

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2005-56361

(P2005-56361A)

(43) 公開日 平成17年3月3日(2005.3.3)

(51) Int.Cl.⁷

G06F 17/30

F I

G06F 17/30

340A

G06F 17/30

240A

テーマコード (参考)

5B075

審査請求 未請求 請求項の数 5 O L (全 50 頁)

(21) 出願番号 特願2003-289468 (P2003-289468)

(22) 出願日 平成15年8月7日(2003.8.7)

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(74) 代理人 100082131

弁理士 稲本 義雄

(72) 発明者 角田 智弘

東京都品川区北品川6丁目7番35号 ソ

ニー株式会社内

(72) 発明者 星野 政明

東京都品川区北品川6丁目7番35号 ソ

ニー株式会社内

Fターム(参考) 5B075 KK07 ND02 NR02 PR08 UU34

(54) 【発明の名称】 情報処理装置および方法、プログラム、並びに記録媒体

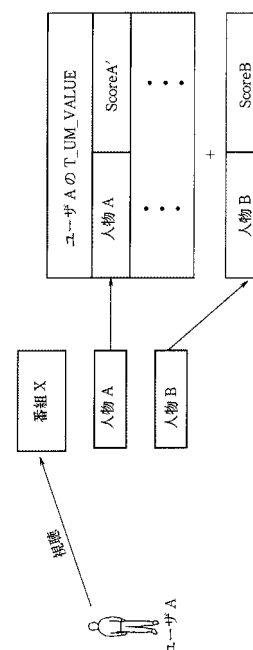
(57) 【要約】

【課題】 ユーザの嗜好に合った情報を作成し、ユーザの嗜好に合った情報を提供できるようにする。

【解決手段】 ユーザAが番組Xを視聴した。番組Xには、人物Aと人物Bが出演していた。ユーザAの嗜好データを管理するT_UM_VALUEデータベースには、人物Aに対するユーザAの数値化された値が書き込まれている。この場合、ユーザAが番組Xを視聴することにより、人物Aに対する値が更新され、ScoreA'に更新される。ユーザAのT_UM_VALUEデータベースには、人物Bに対するユーザAの数値化された値は書き込まれていない。この場合、人物Bという要素と人物Bに割り当てられているScoreBが、新たに、T_UM_VALUEデータベースに書き込まれる。本発明は、EPGを提供する側のサーバに適用することができる。

【選択図】 図36

図 36



【特許請求の範囲】

【請求項 1】

ユーザの嗜好に関する第 1 のデータを管理する第 1 の管理手段と、
情報に関する第 2 のデータを管理する第 2 の管理手段と、
前記ユーザが利用した前記情報に関する前記第 2 のデータを抽出する抽出手段と、
前記抽出手段により抽出された前記第 2 のデータを用いて前記第 1 のデータを更新する更新手段と
を備え、
前記更新手段は、前記第 2 のデータが、前記第 1 のデータ内に存在している場合、その存在している第 1 のデータを第 2 のデータを用いて更新し、前記第 2 のデータが、前記第 1 のデータ内に存在していない場合、前記第 2 のデータを前記第 1 のデータに追加することを特徴とする情報処理装置。 10

【請求項 2】

前記第 1 の管理手段は、前記第 1 のデータとして、第 1 の要素とその第 1 の要素に関して前記ユーザが有する数値化された第 1 の値を関連付けて管理し、
前記第 2 の管理手段は、前記第 2 のデータとして、前記情報に関する第 2 の要素とその第 2 の要素に関して前記情報が有する数値化された第 2 の値を関連付けて管理し、
前記更新手段は、前記第 2 のデータ内の前記第 2 の要素と一致する前記第 1 の要素が、前記第 1 のデータ内に存在している場合、前記第 1 の値と前記第 2 の値を用いて新たな第 1 の値を算出し、その新たな第 1 の値を前記第 1 のデータに書き込み、前記第 2 のデータ 20
内の前記第 2 の要素と一致する前記第 1 の要素が前記第 1 のデータ内に存在していない場合、前記第 2 の要素と前記第 2 の値を、前記第 1 のデータに追加することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】

データを管理するための記憶手段と、記憶手段に記憶されている前記データを用いて所定の処理を実行する制御手段とを少なくとも備えた情報処理装置の情報処理方法であって、
ユーザの嗜好に関する第 1 のデータを管理する第 1 の管理ステップと、
情報に関する第 2 のデータを管理する第 2 の管理ステップと、
前記ユーザが利用した前記情報に関する前記第 2 のデータを抽出する抽出ステップと、 30
前記抽出ステップの処理で抽出された前記第 2 のデータを用いて前記第 1 のデータを更新する更新ステップと
を含み、
前記更新ステップの処理は、前記第 2 のデータが、前記第 1 のデータ内に存在している場合、その存在している第 1 のデータを第 2 のデータを用いて更新し、前記第 2 のデータが、前記第 1 のデータ内に存在していない場合、前記第 2 のデータを前記第 1 のデータに追加する
ことを特徴とする情報処理方法。

【請求項 4】

データを管理するための記憶手段と、記憶手段に記憶されている前記データを用いて所 40
定の処理を実行する制御手段とを少なくとも備えた情報処理装置のプログラムであって、
ユーザの嗜好に関する第 1 のデータを管理する第 1 の管理ステップと、
情報に関する第 2 のデータを管理する第 2 の管理ステップと、
前記ユーザが利用した前記情報に関する前記第 2 のデータを抽出する抽出ステップと、
前記抽出ステップの処理で抽出された前記第 2 のデータを用いて前記第 1 のデータを更新する更新ステップと
を含み、
前記更新ステップの処理は、前記第 2 のデータが、前記第 1 のデータ内に存在している場合、その存在している第 1 のデータを第 2 のデータを用いて更新し、前記第 2 のデータ 50
が、前記第 1 のデータ内に存在していない場合、前記第 2 のデータを前記第 1 のデータに

追加する

ことを特徴とするプログラム。

【請求項 5】

請求項 4 に記載のプログラムを記録している

ことを特徴とする記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、情報処理装置および方法、プログラム、並びに記録媒体に関し、特に、ユーザの嗜好に基づく情報を提供する際に用いて好適な情報処理装置および方法、プログラム、並びに記録媒体に関する。 10

【背景技術】

【0002】

近年、インターネットなどのネットワークに接続できる機器が増加し、従来、ネットワークを介しては提供されていなかったようなサービスが提供され、それらのサービスを受けることが一般的になりつつある。例えば、テレビジョン受像機などは、テレビジョン番組のデータを受信するという従来からある機能に加え、ネットワークに接続し、所定のサーバから EPG (Electronic Program Guide) と称される電子的な番組表を取得する機能を有しているものもある。また、取得された EPG 上から録画予約などもできる機能も実 20
現化されている。

【0003】

EPG は、テレビジョン受像機以外の、例えば、パーソナルコンピュータなどでも取得でき、閲覧できるようになっている。

【0004】

図 1 に EPG の一例を示す。EPG は、上述したように、テレビジョン受像機やパーソナルコンピュータにより取得されるが、その取得された EPG は、表示部 11 に表示される。EPG は、放送局、放送時間、放送される番組が対応付けられたマトリクス状の表で表示されることが多い。例えば、図 1 においては、“放送局 A” の “12 時” 台に放送されている番組は “番組 A - 1” であるといった情報が表示されている。このように、EPG には、番組を放送する放送局、放送される時間、および放送される番組が関連付けられ、その関連付けをユーザが認識しやすい表示が行われる。(例えば、非特許文献 1 参照) 30

【0005】

このように、EPG には、基本的に、放送局、放送時間、放送される番組が関連付けられた状態で表示されるが、さらに、その EPG を提供する側が、ユーザの使い勝手等を考慮して、他の情報も合わせてユーザ側に提示することがある。例えば、図 1 に示した例では、番組名が表示されている領域内に、“予約” という予約ボタン 21 が表示されている。この予約ボタン 21 は、ユーザが、その番組を録画予約したいときに操作されるボタンとして設定されている。EPG を表示している端末が、パーソナルコンピュータのような記憶部 (不図示) を有する端末である場合、この予約ボタン 21 が操作されると、その内蔵する記憶部に、対応する番組が記憶 (録画) されるような設定が行なわれる。 40

【0006】

また、図 1 に示した例では、オススメボタン 22 とランキングボタン 23 も表示されている。オススメボタン 22 は、EPG を提供する側がオススメする番組の情報を閲覧したときに操作されるボタンである。ランキングボタン 23 は、録画予約された数や視聴率などに基づく情報を閲覧したいときに操作されるボタンである。

【0007】

これらのボタンは、ユーザが操作部 (不図示) を操作し、カーソル 31 が所望のボタン上に位置した状態で、クリックなどの所定の操作が行われることにより操作されるように構成されている。

【非特許文献 1】ソニー株式会社、テレビ王国、[online]、[平成 15 年 7 月 30 日検 50

索]、インターネット<URL: <http://www.so-net.ne.jp/tv/index.html>>

【発明の開示】

【発明が解決しようとする課題】

【0008】

図1に示したEPGにおいて、例えば、オススメボタン22が操作されると、図2に示したような画面に、表示部11の表示が切り替えられる。図2に示した画面には、複数の番組名と、その番組名に対応する番組内容が表示されている。このオススメボタン22が操作されたときに表示される画面は、EPGを提供する側が、予め作成したものである。すなわち、オススメの基準は、EPGを提供する側にあり、EPGの提供を受けるユーザ側にはない。

10

【0009】

従って、EPGの1つのサービスとして提供されているオススメ番組の情報などは、複数のユーザを対象としたものであり、ユーザ個人を対象としたものではなかった。すなわち、個々のユーザの嗜好を考慮したオススメ番組の情報を提供することは困難であるという課題があった。このことは、オススメ番組の情報以外の情報を提供する際にも、同様の課題があった。

【0010】

本発明はこのような状況に鑑みてなされたものであり、個々のユーザの嗜好などを考慮し、より適切な情報を、ユーザに提供できるようにすることを目的とする。

【課題を解決するための手段】

20

【0011】

本発明の情報処理装置は、ユーザの嗜好に関する第1のデータを管理する第1の管理手段と、情報に関する第2のデータを管理する第2の管理手段と、ユーザが利用した情報に関する第2のデータを抽出する抽出手段と、抽出手段により抽出された第2のデータを用いて第1のデータを更新する更新手段とを備え、更新手段は、第2のデータが、第1のデータ内に存在している場合、その存在している第1のデータを第2のデータを用いて更新し、第2のデータが、第1のデータ内に存在していない場合、第2のデータを第1のデータに追加することを特徴とする。

【0012】

前記第1の管理手段は、第1のデータとして、第1の要素とその第1の要素に関してユーザが有する数値化された第1の値を関連付けて管理し、第2の管理手段は、第2のデータとして、情報に関する第2の要素とその第2の要素に関して情報が有する数値化された第2の値を関連付けて管理し、更新手段は、第2のデータ内の第2の要素と一致する第1の要素が、第1のデータ内に存在している場合、第1の値と第2の値を用いて新たな第1の値を算出し、その新たな第1の値を第1のデータに書き込み、第2のデータ内の第2の要素と一致する第1の要素が第1のデータ内に存在していない場合、第2の要素と第2の値を、第1のデータに追加するようにすることができる。

30

【0013】

本発明の情報処理方法は、データを管理するための記憶手段と、記憶手段に記憶されているデータを用いて所定の処理を実行する制御手段とを少なくとも備えた情報処理装置の情報処理方法であって、ユーザの嗜好に関する第1のデータを管理する第1の管理ステップと、情報に関する第2のデータを管理する第2の管理ステップと、ユーザが利用した情報に関する第2のデータを抽出する抽出ステップと、抽出ステップの処理で抽出された第2のデータを用いて第1のデータを更新する更新ステップとを含み、更新ステップの処理は、第2のデータが、第1のデータ内に存在している場合、その存在している第1のデータを第2のデータを用いて更新し、第2のデータが、第1のデータ内に存在していない場合、第2のデータを第1のデータに追加することを特徴とする。

40

【0014】

本発明のプログラムは、データを管理するための記憶手段と、記憶手段に記憶されているデータを用いて所定の処理を実行する制御手段とを少なくとも備えた情報処理装置のプ

50

プログラムであって、ユーザの嗜好に関する第1のデータを管理する第1の管理ステップと、情報に関する第2のデータを管理する第2の管理ステップと、ユーザが利用した情報に関する第2のデータを抽出する抽出ステップと、抽出ステップの処理で抽出された第2のデータを用いて第1のデータを更新する更新ステップとを含み、更新ステップの処理は、第2のデータが、第1のデータ内に存在している場合、その存在している第1のデータを第2のデータを用いて更新し、第2のデータが、第1のデータ内に存在していない場合、第2のデータを第1のデータに追加することを特徴とする。

【0015】

本発明の記録媒体は、前記プログラムを記録していることを特徴とする。

【0016】

10

本発明の情報処理装置および方法、並びにプログラムにおいては、ユーザの嗜好に関するデータと情報に関するデータが管理され、ユーザが所定の情報を利用した場合、そのユーザが利用した情報に関連付けられている全ての要素が読み出され、ユーザの嗜好に関するデータが、読み出された要素毎に変更、追加される。よって、ユーザの嗜好に関するデータが、情報を利用する毎に更新され、よりユーザの嗜好に合ったものとされる。

【発明の効果】

【0017】

本発明によれば、情報、特に、テレビジョン放送の番組に関する情報を提供することが可能となる。

【0018】

20

本発明によれば、ユーザの嗜好を考慮した、例えば、テレビジョン放送の番組に関する情報を提供することができる。また、個々のユーザ毎にカスタマイズされた情報を提供することができるため、その情報を提供する側へのアクセスを増加させることが可能となる。

【0019】

本発明によれば、ユーザに対して情報を提供する際、その情報を提供する根拠（キーワードなど）を合わせて提供することが可能となる。よって、ユーザ側は、提供された情報が、どのようなキーワードに基づき選択され、提供されたのかを認識することが可能となる。

【0020】

30

さらに本発明によれば、ユーザ側に提供するキーワードなどは、提供する側で制御することができるため、例えば、ユーザに提供するにはふさわしくないキーワードだが、ユーザに提供する情報を検索する際にはふさわしいキーワードであるといったようなキーワードを管理することが可能となり、より適切な情報提供が可能となる。

【発明を実施するための最良の形態】

【0021】

以下に本発明の実施の形態を説明するが、請求項に記載の発明特定事項と、発明の実施の形態における具体例との対応関係を例示すると、次のようになる。この記載は、請求項に記載されている発明をサポートする具体例が、発明の実施の形態に記載されていることを確認するためのものである。従って、発明の実施の形態中に記載されているが、発明特定事項に対応するものとして、ここには記載されていない具体例があったとしても、そのことは、その具体例が、その発明特定事項に対応するものではないことを意味するものではない。逆に、具体例が発明特定事項に対応するものとしてここに記載されていたとしても、そのことは、その具体例が、その発明特定事項以外の発明特定事項には対応しないものであることを意味するものではない。

40

【0022】

さらに、この記載は、発明の実施の形態に記載されている具体例に対応する発明が、請求項に全て記載されていることを意味するものではない。換言すれば、この記載は、発明の実施の形態に記載されている具体例に対応する発明であって、この出願の請求項には記載されていない発明の存在、すなわち、将来、分割出願、補正、国内優先などにより追加

50

される発明の存在を否定するものではない。

【0023】

本発明の情報処理装置（例えば、図9のサーバ52）は、ユーザの嗜好に関する第1のデータ（例えば、図30のT_UM_VALUEデータベース317）を管理する第1の管理手段（例えば、図11のターゲットデータベース252）と、情報に関する第2のデータ（例えば、図29のT_PRG_VALUEデータベース316）を管理する第2の管理手段（例えば、図11のターゲットデータベース252）と、ユーザが利用した情報に関する第2のデータを抽出する抽出手段（例えば、図33のステップS51乃至52の処理を実行する図9のCPU201）と、抽出手段により抽出された第2のデータを用いて第1のデータを更新する更新手段（例えば、図33のステップS53乃至54の処理を実行する図9のCPU201）とを備え、更新手段は、第2のデータが、第1のデータ内に存在している場合、その存在している第1のデータを第2のデータを用いて更新（例えば、図33のフローチャートの処理）し、第2のデータが、第1のデータ内に存在していない場合、第2のデータを第1のデータに追加する（例えば、図34のフローチャートの処理）ことを特徴とする。

【0024】

以下に、本発明の実施の形態について図面を参照して説明する。

【0025】

以下の説明においては、番組情報を提供するEPG（Electronic Program Guide）を扱うシステムを例に挙げて説明するが、他の情報を扱うシステムにおいても、本発明を適用することは可能である。

【0026】

本発明の実施の形態について説明するが、その説明の順番について記載しておく。まず本発明を適用するシステムの構成について説明を加える。次に、そのようなシステムにおいて授受されるEPGデータについて説明を加えるが、先に、EPGデータを取得したユーザ側のユーザ端末53-1乃至53-N（図3）に表示されるEPGの特徴を説明した後、そのような特徴を実現するためのデータ構成などについて、EPGデータを提供する側のサーバ52（図3）の構成を含めて説明を加える。

【0027】

図3は、本発明を適用したシステムの一実施の形態の構成を示す図である。

【0028】

ネットワーク51は、インターネットやLAN（Local Area Network）で構成され、接続されている機器同士でデータの授受を行う際に用いられる。ネットワーク51には、サーバ52が接続されている。サーバ52は、EPGを作成、提供する側の管理者が管理している端末である。サーバ52からは、ユーザ端末53-1乃至53-Nからの要求に対応して、適宜、EPGのデータが、ネットワーク1を介して提供される。

【0029】

ユーザ端末53-1乃至53-Nは、それぞれ、放送局54から放送されるテレビジョン放送を受信する機能も有する。ユーザ端末53-1乃至53-Nは、それぞれ、ネットワーク51を介してサーバ52と接続する機能を有し、放送局54からの番組を受信する機能を有するテレビジョン受像機やパーソナルコンピュータである。または、ネットワーク51を介してサーバ52と接続する機能を有するパーソナルコンピュータと、放送局54からの番組を受信する機能を有するテレビジョン受像機という2つの端末から構成される。さらにユーザ端末53-1乃至53-Nは、それぞれ、番組を録画するための録画装置が含まれる構成とされていても良い。

【0030】

すなわち、ユーザ端末53-1乃至53-Nは、それぞれ、サーバ52とデータの授受を行う機能を有し、放送局54からのテレビジョン放送用の番組を受信する機能を少なくとも有している、1の装置から構成される端末、または、複数の装置から構成され、それらの装置間でデータの授受を行える構成とされている端末である。

10

20

30

40

50

【 0 0 3 1 】

以下の説明において、ユーザ端末 5 3 - 1 乃至 5 3 - N を個々に区別する必要がない場合、単に、ユーザ端末 5 3 と記述する。また、他の装置においても同様に記述する。さらに、図 3 においては、サーバ 5 2 や放送局 5 4 を、それぞれ 1 台しか図示していないが、説明の都合上、そのように図示しただけであり、実際の構成を限定するものではなく、複数台存在していても、本発明を適用できる事にかわりはない。

【 0 0 3 2 】

図 4 は、ユーザ端末 5 3 の内部構成例を示す図である。ここでは、ユーザ端末 5 3 は、サーバ 5 2 とデータの授受を行い、放送局 5 4 からの番組を受信することができる 1 台の端末であるとして説明する。また、そのような端末は、パーソナルコンピュータにより構成

10

できるため、ここでは、ユーザ端末 5 3 は、パーソナルコンピュータであるとして説明する。なお、パーソナルコンピュータとは、デスクトップ型やノート型のものを含むことはもちろんであるが、PDA (Personal Digital Assistance) や、携帯電話なども含む。

【 0 0 3 3 】

ユーザ端末 5 3 の CPU (Central Processing Unit) 7 1 は、ROM (Read Only Memory) 7 2 に記憶されているプログラムに従って各種の処理を実行する。RAM (Random Access Memory) 7 3 には、CPU 7 1 が各種の処理を実行する上において必要なデータやプログラムなどが適宜記憶される。入出力インタフェース 7 5 は、キーボードやマウスから構成される入力部 7 6 が接続され、入力部 7 6 に入力された信号を CPU 7 1 に出力

