて、携帯電話102のユーザエージェント文字列もモバイル情報検索サーバ103に送信するようにしている。

10

【0162】

モバイル情報検索サーバ103は、GPS位置データすなわち新宿の緯度及び経度を受信する。そして、現在位置情報及びユーザエージェント文字列に基づいて、図38にて後述するリアルタイム処理(1)を実行し(ステップS3502)、携帯電話102の表示部111に図36に示す分類の選択画面3602を表示するhtml文書を作成して、クライアント側へ送信する(ステップS3503)。

20

【0163】

次に、携帯電話102は、後述するリアルタイム処理(1)で作成されたhtml文書を入出力制御部109により取得・解釈する。そして、このhtml文書は表示制御部110を通じて表示部111に入力される。これにより、携帯電話102の表示部111に分類の選択画面3602が表示される。図36に示されるように、分類の選択画面3602には、現在位置の地名「新宿」に対する関連語がクラス番号毎に、すなわちクラス毎に分類されて表示されている。ユーザは、携帯電話102の入力部112を操作して、分類の選択画面3602から、例えば「レストラン」という関連語を選択する。すると、携帯電話102は、「(現在位置の地名)+(スペース)+(関連語)」である「(新宿)+(スペース)+(レストラン)」の文字列をモバイル情報検索サーバ103側に送信する(ステップS3504)。

30

【0164】

モバイル情報検索サーバ103は、「(新宿)+(スペース)+(レストラン)」という文字列を受信する。そして、この文字列に基づいて、図41にて後述するリアルタイム処理(2)を実行し(ステップS3505)、携帯電話102の表示部111に図37(a)に示す検索結果画面3702を表示するhtml文書を作成して、クライアント側へ送信する(ステップS3506)。

【0165】

携帯電話102は、後述するリアルタイム処理(2)で作成されたhtml文書を入出力制御部109により取得・解釈する。そして、このhtml文書は表示制御部110を通じて表示部111に入力され、携帯電話102の表示部111に検索結果画面3702が表示される。このとき、検索結果画面3702には、「(新宿)+(スペース)+(レストラン)」を検索クエリにしてウェブページを検索した検索結果が表示されている。ユーザは、携帯電話102の入力部112を操作して、検索結果画面3702の「(黒四角)」ボタン3703或いは「(白四角)」ボタン3704を選択すると、所定のキャッシュファイル名がモバイル情報検索サーバ103側へ送信される(ステップS3507)。なお、「(黒四角)」ボタン3703は、モバイル情報検索サーバ103側へ送信されるキャッシュファイル名のキャッシュファイル内に住所が含まれているリンクボタン

40

50

であり、「 (白四角)」ボタン3704はキャッシュファイル内に住所が含まれていないリンクボタンである。

【0166】

モバイル情報検索サーバ103は、例えば「 (黒四角)」ボタン3703に対応するキャッシュファイル名を受信したとする。すると、このキャッシュファイル名に基づいて、図42にて後述するリアルタイム処理(3)を行う。そして、携帯電話102の表示部111に地図キャッシュ画面を表示するhtml文書を作成し(ステップS3508)、クライアント側へ送信する(ステップS3509)。

【0167】

携帯電話102は、リアルタイム処理(3)で作成されたhtml文書を入出力制御部109により取得・解釈する。そして、地図サーバ105に地図画像要求を送信する(ステップS3510)。すなわち、このリアルタイム処理(3)で生成されたhtml文書には、地図画像3706の所在を示す地図サーバ105のURLが記載されており、携帯電話102の入出力制御部109は、携帯電話102に当該URLの地図画像3706を送信するように地図サーバ105に要求する。この要求を受けて、地図サーバ105は、地図画像3706を携帯電話102(クライアント)側へ送信する(ステップS3511)。

10

【0168】

携帯電話102は、地図サーバ105から地図画像3706を受信して、表示部111に、図37(b)に示されるような地図付きキャッシュ画面3705を表示する。なお、地図付きキャッシュ画面3705の「 (星印)」記号3707の位置は、キャッシュファイル内に含まれる住所に一致している。具体的には、地図付きキャッシュ画面3705に表示された、東京都新宿区歌舞伎町 - - という住所に当たる。

20

【0169】

[リアルタイム処理(1)フロー]

図38は、図35のS3502のリアルタイム処理(1)の詳細な処理の流れを示すフローチャートである。

【0170】

図35のステップS3502で説明したように、第一のcgiが処理を開始すると(ステップS3801)、第一のcgi3403は携帯電話102から受信したGPS位置データで、GPS地名マスタ308(図3を参照)を検索する(ステップS3802)。そして、GPS地名マスタ308から検索がヒットした地名を取得する。

30

【0171】

次に、地名を検索クエリとして本クラスタディレクトリ318を検索し(ステップS3804)、地名に応じた本クラスタ802(図8、図32及び図33を参照)を取得する。そして、第一のcgi3403は、GPS位置データと共に携帯電話102が取得したユーザエージェント文字列を基にして、携帯端末マスタ3405を検索する。そして、携帯端末マスタ3405の検索によって、携帯電話102の機種名を取得する。

なお、携帯端末マスタ3405の中には、携帯電話102の機種名及びその画面に関する諸情報が記録されている。例えば、等幅フォントであれば、画面の幅はバイト数で決定できる。また、プロポーショナルフォントを使っている場合は、画面に収められる文字数とその文字列によって変化する。そこで、携帯端末マスタ3405で画面に関する諸情報を取得して、本クラスタ802をその携帯端末に最適な表示用本クラスタ3404に変換するのである(ステップS3805)。

40

【0172】

次に、このようにして変換した表示用本クラスタ3404を携帯電話html文書に変換する(ステップS3806)。最後に、第一のcgi3403は、変換したhtml文書をクライアントに送信して(ステップS3807)、一連の処理を終了する(ステップS3808)。なお、等幅フォントを使用する場合は、画面のバイト数で本クラスタ802から表示用本クラスタ3404に変換する文字数が一意的に“1”に決められる。ただ

50

し、プロポーショナルフォントを使う場合は、本クラスタ 8 0 2 で表示しようとする関連語の文字列を、プロポポーショナルフォントファイル 3 4 0 6 を使って参照し、それら文字列を表示するのに必要な画面のピクセル数を得る。画面のピクセル数と携帯電話 1 0 2 の画面の横幅とを比較して最適な文字数を決定する。

【 0 1 7 3 】

図 3 9 は、表示用の本クラスタの一例を示す。

この例は、図 3 2 及び図 3 3 に示す本クラスタ 8 0 2 を携帯端末毎に最適な表示用形態に変換したものである。ここで、図 3 2 及び図 3 3 に示す本クラスタ 8 0 2 の各レコードと、図 3 9 に示す表示用本クラスタ 3 4 0 4 の各レコードとを比較する。すると、表示用本クラスタ 3 4 0 4 は、本クラスタ 8 0 2 から所定のレコードを削除したものであることが確認できる。つまり、携帯電話上で最適化できる表示に変換した結果、表示用本クラスタ 3 4 0 4 は、本クラスタ 8 0 2 の cid=3,8,12,17,33,48,62 のレコードのみを残して削除したもとなっている。なお、本クラスタ 8 0 2 から削除するレコードの条件は、同一の「cid」のレコードの各「term」の文字列数すなわち「len」合計が 2 0 以上 3 0 以下のレコードである。ただし、「term」が複数ある場合は、各「term」の間にはスペースが入るので、「term」の数だけスペースの文字列数が追加される。

10

【 0 1 7 4 】

また、本クラスタ 8 0 2 から表示用本クラスタ 3 4 0 4 に変換する際、cid=17 の各レコードに注目すると、図 3 2 に示される cid=17, tid=4 のレコードのみ削除されている。このレコードを含めてしまうと合計文字列数が 3 1 となり、レコードを削除する条件を満たしてしまふためである。図 3 2 の cid=3, tid=3 のレコードも同様の理由で削除される。

20

【 0 1 7 5 】

図 4 0 は、表示用本クラスタ 3 4 0 4 の h t m l 文書の一例を示した図である。

表示用本クラスタ h t m l 文書 4 0 0 2 は、携帯電話 1 0 2 の表示部 1 1 1 に図 3 6 に示す分類の表示画面を表示させる h t m l 文書である。

以下、表示用本クラスタ h t m l 文書 4 0 0 2 に記載されている単語の並びについて説明する。

【 0 1 7 6 】

図 3 9 に示す表示用本クラスタ 3 4 0 4 の各パラメータの中で、表示用本クラスタ h t m l 文書 4 0 0 2 を生成するのに必要なパラメータは、「cid」、「tid」及び「term」である。

30

ここで、表示用本クラスタ h t m l 文書 4 0 0 2 の一行目に注目する。

この行には、表示用本クラスタ 3 4 0 4 の「cid」が最も小さいレコードの「term」である「カプセルホテル」及び「新宿区役所」が、「tid」の小さい順に左から並んでいる。また、一番左の「カプセルホテル」のさらに左脇には、表示用本クラスタ 3 4 0 4 の中で最も「cid」が小さい、つまり「新宿」という地名に最も関連しているクラスタであることを示す「(0 1)」が付与されている。なお、表示用本クラスタ h t m l 文書 4 0 0 2 の一行目以降も一行目と同様の特徴を持っている。

【 0 1 7 7 】

以上のような特徴を持った表示用本クラスタ h t m l 文書 4 0 0 2 を携帯電話 1 0 2 に送信することにより、携帯電話 1 0 2 は、その表示部 1 1 1 に図 3 6 に示す分類の選択画面 3 6 0 2 を表示する。

40

【 0 1 7 8 】

[リアルタイム処理(2)フロー]

図 4 1 は、図 3 5 の S 3 5 0 5 のリアルタイム処理(2)の詳細な処理の流れを示すフローチャートである。

【 0 1 7 9 】

図 3 5 のステップ S 3 5 0 5 で説明したように、携帯電話 1 0 2 から地名 + 関連語というパラメータが送られると(ステップ S 4 1 0 1)、第二の c g i 3 4 0 7 は、この携帯電話 1 0 2 から受信した地名 + 関連語を検索クエリとして、検索キー・ファイル名対応テ

50

ーブル312を検索する。そして、その結果、検索結果htmlファイル名を取得する(ステップS4102)。

【0180】

この検索結果htmlファイル名の検索結果htmlファイルは、検索エンジン104から取得した検索結果のhtmlファイルである。このhtmlファイルに書かれているAタグを選択すると検索エンジン104にアクセスしてしまうので、そこはモバイル情報検索サーバ103宛のアクセスを示すAタグに書き換えなければならない。

そこで、第二のcgi3407は、この検索結果htmlファイルに記されている検索エンジン104宛のAタグを、モバイル情報検索サーバ103宛のAタグに変換する処理を行う(ステップS4103)。そして、変換されたhtmlファイルを携帯電話(クライアント)102に送信して(ステップS4104)、処理を終了する(ステップS4105)。

10

【0181】

[リアルタイム処理(3)フロー]

図42は、図35のS3508のリアルタイム処理(3)の詳細な処理の流れを示すフローチャートである。

【0182】

図35のステップS3508で説明したように、ユーザが携帯電話102で検索結果htmlファイルのAタグの一つを選択すると、そこに埋め込まれているキャッシュファイル名がモバイル情報検索サーバ103に送信される(ステップS4201)。

20

【0183】

第三のcgi3408は、キャッシュファイル名を検索キーとして、検索キー・ファイル名対応テーブル312を検索する。そして、検索にヒットしたレコードの住所フィールドをみる(ステップS4202)。そこで、この住所フィールドが何らかの文字列が含まれるかどうか、すなわち住所が含まれているかどうか確認する(ステップS4203)。もし住所フィールドに住所が含まれているのであれば、そのレコードの緯度・経度フィールドを読んで、地図サーバ105へのイメージタグを生成する(ステップS4204)。そして、キャッシュファイルを地図サーバ105へのイメージタグを含む携帯用htmlファイルに変換する(ステップS4205)。

そして、このようにして生成した携帯用htmlファイルをクライアントである携帯電話102へ送信して(ステップS4206)、処理を終了する(ステップS4207)。

30

【0184】

もし、ステップS4203の処理において、住所フィールドが空である場合、すなわち住所を含んでいなかったならば、住所がないので先に説明したステップS4204のような変換は一切行わず、単にキャッシュファイルを携帯用htmlファイルに変換する(ステップS4208)。それをクライアントに送信し(ステップS4206)、処理を終了する(ステップS4207)。

【0185】

[連想語検索処理フロー]

図43は、携帯電話102とモバイル情報検索サーバ103との間で行われる連想語検索処理の流れを示すシーケンス図である。

40

ここでは、図35におけるリアルタイム処理(1)から、ユーザがクラスタを選択した場合の動作の流れを示している。

【0186】

図36において関連語をクリックした動作は、図35で説明した動作であったが、この図43に示される動作は、図36の関連語の前にある「w」の記号をクリックした時の動作である。

