スパイダリングを用いた楽曲検索システムの開発

06H091 藤田康孝

目次

s/web/Web/g

概要

自分の欲しい情報が複数の検索サイトに渡ってあった場合、それらをひとつひとつ確認していき、データの集約を手作業でするのは非常に手間であり、時間の無駄である。そこでスパイダリングの技術を用いることによってその手間を少しでも省くことを考えた。今回はその中でも楽曲データの検索についてを考えた。多くの楽曲データの検索サイトでは数多くのデータを取り扱っている所が大半であるが、その中でも有るデータと無いデータと言うものは存在している。また必要なものは楽曲データだけでサイトにある広告等は必要ではない場合がある。その情報を集約し、まとめることは容易では無い。そこで今回は楽曲データを集約し、結果を表示できるプログラムを開発することにした。動作として、ユーザ側から見れば検索したいワードを入力するだけである。あとはプログラムがユーザが入力したワードを基に各 web サイトに正規表現を用いてデータを抽出し、処理が行われ表示される。本システムでは実際にアーティスト名での検索に成功した。それによって 1 つ 1 つのページを見るより手間と時間が省けた。現状の問題としてはアーティスト名での検索に対応していないこととユーザ側から表示方法を選べないことである。今後の課題として、問題点の改善と併せて利用する web サイトを増やした時に正規表現を自動で行うように改良する。

1　はじめに

今日では、インターネットでの情報の取得は容易となった。その中でも検索サイトを使って、自分の欲しい情報を検索し、情報を入手することができる。しかし自分の欲しい情報が複数の検索サイトに渡ってあった場合、それらをひとつひとつ確認していき、データの集約を手作業でするのは非常に手間であり、時間の無駄である。そこでスパイダリングの技術を用いることによってその手間を少しでも省くことを考えた。スパイダ (spider) とは、インターネットから様々な情報を取得するプログラムのこと。スパイダを用いることによって、複数の web サイトにまたがって存在する情報を組み合わせることによって、データ蓄積や独自の検索システムを構築できるなどの様々なことが可能となる。[2]

今回はスパイダを用いて楽曲データの検索についてを考えた。多くの楽曲データの検索サイトでは数多くのデータを取り扱っている所が大半であるが、その中でも有るデータと無いデータと言うものは存在している。また必要なものは楽曲データだけでサイトにある広告等は必要ではない場合がある。その情報を集約し、まとめることは容易では無い。

そこで今回は楽曲データを集約し、結果を表示できるプログラムを開発することにした。本論文では章立てで用語とシステムの内容を説明し、結果と考察で実際の処理結果をまとめる。まとめでは本研究の

2　用語とシステム概要

この章では今回用いたものについての基礎知識、及びシステムの説明を行う。

2.1　スパイダのメリット

スパイダには以下のメリットがある。これにより手動で検索をしたときよりも手間や時間が大幅に節約できる。

2.1.1　リソースへのアクセス自動化

毎日見るサイト等がある場合、機械的な作業はプログラムに任せてしまい、興味のあるコンテンツだけが引き渡されれば、ユーザーの時間、手間を省くことが出来る。データの活用に時間を回せるようになる。

2.1.2　バラバラに存在するデータの集約

web サイト同士はどこかで継っているが、手作業で様々なサイトからデータを集約することは一苦労である。スパイダリングによって作業を自動化することで、情報源をまたがって集約することが出来る。

2.1.3　他のフォーマットへの加工

1 度入手したデータは、ユーザーの思った通りに[*1]加工、変形、再フォーマットすることが出来る。

2.1.4　特定種類の情報を限定的に集約

検索を行う場合、最初に対象を限定しなければならないことがある。スパイダであれば自動的に対象を限定して検索を行い、その結果を集約できる。

[*1] 書いた通りに表示にされるので思った通りに加工、変形できないことがある。

2.2 データの抽出方法

必要なデータを抽出するときに HTML から[*2]正規表現[*3]を用いる。実際に検索サイトから楽曲データを抽出する場合はさらに HTML の中から URL[*4]を抜き出しそこに接続してからデータを抽出することがある。その時相対 URL [*5]は絶対 URL [*6]に直さないと接続できない。

2.3 データベース

データベースは、データを集めて管理し、容易に検索や抽出などの再利用をできるようにしたもの。膨大なデータが存在していても整理・整頓されていなければデータを全て調べないとないらないが、それらを整理・整頓してあるだけでも役にはたつ。さらにデータベース管理システム (DBMS[*7]) があればユーザーがさらに快適にデータベースを利用できる。