20

する。また、入出力インタフェース 7 5 には、ディスプレイやスピーカなどから構成される出力部 7 7 も接続されている。

【 0 0 3 4 】

さらに、入出力インタフェース 7 5 には、ハードディスクなどから構成される記憶部 7 8、および、ネットワーク 5 1 を介して他の装置 (例えば、サーバ 5 2) とデータの授受を行う通信部 7 9 も接続されている。なお、詳細は図示しないが、通信部 7 9 には、放送局 5 4 からの番組を受信し、処理するためのアンテナ、チューナなども含まれる。ドライブ 8 0 は、磁気ディスク 9 1、光ディスク 9 2、光磁気ディスク 9 3、半導体メモリ 9 4 などの記録媒体からデータを読み出したり、データを書き込んだりするときに用いられる。

30

【 0 0 3 5 】

図 5 のフローチャートを参照して、ユーザ端末 5 3 の基本的な動作について説明する。ここでは、サーバ 5 2 により管理されている EPG を取得し、処理する場合を例に挙げて説明する。以下の説明においては、EPG とは、EPG を表示させるためのデータ、および、EPG に付随するデータを含むとする。

【 0 0 3 6 】

ステップ S 1 1 において、ユーザ端末 5 3 は、ネットワーク 5 1 (図 3) を介してサーバ 5 2 にアクセスする。このアクセスの処理は、所定のブラウザが起動され、ダイアルアップ方式など所定の方式 (ユーザが ISP (Internet Service Provider) と契約した内容に従った方式) により行われる。

40

【 0 0 3 7 】

アクセスが完了すると、ステップ S 1 2 において、ログインの処理が行われる。この処理は、ユーザが入力部 7 6 を操作し、ユーザ名やパスワードを入力し、その入力された情報が、通信部 7 9 を介してサーバ 5 2 に対して送信されることにより行われる。

【 0 0 3 8 】

サーバ 5 2 は、ユーザ端末 5 3 からのアクセス (またはログイン) を受けると、管理している EPG の初期画面のデータを、ネットワーク 5 1 を介して、アクセスしてきたユーザ端末 5 3 に対して提供する。もちろん、EPG が提供されるのは、ログインの処理が正常に行われときであり、正常に行われなかったときには、所定のエラー処理が実行される。

50

【 0 0 3 9 】

サーバ 5 2 側から提供されたデータを、ユーザ端末 5 3 は、ステップ S 1 3 の処理として受信する。CPU 7 1 (図 4) は、通信部 7 9 により受信されたデータに基づき、出力部 7 7 としてのディスプレイ 1 0 1 (図 6) 上に、EPG の初期画面 (図 6 を参照して後述する) を表示させる。ユーザは、ディスプレイ 1 0 1 上に表示された初期画面を閲覧し、その画面内に設けられている複数のボタン (リンク先やコマンドが関連付けられているボタン) を操作する。

【 0 0 4 0 】

そのようなボタンの操作が行われると、ステップ S 1 4 において、その操作されたボタンに対応するデータ (指示) が、サーバ 5 2 に送信される。そして、その指示を受信したサーバ 5 2 からの応答としてのデータが、ステップ S 1 5 において受信される。そして、ステップ S 1 3 の処理と同様に、CPU 7 1 が、受信されたデータを処理することにより、ディスプレイ 1 0 1 上に、所定の画面が表示される。ステップ S 1 4 , S 1 5 の処理は、必要に応じ繰り返し行われる。

【 0 0 4 1 】

図 6 は、ステップ S 1 3 においてディスプレイ 1 0 1 上に表示される初期画面の一例を示す図である。図 6 に示したディスプレイ 1 0 1 上には、EPG が表示されているが、その EPG の構成は、基本的に、放送局、放送時間、放送される番組が関連付けられた状態で表示されている。例えば、図 6 に示した EPG においては、“放送局 A”で“12 時”台に放送されるのは、“番組 A - 1”であることが示されている。このように、複数の放送局、それらの放送局で放送される番組、および、それらの番組が放送される時間が、それぞれ関連付けられ、ユーザが、一見して認識できるような表示形態で、表示される。

【 0 0 4 2 】

また、図 6 に示したように、例えば、“番組 A - 1”という文字が表示されている領域内には、予約ボタン 1 1 1 が表示されている。この予約ボタン 1 1 1 が操作されると、対応する番組が、記憶部 7 8 (図 4) に記憶 (録画) されるような設定が行われる。

【 0 0 4 3 】

さらに、図 6 に示したように、例えば、“番組 A - 1”という文字が表示されている領域内には、オススメマーク 1 1 2 が表示されている。このオススメマーク 1 1 2 は、アクセスしてきたユーザに対してオススメする番組に付けられるマークである。ここで、“アクセスしてきたユーザに対してオススメする番組”とは、アクセスしてきたユーザ毎に、そのユーザに適した番組が提供され、ユーザ毎に異なる番組がオススメされる可能性があることを意味する。

【 0 0 4 4 】

換言すれば、オススメされる番組は、サーバ 5 2 にアクセスしてきたユーザ全員に共通したものではなく、よって、サーバ 5 2 側で、予めオススメ番組として設定されている番組ではなく、アクセスしてきたユーザの嗜好に合わせて、ユーザがアクセスしてきたときに、ユーザ毎に作成される情報である。このように、アクセスしてきたユーザの嗜好に合った番組がオススメとして、ユーザ側に提示されるのは、本発明の 1 つの特徴であり、このような特徴を実現するために、サーバ 5 2 は、後述するようなデータベースを有し、そのデータベースを用いた処理を実行する。

【 0 0 4 5 】

このオススメマーク 1 1 2 は、図 6 に示した例では、星マークで示されているが、そのマークの形状などは、どのようなものでも良い。また、星マークなどで示される場合、そのマークの数により、オススメ度が異なるようにしても良い。例えば、図 6 においては、“番組 A - 1”のオススメマーク 1 1 2 は、星マークが 1 つであるが、“番組 B - 1”のオススメマーク 1 1 2 は、星マークが 3 つであり、“番組 B - 1”の方が、“番組 A - 1”よりオススメであることが示されている。

【 0 0 4 6 】

このオススメマーク 1 1 2 (星マークの数) に関するデータは、サーバ 5 2 側で、ユー

10

20

30

40

50

ザの嗜好に関するデータを基に設定され、E P Gの初期画面のデータに含まれて送信される。または、サーバ52側から送信されてきたE P Gの初期画面のデータと、ユーザ端末53側で管理しているユーザの嗜好に関するデータを基に設定されるようにしても良い。このような設定の仕方の差異は、ユーザの嗜好に関するデータ（以下、適宜、嗜好データと記述する）をサーバ52側で管理するか、ユーザ端末53側で管理するかにより生じるものである。ここでは、まず、サーバ52側で、嗜好データを管理するとして説明をする。

【0047】

このような、E P Gの初期画面で、ユーザの嗜好に適した番組に対してオススメマーク112が表示されるようにするのは、サーバ52にユーザ端末53がアクセスした時点で、アクセスしてきたユーザを特定するための認証処理（ログイン）などが行われることを前提としている。しかしながら、例えば、初期画面として図1に示したような画面がユーザに提供され、その後、ログイン処理が実行された時点で、図6に示したような画面がユーザに提供されるようにしても良い。

10

【0048】

オススメマーク112は、オススメの番組（サーバ52側の処理として、所定の条件を満たすとして検出された番組）とされた番組名の領域に必ず表示されるようにしても良い。必ず表示されるようにした場合、同一の放送時間帯に放送される複数の番組にオススメマーク112が、それぞれ表示されといった状況が発生する可能性がある。しかしながら、通常、ユーザは、1つの時間帯には、1つの番組しか視聴できず、また、ユーザが所有する1つの装置では、1の時間帯には1つの番組しか録画できないので、1の時間帯に2以上のオススメマーク112を表示させても、結果としては、その内の2つの番組しか視聴できないと考えられる。

20

【0049】

また、オススメマーク112が表示される番組が、多いと、オススメマーク112を表示させる意味が薄くなると考えられる。極端な例だが、全ての番組に対してオススメマーク112が表示されたような場合（このような状況が発生することもある）、オススメマーク112が、実質上、意味のないものになってしまう。

【0050】

これらのことを考慮すると、オススメマーク112が表示される番組数に、何らかの制限を設けるようにした方が良い。例えば、同一の時間帯には、オススメマーク112は2番組内にする、1日分のE P G内には、オススメマーク112は20番組内にするなどの制御が行われるようにしても良い。

30

【0051】

図6に示したE P Gの画面内には、オススメボタン113とランキングボタン114も表示されている。オススメボタン113は、オススメとされる番組を閲覧したときに操作されるボタンである。ランキングボタン114は、録画予約された数や視聴率などの基づく番組紹介を閲覧したいときに操作されるボタンである。

【0052】

これらのボタンは、ユーザが入力部77（図4）を操作し、カーソル121を所望のボタン上に位置した状態で、クリックなどの所定の操作が行われることにより操作されるように構成されている。

40

【0053】

オススメボタン113が操作されると、図7に示したような画面に、ディスプレイ101上の画面が切り替えられる。図7に示した画面例では1番組分の画面しか図示しないが、オススメとされる1つの番組の詳細情報を提示する画面（以下、適宜、オススメ画面と記述する）は、ディスプレイ101上に複数、同様の形式で表示される。オススメ画面には、番組名表示部151、放送時間表示部152、内容表示部153、出演者表示部154、理由表示部155、および、オススメ度表示部156が設けられている。

【0054】

50

番組名表示部 1 5 1 には、オススめとされる番組名が表示され、放送時間表示部 1 5 2 には、その番組が放送される日時が表示され、内容表示部 1 5 3 には、その番組の内容が簡潔な文章で示され、出演者表示部 1 5 4 には、その番組に出演する主な出演者の名前が表示される。そして、理由表示部 1 5 5 には、なぜその番組がオススめなのかを、ユーザに認識させるようなオススめ理由が表示され、オススめ度表示部 1 5 6 には、オススめ理由から総合的に算出されるオススめ度が、表示される。

【 0 0 5 5 】

図 7 に示した例では、オススめ度表示部 1 5 6 には点数が表示されているが、図 6 に示したオススめマーク 1 1 2 と同様に、星マークなどのマーク（および、そのマークの個数）で表示されるようにしても良い。

10

【 0 0 5 6 】

図 7 に示したオススめ画面のうち、理由表示部 1 5 5 を表示させるのは、本発明の 1 つの特徴である。このような特徴を有することにより、ユーザは、なぜその番組がオススめとして提供されているのか、自分のどのような嗜好に対してマッチングしているのかなどを認識することができる。そして、ユーザは、このような理由を閲覧することにより、視聴（録画）したい番組を選択しやすい（自分の嗜好に合った番組を見つけやすい）ようになる。

【 0 0 5 7 】

また本発明の 1 つの特徴としては、このようなオススめ画面は、既存のものが提供されるのではなく（サーバ 5 2 側が、アクセスしてきたユーザ全てに対して同様のものを提供するのではなく）、アクセスしてきたユーザの嗜好のデータを基に、そのユーザの嗜好に合った番組が選択され、提供される点である。このような特徴のために、ユーザ側としては、提供される情報に信頼性を持つことができ、所望の番組を検索しやすくなり、使い勝手がよい E P G が提供されていると感じることが可能となる。よって、サーバ 5 2 へのアクセスが増加すると考えられ、会員数などを増加させることが可能になると考えられる。

20

【 0 0 5 8 】

図 7 に示したオススめ画面についてさらに説明を加えるに、理由表示部 1 5 5 には、複数の情報が表示される。表示される内容は、例えば、図 7 においては、“よく見る出演者”、“登録ジャンル”、“よく見るキーワード”、“同じ番組を録画した人の興味”などの項目があるが、後述するようにさまざまな項目（情報）が考えられる。これらの理由表示部 1 5 5 に表示される情報に関しては、サーバ 5 2 側で管理されているデータベースの内容などに関わる。

30

【 0 0 5 9 】

図 7 に示した例では、例えば、“よく見る出演者”として「人物 A」が挙げられ、さらに、その「人物 A」に対する点数（図 7 では 1 7 . 0 点）が表示されている。この点数は、オススめとしてサーバ 5 2 側が提供する番組を検索する際、「出演者 A」に対してマッチングしたスコアであり、このスコアが高いほど、ユーザの嗜好に合っていると考えられる。

【 0 0 6 0 】

この点数について説明を加えるに、図 7 に示した例では、単に 1 7 . 0 点と表示されているだけであるが、さらに、次のような詳細な表示の形態も考えられる。例えば、“1 7 . 0 点（8 . 9 , 3 . 8）”といったような括弧書きでさらに詳細な点数が表示される。括弧書きの点数のうち、前半の“8 . 9”は、番組に対する重み付けの値（後述する T_PR G_VALUE の値）であり、後半の“3 . 8”は、ユーザに対する重み付けの値（後述する T_U M_VALUE の値）である。括弧外の点（この場合、1 7 . 0 点）は、この 2 つの重み付けの値の責に所定のパラメータを乗算した値である。

40

【 0 0 6 1 】

ところで、図 7 に示したように、例えば、“よく見る出演者”という同一の項目に対して、「人物 A」、「人物 C」、および「人物 D」という 3 つの情報が表示され、それぞれの情報に対する点数がそれぞれ表示されているような場合、以下のような問題点がある。

50

図 7 に示した例では、理由表示部 155 に表示される項目の内の一部だけを示した状態を示しているが、複数ある項目の全部が表示され、それぞれの項目に対して複数の情報が、複数行にわたって分散的に表示され、さらに、それらの情報毎に点数までも表示されるとなると理由表示部 155 の領域の大きさは大きいものになってしまう。また、情報を羅列してもユーザにとっては、見やすい画面であるとは言えない。さらに、点数を表示しても、その点数からユーザが得る情報量は少なく（点数から得られる情報を判断することはユーザにとっては必ずしも容易なことであるとは考えられないため）、所望の番組の検索には、あまり意味を持たない情報であると考えられる。

【0062】

そこで、オススメ画面を図 8 に示したようにしても良い。図 8 に示したオススメ画面の理由表示部 155 には、1つの項目に対して、複数の情報が行を改めることなく連続的に表示され、点数は表示されないようになっている。従って、同じ項目（例えば、“よく見る出演者”）が、図 7 に示した例のように、何度も表示されるようなことは、図 8 における表示形態ではない。このようにすれば、理由表示部 155 の領域が大きくなってしまいうようなことを防ぐことができる。また、情報がコンパクトにまとまっているので、ユーザは、それらの情報を見やすく、各情報から得られる情報は多いと考えられる。従って、ユーザが所望の番組をより検索しやすくなると考えられる。

【0063】

図 7 または図 8 に示したオススメ画面のうち、ユーザに提示される画面はどちらでも良い。また、表示形態などは、図 7 または図 8 に示した画面に限定されるものではないため、どのような形態、例えば、各表示部の順序が入れ替わった状態で表示される、図示していない表示部が設けられる形態などで情報が提示されるようにしても良い。しかしながら、本発明を適用することにより、オススメ画面、および、そのオススメ画面が表示される前に表示される画面（上述した実施の形態では、初期画面と称した画面）においては、以下のような特徴が少なくともある。

【0064】

オススメ画面のみ、または、初期画面とオススメ画面は、ユーザの嗜好を考慮した画面（情報）となっており、ユーザ毎に適した情報が提示される。その提示される情報は、見やすい形態で提示される。このような特徴を実現するための、サーバ 52 の構成、管理されるデータベース、そのデータベースが用いられた処理などについて、以下に説明を加える。

【0065】

以下の説明の概略を示しておく、まず、サーバ 52 の内部構成について説明を加え、その次に、データベースについての説明を加える。そして、そのデータベースにより管理されるデータの作成について説明を加え、作成されたデータを用いた処理、特に、オススメ画面に関わる処理について説明を加える。

【0066】

図 9 は、サーバ 52 の内部構成例を示す図である。サーバ 52 は、パーソナルコンピュータなどから構成することが可能である。サーバ 52 の CPU 201 は、ROM 202 に記憶されているプログラムに従って各種の処理を実行する。RAM 203 には、CPU 201 が各種の処理を実行する上で必要なデータやプログラムなどが適宜記憶される。入出力インタフェース 205 は、キーボードやマウスから構成される入力部 206 が接続され、入力部 206 に入力された信号を CPU 201 に出力する。また、入出力インタフェース 205 には、ディスプレイやスピーカなどから構成される出力部 207 も接続されている。

【0067】

さらに、入出力インタフェース 205 には、ハードディスクなどから構成される記憶部 208、および、ネットワーク 51 を介して他の装置、例えば、ユーザ端末 53 とデータの授受を行う通信部 209 も接続されている。ドライブ 210 は、磁気ディスク、光ディスク、光磁気ディスク、半導体メモリなどの記録媒体からデータを読み出したり、データ

を書き込んだりするとき用いられる。

【0068】

サーバ52は、上述したようにEPGやオススメ画面を作成し、ユーザ端末53側に提供する機能を有する端末であるが、その機能を実行するためのプログラムは、記憶部208に記憶され、RAM203に展開され、その展開されたプログラムに基づいてCPU201が処理を実行することにより行われる。図10は、サーバ52における機能を説明するための機能ブロック図である。

【0069】

ログ生成機能231は、ユーザが（ユーザ端末53が）アクセスしたときの日時など、また、そのユーザを受けたサービスの利用情報（所定の番組の録画予約）などに基づくログを生成し、記憶部208に記憶させ、その記憶されたログファイルを管理する機能である。ユーザ管理機能232は、ユーザの新規の登録、既に登録されているユーザの削除、ユーザがログインしてきたときの処理を行う機能であり、必要に応じ、ユーザからの操作（指示）を管理する機能である。

【0070】

情報選別機能233は、後述する対象情報メタデータベースとユーザモデルデータベースを参照して、アクセスしてきた（所定の要求を出してきた）ユーザに対して、そのユーザに適した情報を、対象となっている情報データベースから取捨選択し、ユーザ側に提供する機能である。サービス管理機能234は、サービス管理者（サーバ52の管理者や、サーバ52に対してEPGに関わるデータを供給する者）が、各機能や、各データベース、関連情報などの追加、変更、削除などの編集を行うための機能である。

【0071】

ユーザモデル管理機能235は、ユーザあるいはサービスの管理者から、直接的に入力されるか、または、ログデータベースのマイニング（Mining）結果に基づく処理により、ユーザモデルデータベースの登録、更新、削除などを行う機能である。詳細は後述するが、ユーザモデル管理機能235は、ユーザの嗜好に関するデータを取得し、管理するが、その取得には、ユーザ自身が登録したデータを取得する、学習により取得されたデータを取得する、フィルタリング処理により取得された情報を取得するといった方法がある。

【0072】

情報通信機能236は、選別された情報などをユーザ側に提供する機能であり、ユーザ側からのデータを受信する機能である。対象情報メタデータ管理機能237は、ユーザあるいはサービスの管理者からの直接的な入力や、情報データベースやその他のマスターなどの関連情報をマイニングした結果に基づく処理により、情報メタデータベースへの情報の登録、更新、削除などを行う機能である。サービス一般機能238は、サーバ52がサービスを提供するうえで必要となるが、上述した機能では行われない処理を実行するための機能である。

【0073】

このような機能を、サーバ52は備えている。このような機能は、ここでは、サーバ52が備えているとして説明を続けるが、これらの機能のうち、一部の機能は、ユーザ端末53側に持たせることも可能である。また、これらの機能の全て、または、一部をサーバ52以外の端末に持たせ、その端末とデータの授受を行うことにより、結果として、サーバ52が機能を実現できるように構成しても良い。

【0074】

例えば、ユーザ端末53側で、ログ生成機能231や、ユーザモデル管理機能235を有するようにし（ユーザのプライベートな情報や嗜好に関わるデータを管理する機能を、ユーザ側が管理できるようにし）、これらの機能が担当する後述するような処理を、サーバ52側でなく、ユーザ端末53側で行うようにしても良い。