【0187】

リアルタイム処理(1)までの処理は、図35で説明した処理と同じである。

携帯電話102を保有するユーザは、表示用本クラスタ3404html、すなわち分

50

類の選択画面3602を表示した中で、「w」の記号3603を選択する。これは、クラスタの選択に相当する。すると、「w」の記号3603の中に埋め込まれているクラスタ番号が、第四のcgi3409に向けて送信される。そして、第四のcgi3409は、連想語検索を行って連想語表示画面を携帯電話102の表示部111に表示するhtml文書（以下、「連想語検索結果html文書」という）を携帯電話102（クライアント）に送信する。以降、この連想語結果検索html文書に従って、ユーザは連想語のさらに絞り込みを行ったり、連想語の連想語を検索したりする等の操作を継続することとなる。

【0188】

図44は、連想語検索処理の結果として、携帯電話102の表示部111に表示される連想検索結果を示す一例である。

10

図44(a)は、新宿のレストラン及びイタリアンという単語について連想語検索処理を行った結果を示す図である。

ここで、図44(a)に示す関連語の連想語検索結果画面4402上のフレンチの横にあるw記号4403をクリックすると、図44(b)に示す連想語の連想語検索結果画面4404が表示されることとなる。ここでは、新宿のフレンチ、レストラン及びイタリアンの連想語検索結果画面4404が表示されている。すなわち、レストラン及びイタリアンという単語に、さらにフレンチという連想語が追加されて検索されたものとなる。このようにして、連想語の連想語、更に、連想語の連想語の連想語、...、という風に、芋づる式に、連想語検索処理を行うことができる。

20

【0189】

ここから、このような連想語検索処理の動作について、図45に基づいて詳細に説明する。

図45は、図43のS4303の連想語検索処理の詳細な処理の流れを示すフローチャートである。

【0190】

図43のステップS4303で説明したように、携帯電話102からクラスタ番号を受信すると（ステップS4501）、第四のcgi3409は表示用本クラスタ3404から得られた同一クラスタに属する関連語のWAMファイルを全て取得する（ステップS4502）。

30

【0191】

そして、第四のcgi3409は、取得した処理対象のWAMファイルをソートする。そうすると、キャッシュファイル名は上の方に、単語（関連語及び周辺語）はその単語の出現頻度と共に、キャッシュファイル名の下に並ぶこととなる。そこで、単語の部分のみを抜き出して、出現する単語の出現頻度を加算する。そして、単語を連想語テーブル3410の「aterm」のフィールドに、当該単語の出現頻度を加算した結果を連想語テーブル3410の「wcnt」のフィールドに順次書き込む（ステップS4503）。

【0192】

次に、処理対象となるWAMファイルから、関連語テーブルの「aterm」フィールドの各単語が出現するキャッシュファイルの数を、連想語テーブル3410の「pcnt」フィールドに順次書き込む（ステップS4504）。

40

【0193】

続いて、第四のcgi3409は、連想語テーブル3410の各レコード毎に、「wcnt」及び「pcnt」からTFIDF値を計算し、その計算結果を当該各レコードの「tfidf」のフィールドに順次書き込む（ステップS4505）。TFIDF値とは、文章中の特徴的な単語（重要とみなされる単語）を抽出するためのアルゴリズムであり、主に情報検索や文章要約などの分野で利用される。なお、TFIDF値の計算式を以下に示す。計算式において、“10000”は検索エンジン104からモバイル検索サーバ103が取得したキャッシュファイルの総数に対応している。また、対数の項に加算される“1”は、pcntの値が小さいときに、TFIDF値が“0”となることを回避するための調整値で

50

ある。

【 0 1 9 4 】

〔 数 2 〕

$tfidf = wcnt * (1 + \log(10000 / pcnt))$

【 0 1 9 5 】

第四の c g i 3 4 0 9 は、さらに T F I D F 値の大きい順に連想語テーブル 3 4 1 0 の並び替えを行う。そして、連想語テーブル 3 4 1 0 の「aterm」が連想語またはストップワードと一致するレコードを削除する（ステップ 4 5 0 7）。その結果として、第四の c g i 3 4 0 9 は、連想語テーブル 3 4 1 0 から h t m l 文書を作成し、それを送信して（ステップ S 4 5 0 8）、処理を終了する（ステップ S 4 5 0 9）。 10

【 0 1 9 6 】

図 4 6 は、連想語テーブル 3 4 1 0 の中身の一例を示す図である。

図 4 6 (a) は、図 4 5 のステップ S 4 5 0 5 の処理の時点の連想語テーブル 3 4 1 0 の中身である。

図 4 6 (b) は、図 4 5 のステップ 4 5 0 7 の処理を経た連想語テーブル 3 4 1 0 の中身である。

図 4 6 (a) の連想語テーブル 3 4 1 0 に対して、T F I D F 値の大きい順に連想語テーブル 3 4 1 0 の各レコードをソートした後に、「aterm」がストップワード（東京）及び関連語（レストラン、イタリアン）と一致するレコードを削除する。すると、連想語のみが抽出された図 4 6 (b) の連想語テーブル 3 4 1 0 が生成される。なお、以上の処理により生成された連想語テーブル 3 4 1 0 を h t m l 文書に変換して、携帯電話 1 0 2 の表示部 1 1 1 に表示させたものが、図 4 4 (b) に示す連想語検索結果画面である。 20

【 0 1 9 7 】

本実施形態には、以下のような応用例が考えられる。

(1) 例えば、検索エンジン 1 0 4 には、検索対象の言葉のカテゴリの深さを開示する機能がある。そこで、このカテゴリの深さというものを検索結果に反映させることで、より最適な検索結果の表示を実現することができる。

【 0 1 9 8 】

図 4 7 は、深さ付与処理の流れを示すフローチャートである。

深さ付与処理とは、検索対象の関連語（term）の深さを表示用本クラスタ 3 4 0 4 に与える処理である。 30

図 3 8 には図示していないが、この深さ付与処理は、図 3 8 のステップ S 3 8 0 5 の処理とステップ S 3 8 0 6 の処理の間に挿入される処理である。この処理は、オプションで必要に応じて行われる。

【 0 1 9 9 】

ここで、処理を開始すると（ステップ S 4 7 0 1）、表示用本クラスタ 3 4 0 4 の先頭レコードに注目する（ステップ S 4 7 0 2）。そして、変数 d に現在注目しているレコードの「cid」を代入する（ステップ S 4 7 0 3）。

【 0 2 0 0 】

次に、表示用本クラスタ 3 4 0 4 の現在のレコードの「term」を、図示しない深さテーブルで検索して、「term」の深さを得る（ステップ S 4 7 0 4）。なお、深さテーブルとは、表示用本クラスタ 3 4 0 4 の「term」とその深さが対応付けられて書き込まれているテーブルのことである。ただし、この深さの情報は、予め検索エンジン 1 0 4 から取得して、深さテーブルに書き込んでおく。 40

【 0 2 0 1 】

以上のようにして得た「深さ」の情報を、表示用本クラスタ 3 4 0 4 の現在のレコードの「new_tid」に書き込む（ステップ S 4 7 0 5）。

以上の処理が完了すると、表示用本クラスタ 3 4 0 4 のレコードをインクリメントし、現在のレコードの次にあるレコードに注目する（ステップ S 4 7 0 6）。

【 0 2 0 2 】

次に、表示用本クラスタ3404にまだレコードが存在するか否かを確認する(ステップS4707)。レコードがまだ存在する場合には(ステップS4707のY)、現在注目しているレコードの「cid」が変数dと等しいか否かを確認する(ステップS4708)。注目レコードの「cid」が変数dと等しい場合は(ステップS4708のN)、ステップS4704の処理に戻り、再びステップS4704以降の処理を繰り返す。注目レコードの「cid」が変数dと等しくない場合には(ステップS4708のY)、表示用本クラスタ3404のcid=dの各レコードの「new_tid」の最小値を、当該各レコードの「new_cid」に書き込む(ステップS4709)。

【0203】

ステップS4707の処理にて、表示用本クラスタ3404のレコードが存在しない場合には(ステップS4707のN)、「new_cid」、「cid」、「new_tid」の順にソートのキーの優先順位を設定して、表示用本クラスタ3404の各レコードを並び替え(ステップS4710)、処理を終了する(ステップS4711)。

10

【0204】

図48は、図47で説明した深さ付与処理において用いる表示用本クラスタ4802である。

図39の表示用本クラスタ3404と違う点は、「new_cid」及び「new_tid」というフィールドが新たに付与されている点である。これらのフィールドが深さ付与処理において有効に機能することとなる。

図49は、深さで分類した表示用本クラスタの中身を示す図である。すなわち、図49の表示用本クラスタ4902は、図48の表示用本クラスタ4902をステップS4710で処理することによって、並び替えたものである。具体的には、「new_cid」が第一優先、「cid」が第二優先「new_tid」が第三優先となった並び替え結果となる。

20

【0205】

図50は、携帯電話102の表示画面に関連する図である。

図50(a)は、深さ付与処理がなされた表示用本クラスタhtml文書の一例である。

深さで分類した表示用本クラスタhtml文書5002は、携帯電話102の表示部111に図50(b)に示す分類の選択画面5003を表示させるhtml文書である。なお、分類の選択画面5003は、図36に示す分類の選択画面3602と同じ特徴を持っているので、説明は省略する。

30

【0206】

以下、本クラスタhtml文書に記載されている単語の並びについて説明する。

図49に示す本クラスタの各パラメータの中で、深さで分類した表示用本クラスタhtml文書5002を生成するのに必要なパラメータは、「cid」、「new_cid」及び「new_tid」である。

ここで、深さで分類した表示用本クラスタhtml文書5002の一行目に注目する。

この行には、図49に示す表示用本クラスタ4802の「cid」と「new_cid」の合計が最も小さいレコードの「term」である「レストラン」及び「イタリアン」が、「new_tid」の小さい順に左から並んでいる。また、図50(a)に示すように、一番左の「レストラン」のさらに左脇には、表示用本クラスタ4802の中で最も「cid」と「new_cid」の合計が小さい、つまり「新宿」という地名に最も関連しているクラスタであることを示す「(01)」が付与されている。なお、深さで分類した表示用本クラスタhtml文書5002の一行目以降も一行目と同様の特徴を持っている。

40

【0207】

以上のような、深さ付与処理を行うことによって、表示内容が表示される関連語(term)がより最適化されて分類される。

【0208】

(2) 上述の実施形態においては、携帯電話102の現在位置の地名を検索クエリとして、自動的にモバイル情報検索サーバ103に送信した。その代替えとして、検索エンジン

50

104のリアルタイム処理部202に、図51に示すアプリケーション起動時表示画面5102を携帯電話102に表示するhtml文書を作成するcgiを設けておけば、入力部112からユーザの所望する文字列、例えば地名以外の固有名詞等を入力することにより、その文字列を検索クエリとしてモバイル情報検索サーバ103に送信できるようにしてもよい。なお、携帯電話102から地名以外の固有名詞を検索クエリとしてリアルタイムサーバに送信する場合は、GPS地名マスタの代わりに、固有名詞等が書かれたマスタを不揮発性記憶装置に用意しておく必要がある。

【0209】

(3) 上述の実施形態においては、携帯電話102の現在位置の地名を検索クエリとして、自動的にモバイル情報検索サーバ103に送信したが、その代替えとして、携帯電話102の現在位置付近の駅名或いは地名等を選択的にモバイル情報検索サーバ103に送信できるようにしてもよい。

10

【0210】

(4) 上述の実施形態においては、モバイル情報検索サーバ103と検索エンジン104はインターネット106を介して接続されているが、その代替として、モバイル情報検索サーバ103及び検索エンジン104を一体化したサーバを用いることもできる。

【0211】

(5) 上述の実施形態において、モバイル情報検索サーバ103は、検索エンジン104から所定の地名の関連語を取得したが、当該地名に関連するような単語を取得できればいかなるものから取得してもかまわない。例えば、関連語を取得できる場所としては、その地名に関連する情報が記載されているホームページ等があげられる。

20

【0212】

(6) 上記の実施形態においては、WAMファイル生成処理部304は、検索対象として日本語を想定し、形態素解析を行ったが、日本語以外の言語に対しては、接辞処理(stemming)を行うことで、語形変化の多様性を正規化することができる。ここで、接辞処理とは、予め用意された規則にしたがって接尾辞や接頭辞を削除し、語基(stem)のみを残す処理のことである。

【0213】

また、検索エンジン104には、単語を別の単語に翻訳する機能がある。関連語の分類表示は単語の羅列であり、文法構造を含まないため、係り受け構造の解析などの複雑な翻訳処理を必要としない。上記の実施形態においては、携帯電話102の表示言語として日本語を想定しているが、関連語を日本語以外の別の言語の翻訳することにより、言語横断検索を行ってもよい。関連語の翻訳処理は、携帯電話102の表示の直前に行われる処理である。すなわち図47に示す深さ付与処理のステップS4710の処理とステップS4711の処理の間で行われる処理である。この処理は、必要に応じてオプションスイッチとして提供することが望ましい。