[*2] テキストなどでも可能

[*3] ここで言う正規表現はパターンマッチのこと。パターンマッチとは、データを検索する場合に、特定のパターンが出現するかどうか、またどこに出現するかを特定する手法のことである。

[*4] Uniform Resource Locator の略。
ネットワーク内の位置を示してリソースを同定する。

[*5] HTML を解析する際に/hoge/hoge.html といったプロトコル等がない URL のことを相対 URL という。

[*6] し http://www.example.com/hoge/hoge.html といった完全な URL を絶対 URL という

[*7] Database management system の略
データベースを構築するために必要なデータ運用、管理するためのシステム及びそのソフトウェア

3 Result and discussion

3.1 システム概要

システムの概要を図に示す。

3.2 開発環境

- ハードウェア (AT 互換機)
  - CPU:Intel(R) Celeron(R) CPU 2.80GHz
  - Cache Memory:256KB
- OS：Vine Linux 4.2 [研究室内マシン]
- 開発言語：PHP 5.2.6 [研究室内マシン]
- データベース：MySQL 5.0.27 [室内マシン]

3.3 処理の流れ

ユーザーが実際に行うのは検索したいキーワードをテキストボックスに入力するだけである。その後は内部の処理によって楽曲データをまとめて HTML のテーブルを用いて表示する。入力部分は図に示す。

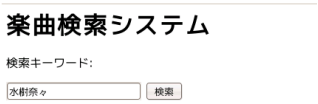


図 1　入力部分

3.4 内部処理

3.4.1 ワードの変換

ユーザーによって入力されたキーワードで検索する際に実際のサイトの検索テキストボックスにそのワードを入れるのではなく直接 URL に渡して HTML を取得してくる。そのときサイトによって入力されたワードは URL 用の文字列に変換する。 URL用の文字列ってなんですか。

3.4.2 URL の抽出

URL から HTML を取得してその中から実際に楽曲データのあるページの URL を正規表現を用いて URL を判別し抽出する。このときタグの中に URL があるが、同じ様なタグのなかに関係のないものがある場合がある。それを考慮しタグの前後に必要部分にマッチする分付け足して抽出する。

3.4.3 **データベース内の処理**

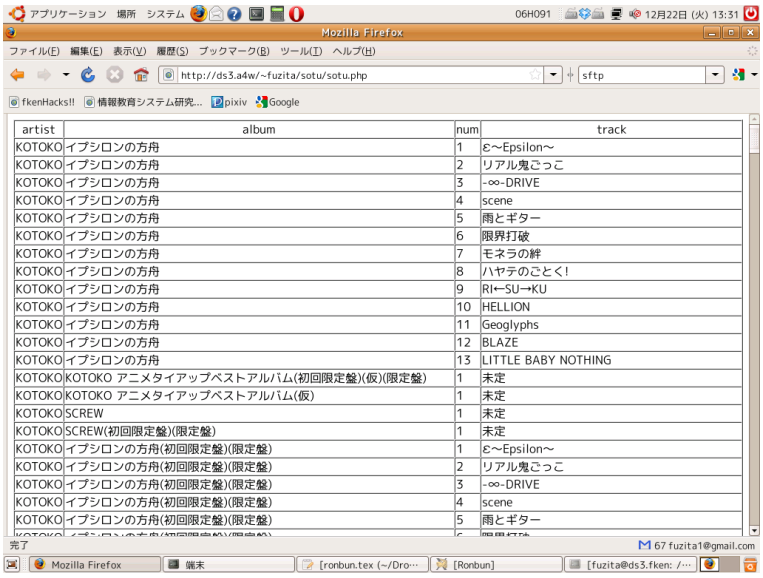
楽曲データを抽出した後データベースに抽出したデータを全て入れる。今回のシステムでは楽曲名、アーティスト名、アルバム名をデータベースに入れる。