【0075】

このような機能が用いる（管理する）データベースについて説明する。データベースは、記憶部208に構築される。図11は、記憶部208に構築されるデータベースを示す

10

20

30

40

50

図である。図 1 1 に示した例では、記憶部 2 0 8 には、データベースとして、マスターデータベース 2 5 1、ターゲットデータベース 2 5 2、ログデータベース 2 5 3、および、ユーザデータベース 2 5 4 が構築されている。

【 0 0 7 6 】

マスターデータベース 2 5 1 は、E P G を作成する際に用いられるデータベースや、ユーザに関する情報（例えば、年齢など）を管理するためのデータベースなどの複数のデータベースにより構成される。ターゲットデータベース 2 5 2 は、オススメ画面などを作成する際に、数値計算などが行われるが、その計算に用いられるデータベースなどの複数のデータベースにより構成される。ログデータベース 2 5 3 は、ログ生成機能 2 3 1 により生成され、管理されるデータベースである。ユーザデータベース 2 5 4 は、会員登録して

10

【 0 0 7 7 】

マスターデータベース 2 5 1 は、図 1 2 乃至図 2 1 の各図に示すデータベースから構成されている。なお、図 1 2 乃至図 2 1 に示すデータベース内のデータは、説明の都合上、一部分のデータのみを図示してある。

【 0 0 7 8 】

図 1 2 に示したデータベースは、M_CATEGORY データベース 2 7 1 であり、放送される番組のカテゴリー（ジャンル）を管理するために設けられている。この M_CATEGORY データベース 2 7 1 には、AttributeID として、“ 0 0 1 ” という識別子が割り当てられている。M_CATEGORY データベース 2 7 1 は、登録されているカテゴリーを識別するための識別子（C

20

【 0 0 7 9 】

図 1 3 に示したデータベースは、M_PERSON データベース 2 7 2 であり、放送される番組に出演する人物（司会者、俳優、女優、文化人、政治家など）を管理するために設けられている。この M_PERSON データベース 2 7 2 には、AttributeID として、“ 0 0 2 ” という識別子が割り当てられている。M_PERSON データベース 2 7 2 は、登録されている人物名を識別するための識別子（PersonID）、人物名（Name）、および、その人物名の読み方（KanaName）を、それぞれ関連付けて管理する。例えば、図 1 3 に示した M_PERSON データベース 2 7 2 において、PersonID として “ 0000000001 ” は、“ 人物 A ” という人物に割り当てられており、その “ 人物 A ” の読み方は、“ ジンブツエー ” であることが関連付けられている。

30

【 0 0 8 0 】

図 1 4 に示したデータベースは、M_PERSON_CHARACTER データベース 2 7 3 であり、放送される番組に出演する人物のキャラクター（特徴）を管理するために設けられている。M_PERSON_CHARACTER データベース 2 7 3 には、AttributeID として、“ 0 0 3 ” という識別子が割り当てられている。M_PERSON_CHARACTER データベース 2 7 3 は、番組に出演する人物のキャラクターを識別するための識別子（CharacterID）とキャラクター（Character）が、関連付けて管理する。なお、キャラクターとは、一般的に使われている意味あい

40

【 0 0 8 1 】

例えば、図 1 4 に示した M_PERSON_CHARACTER データベース 2 7 3 において、CharacterID “ 0 0 1 ” は、“ 司会者系 ” というキャラクターが関連付けられている。なお、図 1 4 に示した例では、1 つの CharacterID には、1 つの言葉が関連付けられている例を示したが、1 つの CharacterID に、複数の言葉（類似する言葉）が関連付けられているようにしても良い。例えば、CharacterID として “ 0 0 4 ” が割り当てられているのは、“ 身近 ” であるが、“ 身近、親しみ ” などと、類似する複数の言葉と関連付けられるようにしても良

50

い。

【0082】

図15に示したデータベースは、M_VIEW_TYPEデータベース274であり、ユーザの視聴タイプ（どのような番組を好むか、どのような形態で視聴するかなど）を表す言葉を管理するために設けられている。このM_VIEW_TYPEデータベース274には、AttributeIDとして、“004”が割り当てられている。M_VIEW_TYPEデータベース274は、視聴タイプを識別するための識別子（ViewID）と視聴タイプ（ViewType）を関連付けて管理する。例えば、図15に示したM_VIEW_TYPEデータベース274において、ViewID“001”は、“笑い刺激型”という視聴タイプに割り当てられている。

【0083】

図16に示したデータベースは、M_KEYWORDデータベース275であり、番組の特徴を表すキーワード（単語）を管理するために設けられている。このM_KEYWORDデータベース275には、AttributeIDとして、“005”が割り当てられている。M_KEYWORDデータベース273は、キーワードを識別するための識別子（KeywordID）とキーワード（Keyword）を関連付けて管理する。なお、キーワードとは、一般的に使われている意味あいをも本発明においても適用するが、例えば、“番組A”の内容を1または複数の単語で表すと、どのような単語が一番適しているか（言い表せているか）を示す言葉であるとする。

【0084】

例えば、図16に示したM_KEYWORDデータベース273において、KeywordID“0000000001”は、“情報”というキーワードに割り当てられている。

【0085】

図17に示したデータベースは、M_KEYWORD_1データベース276であり、上述したM_KEYWORDデータベース275と基本的に同様のデータを管理するために設けられている。このM_KEYWORD_1データベース276は、図16に示したM_KEYWORDデータベース275と比較し、リーゾン（Reason）という項目も、KeywordIDやKeywordと、それぞれ関連付けて管理する。このReasonという項目に記載されている言葉は、例えば、図8において、理由表示部155に表示される言葉であり、ユーザ側に提示される言葉（単語）である。

【0086】

すなわち、Keywordという項目に書かれているキーワードは、マッチングなどの処理を行うとき等に用いられる言葉であり、Reasonという項目に書かれている言葉は、マッチングが行われた結果、そのマッチングのときの得点が高く、ユーザ側にオススメ理由として提示する際に、実際に提示される言葉が書かれている。

【0087】

例えば、KeywordIDが“0000000006”の“エロティック”というキーワードは、マッチングのときに適切な番組を検索するのには適しているとしても、実際に、ユーザに番組をオススメとして提示する際には、あまり好ましくない言葉であると考えられる。そこで、“エロティック”という言葉を表示されないようにするために、Reasonの部分は空欄（Null）とされている。

【0088】

また、例えば、KeywordIDが“0000000008”の“得”というキーワードは、マッチングのときに適切な番組を検索するのには適しているとしても、実際に、ユーザに番組をオススメとして提示する際には、“得”とだけ表示しても具体的な意味がつかみづらく、不親切な表示となってしまうと考えられ、ユーザに提示するには適切ではない。そこで、Reasonの部分には“お得な情報”と記載しておき、“得”というキーワードがユーザに提示される際には、“お得な情報”という言葉が代わりに提示されるようにする。

【0089】

このように、Reasonという項目を設けることにより、マッチングなどの処理に用いられる言葉と、ユーザ側に提示される言葉とを使いわけることが可能となる。

【0090】

図18に示したデータベースは、M_KEYWORD_2データベース277であり、上述したM_K

10

20

30

40

50

KEYWORDデータベース275と基本的に同様のデータを管理するために設けられている。このM_KEYWORD_2データベース277は、図16に示したM_KEYWORDデータベース275と比較し、Displayという項目も、KeywordIDやKeywordと、それぞれ関連付けて管理する。このDisplayという項目に記載されているのはフラグであり、そのフラグは、Keywordの項目に記載されている言葉を、ユーザ側に提示するか否かを表すものとされている。

【0091】

図18に示した例で、“1”は、ユーザ側に提示し、“0”は、ユーザ側に提示しないとして設定されていることを示す。例えば、KeywordIDが“0000000001”と割り当てられている“情報”というキーワードは、オススメ理由として、ユーザ側に提示されるが、KeywordIDが“0000000006”と割り当てられている“エロティック”というキーワードは、オススメ理由として、ユーザ側には提示されない。

10

【0092】

図16乃至図18に示したデータベースのうち、いずれか1つのデータベースがマスターデータベース251に管理されていれば良い。よって、図16乃至図18に示した各データベースのAttributeIDは、“005”と共通のIDにしてある。以下の説明においては、特に断りのない限り、図16に示したM_KEYWORDデータベース275が、マスターデータベース251に管理されているとして説明する。

【0093】

図19に示したデータベースは、M_AGEデータベース278であり、ユーザの年齢層を管理するために設けられている。このM_AGEデータベース278には、AttributeIDとして、“006”が割り当てられている。M_AGEデータベース278は、年齢を識別するための識別子(AgeID)、年齢の幅を示し、その幅を識別するための名前(AgeName)、その年齢の幅の始めの年齢(StartAge)、および、その年齢の幅の終わりの年齢(EndAge)を、それぞれ関連付けて管理する。

20

【0094】

例えば、図19に示したM_AGEデータベース278において、AgeID“001”は、“0から4”というAgeNameに関連付けられ、“0”というStartAgeにも関連付けられ、さらに、“4”というEndAgeにも関連付けられている。

【0095】

図20に示したデータベースは、M_GENDERデータベース279であり、ユーザの性別を管理するために設けられている。このM_GENDERデータベース279には、AttributeIDとして、“007”が割り当てられている。M_GENDERデータベース279は、性別を識別するための識別子(GenderID)と性別(GenderType)を関連付けて管理する。例えば、図20に示したM_GENDERデータベース279において、GenderID“001”は、“男性”というGenderTypeに関連付けられている。

30

【0096】

図21に示したデータベースは、M_REASONデータベース280であり、理由表示部155(図7または図8)に表示される言葉を制御するためのデータを管理する。M_REASONデータベース280は、AttributeID、ValueID、およびReasonを、それぞれ関連付けて管理する。AttributeIDは、上述したように、各データベースを識別するために割り振られたIDであり、例えば、図12に示したM_CATEGORYデータベース271には、AttributeIDとして“001”が割り振られている。

40

【0097】

ValueIDは、各データベースが管理するIDのことであり、CategoryID(図12)、PersonID(図13)、CharacterID(図14)、ViewID(図15)、KeywordID(図16)、AgeID(図19)、および、GenderID(図20)のことである。Reasonは、図17や図18を参照して説明したように、理由表示部155(図7, 8)に表示される言葉のことである。

【0098】

例えば、図21に示したM_REASONデータベース280のAttributeIDが“001”であ

50

る場合、M_CATEGORYデータベース271のことを示し、加えてValueIDが“1000”である場合、M_CATEGORYデータベース271のCategoryIDが“1000”のことを示す。そして、M_REASONデータベース280のAttributeIDが“001”、ValueIDが“1000”で示される言葉が、ユーザに提示される場合、Reasonの項目の対応する欄に記載されている“スポーツ”という言葉が提示されることとなる。

【0099】

このM_REASONデータベース280は、上述した図12乃至図19（図17と図18は除く）にそれぞれ示したデータベースを、理由表示部155（図8）に表示させる言葉を制御するために、1つにまとめたものである。制御するためには、上述したように、ユーザ側に提示するには好ましくない言葉や、そのままではユーザ側に提示するには適していない言葉があり、そのような言葉を適切に処理するためという意味である。 10

【0100】

例えば、ユーザ側に提示するには好ましくない言葉として、上述した例では“エロティック”という言葉が挙げたが、この“エロティック”という言葉は、図16に示したM_KEYWORDデータベース275に管理されており、KeywordIDとして“0000000006”が割り当てられている。M_KEYWORDデータベース275のAttributeIDは“005”であるので、図21に示したM_REASONデータベース280のAttributeIDが“005”で、ValueIDが“0000000006”を見ると、記載がない（削除されている）ことがわかる。M_REASONデータベース280に記載がないということは、ユーザ側には、提示されないということになる。

【0101】

20

よって、M_REASONデータベース280に記載しなければ、ユーザ側に提示するには好ましくない言葉は、提示されないようにするといった制御を行えることになる。

【0102】

また、例えば、そのままではユーザ側に提示するには適していない言葉として、上述した例では“得”という言葉が挙げたが、この“得”という言葉は、図16に示したM_KEYWORDデータベース275に管理されており、KeywordIDとして“0000000008”が割り当てられている。M_KEYWORDデータベース275のAttributeIDは“005”であるので、図21に示したM_REASONデータベース280のAttributeIDが“005”で、ValueIDが“0000000008”を見ると、“お得な情報”という記載がある。M_REASONデータベース280に“お得な情報”と記載があるということは、ユーザ側に“得”という言葉（キーワード）が提示されるときには、実際には“お得な情報”という言葉が提示されることになる。 30

【0103】

よって、M_REASONデータベース280にユーザ側に提示したい言葉を記載しておけば、そのままではユーザ側に提示するには適していない言葉であっても、実際に提示されるときには、ユーザ側に提示するのに適した言葉で提示できるようにするといった制御を行えるようになる。

【0104】

すなわち、番組を推薦する場合を考えると、番組のスコアリング（マッチング）には有効な番組を特徴付ける語であっても、その語自体が、例えば、成人向けの用語、一文字の語、メジャーでないキーワード、一般的過ぎて理由になりにくいキーワード、オススメ理由として成立していないキーワードなどであり、オススメ理由としてユーザに提示するには馴染まないような語である場合、そのような語を表示させない、または、他の語として表示させるといった、自由度の高い表示の制御を行うことが可能となる。 40

【0105】

また、M_REASONデータベース280という、オススメ理由の表示を制御するためのデータベースを設けることにより、図12乃至図20に示したマスターデータベースに基づく表示とは異なる表示が行えるようになるため、例えば、形態素解析の結果、“恋い”、“恋”、“こい”などの全てのValueに対して、「恋愛好きに」といったオススメの理由の表示が可能となる。

【0106】

50

このようなデータを管理するM_REASONデータベース280をマスターデータベース251で管理させることにより、上述したように、適切に、ユーザ側に提示する言葉を制御することができるといった効果の他に、以下のような効果もある。その効果について図22と図23を参照して説明する。

【0107】

図22と図23においては、オススメ理由として、AttributeIDが“001”でCategoryIDが“1000”の言葉（検索言葉Aとする）、AttributeIDが“002”でPersonIDが“000000001”の言葉（検索言葉Bとする）、AttributeIDが“002”でPersonIDが“000000002”の言葉（検索言葉Cとする）、および、AttributeIDが“005”でKeywordIDが“0000000001”の言葉（検索言葉Dとする）を、ユーザ側に提示すると決定された場合を例に挙げて説明する。

10

【0108】

図22を参照して、まず、M_REASONデータベース280が存在していないとき、従って、図12乃至図20（図17、図18を除く）のみが存在しているときの検索の仕方について説明する。検索言葉Aが検索される際、AttributeIDとして“001”が割り当てられているM_CATEGORYデータベース271（図12）内が検索される。そして、CategoryIDが“1000”である“スポーツ”という言葉が検索結果とされる。同様に、検索言葉Bが検索される際、AttributeIDとして“002”が割り当てられているM_PERSONデータベース272（図13）内が検索され、PersonIDが“0000000001”である“人物A”という言葉が検索結果とされる。

20

【0109】

検索言葉Bと同様に、検索言葉Cも、M_PERSONデータベース272が参照され、“人物B”という言葉が検索結果とされる。そして、検索言葉Dは、M_KEYWORDデータベース275が参照され、“情報”という言葉が検索結果とされる。

【0110】

この場合、4つの検索言葉を検索するのに、3つのデータベースを参照しなければならない。

【0111】

これに対し、M_REASONデータベース280が存在する場合、図23を参照して以下に説明するように検索が行われる。検索言葉Aが検索される場合、M_REASONデータベース280が参照される。その参照される位置は、AttributeIDが“001”であり、ValueIDが“1000”のところである。その結果、“スポーツ”という言葉が検索結果とされる。同じように、検索言葉B乃至Dも、M_REASONデータベース280が参照されて、それぞれ検索結果が出される。

30

【0112】

この場合、4つの検索言葉を検索するのに、1つのデータベースのみを参照すればよい。

【0113】

このように、M_REASONデータベース280を設けることにより、検索を実行する際、複数のデータベースを参照する必要はなく、M_REASONデータベース280のみを参照すればよいので、検索に要する時間を短くすることができる。このことは、ユーザがオススメ画面を要求してから、そのユーザに対するオススメ画面を返すまでにかかる時間が短くなることを意味し、リアルタイムに返答を返さなくてはならないシステムにおいて、有効な手段である。

40

【0114】

このように、マスターデータベース251には、EPGの作成、オススメ画面の作成に必要なデータが管理されている。

【0115】

次に、ターゲットデータベース252（図11）について説明する。

【0116】

50

ターゲットデータベース252は、オススメ画面を作成する際、ユーザの嗜好と番組とのマッチングを取るためのデータ、ユーザの嗜好をより詳細なものとするための学習機能を実現するためのデータなどから構成される。

【0117】

図24に示したデータベースは、T_PERSON_TARGET_GENDERデータベース311であり、番組に出演する人物(Person)が、どのような性別(Gender)の人々に人気があるか等を数値化したデータを管理する。T_PERSON_TARGET_GENDERデータベース311は、PersonID、GenderID、および、Scoreを関連付けて管理する。PersonIDは、図13に示したM_PERSONデータベース272が管理するPersonIDと同じIDとされている。すなわち、例えば、PersonID“0000000001”は、“人物A”のことを意味する。

10

【0118】

GenderIDは、図20に示したM_GENDERデータベース279が管理するGenderIDと同じIDとされている。すなわち、例えば、GenderIDが“001”は、“男性”のことを意味する。Scoreは、PersonIDに対応する人物は、GenderIDに対応する性別の人に対して、どの程度の数値(重み付け)を有するかを示す値である。例えば、人物Aが、男性に人気があるなら、数値を高くするまたは低く設定される。

【0119】

数値が高く設定される場合、人気があるので、マッチングが行われるときに検出される可能性を高くするという意図があるときである。逆に、数値が低く設定される場合、人気があるので、沢山の番組に出ている、汎用すぎる等の理由から、マッチングが行われるとき、あまり重要な要素とならないようにするという意図があるときである。どちらに設定するかは、EPGを管理する管理者の考え、利用するユーザの特徴(どの性別のユーザが多いか、どの年代のユーザが多いかなど)、システムへの負荷などを考慮して設定されればよい。

20

【0120】

なお、他のターゲットデータベースも、それぞれ、ターゲットとする要素に関わる数値を管理するが、その数値が高い値として設定されるか、低い値として設定されるかは、上述したようなことが考慮されて設定される。

【0121】

図25に示したデータベースは、T_PERSON_TARGET_AGEデータベース312であり、番組に出演する人物(Person)が、どのような年齢(Age)の人々に人気があるか等を数値化したデータを管理する。T_PERSON_TARGET_AGEデータベース312は、PersonID、AgeID、および、Scoreを関連付けて管理する。このT_PERSON_TARGET_AGEデータベース312は、PersonIDは、図13に示したM_PERSONデータベース272が管理するPersonIDと同じIDとされている。

30

【0122】

AgeIDは、図19に示したM_AGEデータベース278が管理するAgeIDと同じIDとされている。すなわち、例えば、AgeID“001”は、“0から4”歳のことを意味する。Scoreは、PersonIDに対応する人物は、AgeIDに対応する年齢の人に対して、どの程度の数値(重み付け)を有するかを示す値である。例えば、人物Aが、0から4歳(赤ちゃん)に人気があるなら、数値が高くまたは低く設定される。

40

【0123】

図26に示したデータベースは、T_PERSON_VIEW_TYPEデータベース313であり、番組に出演する人物が、どのような視聴タイプに適しているか等を数値化したデータを管理する。T_PERSON_VIEW_TYPEデータベース313は、PersonID、ViewID、および、Scoreを関連付けて管理する。このPersonIDも、図13に示したM_PERSONデータベース272が管理するPersonIDと同じIDとされている。

【0124】

ViewIDは、図15に示したM_VIEW_TYPEデータベース274が管理するViewIDと同じIDとされてる。すなわち、例えば、ViewID“001”は、“笑い刺激型”のことを意味する

50

。Scoreは、PersonIDに対応する人物は、ViewIDに対応する視聴タイプに対して、どの程度の数値（重み付け）を有するかを示す値である。例えば、人物Aが、笑い刺激型である（お笑い番組に良く出る芸人）なら、数値が高くまたは低く設定される。