30

【0214】

(7) 上記の実施形態においては、〔数1〕により、単語間の関連性の強さを計算している。これは、発明者の行ったデータ実験において、〔数1〕が他の計算式よりも、良い精度で共起関係を抽出できることが確認されたためである。〔数1〕の代替として、共起関係を抽出するための計算式として情報検索の分野でよく知られている、相互情報量、カイ二乗値、シン普森係数などの計算式を用いても良い。なお、これらの計算式によっても、ある程度の精度で、関連語の共起関係を抽出することができることが発明者によって確認されている。

40

【0215】

(8) また、WAMファイルに対し、全文検索を実行してもよい。このためには、予めインデックス作成プログラムを用いて、WAMファイル内に含まれるキャッシュファイル名、検索結果htmlファイル名及びWAMファイル内の単語の出現頻度等を高速に検索することができる、インデックスファイルを作成しておく。そして、全文検索実行時には、全文検索エンジンがインデックスファイルにアクセスして、高速な検索を実現する。こ

50

れにより、図16に示すステップS1611、ステップS1613及びステップS1615のループ処理をする必要がなくなり、処理の高速化を実現できる。

【0216】

また、WAMファイルに対して全文検索エンジンによる全文検索を用いることにより、第二のcgi3407の処理を一部代替することができる。具体的には、図41のステップS4102を、「地名+関連語で全てのWAMファイルを対象に全文検索を実行する」、とする。そして、ステップS4103は、「検索にヒットしたWAMファイル名からキャッシュファイル名へのAタグを生成する」、とする。

【0217】

また、図41のステップS4102は、上記WAMファイルに対する全文検索エンジンによる全文検索に代えて、キャッシュファイルに対する全文検索エンジンによる全文検索を用いた処理とすることができる。この場合は、検索結果にノイズがやや増えるものの、詳細な検索を行いたい場合に向いている。

10

【0218】

なお、このような全文検索エンジンを用いることで、地名+関連語に限らず、連想語を含む任意の単語を用いた検索も実現できる。

【0219】

(9)また、第二のcgi3407は、図13の検索キー・ファイル名対応テーブル312から取得した検索結果htmlファイル1003を、検索クエリのTFIDF値の大きいものの順で順序付けしてから、携帯電話102に検索結果画面を表示するためのhtml文書を作成してもよい。

20

【0220】

また、検索結果htmlファイル1003は、図13の検索キー・ファイル名対応テーブル312から取得した(緯度、経度)とユーザの現在位置の(緯度、経度)のユークリッド距離の小さいものの順で順序付けしてもよい。

【0221】

検索結果htmlファイル1003の順序付けは、必要に応じてオプションスイッチとして提供することが望ましい。また、順序付けは、ユーザからの要求に応じて選択的に行うようにしてもよい。

【0222】

(10)上記の実施形態においては、GPS受信部108から得られる位置情報に対応する地名を取得しているが、携帯電話102の保有する音声認識の機能により地名等の固有名詞を取得してもよい。

30

また、ユーザが受信したメールからコピーアンドペーストの機能により切り出した固有名詞を取得してもよい。これらの処理は、図38のステップS3802及びステップS3803の処理、すなわち地名(固有名詞)を取得する手段の代替となる。なお、これらの処理は、必要に応じてオプションスイッチとして提供することが望ましい。

【0223】

以上説明したように、本実施形態では、検索エンジンが提供する、地名に対して強い結びつきを示す関連語を出力するサービスを有効活用している。「地名+関連語<1>」、「地名+関連語<2>」、...、で検索して、そこから得られる各ウェブサイトのキャッシュファイルに対し、各関連語の前後の数単語だけに情報を絞り込んでいる。この情報を基に、各キャッシュファイルにおける各関連語同士の出現頻度を計算して、各関連語の分類を行う。情報が絞り込まれているから、情報の純度が高い、逆に言えば、ノイズが少ない。こうして、分類分けされた各関連語の群をユーザに提示することで、情報をその地名に由来するクラス毎に整理して提供できる。

40

それにより、ユーザは目的地に立って、携帯電話に所定の操作を行うだけで、地名入力せずとも、その土地に由来する情報が整理された状態で提供される。したがって、ユーザの利便性が向上するという利点がある。

【0224】

50

また、本実施形態では、本クラスタを作成する際に、仮クラスタの各レコードにストップワードのフラグを設定する。つまり、ストップワードを含むレコードを最後に削除する。このため、ユーザにとって不要な情報を除外できるので、関連語を各カテゴリに分類する際の分類精度を向上させることができる。

【0225】

また、本実施形態は、携帯画面の表示可能文字数に合わせて、適切な文字列長で各クラスタに分類した関連語を表示できるように、本クラスタを表示用本クラスタに変換する。これにより、関連語の意味的な類似性を保ちつつ、携帯電話の小さな画面を有効活用した、分類表示が実現できる。

【0226】

また、本実施形態は、携帯画面に限らず、表示可能文字数が制限されている入出力機器一般、任意文字列の入力が制限されるタッチパネルモニタ付情報端末、高齢者、障害者、幼児など入出力の自由度が制限されたユーザ向けの端末においても、適応可能である。その際、本実施形態の特色である、ユーザの直感に合致する、適切なキーワード（検索クエリ）をクラスタリングして提供できるので、ユーザにとって易しい操作性を提供できると共に、検索出力結果もユーザが希望する内容に合致し易い。

【0227】

以上、本発明の実施形態の例について説明したが、本発明は上記実施形態例に限定されるものではなく、特許請求の範囲に記載した本発明の要旨を逸脱しない限りにおいて、他の変形例、応用例を含むことはいうまでもない。

【図面の簡単な説明】

【0228】

【図1】本発明の一実施の形態による、情報検索システムの全体概略図及び携帯電話の機能ブロック図である。

【図2】モバイル情報検索サーバの機能ブロック図である。

【図3】バッチ処理部及び不揮発性データ記憶部の機能ブロック図である。

【図4】データ取得処理部を中心とする機能ブロック図である。

【図5】WAMファイル生成処理部を中心とする機能ブロック図である。

【図6】関連語出現テーブル生成処理部を中心とする機能ブロック図である。

【図7】ベストスコアテーブル生成処理部を中心とする機能ブロック図である。

【図8】本クラスタ生成処理部を中心とする機能ブロック図である。

【図9】モバイル情報検索サーバと検索エンジンとの間で行われる動作を示すシーケンス図である。

【図10】検索エンジンから送信される関連語及び検索結果htmlファイルを示す図である。

【図11】検索エンジンから送信されるキャッシュファイルを示す図である。

【図12】モバイル情報検索サーバを構成するバッチ処理部の動作の流れを示すフローチャートである。

【図13】GPS地名マスタ及び検索キー・ファイル名対応テーブルを示す図である。

【図14】バッチ処理部を構成するWAMファイル生成処理部の動作の流れを示すフローチャートである。

【図15】WAMファイル示す図である。

【図16】バッチ処理部を構成する関連語出現テーブル生成処理部の動作の流れを示すフローチャートである。

【図17】関連語出現テーブルを示す図である。

【図18】バッチ処理部を構成するベストスコアテーブル生成処理部の動作の流れを示すフローチャートである。

【図19】バッチ処理部を構成するベストスコアテーブル生成処理部の動作の流れを示すフローチャートの続きである。

【図20】仮ベストスコアテーブルの更新を示す図である。

10

20

30

40

50

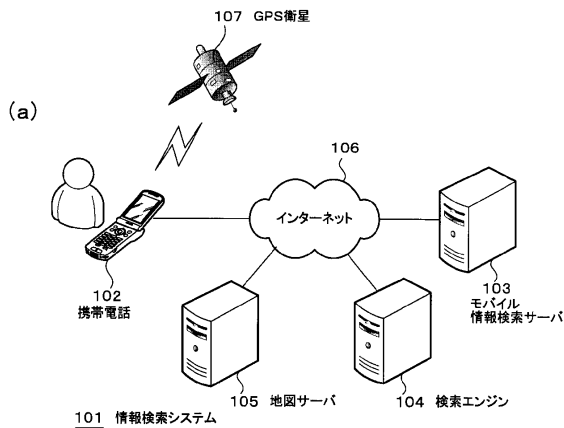
- 【図 2 1】本ベストスコアテーブルを示す図である。
- 【図 2 2】バッチ処理部を構成する本クラスタ生成処理部の動作の流れを示すフローチャートである。
- 【図 2 3】仮クラスタの生成手順を示す図である。
- 【図 2 4】仮クラスタを示す図である。
- 【図 2 5】本クラスタ生成処理部の `cnt` 付与処理の詳細な処理の流れを示すフローチャートである。
- 【図 2 6】`cnt` が付与された仮クラスタを示す図である。
- 【図 2 7】本クラスタ生成処理部の `valid` 付与処理の詳細な処理の流れを示すフローチャートである。 10
- 【図 2 8】`valid` が付与された仮クラスタを示す図である。
- 【図 2 9】本クラスタ生成処理部の `alt` 付与処理の詳細な処理の流れを示すフローチャートである。
- 【図 3 0】`alt` が付与された仮クラスタを示す図である。
- 【図 3 1】ストップワードが付与された仮クラスタを示す図である。
- 【図 3 2】本クラスタを示す図である。
- 【図 3 3】本クラスタを示す図の続きである。
- 【図 3 4】リアルタイム処理部及び不揮発性データ記憶部の機能ブロック図を示す図である。
- 【図 3 5】携帯電話とモバイル情報検索サーバと地図サーバとの間で行われる動作の流れを示すシーケンス図である。 20
- 【図 3 6】携帯電話の画面に表示される分類の表示画面を示す図である。
- 【図 3 7】携帯電話の画面に表示される検索結果画面及び地図付きキャッシュ画面を示す図である。
- 【図 3 8】リアルタイム処理部を構成する第一の `cgi` で行われる動作の流れを示すフローチャートである。
- 【図 3 9】表示用本クラスタを示す図である。
- 【図 4 0】表示用本クラスタ `html` 文書を示す図である。
- 【図 4 1】リアルタイム処理部を構成する第二の `cgi` で行われる動作の流れを示すフローチャートである。 30
- 【図 4 2】リアルタイム処理部を構成する第三の `cgi` で行われる動作の流れを示すフローチャートである。
- 【図 4 3】携帯電話とモバイル情報検索サーバとの間で行われる連想語検索に関する動作の流れを示すシーケンス図である。
- 【図 4 4】携帯電話に表示される連想語検索結果画面を示す図である。
- 【図 4 5】リアルタイム処理部を構成する第四の `cgi` で行われる動作の流れを示すフローチャートである。
- 【図 4 6】連想語テーブルを示す図である。
- 【図 4 7】リアルタイム処理部を構成する第一の `cgi` で行われる深さ付与処理の流れを示すフローチャートである。 40
- 【図 4 8】深さを付与した表示用本クラスタを示す図である。
- 【図 4 9】深さで分類した表示用本クラスタを示す図である。
- 【図 5 0】深さで分類した分類の選択画面を示す図である。
- 【図 5 1】携帯電話に表示されるアプリケーション起動時の表示画面を示す図である。
- 【符号の説明】
- 【 0 2 2 9】
- 1 0 1 ... 情報検索システム、1 0 2 ... 携帯電話、1 0 3 ... モバイル情報検索サーバ、1 0 4 ... 検索エンジン、1 0 5 ... 地図サーバ、1 0 6 ... インターネット、1 0 7 ... GPS 衛星、1 0 8 ... GPS 受信部、1 0 9 ... 入出力制御部、1 1 0 ... 表示制御部、1 1 1 ... 表示部、1 1 2 ... 入力部、1 1 3 ... `web` ブラウザプログラム、2 0 2 ... バッチ理部、2 0 3 50

...リアルタイム処理部、204...不揮発性データ記憶部、302...非対話型ウェブクライアント、303...データ取得処理部、304...WAMファイル生成処理部、305...関連語出現テーブル生成処理部、306...ベストスコアテーブル生成処理部、307...本クラスタ生成処理部、308...GPS地名マスタ、309...地名関連語リスト、310...検索結果ファイルディレクトリ、311...キャッシュファイルディレクトリ、312...検索キー・ファイル名対応テーブル、313...WAMファイルディレクトリ、314...関連語出現テーブル、315...仮ベストスコアテーブル、316...本ベストスコアテーブル、317...仮クラスタ、318...本クラスタディレクトリ、319...本クラスタ作成部、502...WAMファイル、802...本クラスタ、1003...検索結果htmlファイル、1004...リンク先URL、1102...キャッシュファイル、1502...括弧、2002...ソート前仮ベストスコアテーブル、2003...ソート済仮ベストスコアテーブル(仮ベストスコアテーブル)、2102...2レコード、2103...1レコード、3402...webサーバプログラム、3403...第一のcgi、3404...表示用本クラスタ、3405...携帯端末マスタ、3406...プロポーショナルフォントファイル、3407...第二のcgi、3408...第三のcgi、3409...第四のcgi、3410...連想語テーブル、3602...分類の選択画面、記号...3603、3702...検索結果画面、3703...ボタン、3704...ボタン、3705...地図付きキャッシュ画面、3706...地図画像、3707...記号、4002...表示用本クラスタhtml文書、4402...関連語の連想語検索結果画面、4403...記号、4404...連想語の連想語検索結果画面、4802...表示用本クラスタ、5002点深さで分類した表意洋本クラスタhtml文書、5003...深さで分類した分類の選択画面、5102...アプリケーション起動時表示画面