3.4.4 **出力**

データベースのものをそのまま出力する。このときデータベース内で重複したものがある場合は出力する時に削除し、出力する。

3.5 結果と考察

今回の研究で実際にスパイダリングを用いてアーティスト名で検索をし、その結果を表示することができた。これは目的としていた検索での時間を短縮することができ、また、複数の web サイトを用いることによって情報量を増やすことにも成功したということになる。これにより複数検索する手間なども削減できた。



artist	album	num	track
KOTOKO	イプシロンの方舟	1	ε〜Epsilon〜
KOTOKO	イプシロンの方舟	2	リアル鬼ごっこ
KOTOKO	イプシロンの方舟	3	-∞-DRIVE
KOTOKO	イプシロンの方舟	4	scene
KOTOKO	イプシロンの方舟	5	雨とギター
KOTOKO	イプシロンの方舟	6	限界打破
KOTOKO	イプシロンの方舟	7	モネラの絆
KOTOKO	イプシロンの方舟	8	ハヤテのごとく!
KOTOKO	イプシロンの方舟	9	RI←SU→KU
KOTOKO	イプシロンの方舟	10	HELLION
KOTOKO	イプシロンの方舟	11	Geoglyphs
KOTOKO	イプシロンの方舟	12	BLAZE
KOTOKO	イプシロンの方舟	13	LITTLE BABY NOTHING
KOTOKO	KOTOKO アニメタイアップベストアルバム(初回限定盤)(仮)(限定盤)	1	未定
KOTOKO	KOTOKO アニメタイアップベストアルバム(仮)	1	未定
KOTOKO	SCREW	1	未定
KOTOKO	SCREW(初回限定盤)(限定盤)	1	未定
KOTOKO	イプシロンの方舟(初回限定盤)(限定盤)	1	ε〜Epsilon〜
KOTOKO	イプシロンの方舟(初回限定盤)(限定盤)	2	リアル鬼ごっこ
KOTOKO	イプシロンの方舟(初回限定盤)(限定盤)	3	-∞-DRIVE
KOTOKO	イプシロンの方舟(初回限定盤)(限定盤)	4	scene
KOTOKO	イプシロンの方舟(初回限定盤)(限定盤)	5	雨とギター

図 2　結果画面

しかし現在の仕様ではアーティスト名での検索にしか対応していない。今回用いた web サイトではアーティスト名の他にもディスク名や曲名でも検索をすることができたが、それらに対応することができなかった。また、web サイトによって HTML の構成が違い、実際に抽出する URL やデータの部分を抜き出すための正規表現を web サイト別に用意しなくてはならなかった。しかしこれは web サイトを追加したい場合にその web サイトの URL だけではなく、データを抽出する正規表現も追加で記入しなくてはならない状態である。

4　まとめ

今回の研究でスクレーピングを用いた楽曲検索システムを開発した。これにより複数の web サイトの楽曲データをまとめることによって、実際にページを見る手間と時間の短縮につながった。また PHP での作成なので、インターネット環境があれば使える。

今回の研究では実験的に 2 つのサイトを用いた。その結果欲しい情報を収集することができた。しかし重複処理の時に空白文字の有無で同じ内容の結果が表示されてしまうことがあった。これにより少し見づらくなってしまった。

4.1 今後の課題

- 今回はアーティスト名の検索しか対応していない。しかし検索サイトでは曲名検索やアルバムのタイトルなどでも検索できる。よって今回のシステムでもそれらの検索方法を追加する。
- 同じ結果の処理方法同じ結果のものがあった場合、曲名やアルバム名に空白文字の有無で重複処理に引っかからず、そのまま表示されてしまう。よって空白文字の有無でも重複処理を行い表示されないようにする。
- 表示方法の変更今回は表示で HTML のテーブルを用いてアーティスト名、アルバム名、トラック番号、曲名を表示させた。しかしこれでは見づらいので見やすい表示に変更する。またユーザーが欲しい情報だけ表示出きるように改良を行う。
- 検索対象の増加今回は検索サイトを 2 つしか使用しなかった。しかし 2 つでは少ないので、他のサイトも検索対象に入れる。
- 正規表現の自動化プログラムに検索対象のサイトを追加した時等に正規表現を 1 つずつ追加するのは手間である。なのでサイトごとに自動で欲しいデータ部分を抽出する正規表現を、自動で作成するようにする。

Acknowledgment

4.2 謝辞

本研究を行うに当たり、ご指導及びご協力頂いた大垣斉講師、藤井信夫教授、水野貴弘先輩、小山翔平先輩、情報安全工学実験室のメンバー及び卒業生の方々、その他、見守ってくださった方々に深く感謝の意を表します。

参考文献

[1] RFC792, Internet Control Message Protocol

[2] Spider, [http://mikilab.doshisha.ac.jp/dia/research/report/2005/0813/008/report20050813008.html]

付録 A　付録 1

A.1　付録 1.1