【0125】

図27に示したデータベースは、T_PERSON_CHARACTERデータベース314であり、番組に出演する人物が、どのようなキャラクタ（特徴）を有しているか等を数値化したデータを管理する。T_PERSON_CHARACTERデータベース314は、PersonID、CharacterID、および、Scoreを関連付けて管理する。このPersonIDも、図13に示したM_PERSONデータベース272が管理するPersonIDと同じIDとされている。

【0126】

CharacterIDは、図14に示したM_PERSON_CHARACTERデータベース273が管理するCharacterIDと同じIDとされてる。すなわち、例えば、CharacterID“001”は、“司会者系”のことを意味する。Scoreは、PersonIDに対応する人物は、CharacterIDに対応するキャラクタに対して、どの程度の数値（重み付け）を有するかを示す値である。例えば、人物Aが、バラエティ番組などで司会者をつとめることが多い司会者系であるなら、数値が高くまたは低く設定される。

【0127】

図28に示したデータベースは、T_PRG_VALUデータベース315であり、所定の番組と、その番組に関連する事項を関連付けて管理する。T_PRG_VALUデータベース315は、ProgramID、Attribute、および、ValueIDを関連付けて管理する。ProgramIDは、各番組を識別するために各番組に割り当てられている識別子である。

【0128】

Attributeは、どのデータベースに対する特性であることを示すデータである、AttributeIDで記載することも可能である。ValueIDは、図24に示したM_REASONデータベース280が管理するValueIDと同じIDとされてる。すなわち、例えば、AttributeIDが“002”（すなわち、AttributeとしてはPerson）で、ValueIDが“1000”は、“人物A”のことを意味する。このように、T_PRG_VALUEデータベース315は、所定の番組と、その番組に出演している人物（Person）や、その番組を特徴付けるキーワード（Keyword）などを関連付けるためのデータベースとして用いられる。

【0129】

図29に示しデータベースは、T_PRG_VALUEデータベース316であり、番組自体の重み付けの数値を管理する。このT_PRG_VALUEデータベース316は、図28に示したT_PRG_VALUデータベース315に、Scoreをさらに関連付けて管理する構成とされている。ただし、図29に示したT_PRG_VALUEデータベース316では、Attributeではなく、AttributeIDを管理する。

【0130】

例えば、図29に示したT_PRG_VALUEデータベース316において、ProgramIDとして“1000000030000029”というIDが割り当てられている番組は、AttributeIDが“001”、ValueIDが“1000”に対して“0.65193365995972070000”という数値が与えられている。

【0131】

スコアリングの計算のときに、このScoreの値が用いられるが、そのスコアリングの計算時には、AttributeIDとValueIDの組から一意にScore値が求められるため、AttributeIDとValueIDの組が、どのような語に関連付けられているかを判別する必要はないが、説明のために判別すると、AttributeIDが“001”、ValueIDが“1000”は、図21に示したM_REASONデータベース280を参照するに“スポーツ”である。よって、ここでは、ProgramIDとして“1000000030000029”というIDが割り当てられている番組は、“スポーツ”という要素に対しては、“0.65193365995972070000”という数値を有することになる。

【0132】

10

20

30

40

50

このように、このT_PRG_VALUEデータベース316も、複数のAttributeIDを含むため、それぞれの要素を検出する必要がある場合、複数の対応するマスターデータベースを参照しなくてはならないが、M_REASONデータベース280を設けることにより、このM_REASONデータベース280のみを参照するだけで、必要な要素を検出することが可能となる。ただし、M_REASONデータベース280には、ユーザ側に提示するにはふさわしくないキーワードなどは削除されている（記載されていない）ため、必要に応じてM_REASONデータベース280を参照するようにしても、削除されている語は検出することはできないので、処理に必要な場合は、M_REASONデータベース280以外のマスターデータベースを参照する必要がある。

【0133】

10

図30に示したデータベースは、T_UM_VALUEデータベース317であり、ユーザの嗜好に関するデータを管理するために設けられている。T_UM_VALUEデータベース317は、MemberID、UMTypeID、AttributeID、ValueID、および、Scoreを関連付けて管理する。MemberIDは、ユーザ登録しているユーザに対して割り当てられた識別子である。UMTypeIDは、この要素は、何に基づいて設定されたものであるかを示す。“何に基づいて”とは、ここでは、登録、学習、フィルタリングに基づく場合が考えられる。

【0134】

例えば、図30を参照するに、1行目のAttributeIDが“001”でValueIDが“1000”の要素は、図21のM_REASONデータベース280を参照するに、“スポーツ”という要素であるが、その要素を、ユーザが、自分は“スポーツ”が好きであると登録してあった場合には、UMTypeIDは“1”と設定される。

20

【0135】

図30を参照するに、2行目のAttributeIDが“001”でValueIDが“1001”の要素は、図21のM_REASONデータベース280を参照するに、“スポーツ：野球”という要素であるが、その要素は、ユーザが良く視聴している番組を解析した結果、スポーツの中でも“野球”が好きであると判断され、その結果設定された要素である場合には、UMTypeIDは“2”と設定される。

【0136】

図30を参照するに、3行目のAttributeIDが“002”でValueIDが“000000001”の要素は、図21のM_REASONデータベース280を参照するに、“人物A”という要素であるが、その要素を、後述するフィルタリングの処理の結果、“人物A”が好きであろうと判断され、その結果設定された要素である場合には、UMTypeIDは“3”と設定される。

30

【0137】

本実施の形態においては、このように、ユーザにオススメする番組を検出する際に用いられる要素は、ユーザ自身による登録により設定され、ユーザの嗜好を学習することにより設定され、さらに、フィルタリングの処理により設定される。このようにすることで、よりユーザの嗜好を的確に判断し、オススメの番組を提示できるようになる。例えば、ユーザ自身による登録だけに依存すると、あまり多くの要素を登録することができない。例えば、3万個の要素がある場合、その3万個の要素の中から、ユーザ自身が、自分に合う要素を登録するというのは、不可能なこと（可能であってもユーザにとって煩雑な処理）となってしまう。

40

【0138】

よって、例えば、“スポーツ”（AttributeIDが001でValueIDが1000）などという広い範囲のカテゴリーは設定できても、スポーツの中でも“野球”といった細かい設定はできない可能性がある。しかしながら、ユーザが“スポーツ”という要素しか設定していなくても、そのユーザが“野球”をよく視聴していれば、学習という機能により、“スポーツ：野球”（AttributeIDが001でValueIDが1001）の要素も設定されることになる。さらにフィルタという機能を設けることにより、後述するように、ユーザの嗜好に合っているはずと推測されるが、ユーザが登録もせず、学習の結果でも設定されないような要素も設定することが可能となる。

50

【0139】

なお、このように登録、学習、フィルタという3つの処理で要素が設定されるわけだが、ユーザ自身が登録した要素の方が、他の処理で設定された要素よりも重要視され、学習により設定された要素の方が、フィルタという処理で設定された要素よりも重要視される方が好ましい。そのような重み付けに関することは、図30に示したT_UM_VALUEデータベース317のScoreの数値を変えることにより、制御することができる。

【0140】

図22乃至図30を参照して説明したようなデータベースが、ターゲットデータベース252に管理されている。

【0141】

ここで、T_PRG_VALUEデータベース315の作成について、図31のフローチャートを参照して説明する。図31に示したフローチャートに基づく処理は、EPGのデータが作成された際に行われる（放送する番組やその番組に出演する出演者などが決定し、EPGを作成するためのデータがそろった時点で行われる）。また、図31に示したフローチャートに基づく処理は、サーバ52のCPU201（図9）が実現する対象情報メタデータ管理機能237（図10）が実行する。または、管理者が手作業で行いデータを入力することも可能である。ここでは、CPU201が実行するとして説明する。他のフローチャートの処理も、断りのないかぎり、CPU201（CPU201が実現する機能）が実行するとして説明する。

【0142】

ステップS31において、CPU31は、T_PRG_VALUEデータベース315に追加するデータを取得する。ステップS31において取得されるデータは、番組に出演する出演者、番組のキーワードなどの要素に対応するT_PRG_VALUEである。各要素は、主に、EPG、例えば、図6に示したEPGに表示される情報であり、そのために、EPGを作る時に必要とされるデータであり、そのEPGを作成する側から供給されるものである。

【0143】

例えば、図32に示したようなEPGがユーザ側に提供される場合を考える。図32の左側に示したEPGでは、“番組A-1”という番組名と、その下に出演者として“人物A”と“人物B”が表示されている。さらに下側に、“情報”と“買い物”という2つのキーワードが表示されている。このようなEPGを表示させるのに必要な情報としては、図32の右側に示したように、まず、“人物A”というのを表示させるために、AttributeIDが“002”でValueIDが“0000000001”という情報が必要である。同じように、“人物B”、“情報”、および“買い物”という表示をさせるのに必要な情報としては、AttributeIDとValueIDの組み合わせの情報が必要である。

【0144】

図32の図中右側に示したような情報が、ステップS31において取得されるわけだが、そのデータは、図28に示したT_PRG_VALUEデータベース315として提供されるため、対応するT_PRG_VALUEデータベース315内のデータを取得することにより行われる。ステップS32において、マップデータから、ステップS31において取得された各要素に対応する属性データと、その属性データに対応するスコアが取得される。マップデータとは、ターゲットデータベース252で管理されている各データベースのことである。

【0145】

ステップS33において、出演者や出現キーワードのT_PRG_VALUEのスコアに、ステップS32において取得されたマップデータから取得されたスコアを乗算し、その値を属性データのスコアとして設定する。そして、ステップS34において、設定されたスコアが、T_PRG_VALUEデータベース316の1つの要素として追加、記憶される。要素が追加される際、その追加される要素には、属性に対応するAttributeIDが割り当てられる。また、その要素には、ValueIDとして、当該属性のマスタデータで定められている属性を一意に識別するIDが割り当てられる。

【0146】

10

20

30

40

50

このようにして、図 2 9 に示した T_PRG_VALUE データベース 3 1 6 が作成される。

【 0 1 4 7 】

次に、図 3 0 に示した T_UM_VALUE データベース 3 1 7 の作成に関わる処理について説明する。図 3 0 に示した T_UM_VALUE データベース 3 1 7 のところの説明で、T_UM_VALUE データベース 3 1 7 の各要素は、登録、学習、またはフィルタといういずれかの処理により設定されたものであると説明した。ここではまず、図 3 3 に示したフローチャートを参照して、ユーザ自身の登録による要素の追加処理について説明する。T_UM_VALUE データベース 3 1 7 の作成に関わる処理は、CPU 2 0 1 が実現する 1 つの機能であるユーザモデル管理機能 2 3 5 により実行される。

【 0 1 4 8 】

ステップ S 5 1 において、CPU 2 0 1 は、T_UM_VALUE データベース 3 1 7 に追加するデータを取得する。ステップ S 5 1 において取得されるデータは、編集対象となっているユーザ（視聴者）の嗜好に関わる出演者、キーワードなどの各要素に対する T_UM_VALUE である。ステップ S 5 2 において、マップデータから、ステップ S 5 1 において取得された各要素に対応する属性データと、その属性データに対応するスコアが取得される。

【 0 1 4 9 】

ステップ S 5 3 において、出演者や出現キーワードの T_UM_VALUE のスコアに、ステップ S 5 2 において取得されたマップデータからの取得されたスコアを乗算し、その値を属性データのスコアとして設定する。そして、ステップ S 5 4 において、設定されたスコアが、T_UM_VALUE データベース 3 1 7 の 1 つの要素として追加、記憶される。要素が追加される際、その追加される要素には、属性に対応する AttributeID が割り当てられる。また、その要素には、ValueID として、当該属性のマスターデータで定められている属性を一意に識別する ID が割り当てられる。

【 0 1 5 0 】

このようにして、ユーザ自身が登録した要素に対する処理が行われることにより図 3 0 に示した T_UM_VALUE データベース 3 1 7 の一部が作成される。

【 0 1 5 1 】

次に、T_UM_VALUE データベース 3 1 7 に、学習により要素が追加される場合の処理について、図 3 4 のフローチャートを参照して説明する。ステップ S 7 1 において、処理対象となっているユーザが視聴した番組（録画予約した番組）に関するデータが、ログデータベース 2 5 3（図 1 1）から読み出される。ログデータベース 2 5 3 には、ユーザを識別するための識別子（MemberID）、そのユーザが視聴した番組を識別するための識別子（ProgramID）が、少なくとも関連付けられて管理されている。

【 0 1 5 2 】

ステップ S 7 1 においては、所定のユーザが視聴した番組に関わるデータが取得されるわけだが、まず、処理対象とされているユーザの MemberID が取得される。そして、その MemberID を基に、ログデータベース 2 5 3 からユーザが利用した番組の ProgramID が取得される。ProgramID が取得されると、T_PRG_VALUE データベース 3 1 6 が参照され、取得された ProgramID に対応する、それぞれの要素毎に、T_PRG_VALUE が取得される。

【 0 1 5 3 】

ステップ S 7 2 において、取得された T_PRG_VALUE のスコアが、同一の AttributeID、ValueID の T_UM_VALUE の要素に加算される。この処理について、図 3 5 を参照して説明を加える。ログデータベース 2 5 3 に、MemberID が “ 200241723056783 ”、ProgramID が “ 1000 0000300000029 ” というデータが残されている状況で、このデータが読み出されると、まず、同一の ProgramID を有する要素が、T_PRG_VALUE データベース 3 1 6 から読み出される。図 3 5 に示した例では、AttributeID が “ 0 0 1 ”、ValueID が “ 1000 ”、Score が “ 1.651 93365995972070000 ” という要素が読み出される。

【 0 1 5 4 】

次に、この読み出された AttributeID と ValueID が一致する要素が、T_UM_VALUE データベース 3 1 7 内に存在するか否かが判断される。図 3 5 に示したように、一致する要素が存

10

20

30

40

50

在する場合、その存在していると判断された要素のScore、図35に示した例では、“.84296419427714530000”が読み出される。その読み出されたScoreに、先に、T_PRG_VALUEデータベース316から読み出されているScore(この場合、“1.65193365995972070000”)が加算される。なお、単に加算されるだけでなく、所定の値が乗算される(重み付けが行われる)などするようにしても良い。

【0155】

このように、ユーザが番組を視聴する(録画予約する)たびに、T_UM_VALUEデータベース317の各要素のScoreが変更される。すなわち、よりユーザの嗜好に近いものになるように、学習が行われていることになる。

【0156】

図35に示した例では、T_UM_VALUEデータベース317に、T_PRG_VALUEデータベース316から読み出されたAttributeIDとValueIDが一致する要素が存在する場合を例に挙げたが、存在していない場合も考えられる。例えば、ユーザAが視聴した番組に“人物A”が出演していたが、その“人物A”という要素は、ユーザAに対応するT_UM_VALUEデータベース317には登録されていないこともある。

【0157】

そのような場合、新たな要素がT_UM_VALUEデータベース317に追加される。図36を参照して、このような状況について説明を加える。

【0158】

ユーザAが、番組Xを視聴し、その番組Xには、人物Aと人物Bが出演していた。人物Aは、ユーザAが好みであり、人物Aが出演していた番組は、よく視聴していたが、人物Bのことは知らなく、人物Bが出演している番組X以外の他の番組を視聴したことはなかった。このような状況の時には、ユーザAのT_UM_VALUEデータベース317には、人物Aに関するScoreAは既に存在しているので、そのScoreAに、新たな数値が加算され、ScoreA'とされる。

【0159】

一方、人物Bに関する要素は、ユーザAのT_UM_VALUEデータベース317には存在していないため、新たに人物Bに対する要素が追加される。

【0160】

ここでは、人物Aと同じ番組に出演していた人物Bに関する要素が追加されるとしたが、例えば、人物Aのキャラクタや視聴タイプなどの要素も追加される場合がある。また、人物Aのキャラクタや視聴タイプなどが、既に存在しているときには、そのScoreに、新たな値が加算されるなどして、その値が変化することはもちろんある。

【0161】

例えば、ユーザAが視聴し、人物Aが出演していた番組Xは、バラエティ番組であっても、その人物A自体は、時代劇に出演することが多い人物であるような場合、人物Aのキャラクタとして、“時代劇”が関連付けられている(マッピングされている)。よって、ユーザAが視聴していた番組Xが、バラエティ番組であっても、上述したような処理が実行されるため、番組Xを視聴することにより、人物Aに関連付けられている“時代劇”という要素のスコア値があがり、次回から、“時代劇”もオススメ番組として推薦される可能性が高くなる。

【0162】

また、人物Aが“たくましい”といった属性を有する場合、その“たくましい”といった属性を有する人物C(不図示)の要素がユーザAのT_UM_Valueデータベース317に追加される。もちろん、人物Aが“たくましい”といった属性を有する場合、かつ、ユーザAのT_UM_VALUEデータベース317に“たくましい”といった要素が存在している場合、その要素のScoreは、更新される。

【0163】

さらに、ここでは、番組Xには、人物Aと人物Bが出演しているとしたが、例えば、人物Aのみが出演していた場合でも、同様なことが実行される。すなわち、人物Aと人物B

10

20

30

40

50

は、“時代劇”というキーワード（属性）で関連付けられている（マッピングされている）ような場合、人物Aしか出演していない番組Xが視聴された時点で、人物Aにマッピングされている人物Bが読み出され、その人物Bの要素が追加されることになる。

【0164】

すなわち、このような学習が行われることにより、この場合、番組Xの属性や人物Aの属性に関連付けられている要素も、新たに、T_UM_VALUEデータベース317に追加される、または、その要素のScoreが更新される。

【0165】

このように、ユーザが録画予約や視聴などを行うことにより、Score値は更新され、新たな要素が追加される。よって、徐々に、ユーザの嗜好に関するデータが、サンプル数が増えることにより詳細なものとなり、そのようなデータを用いて行われるオススメ番組の検索を、よりユーザに適した番組（ユーザが求める番組）を検索することができるものとする。 10

【0166】

図37は、このような学習の処理を行わない場合と、行った場合とを比較するために作成したグラフである。グラフAは、属性によるオススメを行った時を示し、グラフBは、詳細メタデータのみによりオススメを行った時を示す。図37に示したように、あまり時間が経過していない初期段階では、属性によるオススメを行う方（グラフA）が、学習度が高く、一定の時間が経過した段階では、詳細メタデータのみによりオススメを行う方（グラフB）の方が、学習度が高いといった結果が得られる。 20

【0167】

この結果を考慮すると、ユーザ登録して間もないユーザに対しては、学習という処理を行う方が、より早い時期に、学習度を高めることができ、ユーザの嗜好のデータの信頼性を高めることができる。そして、所定の時間が経過した後（所定の学習度に達成した後）は、学習をやめる（または、学習の回数を少なくする）などした方が好ましい。

【0168】

いずれにしても、学習を行うことにより、より早い時期に、ユーザの嗜好に関する、信頼性の高いデータを作成することができる。よって、より早い時期に、適切にユーザの嗜好に適した番組をオススメすることができるようになり、ユーザが満足するような結果を提供することが可能となる。よって、登録したユーザが、離れてしまう（脱会する）といったようなことを防ぐことが可能になる。 30

【0169】

ユーザの嗜好に関するデータを、学習により収集するといった方法の他に、ユーザ自身で、登録するといった方法があることは既に上述した。これらの方法は、よりユーザの嗜好を正しいものとするのが1つの目的としてある。そこで、収集されるデータ（要素）についてだが、この場合、テレビジョン放送として放送される番組内から、ユーザの嗜好に合った番組をオススメするというのが1つの目的としているため、主に、番組の種類、番組に出演する出演者などに関する要素が収集される。

【0170】

もちろん、このような要素だけでも良いが、さらに、ユーザのライフスタイルに関する情報を収集し、そのような情報に基づく要素も考慮して番組の推薦が行われるようにしても良い。ライフスタイルとは、例えば、日中はテレビジョン放送を視聴しない等である。しかしながら、このような、ある意味、ユーザのプライバシーに関わることは、また、ユーザにとって、番組の推薦にどのような関連があるのか直接的に導き出せないプライバシーな情報を、簡単に提供することを快くは思わないユーザもいると考えられる。 40