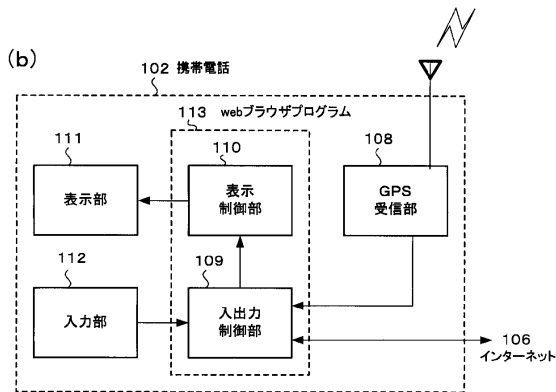
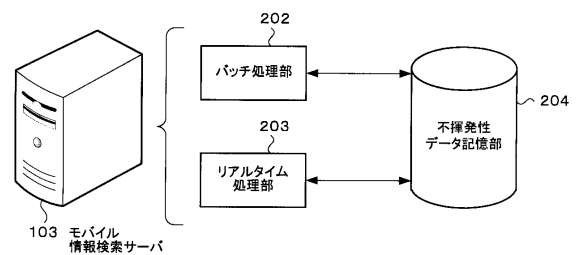
10

20

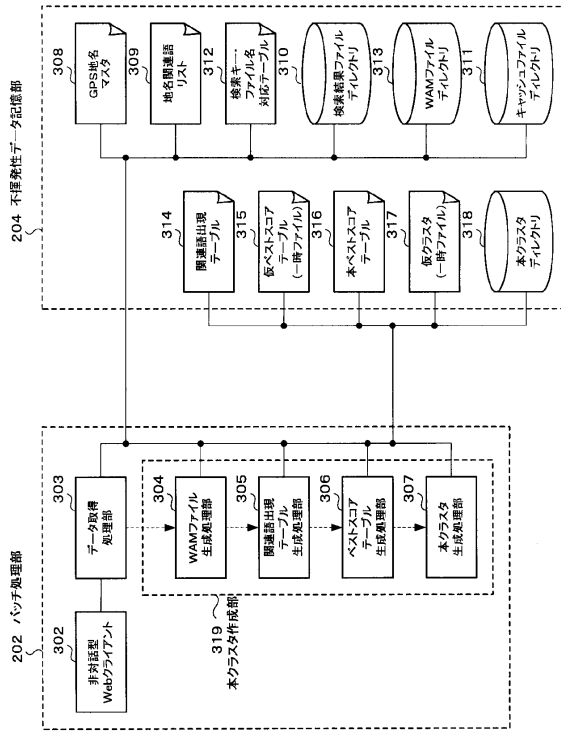
【図1】



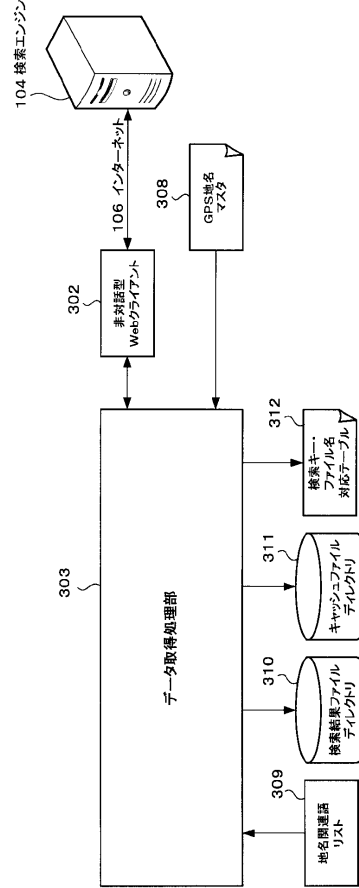
【図2】



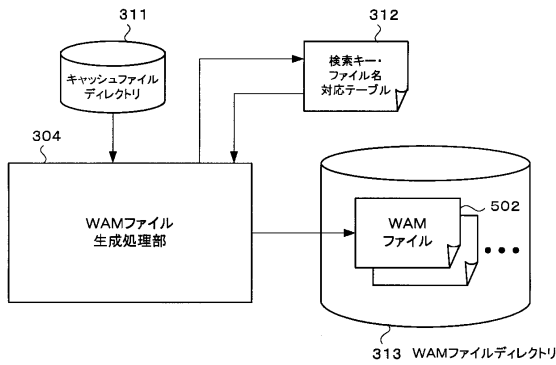
【図3】



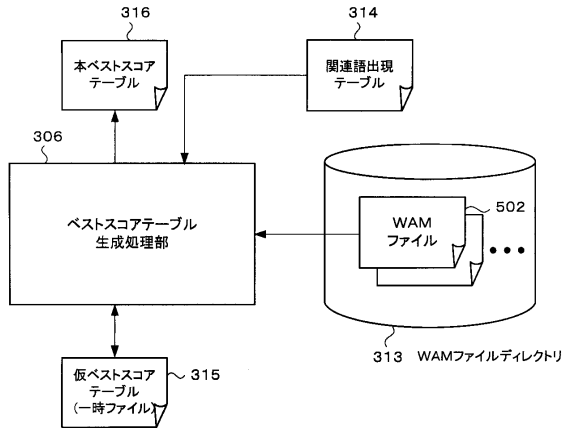
【図4】



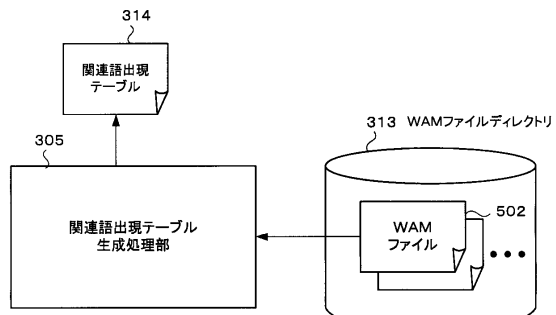
【図5】



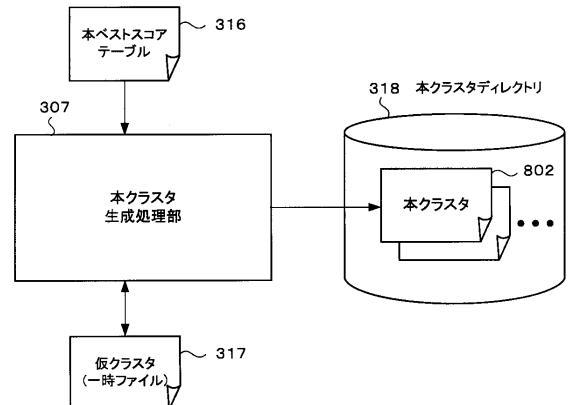
【図7】



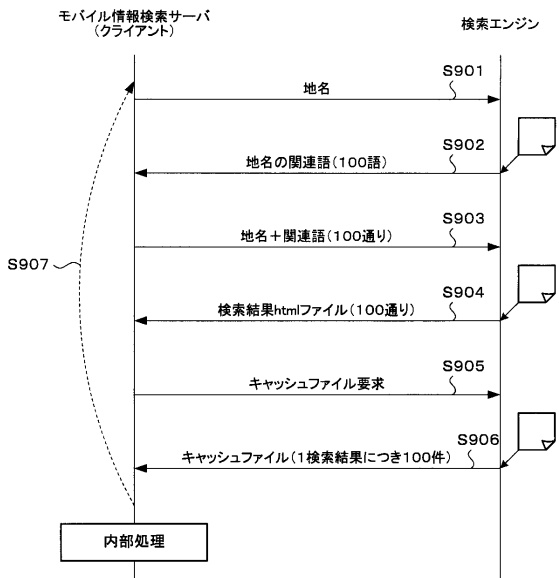
【図6】



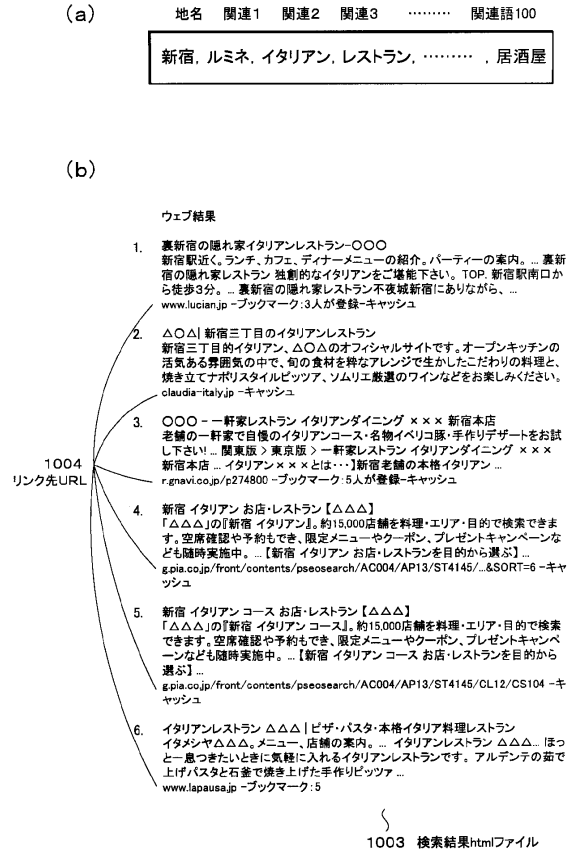
【図8】



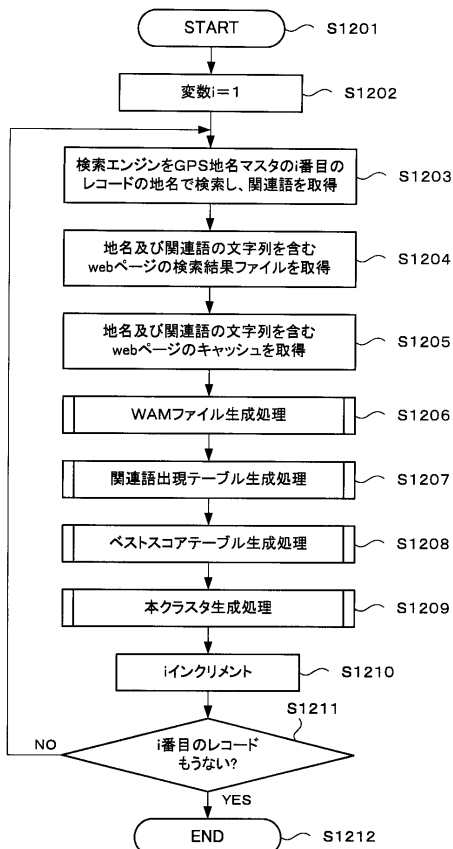
【図9】



【図10】



【図12】



バッチ処理・メインフロー

【図13】

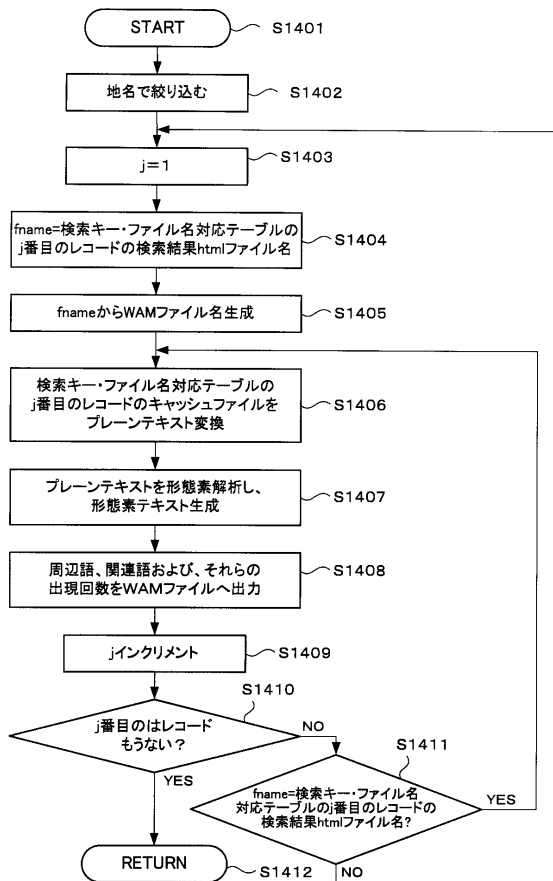
(a) 308 GPS地名マスタ

地名	緯度範囲(緯度1, 経度2)	経度範囲(緯度1, 経度2)	...
新宿	(×, ○)	(△, ■)	...
銀座	(◎, □)	(▲, ▼)	...
有楽町	(▼, ●)	(★, ☆)	...