【0171】

そこで、このような情報は、アンケート形式や占い形式で、ユーザ側には、プライベートな情報を提供して感じさせないような形式で収集する。このようにすることで、ユーザは、ゲーム感覚でプロフィールを登録することができ、そのようなプロフィールを管理する（収集する）側（この場合、サーバ52、および管理者）は、プロフィールを収 50

集しやすくなるといった効果を望むことができる。

【0172】

このようなライフスタイルといった情報も、オススメ番組の検索が実行される際に用いられる要素とすることにより、よりユーザに適した番組を推薦することが可能となる。

【0173】

次に、T_UM_VALUEデータベース317に対するフィルタリング処理（またはマイニング（mining）処理）について説明する。このフィルタリング処理は、主に図8に示したオススメ画面のうち、理由表示部155の“同じ番組を録画した人の興味”という項目を表示させるために行われる処理である。このような表示を行うのは、以下のような状況、理由からである。

10

【0174】

図38を参照して説明する。例えば、ユーザAが、番組Xを録画予約し、ユーザBも番組Xを録画予約したとする。ユーザAもユーザBも、番組Xに対して興味があると考えられる。例えば、その番組Xが、お笑い番組だとすると、ユーザAとユーザBは、お笑い系が好みであると考えられる。

【0175】

ユーザBは、お笑い系に属する番組Zも録画予約している。番組Xには、人物Aが出演し、番組Zには、人物Bが出演している。このような場合、番組Zや人物Bは、ユーザAが行った録画予約のログ内やユーザAの嗜好データには含まれていないが（従って、オススメ番組として推薦されることはないが）、お笑い系の好きなユーザAは、同じお笑い系に属する番組Zや人物Bのことも好みであると考えられる。換言すれば、ユーザAは、好みであるはずの番組Zや人物Bのことを単に知らないだけとも考えられる。

20

【0176】

そこで、番組Xを録画予約したユーザBが録画予約した番組Z、および、番組Zに出演する人物Bが、ユーザAに推薦されるようにする。このようにすれば、ユーザA側としては、その時点まで認識していなかった番組Zや人物Bのことを、新たに認識することができる。認識していなかった新たな情報が提示されるということは、ユーザ側にとって、有意な情報であり、サーバ52から提供される情報の信頼性を高めることができると考えられる。

【0177】

このようなことは、従来でも、似たようなことは行われていた。例えば、書籍を販売しているサイトにおいて、購入者Aが書籍Aを購入した場合、同じくその書籍Aを購入した購入者Bが購入した書籍Bを、購入者Aに情報として提供するといったことは行われていた。このような場合、購入者Aの嗜好に関しては考慮されておらず、単に、購入者Aより前の時点で書籍Aを購入した購入者Bが存在し、その購入者Bが購入したのが書籍Bであるので、その書籍Bがオススメされるという結果になっていた。

30

【0178】

ここで、従来行われていたオススメ（フィルタリング処理）と、本実施の形態で行われるオススメ（フィルタリング処理）を比較し、その違いを説明する。図39は、従来行われていたフィルタリング処理について説明するための図である。図39に示したように、ユーザA乃至ユーザZが、視聴した番組（コンテンツ）の履歴が管理されている状況を考える。例えば、ユーザAにコンテンツをオススメする場合、ユーザB乃至ユーザZから構成されるユーザグループの中から、ユーザAと同じコンテンツを沢山視聴した人、すなわち、コンテンツの利用の相関度が高いユーザが検索される。

40

【0179】

そして、相関度が高いとして検索されたユーザが利用したコンテンツの中で、ユーザAがまだ利用したことがないコンテンツが、オススメとしてユーザAに提示される。

【0180】

このようにしてユーザ側に情報を提示する従来の方法の場合、過去に相関度の高い人が利用したコンテンツ（番組）しかオススメできないといった問題があった。

50

【 0 1 8 1 】

図 4 0 は、本実施の形態において行われるフィルタリング処理について説明するための図である。図 4 0 に示したように、ユーザ A 乃至ユーザ Z が、利用したコンテンツの履歴が管理されている状態を考えるが、履歴の中には、各コンテンツのメタ情報も含まれている。さらに本実施の形態においては、コンテンツレポジトリも管理される。コンテンツレポジトリとは、ここでは、ターゲットデータベース 2 5 2 で管理されているデータのことである。

【 0 1 8 2 】

例えば、ユーザ A にコンテンツをオススメする場合、ユーザ B 乃至ユーザ Z から構成されるユーザグループの中から、ユーザ A と同じメタ情報（例えば、出演者やキーワードなどを情報として含んでいる）を沢山利用した人、すなわち、メタ情報の利用の相関度が高いユーザ（すなわち、嗜好が似ていると判断されるユーザ）が検索される。そして、相関度が高いとして検索されたユーザが利用したメタ情報の中で、ユーザ A が、まだ利用したことがないメタ情報、または、利用頻度が低いメタ情報がさらに検索される（従って、ユーザ A 自体は、認識していない出演者やキーワード）。そして、その検索されたメタ情報を有するコンテンツ（番組）が、コンテンツレポジトリから読み出され、そのコンテンツが、オススメとしてユーザ A に提示される。

【 0 1 8 3 】

このようにしてユーザ側に情報を提示することにより、例えば、再利用されることのないコンテンツや、期間限定で供給されるコンテンツなどの推薦が行えるようになると共に、通常のコンテンツなどの推薦も、より適切に行えるようになる。さらに例えば、コンテンツの推薦などにより得られたキーワードの相関度を用いて、他のコンテンツの推薦等を行うことも可能となる。

【 0 1 8 4 】

図 4 1 に示したフローチャートを参照して、本実施の形態におけるフィルタリング処理について説明する。ステップ S 9 1 において、ユーザ間の相関度の計算が行われる。このユーザ間の相関度の計算は、次式（ 1 ）に基づいて行われる。

【 数 1 】

$$\text{Sim}_{xy} = \frac{\sum_v (X_v - X_A) \cdot (Y_v - Y_A)}{\sqrt{\sum_v (X_v - X_A)^2} \cdot \sqrt{\sum_v (Y_v - Y_A)^2}} \dots\dots\dots (1)$$

【 0 1 8 5 】

式（ 1 ）において、 X_v は、Value v に対するユーザ X の評価（Score）を示し、 X_A は、ユーザ X の評価の平均値を示す。同様に、 Y_v は、Value v に対するユーザ Y の評価を示し、 Y_A は、ユーザ Y の評価の平均値を示す。式（ 1 ）に基づく計算を行うことにより、ユーザ X とユーザ Y の相関度（T_UM_SIMILARITY）を求めることができる。式（ 1 ）における各値は、T_UM_VALUEデータベース 3 1 7 で管理されている値が代入される。

【 0 1 8 6 】

このような計算が、ステップ S 9 1 において行われると、ステップ S 9 2 において、予測ベクトルの算出が行われる。ユーザ X の Value v に対する予測ベクトルの算出は、次式（ 2 ）に基づいて行われる。

【数 2】

$$\text{Expect}X_v = X_A + \frac{\sum_N (N_v - N_A) \text{Sim}_{XN}}{\sum_N |\text{Sim}_{XN}|} \quad \text{-----} (2)$$

【0187】

式(2)において、 X_A は、ユーザXの評価の平均値を示し、 N は、 Value_v に対するユーザXとの相関度を持つユーザNの評価を示し、 N_A は、ユーザNの評価の平均値を示し、 Sim_{XN} は、ステップS91において算出された値であり、ユーザXとユーザNの相関度を示す。 10

【0188】

このようにして算出された予測ベクトルは、T_UM_VALUEデータベース317に記憶される。図30に示したT_UM_VALUEデータベース317のところで説明したように、UMTypeIDの“3”は、このようなフィルタリング処理により算出された予測ベクトルであることを示している。

【0189】

ステップS91において算出された相関度を管理するデータベースを、記憶部208(図11)に設けるようにしても良い。また、ステップS92において算出された予測ベクトルの値は、T_UM_VALUEデータベース317に記憶されるようにしたが、記憶されない構成とすることも可能である。相関度や予測ベクトルは、会員登録しているユーザの内の1人が、録画予約や視聴をした時点で新たなデータが追加された(新たに計算の対象となるデータが追加された)ことになり、その結果、基本的に算出し直さなくてはならなくなる。 20

【0190】

このようなことを考慮すると、必要が生じた時点で算出し、常に、最新の予測ベクトルの値を用いることができるようにしておいた方が良いとも考えられたため、記憶させないようにすることも可能である。よって、相関度を管理するためのデータベースを設けなかった場合や、予測ベクトルの値をT_UM_VALUEデータベース317に記憶させるようにしなかった場合、必要に応じて、例えば、ユーザにオススメ画面を提供する時点で、相関度や予測ベクトルを算出するようにすればよい。 30

【0191】

また、相関度や予測ベクトルを事前に記憶させるようにした場合であっても、その記憶されているデータを、全て再度算出し直すといったようなことはせず、新たにログが生成されたユーザに対してのみ(そのユーザのデータに対してのみ)、相関度や予測ベクトルを算出するといった処理を行うようにしても良い。このようにすれば、相関度や予測ベクトルを算出するために要する時間を短縮することができる。また、サーバ52への負担を軽くすることができる。 40

【0192】

図41に示したフローチャートの処理のように、予測ベクトルを算出するようにしても良いが、例えば、1万人の会員が存在すると、約1万×1万の計算が最低限必要となる。会員数が増せば、それだけ、計算回数が増加し、システムにかかる負担が増大する。そこで、システムにかかる負担を軽減するとともに、軽減しても適切なフィルタリング処理が行えるようにする(図41に示したフローチャートの処理を実行したときとほぼ同様の効果が得られるようにする)ための、他のフィルタリング処理について、図42のフローチャートを参照して説明する。

【0193】

図42に示したフローチャートの処理では、ステップS101においてユーザ間の相関 50

度が計算され、ステップS 1 0 2において、ユーザグループの生成の処理が行われ、ステップS 1 0 3において、予測ベクトルの算出が行われる。図4 1に示したフローチャートの処理と比較すると、図4 2に示したフローチャートの処理では、ステップS 1 0 2におけるユーザグループの生成処理という処理が追加された構成とされている。このユーザグループの生成処理について、図4 3のフローチャートを参照して説明する。なお、ステップS 1 0 1における相関度の計算に関わる処理は、ステップS 9 1（図4 1）における相関度の計算に関わる処理と基本的に同様である。

【0 1 9 4】

ステップS 1 1 1において、全ユーザが各グループに属しているとしてグループをユーザの人数分だけ生成する。すなわち、構成人数が1人のグループが、ユーザ数分だけ生成される。例えば、1万人のユーザが存在していると、構成人数が1人のグループが1万グループ作成される。

【0 1 9 5】

ステップS 1 1 2において、相関度データベース（不図示）が参照され、相関度の高い順にソート処理が実行される。このステップS 1 1 2における処理においては、ステップS 1 0 1（図4 2）において算出された相関度が用いられる。相関度データベースが、設けられる構成とされている場合、ステップS 1 0 1において算出された相関度が、相関度データベースに記憶され、その記憶された相関度が用いられてステップS 1 1 2の処理が実行される。一方、相関度データベースが設けられない構成とされた場合、ステップS 1 0 1において相関度が算出される毎に、ソート処理が繰り返され、ステップS 1 0 1における相関度の処理が終了した時点で、ステップS 1 1 2におけるソートの処理も終了される。

【0 1 9 6】

ステップS 1 1 1における処理と、ステップS 1 1 2における処理は、時間的に前後関係を有しなくてはならない処理ではないため、ステップS 1 1 2における処理が、ステップS 1 1 1における処理より前の時点で行われても良いし、ステップS 1 1 1の処理とステップS 1 1 2の処理が、並列的に処理が行われても良い。

【0 1 9 7】

ステップS 1 1 2までの処理が終了されると、ステップS 1 1 3に処理が進められる。ステップS 1 1 3において、相関度の高い順に、計算対象とされている2人のユーザが属するグループがまとめて1つのグループとして設定される。この処理により、相関度がほぼ同じユーザ同士が、1つのグループとして設定されることになる。例えば、1万のユーザ存在し、1万のグループが設定されているときには、このステップS 1 1 3における処理で、半分の5千グループに設定し直されることになる。

【0 1 9 8】

ステップS 1 1 4において、設定し直されたグループ数は、最終的に求めたいグループ数以下になったか否かが判断される。ステップS 1 1 4において、最終的に求めたいグループ数以下になったと判断されるまで、ステップS 1 1 3とステップS 1 1 4の処理が繰り返される。

【0 1 9 9】

例えば、最終的に求めたいグループが2千グループ以下である場合、1万グループが存在しているときを考える。まず、1回目のステップS 1 1 3における処理で5 0 0 0グループ（1グループは、2名で構成される）とされ、2回目のステップS 1 1 3の処理で2 5 0 0グループ（1グループは、4名で構成される）とされ、3回目のステップS 1 1 3の処理で1 2 5 0グループ（1グループは、8名で構成される）とされる。この時点で、2 0 0 0グループ以下となるので、ステップS 1 1 4の処理で、最終的に求めたいグループ数に達したと判断され、ユーザグループの生成処理は終了される。

【0 2 0 0】

ステップS 1 1 3とステップS 1 1 4における処理は、所定のグループ数以下にユーザグループを設定するために行われる処理であり、上述したような、存在するグループ数を

10

20

30

40

50

半分のグループ数にするという処理を繰り返す方法に限定されるものではない。例えば、全ユーザ数（ステップS 1 1 1において作成された全グループ数）を、目標としているグループ数で除算し、その結果得られた数値を、1グループの人数として設定し、相関度の高い順に、その設定された人数毎に、グループを設定し直すようにしても良い。

【0201】

例えば、1万のグループが存在し、2千のグループ数以下にしたい場合、まず、1万を2千で除算し、5という結果を得る。よって、1グループを5人と設定し、5人で構成されるグループを、相関度の高い順に作成すれば、2千グループが生成されることになる。

【0202】

ユーザグループをどのような方法を用いて生成しても良いが、相関度の近いユーザ同士が、1つのグループとされ、グループ数の総数が、所定の数以下になれば良い。このグループの最終的な数は、システムへの負荷などを考慮して設定されればよい。

【0203】

このようにして生成されたグループ毎に、予測ベクトルが、ステップS 1 0 3において算出される。このステップS 1 0 3における処理は、基本的に、ステップS 9 2（図4 1）の処理と同様であるが、ステップS 9 2においては、例えば、ユーザXとユーザYというように、所定のユーザと他のユーザとの間の予測ベクトルが求められる処理であった。

【0204】

これに対し、ステップS 1 0 3においては、例えば、グループXとグループYというように、所定のグループと他のグループとの間の予測ベクトルが求められる。このようにした場合、例えば、ユーザXの予測ベクトルは、そのユーザXが属するグループXとグループYの予測ベクトルとされる。

【0205】

または、ステップS 1 0 3においては、例えば、グループXに属するユーザXと、そのユーザXが属していない他のグループYというように、所定のユーザと所定のグループとの間の予測ベクトルが求められるようにしても良い。

【0206】

いずれの方法によっても、グループという概念を用いて予測ベクトルを算出する方が、ユーザ同士で予測ベクトルを求める方法よりも、その算出にかかる処理を軽くすることができるため、算出にかかる時間を短縮し、システムにかかる負荷を低減させることができる。

【0207】

図4 4は、他のフィルタリング処理について説明するためのフローチャートである。図4 4に示したフローチャートの処理も、グループ化を前提としているが、相関度を用いない点が、図4 2に示したフィルタリング処理と異なる。ステップS 1 3 1において、ユーザグループの生成が行われる。このステップS 1 3 1におけるユーザグループの生成の処理は、例えば、ログデータベース2 5 3（図1 1）が参照されて行われる。ログデータベースが参照され、近々に同じ情報を利用したユーザ同士がグループ化される。このようにすれば、このフィルタリング処理が実行されている時点で、同じような興味を持っているユーザ同士をグループ化できると考えられる。

【0208】

この際、1グループに属する人数などは、予め設定されている、または、上述したような方法（例えば、全ユーザ数を、目標とするグループ数で除算する）により設定される。その設定されている人数で1グループが構成されるように、グループ化の処理が実行される。なお、このグループ化の処理は、管理者が手作業により行うことも当然可能である。

【0209】

このようなグループ化（ユーザグループの生成）が終了されると、ステップS 1 3 2において予測ベクトルが算出される。この予測ベクトルの算出は、ステップS 1 0 3（図4 2）における処理と、基本的に同様に行われるので、ここでは、その説明を省略する。この場合、グループに属するユーザの各Valueに対する平均値が、そのグループに属する全

10

20

30

40

50

ユーザに対する予測ベクトルとして設定される。

【0210】

グループ化や、予測ベクトルの算出を、必要がある時に、その時点で行うのではなく、事前に行い、記憶させるようにした場合、その記憶されているデータを再度算出し直すといったようなことはせず、新たにログが生成されたユーザに対してのみ（そのユーザのデータに対してのみ）、処理を実行するようにしても良い。このようにすれば、予測ベクトルを算出するために要する時間を短縮することができる。また、サーバ52への負担を軽くすることができる。

【0211】

図41乃至図44に示した各フローチャートの処理のうち、いずれの処理に基づいて予測ベクトルが求められても良い。 10

【0212】

なお、上述したフィルタリングの処理において、計算に利用する相関度（ $S i m_{X N}$ ）に関して、所定の値以上の相関度を持つユーザのみ計算に用いる（閾値を設定し、その閾値以上の相関度を有するユーザのみを計算対象のユーザに設定する）ようにしても良い。このようにすれば、予測ベクトルの算出にかかる時間を短縮することができ、また、算出のためにかかる負荷を軽減させることができる。

【0213】

また、予測ベクトルが所定のユーザに適用される際、そのユーザが有していない全てのValueに関する要素を、T_UM_VALUEデータベース317に登録するのではなく、上位の所 20
定数のみを登録するようにしても良い。このように、登録される要素に制限を加えることにより、データベースのリソースの節約、計算速度の向上をはかることが可能となる。

【0214】

仮に、予測ベクトルの情報のみを用いて、ユーザにオススメの番組を推薦するような場合、ユーザが既に有しているValueに関しても予測ベクトルを求め、オススメ番組の検索などに用いることができる。ユーザの現在のプロフィール（学習や登録により求められたプロフィール）が、“本来なら、このようになるのではないか”、“将来的には、こういう風になるはず”といったような観点からのオススメが可能となる。

【0215】

このように、本実施の形態においては、ユーザ側に推薦する番組を検索する際の情報（ 30
T_UM_VALUEデータベース317のデータ（要素））は、登録、学習、フィルタリングといった、それぞれの処理が総合的に実行されることにより作成される。

【0216】

次に、作成されたこれらのデータベースを用いて、推薦する番組の検索（オススメ画面の作成）について説明する。図45のフローチャートを参照し、所定のユーザに対して推薦する番組を検索し、その検索された番組の情報を取得するまでの処理について説明する。

【0217】

ステップS161において、ユーザがログインしてくるため、そのログインに対応するための処理が実行される。このログインに関わる処理は、ユーザ管理機能232（図10 40
）において行われる。ユーザは、ユーザ端末53（図3）を用いて、自己のユーザIDとパスワードをサーバ52に対して送信する。その送信されたユーザIDとパスワードは、サーバ52の情報通信機能236（図10）の制御により受信され、ユーザ管理機能232に供給される。ユーザ管理機能232は、供給されたユーザのユーザIDとパスワードの組が、ユーザデータベース254に記憶されているか否かを判断する。

【0218】

供給されたユーザIDとパスワードの組が、ユーザデータベース254に記憶されていると判断された場合、ログインの処理は正常に行われることになり、ステップS162に処理が進められるが、ログインの処理が正常に行われなかった場合は、ステップS162に処理が進められることなく、エラー処理、例えば、ログインを要求してきたユーザに対し 50

て、ログインが正常に行われなかったことを示すメッセージを送付するなどの処理が実行され、図 45 に示したフローチャートの処理は終了される。

【0219】

ステップ S 1 6 2 において、対象ユーザモデルが取得される。対象ユーザモデルの取得とは、T_UM_VALUEデータベース 3 1 7 から、ログインしてきたユーザに関わる情報が読み出されることを意味する。ステップ S 1 6 1 におけるログインの処理で、ログインしてきたユーザが判別されているため、MemberIDを取得することができる状態とされている。そこで、ログインしてきたユーザのMemberIDを、ユーザデータベース 2 5 4 から読み出し、そのMemberIDに対応するデータが、T_UM_VALUEデータベース 3 1 7 から読み出される。

【0220】

このステップ S 1 6 2 以降の処理は、情報選別機能 2 3 3 により行われる。ログインの処理を実行するのは、上述したようにユーザ管理機能 2 3 2 であるが、ユーザ管理機能 2 3 2 は、ログインの処理を実行した結果、取得したMemberIDを、情報選別機能 2 3 3 に供給する。情報選別機能 2 3 3 は、供給されたMemberIDを用いて、ステップ S 1 6 2 における処理を実行する。

【0221】

情報選別機能 2 3 3 は、ステップ S 1 6 2 の処理を実行すると共に、並列的に、ステップ S 1 6 3 の処理も実行する。ステップ S 1 6 3 において、ユーザ情報から、対象情報を絞る必要があるか否かが判断される。“ユーザ情報”とは、ここでは、ユーザが住んでいる地域、ユーザが有する視聴権限、ユーザの年齢や性別などである。また、“対象情報を絞る必要があるか否か”とは、例えば、一部地域では放送されないような番組が存在する
20

【0222】

視聴権限により対象情報を絞る必要がある場合、例えば、有料放送を視聴する権限があるか否か、そのような権限があると判断されたときであっても、どのような範囲で権限を有しているのか等が判断され、対象情報の絞り込みが行われる。また、年齢により対象情報を絞る必要がある場合、例えば、成人向けの番組が、成人でないユーザに対してはオス
30

【0223】

性別により対象情報を絞る必要がある場合、例えば、男性向けの番組を女性に対してオス
30

【0224】

ステップ S 1 6 3 において、ユーザ情報から対象情報を絞る必要があると判断された場合、ステップ S 1 6 4 に処理が進められ、対象情報を絞る必要がないと判断された場合、
40

【0225】

一方、ステップ S 1 6 4 の処理がスキップされてステップ S 1 6 5 に処理が進められた場合は、情報データを選別する必要がないので、全ての番組が処理対象とされる。ステップ S 1 6 5 において、処理対象とされている番組の情報メタデータが取得される。この情報メタデータは、T_PRG_VALUEデータベース 3 1 6 (図 2 9) で管理されているデータのことである。

【0226】

10

20

30

40

50

ステップ S 1 6 5 において、処理対象とされた番組の情報メタデータが取得されると、ステップ S 1 6 6 において、推薦対象情報の得点計算および選別の処理が実行される。この処理は、ステップ S 1 6 2 で取得された対象ユーザモデル (T_UM_VALUE データベース 3 1 7) の Score と、ステップ S 1 6 5 において取得された情報メタデータ (T_PRG_VALUE データベース 3 1 6) の Score が用いられて行われる。例えば、ベクトル演算などの得点計算方法により値が算出され、その値を元に、番組のランキングが付けられる。この処理により演算された得点は、例えば、図 8 に示したオススメ画面内のオススメ度表示部 1 5 6 に表示される得点とされる。

【 0 2 2 7 】

このようにして、各番組の得点が演算され、その得点の高い順にランキング付けが行われると、ステップ S 1 6 7 に処理が進められる。ステップ S 1 6 7 において、推薦対象とされた番組の情報の取得が行われる。推薦対象とされた番組とは、ランキングの上位に位置する、所定数の番組 (例えば、10 番組) である。その所定数の番組の情報が取得されるわけだが、この取得は、情報選別機能 2 3 3 により行われる。情報選別機能 2 3 3 は、ランキングの上位に位置する番組の識別子 (ProgramID) を取得する。

【 0 2 2 8 】

その ProgramID を元に、出演者 (PersonID) やキーワード (KeywordID) といった番組に関する情報を取得する。このようにして、複数の番組の情報が順次取得される。

【 0 2 2 9 】

ユーザにオススメとして推薦される複数の番組の情報が取得された後の処理について、図 4 6 のフローチャートを参照して説明する。ステップ S 1 8 1 において、推薦情報の取得が行われる。この処理は、ステップ S 1 6 7 (図 4 5) のことである。ステップ S 1 8 2 において、推薦情報の推薦度を算出する際に、算出対象とされた Value (ValueID により識別される要素) に関して、各 Value 毎のスコアでソート処理が実行される。

【 0 2 3 0 】

ステップ S 1 8 3 において、ソートされた上位の Value に対して、その Value に関連付けられている AttributeID と ValueID の組を用いて、M_REASON データベース 2 8 0 (図 2 1) から、オススメ理由が取得される。例えば、ソートされた結果、上位の Value とされた AttributeID が “ 0 0 1 ”、ValueID が “ 1 0 0 0 ” であった場合、“スポーツ”というオススメ理由が取得される。

【 0 2 3 1 】

ステップ S 1 8 4 において、ReasonType の表示上限個数と、ReasonType 毎に指定された条件を満たしているか否かが判断される。ここで、ReasonType に関わる事項について説明する。ReasonType とは、図 8 に示したオススメ画面のうち、理由表示部 1 5 5 に表示される理由 (Reason) の個数などを制限するために設定されているものである。図 4 7 に ReasonType を設定するためのデータを示し、図 4 8 に各 ReasonType 毎の表示個数を設定するためのデータを示す。図 4 7 や図 4 8 に示した表示個数を制御するためのデータに基づく処理は、例えば、情報選別機能 2 3 3 の 1 機能として組み込まれ、実行される。

【 0 2 3 2 】

図 4 7 に示したデータのうち、1 行目は、ReasonType を指定するデータであることを宣言するため、そして、2 行目以降のデータ形式を規定するためのデータである。2 行目以降のデータにおいて、“Normal” は、UMType を示し、図 3 0 に示した T_UM_VALUE データベース 3 1 7 の説明においては、UMTypeID = “ 2 ” と記載した事項と同じ意味合いを有する。すなわち、“Normal” は、学習により登録された要素であることを示す。

【 0 2 3 3 】

“Prior” は、図 3 0 に示した T_UM_VALUE データベース 3 1 7 の説明においては、UMTypeID = “ 1 ” と記載した事項と同じ意味合いを有する。すなわち、“Prior” は、ユーザ自身が登録した要素であることを示す。さらに、“Expected” は、図 3 0 に示した T_UM_VALUE データベース 3 1 7 の説明においては、UMTypeID = “ 3 ” と記載した事項と同じ意味合いを有する。すなわち、“Expected” は、フィルタリング処理により登録された要素を意

味する。

【 0 2 3 4 】

2 行目以降は、同一の形態でデータが記載されているので、2 行目を例に挙げてそのデータについて説明する。2 行目には、“ ReasonType . Normal . Category = 4 ”との記載がある。まず、“ ReasonType ”は、その行のデータがReasonTypeを設定するものであることを宣言している。次の“ Normal ”は、UMTypeを示している。そして、“ Category = 4 ”は、“ Category ”は、“ 4 ”というReasonTypeに属することを示している。まとめると、2 行目の記載が意味するものは、この場合、学習により登録され、Attributeとして“ Category ”に属するものは、ReasonTypeが“ 4 ”と設定されることを示している。

【 0 2 3 5 】

10

図 4 8 は、このようなReasonType毎のReasonType表示個数を設定するためのデータである。2 行目の“ # [0] 登録キーワード、人物（合計 5 件以内） ”という記載を例に挙げて、図 4 8 に示したデータについて説明する。“ # [0] ”は、ReasonTypeが“ 0 ”の内容を示すデータが、2 行目には記載されていることを示している。“ # [0] ”に続く“ 登録キーワード、人物（合計 5 件以内） ”という記載は、“ 登録キーワード、人物 ”という項目が、理由表示部 1 5 5 に表示される場合、この項目に対応するキーワードや人物は、合わせて 5 件以内しか表示してはいけないという規定である。換言すれば、この場合、Attributeが“ Keyword ”や“ Person ”であるReason (図 2 1) は、合計 5 件までしか、この項目の欄には表示させてはいけないという制限が規定されている。

【 0 2 3 6 】

20

図 4 7 と図 4 8 に示したデータにより、図 8 に示したオススメ画面内の理由表示部 1 5 5 の表示に制御が加えられることになる。ここで、オススメ画面の理由表示部 1 5 5 の表示とあわせて、図 4 7 と図 4 8 に示したデータについての説明をさらに加える。まず、ステップ S 1 8 3 における処理において、AttributeIDとして“ 0 0 1 ”、ValueIDとして“ 1 0 0 0 ”が取得されたとする。AttributeID“ 0 0 1 ”、ValueID“ 1 0 0 0 ”の組をM_REASONデータベース 2 8 0 (図 2 1) を参照すると、“ スポーツ ”という言葉が関連付けられているため、オススメ理由 (Reason) として、“ スポーツ ”という言葉がユーザ側に提示されるとして決定される。

【 0 2 3 7 】

さらに、AttributeID“ 0 0 1 ”、ValueID“ 1 0 0 0 ”の組を、T_UM_VALUEデータベース 3 1 7 (図 3 0) を参照して検出すると、UMTypeIDが“ 1 ”であることが分かる。UMTypeID“ 1 ”は、UMTypeとしては“ Prior ”である。また、AttributeID“ 0 0 1 ”は、M_CATEGORYデータベース 2 7 1 (図 1 2) に割り振られたIDであるため、Attributeとしては“ Category ”であることが分かる。ここまでの処理で、オススメ理由として表示させようとしている“ スポーツ ”という言葉は、“ Prior ”であり、“ Category ”であることが分かる。

30

【 0 2 3 8 】

図 4 7 を参照するに、“ Prior ”、“ Category ”の組み合わせは、1 2 行目に存在する。1 2 行目の記載より、ReasonTypeが“ 1 ”であることが分かる。そこで、図 4 8 のデータを参照するに、ReasonType = 1 については、4 , 5 行目に記載がある。4 行目の記載より、“ 登録ジャンル ”という項目の欄に表示されることがわかる。また、この項目には、1 件しか表示してはいけないという規定になっていることが分かる。

40

【 0 2 3 9 】

結果として、この場合、理由表示部 1 5 5 には、
登録ジャンル：「スポーツ」
といった表示がされることになる。

【 0 2 4 0 】

なお、このような処理が行われた後に、“ Prior ”、“ Category ”、および、ReasonType“ 1 ”のValueがステップ S 1 8 3 で取得された場合、既に、“ スポーツ ”という言葉が割り当てられているので、その取得されたValueは、処理対象とはならず破棄される。ま

50

た、ステップ S 1 8 2 において、ソートをかけているため、破棄される Value は、常に、既に処理済みの Value よりも低い値を有するものであるため、結果として、ユーザに提示されるのは、常に、Value が高いものだけになる。

【 0 2 4 1 】

この説明は、ステップ S 1 8 3 とステップ S 1 8 4 の処理についての説明であるが、実際に、上述したような順番で、データベースなどが参照されつつ行われる必要はない。すなわち、ステップ S 1 8 3 の処理が実行される前の段階で、例えば、UMTypeIDなどは判断されているため、または、判断されるようにすれば、ステップ S 1 8 3 , S 1 8 4 の処理において T_UM_VALUE データベース 3 1 7 を再度参照する必要性はない。よって、適宜、処理の重複がないように、各処理が実行されれば良く、どのタイミングで、どのようなデータベースが参照されるか等は上述した説明に限定されるものではない。

10

【 0 2 4 2 】

このように図 4 7 と図 4 8 にそれぞれ示したデータに基づく制御により、ステップ S 1 8 4 (図 4 6) の判断が行われる。ステップ S 1 8 4 において、ReasonType の表示上限個数 (図 4 7 に示したデータにより制限が加えられる) と、ReasonType 毎に指定されている条件 (図 4 8 に示したデータにより制限が加えられる) を満たしていると判断されるまで、ステップ S 1 8 3 とステップ S 1 8 4 の処理が繰り返され、満たしていると判断されると、図 4 6 に示したフローチャートの処理は終了される。

【 0 2 4 3 】

すなわち、図 4 6 に示したフローチャートの処理が終了されると、ユーザ側にオススメ画面を提示するためのデータがそろったことになるので、そのそろえられたデータが、ユーザ側にユーザ端末 5 3 に送信される。

20

【 0 2 4 4 】

ここで、さらに、ステップ S 1 8 3 における処理について説明を加える。ステップ S 1 8 3 においては、M_REASON データベース 2 8 0 が参照される。例えば、AttributeID として、“ 0 0 5 ” が、ValueID として “ 0000000006 ” が取得された場合を考える。この AttributeID と ValueID の組み合わせと関連付けられているのは、図 1 6 の M_KEYWORD データベース 2 7 5 を参照すると、“ エロティック ” という言葉であることが分かる。しかしながら、既に説明したように、この “ エロティック ” という言葉は、ユーザ側に提示する言葉としてはふさわしくないと判断されているため、M_REASON データベース 2 8 0 (図 2 1) に

30

【 0 2 4 5 】

よって、ステップ S 1 8 3 における処理で、AttributeID “ 0 0 5 ”、ValueID “ 0000000006 ” の組が取得されたとしても、オススメ理由として M_REASON データベース 2 8 0 から読み出されるデータはないことになる。このような場合、ステップ S 1 8 4 の処理を行う必要がないので、行わないで、次の要素 (Value) に対してステップ S 1 8 3 の処理が実行されるようにしても良い。いずれにしても、M_REASON データベース 2 8 0 に記載がない Reason は、ユーザ側には提示されないという制御が行われることになる。

【 0 2 4 6 】

図 4 6 に示したフローチャートの処理では、図 2 1 に示したような M_REASON データベース 2 8 0 が参照されるとして説明した。しかしながら、図 2 1 に示したような M_REASON データベース 2 8 0 を有してなく、図 1 7 に示した M_KEYWORD_1 データベース 2 7 6 を有しているような場合でも、図 4 9 のフローチャートの処理に従えば、基本的に、図 4 6 に示した処理を実行したときと同様の処理を行うことができる。

40

【 0 2 4 7 】

図 4 9 におけるステップ S 2 0 1、ステップ S 2 0 2、および、ステップ S 2 0 4 の処理は、図 4 6 のステップ S 1 8 1、ステップ S 1 8 2、および、ステップ S 1 8 4 の処理と基本的に同様であるので、その説明は省略する。ステップ S 2 0 3 において、ソートされた上位の Value に対して、当該マスターテーブルが参照され、当該 ValueID が用いられて、オススメ理由 (Reason) が取得される。この処理も、基本的には、図 4 6 に示したフロ

50

ーチャートのステップS 1 8 3の処理と同様であるが、参照されるデータベースが異なる。

【0 2 4 8】

ステップS 2 0 3においては、M_KEYWORD_1データベース2 7 6が参照される。例えば、ValueID(この場合、KeywordIDになる)が“0000000008”の場合、M_KEYWORD_1データベース2 7 6内のReasonの欄が参照され、“お得な情報”というのが読み出される。また、例えば、ValueIDが“0000000006”の場合、M_KEYWORD_1データベース2 7 6内のReasonの欄が参照されるが、空欄(NULL)とされているため、読み出されるデータなく、このValueID“0000000006”に対する処理は終了され、次のValueIDに対する処理に移行される。

【0 2 4 9】

このように、M_KEYWORD_1データベース2 7 6を用い、図4 9に示したフローチャートの処理を実行した場合も、ユーザに提示される言葉として不適切な言葉は、提示されない、または、他の言葉に変換されて提示されるように制御することが可能である。

【0 2 5 0】

M_KEYWORD_1データベース2 7 6の代わりに、M_KEYWORD_2データベース2 7 7(図1 8)が備えられている場合、図5 0のフローチャートに基づく処理が実行される。この場合、ステップS 2 1 1、ステップS 2 1 2、および、ステップS 2 1 4は、図4 6のステップS 1 8 1、ステップS 1 8 2、および、ステップS 1 8 4の処理と基本的に同様であるので、その説明は省略する。

【0 2 5 1】

ステップS 2 1 3においては、M_KEYWORD_2データベース2 7 7が参照される。例えば、ValueID(この場合、KeywordIDになる)が“0000000001”の場合、M_KEYWORD_2データベース2 7 7内のKeywordの欄が参照され、“情報”というのが読み出され、Displayの欄の“1”という値が読み出される。このDisplayの欄の“1”という値(フラグ)は、ユーザ側に提示されるとして設定されていることを示す。

【0 2 5 2】

例えば、ValueID(この場合、KeywordIDになる)が“0000000008”の場合、M_KEYWORD_2データベース2 7 7内のKeywordの欄が参照され、“得”というのが読み出され、Displayの欄の“0”という値が読み出される。このDisplayの欄の“0”という値(フラグ)は、ユーザ側に提示されないとして設定されていることを示す。よって、結果としてユーザ側には提示されないので、このような処理を行う必要もなく、Displayの欄が“0”のときには、“0”と判断された時点で、処理は、次のValueIDの要素へと移行される。

【0 2 5 3】

このように、M_KEYWORD_1データベース2 7 6を用い、図5 0に示したフローチャートの処理を実行した場合も、ユーザに提示される言葉として不適切な言葉は、提示されないように制御することが可能である。しかしながら、図1 8に示したM_KEYWORD_2データベース2 7 7を用いた処理では、ユーザに提示するにはふさわしくない言葉を、他の言葉に変換して提示するといった処理はできない。よって、他の言葉に変換しなくても良いようなシステムにおいては、有効な処理の仕方である。

【0 2 5 4】

このように、本発明においては、ユーザ側に提示する言葉を制御することができる。本実施の形態においては、主に、番組を推薦する場合を例に挙げて説明したが、番組以外の情報を推薦するような場合も本発明を適用することはできる。所定の情報をユーザに推薦する際、なぜその情報が推薦されたのか、その根拠をユーザ側に提示することができ、また、その提示される内容は、ユーザが認識しやすい言葉で提示されるので、ユーザは、提示される内容を理解しやすくくなり、提示される内容に信頼性をもつことができるようになる。

【0 2 5 5】

また、本発明によれば、その提示される内容は、ユーザの嗜好を良く考慮したものとすることができる。その理由は、上述したように、ユーザの嗜好に関するデータを、登録、

10

20

30

40

50

学習、フィルタリングといった複数の処理を総合的に行うようにしたので、より、詳細に、ユーザの嗜好を解析することができるようになる。

【0256】

上述した一連の処理は、それぞれの機能を有するハードウェアにより実行させることもできるが、ソフトウェアにより実行させることもできる。一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行することが可能な、例えば汎用のパーソナルコンピュータなどに、記録媒体からインストールされる。

【0257】

記録媒体は、図4に示すように、パーソナルコンピュータとは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク91（フレキシブルディスクを含む）、光ディスク92（CD-ROM（Compact Disc-Read Only Memory）、DVD（Digital Versatile Disc）を含む）、光磁気ディスク93（MD（Mini-Disc）（登録商標）を含む）、若しくは半導体メモリ94などよりなるパッケージメディアにより構成されるだけでなく、コンピュータに予め組み込まれた状態でユーザに提供される、プログラムが記憶されているROM72や記憶部78が含まれるハードディスクなどで構成される。

【0258】

なお、本明細書において、媒体により提供されるプログラムを記述するステップは、記載された順序に従って、時系列的に行われる処理は勿論、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理をも含むものである。