(b) 312 検索キーワードファイル名対応テーブル

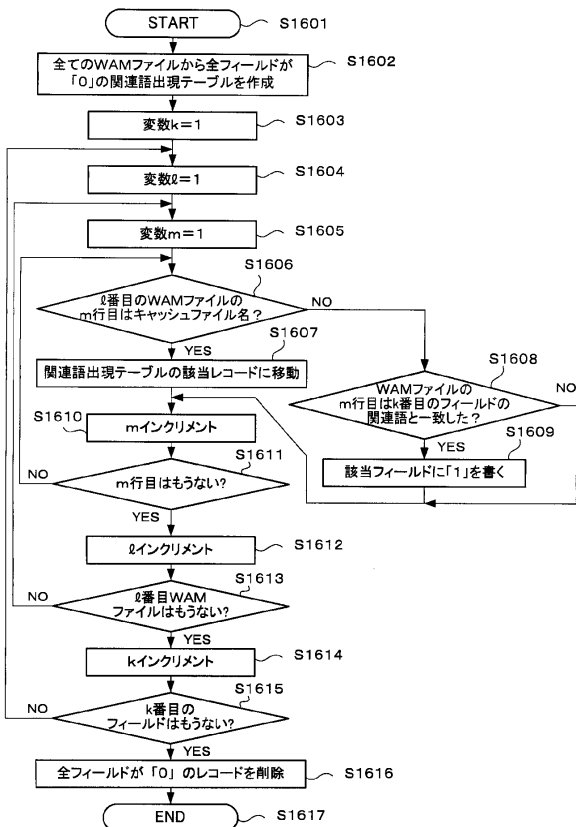
地名	関連語	検索結果htmlファイル名	キャッシュファイル名	WAMファイル名	住所 (緯度, 経度)
新宿	新宿二丁目	shinjyuku_001	shinjyuku_001	shinjyuku_001.wam	東京都新宿区新宿2-●-●
新宿	新宿二丁目	shinjyuku_001	shinjyuku_001	shinjyuku_001.wam	空
新宿	新宿二丁目	shinjyuku_001	shinjyuku_001	shinjyuku_001.wam	東京都新宿区新宿2-▲-◎
新宿	イタリアン	shinjyuku_002	shinjyuku_002	shinjyuku_002.wam	空
新宿	イタリアン	shinjyuku_002	shinjyuku_002	shinjyuku_002.wam	東京都新宿区新宿2-★-★
新宿	銀座	shinjyuku_100	shinjyuku_100	shinjyuku_100.wam	東京都新宿区△-■-●
新宿	銀座	ginza_001	ginza_001	ginza_001.wam	東京都中央区銀座●-△-□
新宿	銀座	ginza_100	ginza_100	ginza_100.wam	東京都中央区銀座▽-★-☆
新宿	有楽町	yurakucho_001	yurakucho_001	yurakucho_001.wam	東京都千代田区有楽町◎-★-△

【図14】



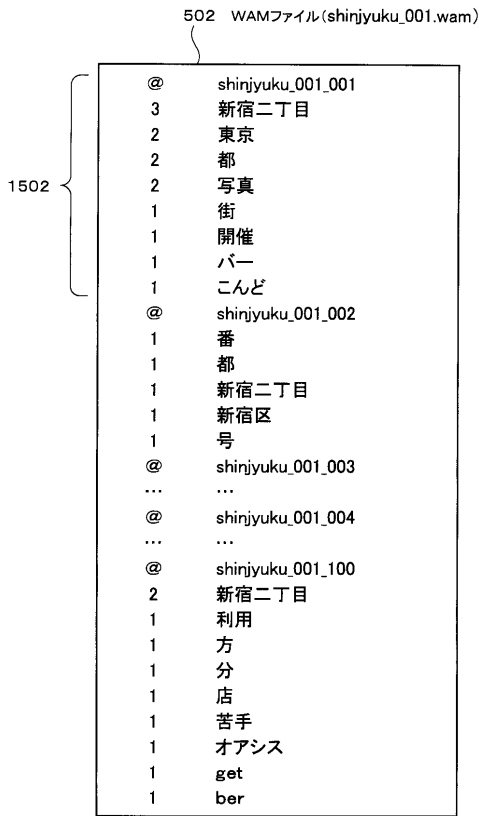
WAMファイル生成処理(ステップS1206)

【図16】



関連語出現テーブル生成処理(ステップS1207)

【図15】



【図17】

314 関連語出現テーブル

(a)

キャッシュファイル名	関連語	term(1)	term(2)	term(3)	...	term(100)
shinjiyuku_001_001		0	0	0	...	0
shinjiyuku_001_002		0	0	0	...	0
...	
shinjiyuku_001_100		0	0	0	...	0
shinjiyuku_002_001		0	0	0	...	0
shinjiyuku_002_002		0	0	0	...	0
...	
shinjiyuku_002_100		0	0	0	...	0
...	
shinjiyuku_100_001		0	0	0	...	0
...	
shinjiyuku_100_100		0	0	0	...	0

314 関連語出現テーブル

(b)

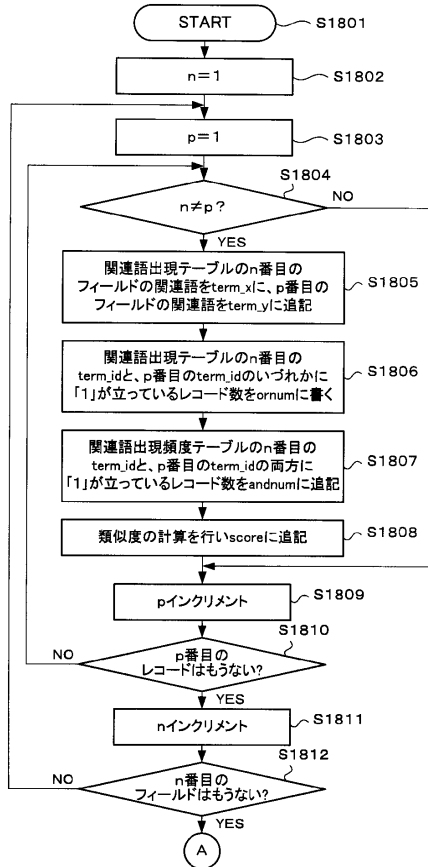
キャッシュファイル名	関連語	term(1)	term(2)	term(3)	...	term(100)
shinjiyuku_001_001		0	0	0	...	0
shinjiyuku_001_002		0	0	0	...	1
...	
shinjiyuku_001_100		1	0	1	...	1
shinjiyuku_002_001		1	0	0	...	0
shinjiyuku_002_002		1	1	1	...	0
...	
shinjiyuku_002_100		1	1	0	...	1
...		0
shinjiyuku_100_001		0	0	1	...	1
...	
shinjiyuku_100_100		0	1	0	...	0

314 関連語出現テーブル

(c)

キャッシュファイル名	関連語	term(1)	term(2)	term(3)	...	term(100)
shinjiyuku_001_002		0	0	0	...	1
...	
shinjiyuku_001_100		1	0	1	...	1
shinjiyuku_002_001		1	0	0	...	0
shinjiyuku_002_002		1	1	1	...	0
...	
shinjiyuku_002_100		1	1	0	...	1
...		0
shinjiyuku_100_001		0	0	1	...	1
...	
shinjiyuku_100_100		0	1	0	...	0

【図18】



ベストスコアテーブル生成処理(ステップS1208)

【図20】

2002 ソート前仮ベストスコアテーブル

(a)

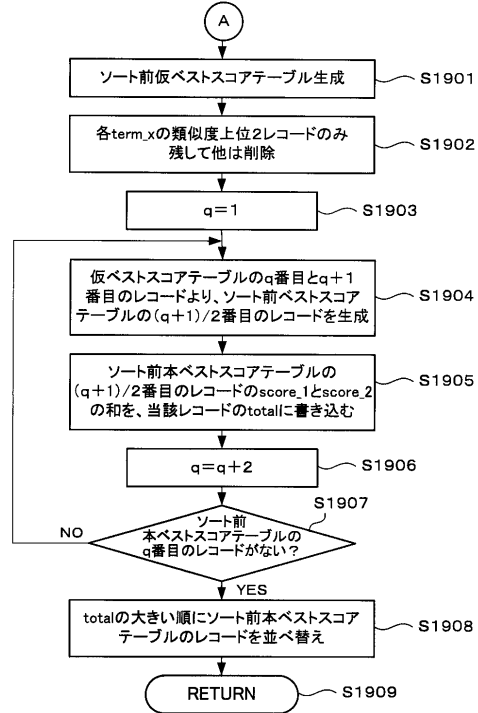
score	andnum	ornum	term_x	term_y
0.013	12	2375	店	郵便局
0.764	396	3102	店	新宿区
...
0.427	200	2481	店	グルメ
...
0.163	14	226	郵便局	住友ビル
...
0.167	14	221	郵便局	野村ビル
...
0.05	22	1364	新宿区	ラーメン
...

2003 ソート済仮ベストスコアテーブル(仮ベストスコアテーブル)

(b)

score	andnum	ornum	term_x	term_y
0.764	396	3102	店	新宿区
0.427	200	2481	店	グルメ
0.167	14	221	郵便局	野村ビル
0.163	14	226	郵便局	住友ビル
...

【図19】



ベストスコアテーブル生成処理(ステップS1208)続き

【図21】

2102 2レコード

(a)

score	andnum	ornum	term_x	term_y
0.764	396	3102	店	新宿区
0.427	200	2481	店	グルメ

2103 1レコード

(b)

total	term	score_1	term_1	score_2	term_2
1.191	店	0.764	新宿区	0.427	グルメ

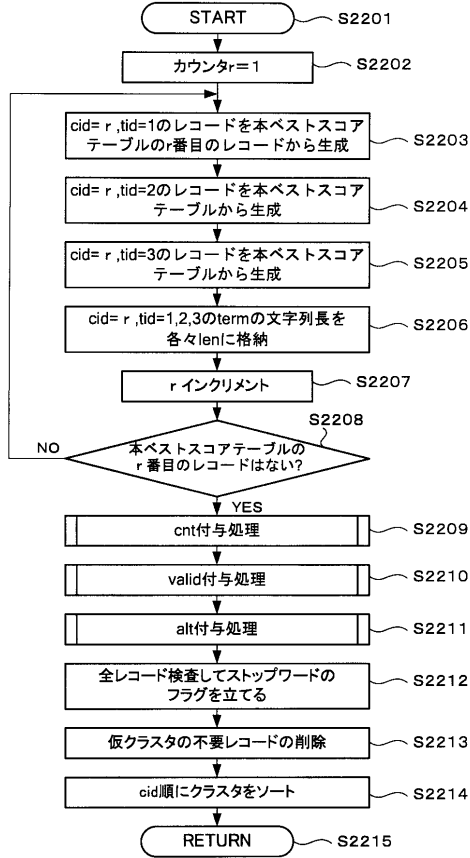
316 本ベストスコアテーブル

(c)

total	term	score_1	term_1	score_2	term_2
1.697	新宿駅	0.965	南口	0.732	西口
1.479	西口	0.747	ビックカメラ	0.732	新宿駅
1.297	カプセルホテル	1.006	新宿区役所	0.291	ビジネスホテル
...
1.191	店	0.764	新宿区	0.427	グルメ
...

注: 以下は登録商標
ビックカメラ

【図 2 2】



本クラスタ生成処理 (ステップS1209)

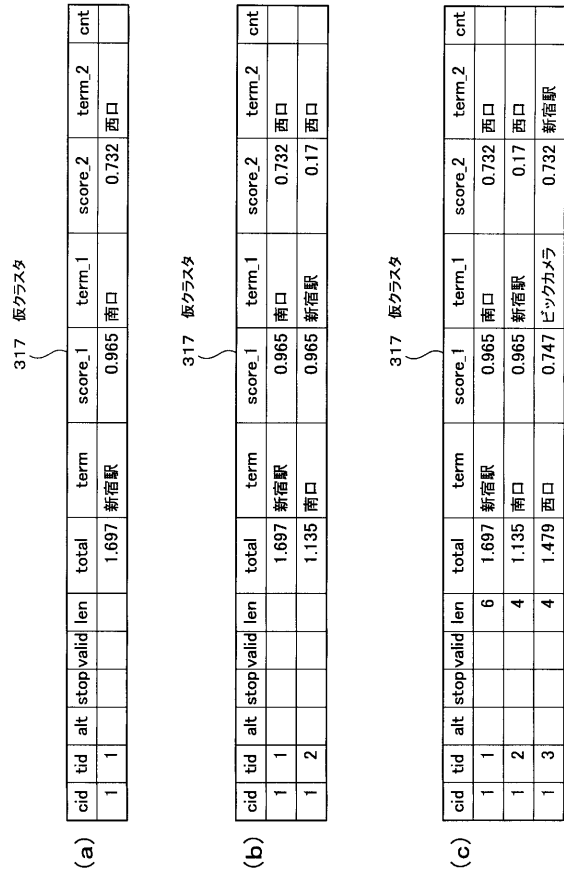
【図 2 4】

317 仮クラスタ

cid	tid	alt	stop	valid	len	total	term	score_1	term_1	score_2	term_2	cnt
1	1				6	1.697	新宿駅	0.965	南口	0.732	西口	
1	2				4	1.135	南口	0.965	新宿駅	0.17	西口	
1	3				4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	
2	1				4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	
2	2				12	1.2	ビックカメラ	0.747	西口	0.435	東口	
2	3				6	1.697	新宿駅	0.965	南口	0.732	西口	
3	1				14	1.297	カブセルホテル	1.006	新宿区役所	0.291	ビックカメラ	
...