【0259】

また、本明細書において、システムとは、複数の装置により構成される装置全体を表すものである。

【図面の簡単な説明】

【0260】

【図1】従来のEPG画面について説明するための図である。

【図2】従来の番組のオススメ画面について説明するための図である。

【図3】本発明を適用したシステムの一実施の形態の構成を示す図である。

【図4】ユーザ端末の内部構成例を示す図である。

【図5】ユーザ端末の基本的な処理を説明するためのフローチャートである。

【図6】EPGの画面例を示す図である。

【図7】オススメ画面の一例を示す図である。

【図8】オススメ画面の他の例を示す図である。

【図9】サーバの内部構成例を示す図である。

【図10】サーバの機能について説明する機能ブロック図である。

【図11】記憶部に記憶されているデータベースについて説明するための図である。

【図12】M_CATEGORYデータベースについて説明するための図である。

【図13】M_PERSONデータベースについて説明するための図である。

【図14】M_PERSON_CHARACTERデータベースについて説明するための図である。

【図15】M_VIEW_TYPEデータベースについて説明するための図である。

【図16】M_KEYWORDデータベースについて説明するための図である。

【図17】M_KEYWORD_1データベースについて説明するための図である。

【図18】M_KEYWORD_2データベースについて説明するための図である。

【図19】M_AGEデータベースについて説明するための図である。

【図20】M_GENDERデータベースについて説明するための図である。

【図21】M_REASONデータベースについて説明するための図である。

【図22】マスターデータベースから情報を検出する際の処理について説明するための図である。

10

20

30

40

50

【図 2 3】マスターデータベースから情報を検出する際の処理について説明するための図である。

【図 2 4】T_PERSON_TARGET_GENDERデータベースについて説明するための図である。

【図 2 5】T_PERSON_TARGET_AGEデータベースについて説明するための図である。

【図 2 6】T_PERSON_TARGET_GENDERデータベースについて説明するための図である。

【図 2 7】T_PERSON_VIEW_TYPEデータベースについて説明するための図である。

【図 2 8】T_PERSON_CHARACTERデータベースについて説明するための図である。

【図 2 9】T_PRG_VALUEデータベースについて説明するための図である。

【図 3 0】T_UM_VALUEデータベースについて説明するための図である。

【図 3 1】T_PRG_VALUEデータベースを作成する際の処理について説明するためのフローチャートである。 10

【図 3 2】表示される情報と、その情報に関するデータの関係について説明するための図である。

【図 3 3】T_UM_VALUEデータベースを作成する際の処理について説明するためのフローチャートである。

【図 3 4】T_UM_VALUEデータベースを作成する際の他の処理について説明するためのフローチャートである。

【図 3 5】Scoreの変更に関わる処理について説明するための図である。

【図 3 6】要素の追加に関する処理について説明するための図である。

【図 3 7】学習処理をした場合と、学習処理をしない場合とを比較するためのグラフである。 20

【図 3 8】フィルタリング処理を行う理由について説明するための図である。

【図 3 9】他のユーザの履歴から所定のコンテンツを推薦する場合の処理について説明するための図である。

【図 4 0】他のユーザの履歴から所定のコンテンツを推薦する場合の処理について説明するための図である。

【図 4 1】フィルタリング処理について説明するためのフローチャートである。

【図 4 2】フィルタリング処理について説明するためのフローチャートである。

【図 4 3】ユーザグループの作成について説明するためのフローチャートである。

【図 4 4】フィルタリング処理について説明するためのフローチャートである。 30

【図 4 5】推薦する番組の検索に関わる処理について説明するためのフローチャートである。

【図 4 6】推薦理由を取得する際の処理について説明するためのフローチャートである。

【図 4 7】ReasonTypeに関わる処理を実行するデータについて説明するための図である。

【図 4 8】ReasonTypeに関わる処理を実行するデータについて説明するための図である。

【図 4 9】推薦理由を取得する際の処理について説明するためのフローチャートである。

【図 5 0】推薦理由を取得する際の処理について説明するためのフローチャートである。

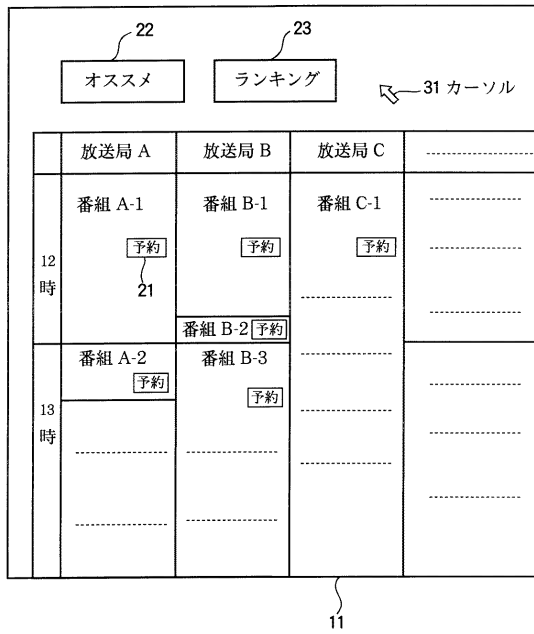
【符号の説明】

【0 2 6 1】

5 1 ネットワーク, 5 2 サーバ, 5 3 ユーザ端末, 5 4 放送局, 1 0 40
1 ディスプレイ, 1 1 2 オススメマーク, 1 1 3 オススメボタン, 1 1 4
ランキングボタン, 1 5 1 番組表示部, 1 5 2 放送時間表示部, 1 5 3 内容
表示部, 1 5 4 出演者表示部, 1 5 5 理由表示部, 1 5 6 オススメ度表示部
, 2 3 1 ログ生成機能, 2 3 2 ユーザ管理機能, 2 3 3 情報選別機能, 2
3 4 サービス管理機能, 2 3 5 ユーザモデル管理機能, 2 3 6 情報通信機能,
2 3 7 対象情報メタデータ管理機能, 2 3 8 サービス一般機能, 2 5 1 マス
ターデータベース, 2 5 2 ターゲットデータベース, 2 5 3 ログデータベース,
2 5 4 ユーザデータベース

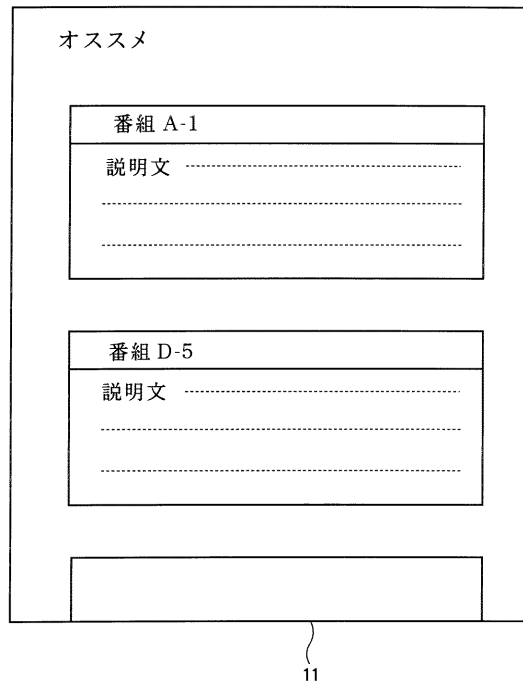
【図 1】

図 1



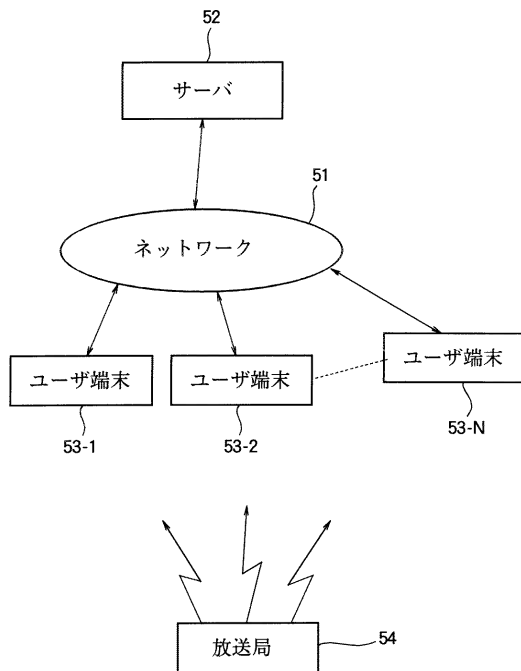
【図 2】

図 2



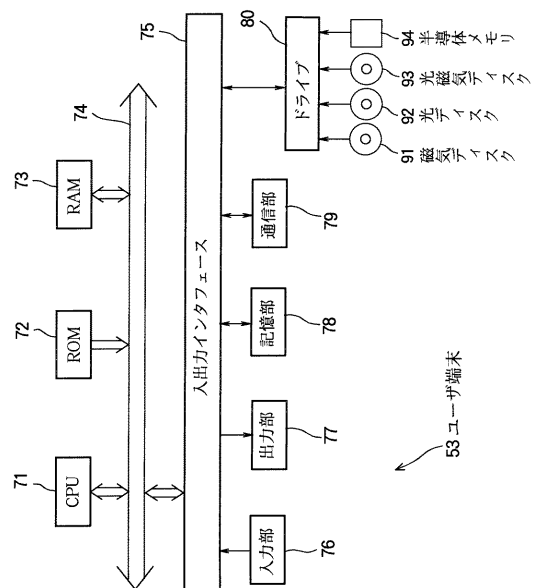
【図 3】

図 3



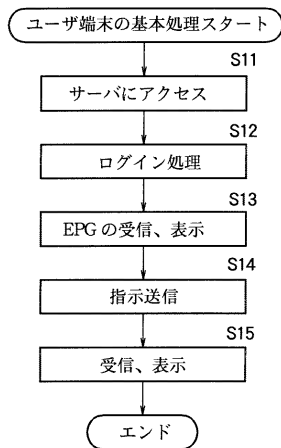
【図 4】

図 4



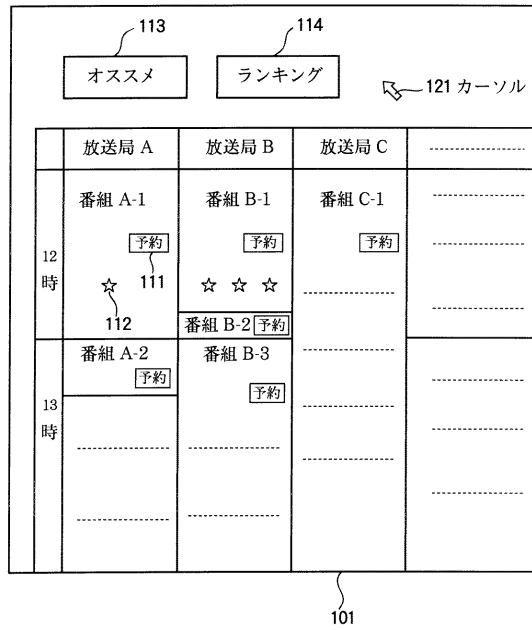
【 図 5 】

図 5



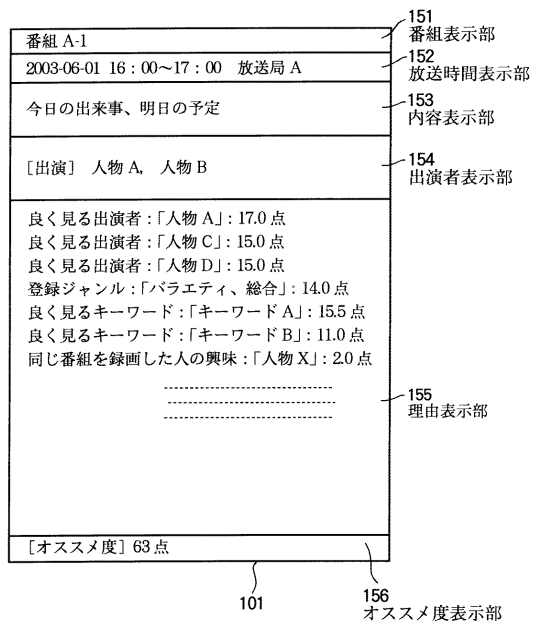
【 図 6 】

図 6



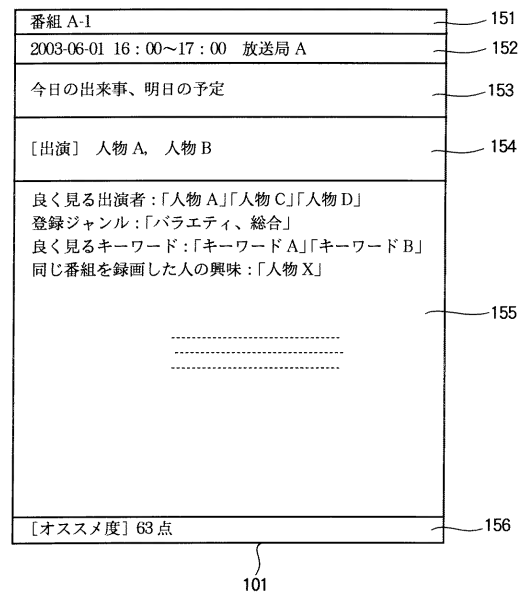
【 図 7 】

図 7



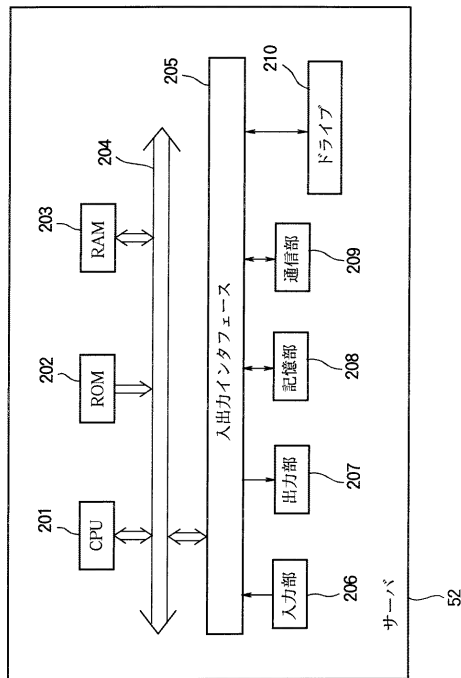
【 図 8 】

図 8



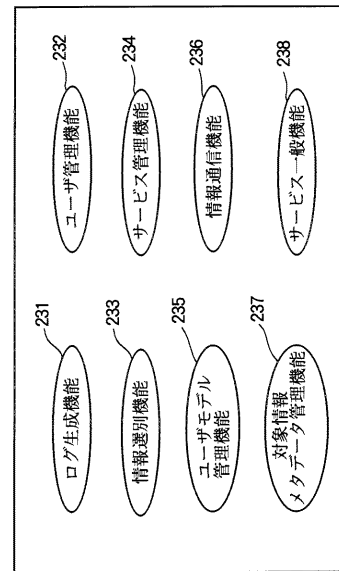
【図 9】

図 9



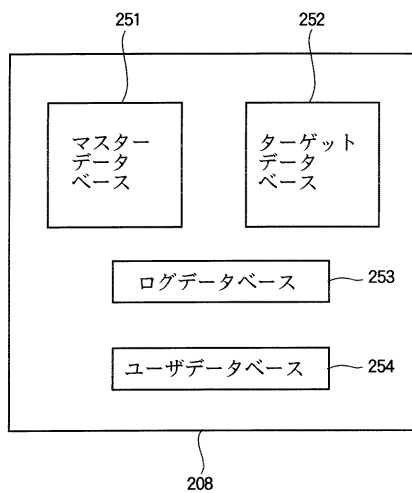
【図 10】

図 10



【図 11】

図 11



【図 12】

図 12

M_CATEGORY		001
category ID	category	
1000	スポーツ	
1001	スポーツ：野球	
1002	スポーツ：サッカー	
1003	スポーツ：ゴルフ	

271

【図 1 3】

図 13

M_PERSON			002
Person ID	Name	KanaName	
0000000001	人物 A	ジンプツエー	
0000000002	人物 B	ジンプツビー	
0000000003	人物 C	ジンプツシー	
0000000004	人物 D	ジンプツデー	
0000000005	人物 E	ジンプツイー	
0000000006	人物 F	ジンプツエフ	
0000000007	人物 G	ジンプツジー	
0000000008	人物 H	ジンプツエイチ	

272

【図 1 4】

図 14

M_PERSON_CHARACTER		003
CharacterID	Character	
001	司会者系	
002	恋愛	
003	知的	
004	身近	
005	情熱的	
006	クール	

273

【図 1 5】

図 15

M_VIEW_TYPE		004
ViewID	ViewType	
001	笑い刺激型	
002	熱中トレンド型	
003	ロマン・フィクション型	
004	息抜きザッピング	

274

【図 1 6】

図 16

M_KEYWORD		005
KeywordID	Keyword	
0000000001	情報	
0000000002	買物	
0000000003	テレビ	
0000000004	ニュース	
0000000005	ショッピング	
0000000006	エロティック	
0000000007	スポーツ	
0000000008	得	
0000000009	スーパー	
0000000010	旅	

275

【図 17】

図 17

M_KEYWORD_1		005
KeywordID	Keyword	Reason
0000000001	情報	情報
0000000002	買物	買物
0000000003	テレビ	テレビ
0000000004	ニュース	ニュース
0000000005	ショッピング	ショッピング
0000000006	エロティック	
0000000007	スポーツ	スポーツ
0000000008	得	お得な情報
0000000009	スーパー	スーパー
0000000010	旅	旅

276

【図 18】

図 18

M_KEYWORD_2		005
KeywordID	Keyword	Display
0000000001	情報	1
0000000002	買物	1
0000000003	テレビ	1
0000000004	ニュース	1
0000000005	ショッピング	1
0000000006	エロティック	0
0000000007	スポーツ	1
0000000008	得	0
0000000009	スーパー	1
0000000010	旅	1

277

【図 19】

図 19

M_AGE			006
AgeID	AgeName	StartAge	EndAge
001	0 から 4	0	4
002	5 から 9	5	9
003	10 から 14	10	14
004	15 から 19	15	19
005	20 から 24	20	24
006	25 から 29	25	29
007	30 から 34	30	34
008	35 から 39	35	39
009	40 から 44	40	44
010	45 から 49	45	49
011	50 から 54	50	54
012	55 から 59	55	59
013	60 から	60	100

278

【図 20】

図 20

M_GENDER		007
GenderID	GenderType	
001	男性	
002	女性	
003	男女共	

279

【図 2 1】

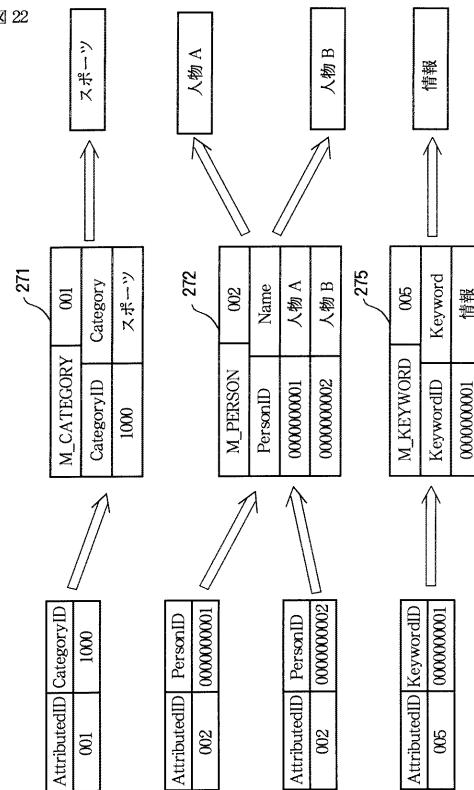
図 21

M_REASON		
Attributeld	ValueId	Reason
001	1000	スポーツ
001	1001	スポーツ：野球
001	1002	スポーツ：サッカー
001	1003	スポーツ：ゴルフ
002	0000000001	人物 A
002	0000000002	人物 B
002	0000000003	人物 C
002	0000000004	人物 D
002	0000000005	人物 E
002	0000000006	人物 F
002	0000000007	人物 G
002	0000000008	人物 H
003	001	司会者系
003	002	恋愛
003	003	知的
003	004	身近
003	005	情熱的
003	006	クール
004	001	笑い刺激型
004	002	熱中トレンド型
004	003	ロマン・フィクション型
004	004	息抜きザッピング
006	0000000001	情報
006	0000000002	買い物
006	0000000003	テレビ
006	0000000004	ニュース
006	0000000005	ショッピング
006	0000000007	スポーツ
006	0000000008	お得な情報
006	0000000009	スーパー
006	0000000010	旅