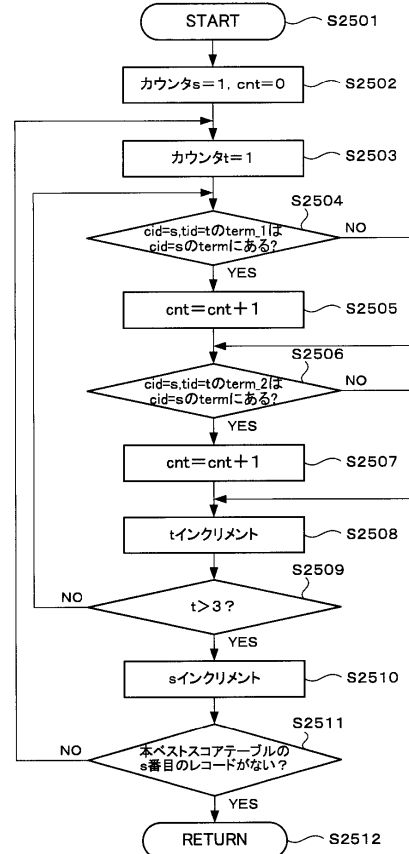
注: 以下は登録商標
ビックカメラ

【図 2 3】



cnt付与処理 (ステップS2209)

【図 2 5】



cnt付与処理 (ステップS2209)

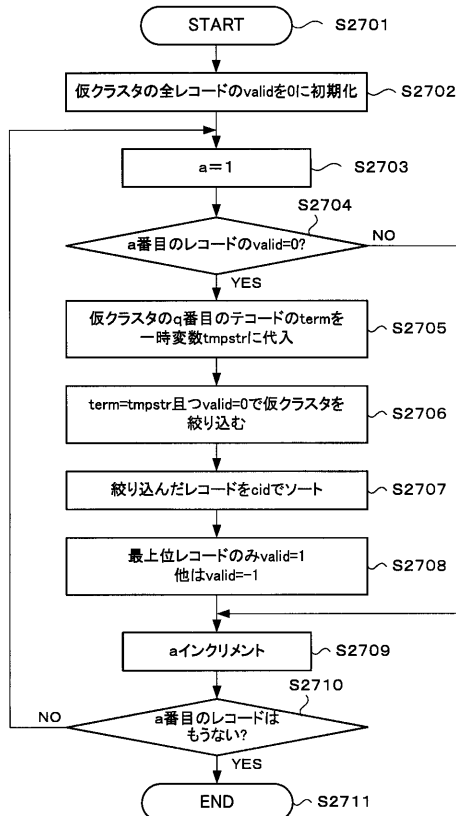
【図26】

317 仮クラスタ

cid	tid	alt	stop	valid	len	total	term	score_1	term_1	score_2	term_2	cnt
1	1				6	1.697	新宿駅	0.965	南口	0.732	西口	2
1	2				4	1.135	南口	0.965	新宿駅	0.17	西口	2
1	3				4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	1
2	1				4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	2
2	2				12	1.2	ビックカメラ	0.747	西口	0.435	東口	1
2	3				6	1.697	新宿駅	0.965	南口	0.732	西口	1
3	1				14	1.297	カゲルホテル	1.006	新宿区役所	0.291	ピジネスホテル	2
...
6	1				10	1.136	新宿区役所	1.006	カゲルホテル	0.13	郵便局	2
6	2				14	1.297	カゲルホテル	1.006	新宿区役所	0.291	ピジネスホテル	1
6	3				6	0.33	郵便局	0.167	野村ビル	0.163	住友ビル	0
...

注:以下は登録商標
ビックカメラ

【図27】



valid付与処理(ステップS2210)

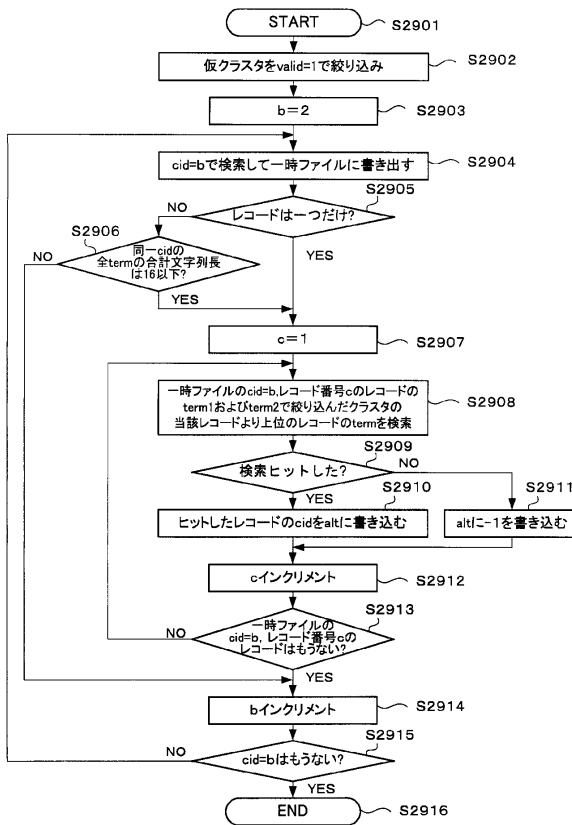
【図28】

317 仮クラスタ

cid	tid	alt	stop	valid	len	total	term	score_1	term_1	score_2	term_2	cnt
1	1				6	1.697	新宿駅	0.965	南口	0.732	西口	2
1	2				4	1.135	南口	0.965	新宿駅	0.17	西口	2
1	3				4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	1
2	1				4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	2
2	2				12	1.2	ビックカメラ	0.747	西口	0.435	東口	1
2	3				6	1.697	新宿駅	0.965	南口	0.732	西口	1
3	1				14	1.297	カゲルホテル	1.006	新宿区役所	0.291	ピジネスホテル	2
...

注:以下は登録商標
ビックカメラ

【図29】



alt付与処理(ステップS2211)

【 30】

cid	tid	alt	stop	valid	len	total	term	score_1	term_1	score_2	term_2	cnt
1	1	0	0	1	6	1.697	新宿駅	0.965	南口	0.732	西口	2
1	2	0	0	1	4	1.135	南口	0.965	新宿駅	0.17	西口	2
1	3	0	0	1	4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	1
2	1	0	0	1	4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	2
2	2	0	0	1	12	1.2	ビックカメラ	0.747	西口	0.435	東口	1
2	3	0	0	1	6	1.697	新宿駅	0.965	南口	0.732	西口	1
3	1	0	0	1	14	1.297	カゼルホテル	1.006	新宿区役所	0.291	ビジネスホテル	2
...
9	1	-1	1	1	6	1.079	ホテル	0.621	ビジネスホテル	0.458	ビルトン	1
...
17	1	0	0	1	4	0.811	映画	0.599	映画館	0.212	新宿ハルト9	2
17	2	0	0	1	6	0.775	映画館	0.599	映画	0.176	新宿ハルト9	2
17	3	0	0	1	12	0.388	新宿ハルト9	0.212	映画	0.176	映画館	2
...
61	1	17	1	6	0.264	バルト	0.169	映画館	0.095	映画	映画	2
61	2	0	0	-1	6	0.775	映画館	0.599	映画	0.176	新宿ハルト9	1
61	3	0	0	1	4	0.811	映画	0.599	映画館	0.212	新宿ハルト9	1
...

注:以下は登録商標
ビックカメラ

【 31】

cid	tid	alt	stop	valid	len	total	term	score_1	term_1	score_2	term_2	cnt
1	1	0	1	1	6	1.697	新宿駅	0.965	南口	0.732	西口	2
1	2	0	1	1	4	1.135	南口	0.965	新宿駅	0.17	西口	2
1	3	0	1	1	4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	1
2	1	0	1	1	4	1.479	西口	0.747	ビックカメラ	0.732	新宿駅	2
2	2	0	1	1	12	1.2	ビックカメラ	0.747	西口	0.435	東口	1
2	3	0	1	1	6	1.697	新宿駅	0.965	南口	0.732	西口	1
3	1	0	0	1	14	1.297	カゼルホテル	1.006	新宿区役所	0.291	ビジネスホテル	2
...

注:以下は登録商標
ビックカメラ

【 32】

802 ネクラスタ

cid	tid	alt	stop	valid	len	total	term	score_1	term_1	score_2	term_2	cnt
3	1	0	0	1	14	1.297	カゼルホテル	1.006	新宿区役所	0.291	ビジネスホテル	2
3	2	0	0	1	10	1.136	新宿区役所	1.006	カゼルホテル	0.13	郵便局	1
3	3	0	0	1	14	0.912	ビジネスホテル	0.621	ホテル	0.291	カゼルホテル	1
4	3	-1	0	1	4	1.05	東口	0.597	新宿駅	0.453	ビックカメラ	1
8	1	0	0	1	6	1.086	小田急	0.564	新宿駅	0.522	ミロード	2
8	2	0	0	1	4	0.728	京王	0.564	小田急	0.164	新宿駅	1
8	3	0	0	1	8	0.634	ミロード	0.522	小田急	0.112	店	1
9	1	-1	0	1	6	1.079	ホテル	0.621	ビジネスホテル	0.458	ビルトン	2
9	3	-1	0	1	8	0.543	ビルトン	0.458	ホテル	0.076	発	1
12	3	0	0	1	10	0.962	イスタラン	0.437	グルメ	0.437	イスタラン	2
15	1	-1	0	1	14	0.882	ヨドバシカメラ	0.708	西口	0.174	ビックカメラ	2
16	1	0	0	1	10	0.829	ラブホテル	0.544	ラブホ	0.285	ホテル	2
16	2	0	0	1	6	0.64	ラブホ	0.544	ラブホテル	0.096	ホテル	2
17	1	0	0	1	4	0.811	映画	0.599	映画館	0.212	新宿ハルト9	2
17	2	0	0	1	6	0.755	映画館	0.599	映画	0.176	新宿ハルト9	2
17	3	0	0	1	12	0.388	新宿ハルト9	0.212	映画	0.176	映画館	2
17	4	61	0	1	6	0.264	バルト	0.169	映画館	0.095	映画	2
18	1	0	0	1	10	0.755	都営新宿線	0.656	三丁目	0.119	京王	2
18	2	0	0	1	6	0.721	三丁目	0.656	都営新宿線	0.085	店	1
22	3	-1	0	1	4	0.574	和食	0.319	居酒屋	0.255	レストラン	1
23	1	-1	0	1	4	0.671	焼肉	0.489	食べ放題	0.222	グルメ	2
23	2	-1	0	1	8	0.562	食べ放題	0.489	焼肉	0.213	ランチ	1