280

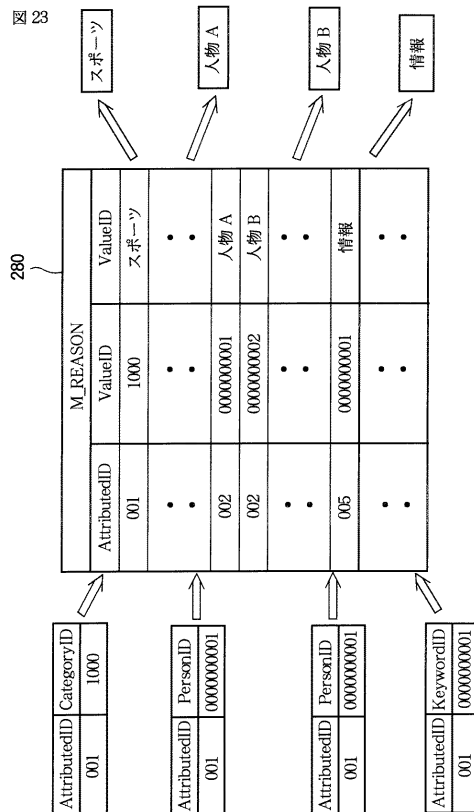
【図 2 2】

図 22



【図 2 3】

図 23



【図 2 4】

図 24

T_PERSON_TARGET_GENDER		
PersonID	GenderID	Score
000000001	001	.3333333333333331483
000000001	003	.6666666666666662966
000000002	001	.3333333333333331483
000000002	002	.6666666666666662966
000000003	002	.3333333333333331483
000000004	003	.3333333333333331483
000000005	001	.3333333333333331483
000000006	003	.3333333333333331483
000000007	001	.3333333333333331483
000000008	003	.6666666666666662966

311

【図 25】

図 25

T_PERSON_TARGET_AGE		
PersonID	AgeID	Score
000000001	005	.66666666666666662966
000000002	006	.66666666666666662966
000000002	007	.66666666666666662966
000000002	013	.66666666666666662966
000000003	007	.33333333333333331483
000000003	008	.33333333333333331483
000000004	002	.33333333333333331483
000000004	003	.33333333333333331483
000000004	004	.33333333333333331483
000000005	011	.66666666666666662966
000000005	012	.66666666666666662966
000000005	013	.66666666666666662966
000000006	007	.33333333333333331483
000000006	008	.33333333333333331483
000000006	009	.33333333333333331483
000000006	010	.33333333333333331483
000000006	011	.33333333333333331483
000000007	012	.33333333333333331483
000000007	013	.33333333333333331483
000000008	006	.33333333333333331483

312

【図 26】

図 26

T_PERSON_VIEW_TYPE		
PersonID	ViewID	Score
000000001	003	1.66666666666666674068
000000002	006	.33333333333333331483
000000003	006	.66666666666666662966
000000004	001	1.33333333333333325932
000000004	004	.66666666666666662966
000000004	005	1.66666666666666674068
000000005	001	.66666666666666662966
000000005	003	.33333333333333331483
000000005	006	.33333333333333331483
000000006	003	.66666666666666662966
000000007	006	.33333333333333331483
000000008	001	.66666666666666662966

313

【図 27】

図 27

T_PERSON_CHARACTER		
PersonID	CharacterID	Score
000000001	006	1.33333333333333325932
000000001	007	.66666666666666662966
000000002	004	1.33333333333333325932
000000002	005	1.33333333333333325932
000000002	007	1.33333333333333325932
000000003	010	.33333333333333331483
000000004	001	.33333333333333331483
000000004	002	.33333333333333331483
000000004	014	1.33333333333333325932
000000005	003	.66666666666666662966
000000005	004	.66666666666666662966
000000005	014	1.33333333333333325932
000000006	006	.33333333333333331483
000000007	012	.33333333333333331483
000000007	013	.33333333333333331483
000000008	003	.66666666666666662966
000000008	004	.66666666666666662966

314

【図 28】

図 28

T_PRG_VALUE		
ProgramID	Attribute	ValueID
1000000030000029	Person	0000082163
1000000030000029	Person	0000216595
1000000030000029	Person	0000244529
1000000030000029	Person	1000001790
1000000030000029	Person	1000002204
1000000030000029	Keyword	0000068833
1000000030000029	Keyword	0000016317
1000000030000029	Keyword	0000016365
1000000030000029	Keyword	0000017540
1000000030000029	Keyword	0000026674

315

【図 29】

図 29

T_PRG_VALUE			
ProgramID	AttributeID	ValueID	Score
1000000030000029	001	1000	1.65193365995972070000
1000000030000029	002	0000082163	9.26402330889236800000
1000000030000029	002	0000216595	9.95717048945231300000
1000000030000029	002	0000244529	8.16541102022425800000
1000000030000029	002	1000001790	8.16541102022425800000
1000000030000029	002	1000002204	9.95717048945231300000
1000000030000029	003	002	2.73899365104890520000
1000000030000029	003	003	2.54803404553218500000
0000000030000029	004	001	2.64528432537514800000
1000000030000029	004	004	2.44346124161260800000
1000000030000029	005	006	2.00702060180029470000
1000000030000029	005	007	1.95849912843653650000
1000000030000029	005	008	1.95682099412762900000
1000000030000029	006	0000068833	4.90092468410400400000
1000000030000029	006	0000016317	7.55927521665394200000
1000000030000029	006	0000016365	8.57087612833242200000
1000000030000029	006	0000017540	4.69967511742453150000
1000000030000029	006	0000026674	5.38245951094893000000

316

【図 30】

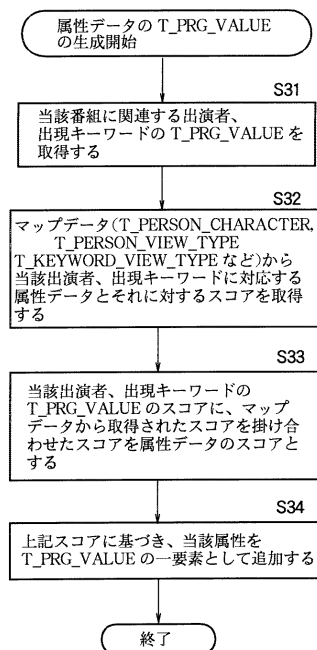
図 30

T_UM_VALUE				
MemberID	UMTypeID	AttributeID	ValueID	Score
200241723056783	1	001	1000	.84296419427714530000
200241723056783	2	001	1002	.66461444765003500000
200241723056783	3	002	0000000001	2.18230298324329500000
200241723056783	1	002	0000000173	1.99001971135882830000
200241723056783	1	002	00000004651	2.19931634905454900000
200241723056783	1	002	0000088887	2.05991592691361850000
200241723056783	1	002	014	6.94115864459585000000
200241723056783	1	003	001	.88080951593773930000
200241723056783	1	003	0000007863	1.13958606003961480000
200241723056783	1	006	0000021755	2.26681335996475970000
200241723056783	1	006	0000020568	1.90854932820777700000
200241723056783	1	006	0000018543	3.13485633400288370000
200241723056783	1	006	0000015737	1.69909282252661780000
200241723056783	1	006	0000015705	1.98230965083122500000
200241723056783	1	006	0000013050	3.43226393316623570000

317

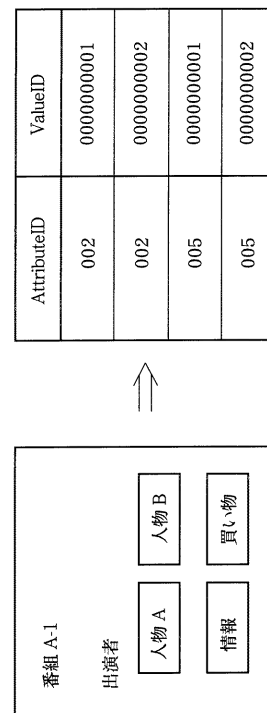
【図 31】

図 31



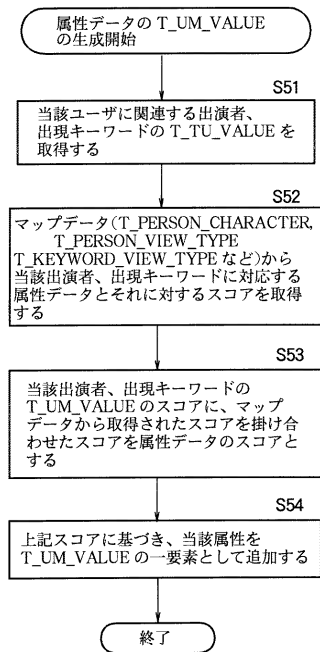
【図 32】

図 32



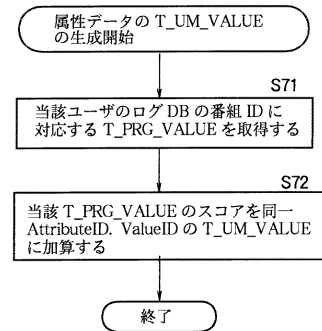
【図 33】

図 33



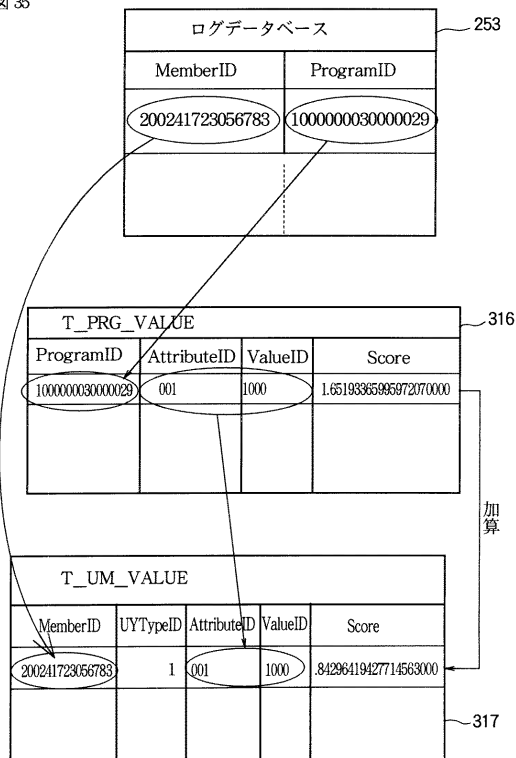
【図 34】

図 34



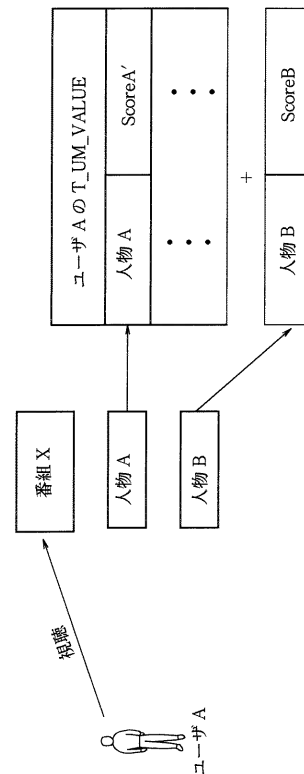
【図 35】

図 35



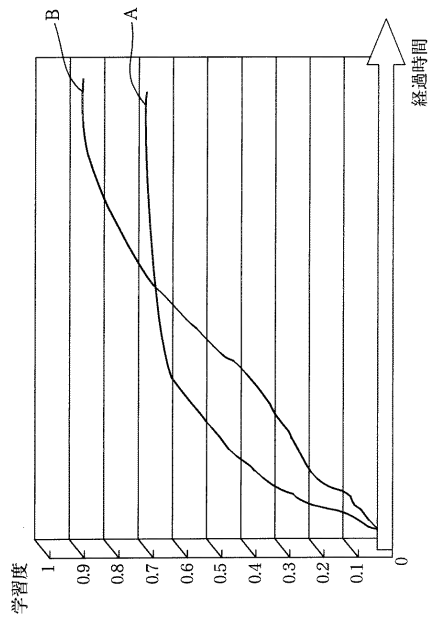
【図 36】

図 36



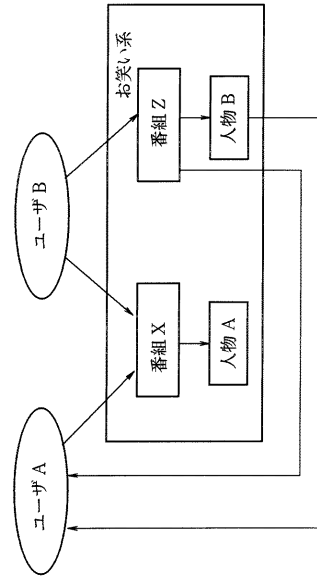
【図 37】

図 37



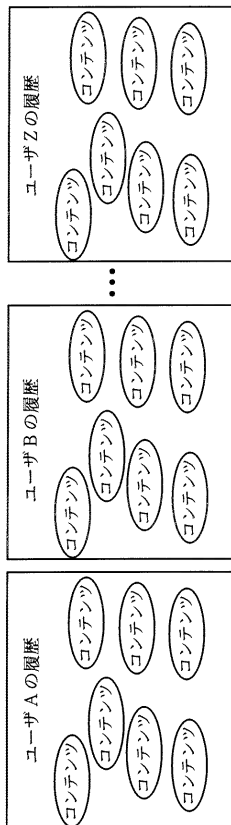
【図 38】

図 38



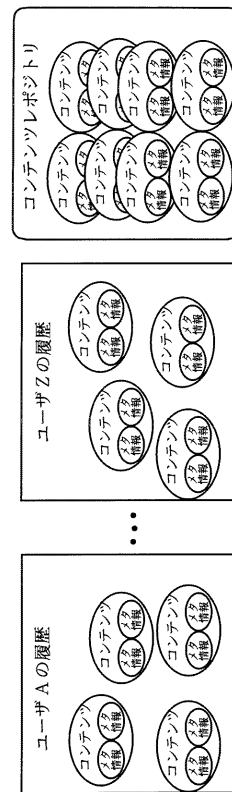
【図 39】

図 39



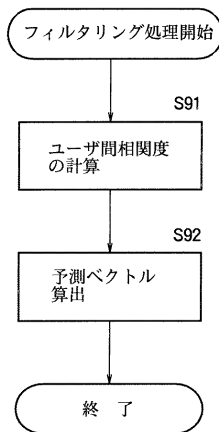
【図 40】

図 40



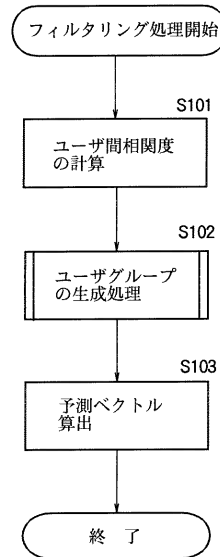
【図 4 1】

図 41



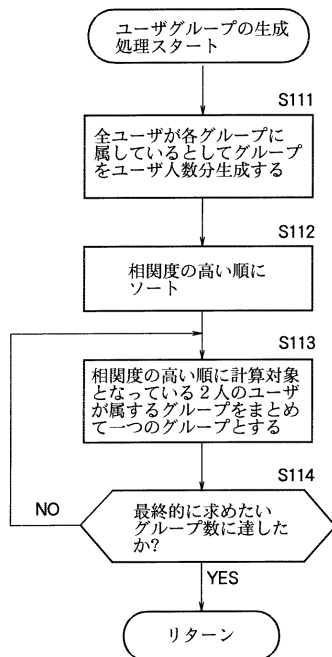
【図 4 2】

図 42



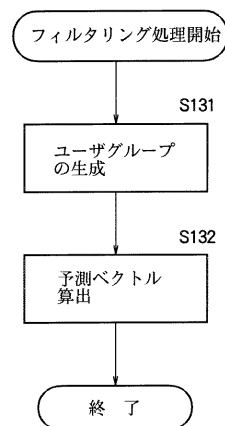
【図 4 3】

図 43



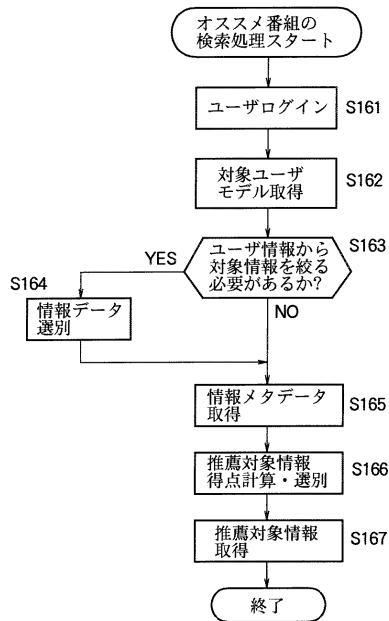
【図 4 4】

図 44



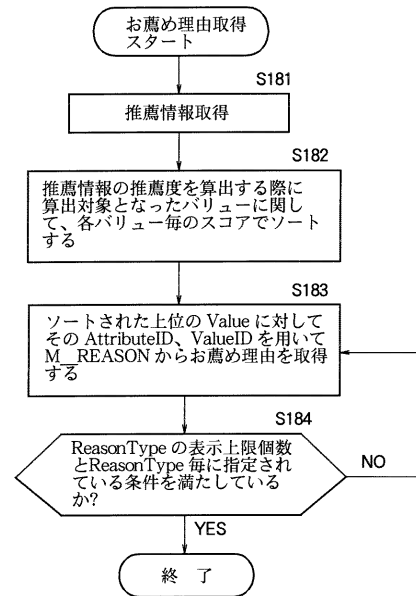
【図 45】

図 45



【図 46】

図 46



【図 47】

図 47

```

1  ## Key = [UMType, アトリビュート]毎の ReasonType
2  ReasonType. Normal. Category=4
3  ReasonType. Normal. Person=3
4  ReasonType. Normal. PersonCharacter=8
5  ReasonType. Normal. ViewType=7
6  #ReasonType. Normal. Age=
7  ReasonType. Normal. Keyword=2
8  #ReasonType. Normal. BranchStation=
9  ReasonType. Normal. Time=5
10 ReasonType. Normal Authority=9
11 #ReasonType. Normal Gender=
12 ReasonType. Prior. Category=1
13 ReasonType. Prior. Person=0
14 ReasonType. Prior. PersonCharacter=8
15 ReasonType. Prior. ViewType=7
16 #ReasonType. Prior. Age=
17 ReasonType. Prior. Keyword=0
18 ReasonType. Prior. Gender=
19 ReasonType. Expected. Person=6
20 ReasonType. Expected. keyword=6
  
```

【図 48】

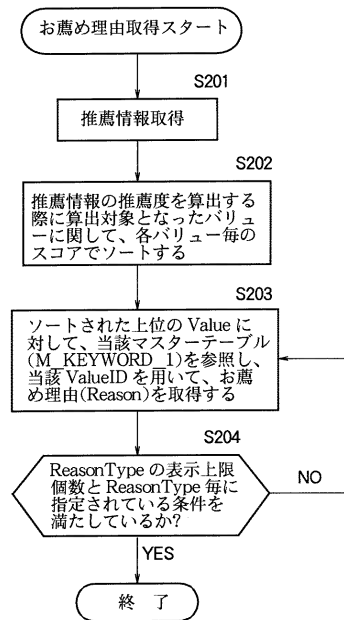
図 48

```

1  # ReasonType 毎の Reson 表示個数の設定
2  # [0] 登録キーワード、人物 (合計 5 件以内)
3  Num. ReasonType. 0=5
4  # [1] 登録ジャンル (1 件以内)
5  Num. ReasonType. 1=1
6  # [2] よくみるキーワード (5 件以内)
7  Num. ReasonType. 2=5
8  # [3] よくみる出演者含 (5 件以内)
9  Num. ReasonType. 3=5
10 # [4] よくみるジャンル (1 件以内)
11 Num. ReasonType. 4=1
12 # [5] よくみる時間帯 (1 件以内)※24時間の中でスコアの高い 2 つの時間帯
13 Num. ReasonType. 5=1
14 # [6] 協調フィルタリングによるキーワード、出演者 (合計 5 件以内)
15 Num. ReasonType. 6=5
16 # [7] 視聴出演者による視聴型 (1 件以内)
17 Num. ReasonType. 7=1
18 # [8] 視聴出演者による性格名 (1 件以内)
19 Num. ReasonType. 8=1
20 # [9] 権威者によるオススメ (1 件以内)
21 Num. ReasonType. 9=1
  
```

【図 49】

図 49



【図 50】

図 50