注:以下は登録商標
小田急、京王、ミロード、ビルトン、ヨドバシカメラ

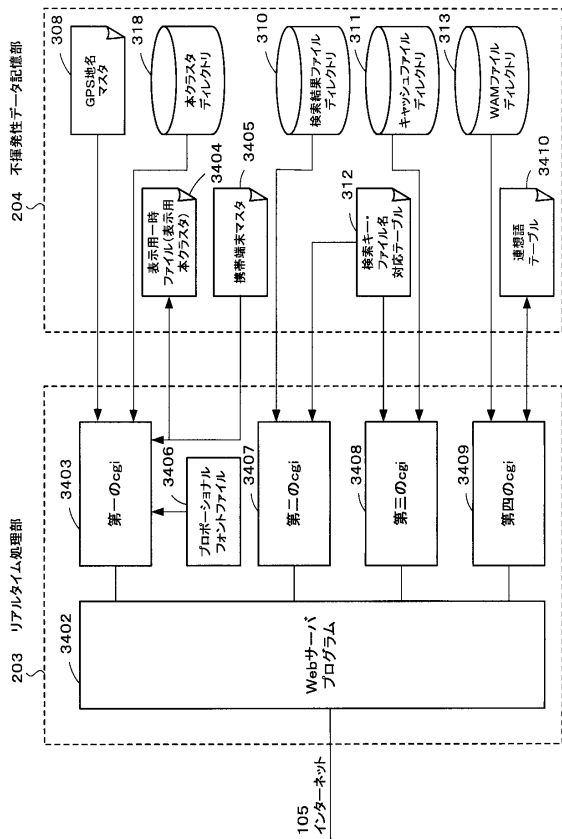
【 33】

802 ネクラスタ

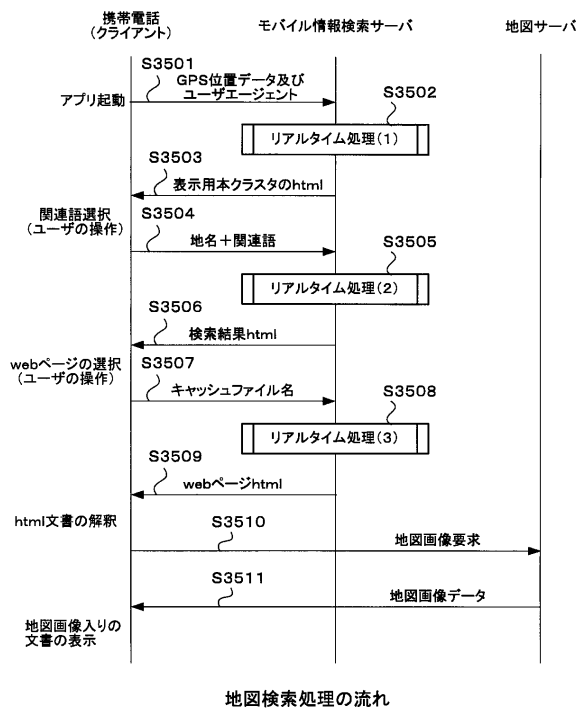
cid	tid	alt	stop	valid	len	total	term	score_1	term_1	score_2	term_2	cnt
24	1	0	0	1	2	0.667	鍋	0.5	もつ鍋	0.167	居酒屋	2
24	2	0	0	1	6	0.632	志年会	0.5	鍋	0.132	居酒屋	1
24	3	0	0	1	6	0.282	志年会	0.167	鍋	0.115	もつ鍋	2
29	1	-1	0	1	6	0.609	居酒屋	0.319	和食	0.29	グルメ	2
33	1	0	0	1	6	0.5	伊勢丹	0.279	三越	0.221	デパート	2
33	2	0	0	1	4	0.446	伊勢丹	0.279	伊勢丹	0.167	三越	2
34	1	-1	0	1	6	0.338	デパート	0.221	伊勢丹	0.167	三越	2
34	2	0	0	1	6	0.497	カフェ	0.275	レストラン	0.222	イタリアン	2
35	1	-1	0	1	6	0.49	ランチ	0.275	グルメ	0.215	カフェ	2
12	3	-1	0	1	8	0.341	野村ビル	0.174	新宿センタービル	0.143	郵便局	1
36	1	-1	0	1	6	0.43	マルイ	0.287	丸井	0.143	伊勢丹	2
38	2	-1	0	1	4	0.427	丸井	0.287	マルイ	0.14	伊勢丹	2
38	1	0	0	1	16	0.353	京王プラザホテル	0.243	ホテル	0.11	ホテルインナー	2
46	3	0	0	1	24	0.243	ホテルインナー	0.133	栄	0.11	京王プラザホテル	1
47	1	-1	0	1	8	0.345	ぐるなび	0.188	レストラン	0.157	グルメ	2
48	1	-1	0	1	20	0.344	新宿ワシントンホテル	0.241	ホテル	0.103	京王プラザホテル	2
49	3	-1	0	1	6	0.33	郵便局	0.163	野村ビル	0.163	住友ビル	1
51	3	-1	0	1	8	0.319	住友ビル	0.163	郵便局	0.156	野村ビル	2
52	1	-1	0	1	6	0.323	ケーキ	0.169	食べ放題	0.154	カフェ	2
53	1	-1	0	1	18	0.32	新宿ワシントンホテル	0.213	ホテル	0.107	京王プラザホテル	2
62	1	0	0	1	14	0.26	ホテルベッパ	0.153	グルメ	0.107	美容院	2
62	3	0	0	1	6	0.213	美容院	0.107	ホテルベッパ	0.106	ネイルサロン	1
...

注:以下は登録商標
小田急、京王、ミロード、ビルトン、ヨドバシカメラ

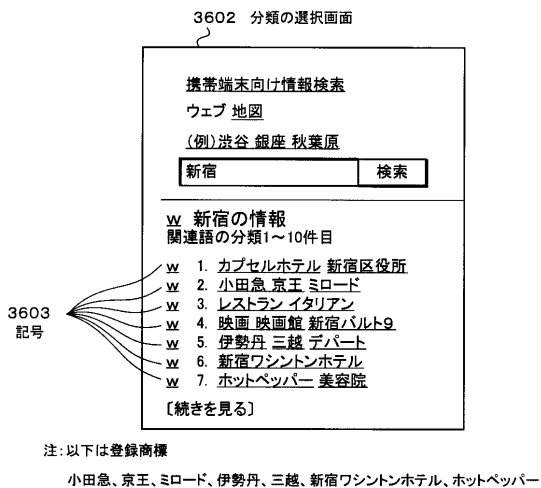
【図 3 4】



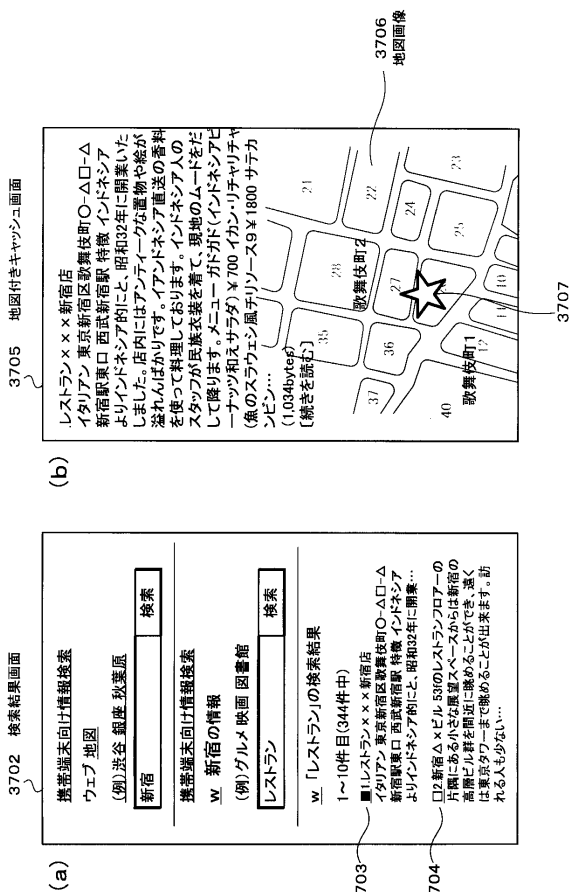
【図 3 5】



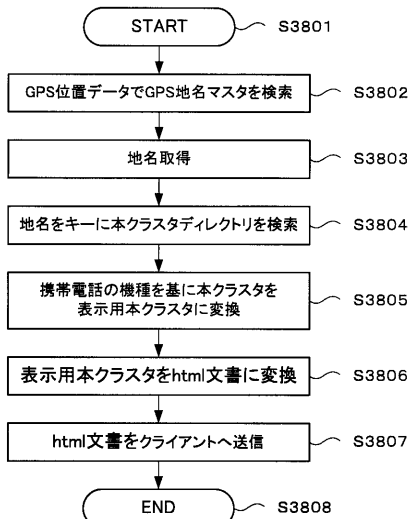
【図 3 6】



【図 3 7】



【図38】



第一のcgi(リアルタイム処理(1))

【図39】

3404 表示用本クラスタ

cid	tid	alt	stop	valid	len	total	term	score.1	term.1	score.2	term.2	cnt
3	1	0	0	1	14	1.297	カプセルホテル	1.006	新宿区役所	0.291	ビジネスホテル	2
3	2	0	0	1	10	1.136	新宿区役所	1.006	カプセルホテル	0.13	郵便局	1
8	1	0	0	1	6	0.086	小田急	0.564	京王	0.522	ミロード	2
8	2	0	0	1	4	0.728	京王	0.584	小田急	0.164	新宿駅	1
8	3	0	0	1	8	0.634	ミロード	0.522	小田急	0.112	店	1
12	1	0	0	1	10	0.914	レストラン	0.477	グルメ	0.437	イタリアン	2
12	3	0	0	1	10	0.692	イタリアン	0.437	レストラン	0.255	和食	1
17	1	0	0	1	4	0.811	映画館	0.589	映画館	0.212	新宿バルト9	2
17	2	0	0	1	6	0.775	映画館	0.589	映画	0.176	新宿バルト9	2
17	3	0	0	1	12	0.388	新宿バルト9	0.212	映画	0.176	映画館	2
33	1	0	0	1	6	0.5	伊勢丹	0.279	三越	0.221	デパート	2
33	2	0	0	1	4	0.446	三越	0.279	伊勢丹	0.167	デパート	2
33	3	0	0	1	8	0.388	デパート	0.221	伊勢丹	0.167	三越	2
48	1	-1	0	1	20	0.344	新宿ワシントンホテル	0.241	ホテル	0.103	京王プラザホテル	2
62	1	0	0	1	14	0.26	ホットペッパー	0.153	グルメ	0.107	美容院	2
62	3	0	0	1	6	0.213	美容院	0.107	ホットペッパー	0.106	ネイルサロン	1
...

注: 以下は登録商標
小田急、京王、ミロード、伊勢丹、三越、新宿ワシントンホテル、ホットペッパー

【図40】

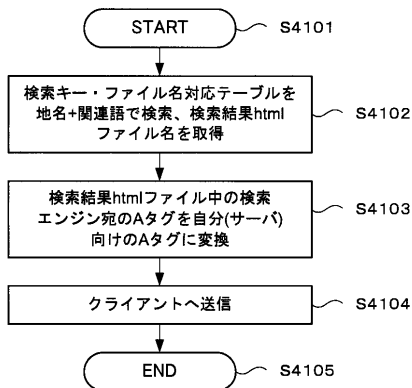
- 新宿
- (01)カプセルホテル 新宿区役所
 - (02)小田急 京王 ミロード
 - (03)レストラン イタリアン
 - (04)映画 映画館 新宿バルト9
 - (05)伊勢丹 三越 デパート
 - (06)新宿ワシントンホテル
 - (07)ホットペッパー 美容院

4002 表示用本クラスタhtml文書

注: 以下は登録商標

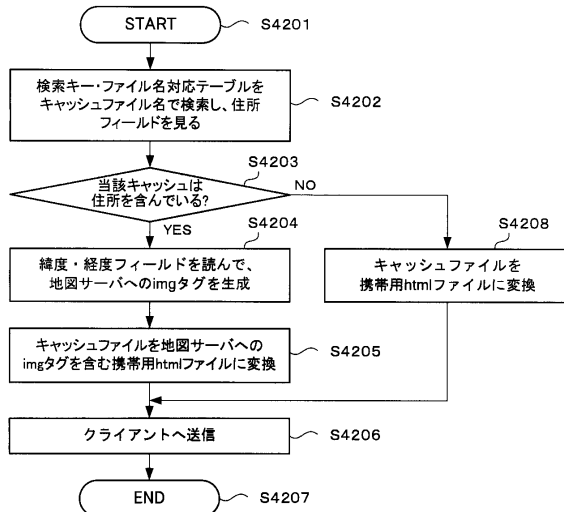
小田急、京王、ミロード、伊勢丹、三越、新宿ワシントンホテル、ホットペッパー

【図41】



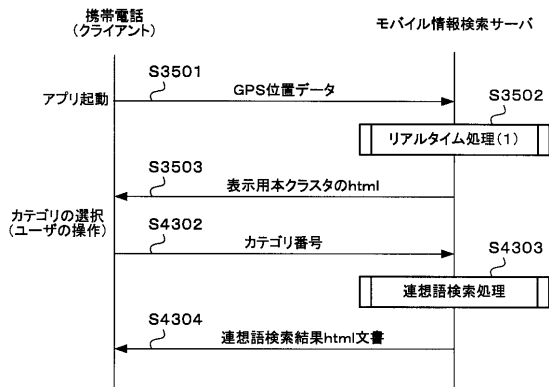
第二のcgi(リアルタイム処理(2))

【図42】



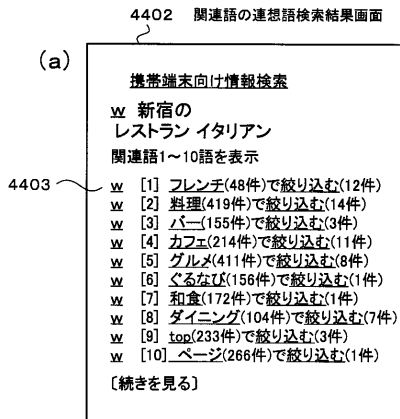
第三のcgi(リアルタイム処理(3))

【図43】

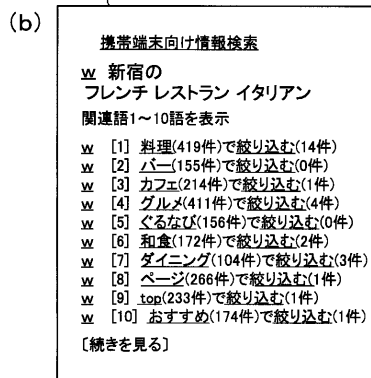


連想語検索処理の流れ

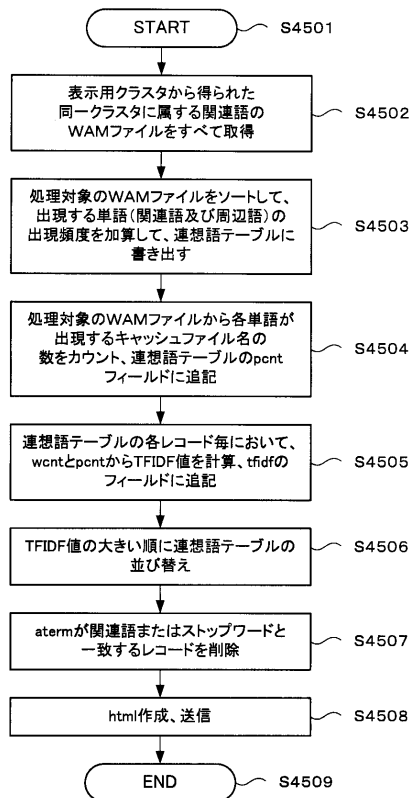
【図44】



4404 連想語の連想語検索結果画面

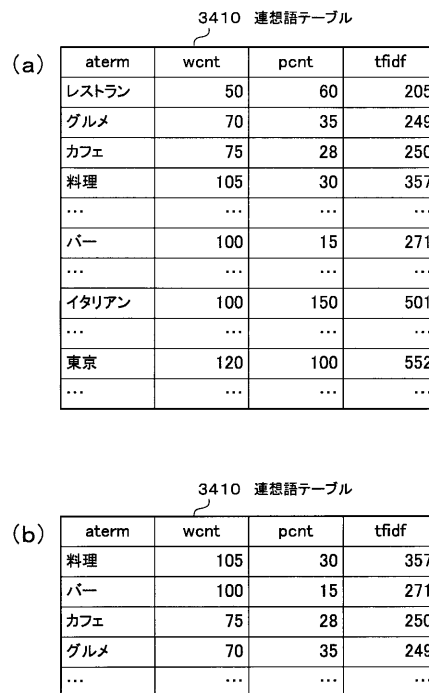


【図45】

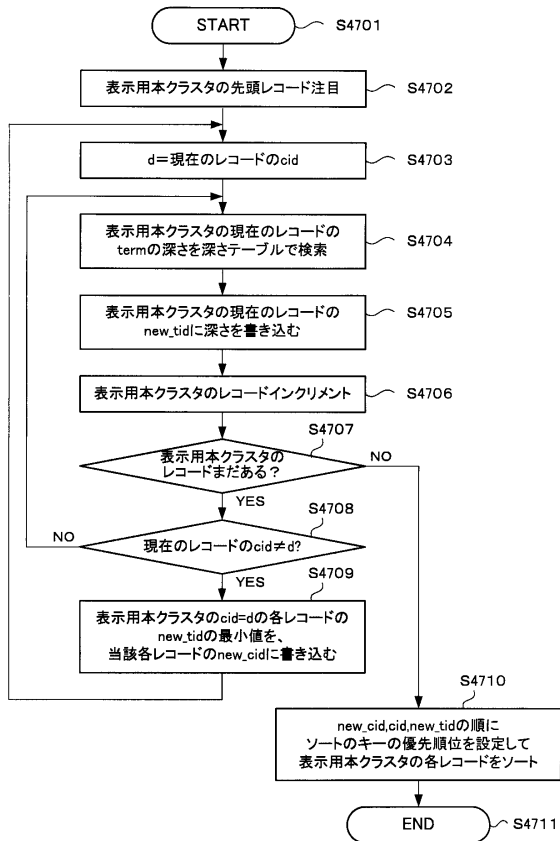


第四のcgi(連想語検索処理)

【図46】



【図47】



深さ付与処理

【図49】

4802 表示用本クラスタ

cid	tid	alt	stop	valid	len	total	term	score.1	term.1	score.2	term.2	ont	new_cid	new_tid
12	1	0	0	1	10	0.914	レストラン	0.477	グルメ	0.437	イタリアン	2	5	5
12	3	0	0	1	10	0.692	レストラン	0.437	レストラン	0.255	和食	1	5	6
33	3	0	0	1	8	0.388	デパート	0.221	伊勢丹	0.167	伊勢丹	2	5	5
33	1	0	0	1	6	0.5	伊勢丹	0.279	三越	0.221	デパート	2	5	6
33	2	0	0	1	4	0.446	三越	0.179	伊勢丹	0.167	デパート	2	5	6
62	1	0	0	1	14	0.26	ホテル	0.153	グルメ	0.107	美容院	2	5	5
62	3	0	0	1	6	0.213	美容院	0.107	ホットベッパ	0.106	ネイルサロン	1	5	6
8	1	0	0	1	4	1.086	小田急	0.564	京王	0.164	京王	1	6	6
8	2	0	0	1	4	0.728	京王	0.564	小田急	0.164	京王	2	6	6
8	3	0	0	1	8	0.634	ミロード	0.522	小田急	0.112	店	1	6	7
17	1	0	0	1	4	0.811	映画館	0.599	映画館	0.212	新宿バルト9	2	6	6
17	2	0	0	1	6	0.775	映画館	0.599	映画館	0.176	新宿バルト9	2	6	9
17	3	0	0	1	12	0.388	新宿バルト9	0.212	映画館	0.176	映画館	2	6	10
17	4	61	0	1	6	0.446	バルト	0.169	映画館	0.095	映画館	2	6	10
3	2	0	0	1	10	0.264	カプセルホテル	1.006	カプセルホテル	0.13	郵便局	1	9	9
3	1	0	0	1	14	0.297	カプセルホテル	1.006	新宿区役所	0.291	ビジネスホテル	2	9	10
3	3	0	0	1	14	0.912	ビジネスホテル	0.621	ホテル	0.291	カプセルホテル	1	9	10
48	1	-1	0	1	20	0.344	新宿ワシントンホテル	0.241	ホテル	0.103	京王プラザホテル	2	10	10

注:以下は登録商標
小田急、京王、ミロード、伊勢丹、三越、新宿ワシントンホテル、ホットベッパ

【図50】

5002 深さで分類した表示用本クラスタhtml文書

(a)

- (01)レストラン イタリアン
- (02)デパート 伊勢丹 三越
- (03)ホットベッパ 美容院
- (04)小田急 京王 ミロード
- (05)映画 映画館 新宿バルト9
- (06)新宿区役所 カプセルホテル
- (07)新宿ワシントンホテル

5003 深さで分類した分類の選択画面

(b)

携帯端末向け情報検索
ウェブ 地図
(例)渋谷 銀座 秋葉原

新宿 検索

W 新宿の情報
関連語の分類1~10件目

- W 1. レストラン イタリアン
- W 2. デパート 伊勢丹 三越
- W 3. ホットベッパ 美容院
- W 4. 小田急 京王 ミロード
- W 5. 映画 映画館 新宿バルト9
- W 6. 新宿区役所 カプセルホテル
- W 7. 新宿ワシントンホテル

[続きを見る]

注:以下は登録商標
小田急、京王、ミロード、伊勢丹、三越、新宿ワシントンホテル、ホットベッパ

【図48】

4802 表示用本クラスタ

cid	tid	alt	stop	valid	len	total	term	score.1	term.1	score.2	term.2	ont	new_cid	new_tid
3	1	0	0	1	14	1.297	カプセルホテル	1.006	新宿区役所	0.291	ビジネスホテル	2	9	10
3	2	0	0	1	10	1.136	カプセルホテル	1.006	カプセルホテル	0.13	郵便局	1	9	9
3	3	0	0	1	14	0.912	ビジネスホテル	0.621	ホテル	0.291	カプセルホテル	1	9	10
8	1	0	0	1	6	0.086	小田急	0.564	京王	0.522	ミロード	2	6	6
8	2	0	0	1	4	0.728	京王	0.564	小田急	0.164	新宿駅	1	6	6
8	3	0	0	1	8	0.634	ミロード	0.522	小田急	0.112	店	1	6	7
12	1	0	0	1	10	0.914	レストラン	0.477	グルメ	0.437	イタリアン	2	5	5
12	3	0	0	1	10	0.692	レストラン	0.437	レストラン	0.255	和食	1	5	6
17	1	0	0	1	4	0.811	映画館	0.599	映画館	0.212	新宿バルト9	2	6	6
17	2	0	0	1	6	0.775	映画館	0.599	映画館	0.176	新宿バルト9	2	6	9
17	3	0	0	1	12	0.388	新宿バルト9	0.212	映画館	0.176	映画館	2	6	10
17	4	61	0	1	6	0.264	バルト	0.169	映画館	0.095	映画館	2	6	10
33	1	0	0	1	6	0.5	伊勢丹	0.279	三越	0.221	デパート	2	5	6
33	2	0	0	1	4	0.446	三越	0.279	伊勢丹	0.167	デパート	2	5	6
33	3	0	0	1	8	0.388	デパート	0.221	伊勢丹	0.167	三越	2	5	5
48	1	-1	0	1	20	0.344	新宿ワシントンホテル	0.241	ホテル	0.103	京王プラザホテル	2	10	10
62	1	0	0	1	14	0.26	ホテル	0.153	グルメ	0.107	美容院	2	5	6
62	3	0	0	1	6	0.213	美容院	0.107	ホットベッパ	0.106	ネイルサロン	1	5	5

注:以下は登録商標
小田急、京王、ミロード、伊勢丹、三越、新宿ワシントンホテル、ホットベッパ

【図 5 1】

5102 アプリケーション起動時表示画面

携帯端末向け情報検索
ウェブ 地図
(例) 渋谷 銀座 秋葉原

新宿	検索
----	----

【 図 1 1 】

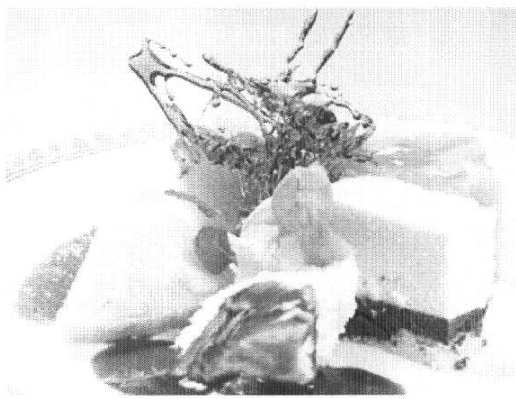
1102 キャッシュファイル

}

新宿 エリア / イタリアン・フレンチ

エリア 新宿 選択すべて | 東口・歌舞伎町方面 | 南口・代々木方面 | 西口・都庁方面 | 新大久保
 お店ジャンル イタリアン・フレンチ 選択すべて | イタリアン | フレンチ | パスタ・ピザ

18件 ありました



オーガニック&チーズ フレンチ個室レストラン ○×○×

TVでおなじみ◎◎シェフ【フレンチ会席】が人気!
4つの貸切できる部屋あり

6名～120名様まで貸切OKの個室パーティーが大好評!! 料理は気取らずに本格的なカジュアルフレンチが楽しめます☆月～木ならなんと10名～でも貸切OK!! 同じビルにチャペル、写真スタジオ、ウェディングドレスなどが完備されているため、ウェディングパーティーだけで年間400組が利用する超人気店でもあります。満足度の高さはリーズナブルに利用できることと、ウェディング利用者からの紹介(!!)が多いことでわかります。もちろん会社単位のご宴会や友達同士の飲み会・合コンにもお使いいただけます。マイク・カラオケ・プロジェクター完備で幹事さんには嬉しいお店。特に週末はすぐ予約でうまってしまおうか。迷う前にまず気軽にご相談ください♪また、当店では各種チーズの取り扱いもあり★こちらも詳しくはお問合せ下さい!

アクセス: JR新宿駅東口徒歩5分 / 地下鉄新宿三丁目駅徒歩4分

フロントページの続き

- (56)参考文献 特開2001-264101(JP,A)
特開2005-122396(JP,A)
特開平11-203311(JP,A)
安川美智子 外1名, Web検索結果のクラスタリングと特徴語の抽出, FIT2005 第4
回情報科学技術フォーラム, 社団法人情報処理学会 社団法人電子情報通信学会, 2005年
8月22日, 第61~62頁

- (58)調査した分野(Int.Cl., DB名)
G06F 17/30