**商品検索支援プログラム**

**Something I Want**

学籍番号：15FI116

氏名：萬田一沙

1. はじめに

パソコンやスマートフォンの普及により、インターネットを利用することの増えた昨今、ユーザによるYahoo!や楽天、Amazonといった大手ECサイトの利用率は大きくなっている。(図1)

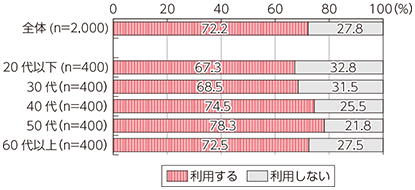


図1　ネットショッピングの利用率(出典：総務省「情報通信白書」[[1]](#footnote-1))

そこで、本システムは特定のECサイトにより提供されているAPIを利用して、商品検索を手軽に、ユーザの要求を検索結果に反映させられるように設計された。ユーザは、本システムを利用することで、任意の条件による商品検索を行うことができ、また一目で見やすいように表示させることができる。

1. 情報抽出/検索/推薦システムの動向

ユーザがインターネットを利用して情報を検索するために、現在利用されている手法として、サイト内検索が挙げられる。サイト内検索は「クローラ型」と「情報連絡型」の2つのタイプに分けられる。クローラ型は、HTMLサイト内を巡回して情報を集めてくれるクローラを利用することで、ユーザが必要としている情報を取得できる。情報連携型は、あらかじめデータベースに情報を登録しておくことで、ユーザが必要としている情報をデータベースから取得できる。

本システムはユーザの指定した条件からYahoo!ショッピングにリクエストを送り、データベースから商品情報を返してもらうシステムであるため、情報連携型を利用している。

1. システムの設計と実装
   1. 実現した機能

本システムは特定のECサイト(今回ではYahoo!ショッピングを利用している)を通して、任意のキーワードによる商品検索を行えるシステムである。特徴として、本システムを利用して商品検索を行う際に、特定の形式で提供される検索条件を付加することで、検索結果に対して幾つかの条件(ソート、価格帯等)を設けることができる。

商品を検索する際に、

Python siw.py query:’検索したいワード’

と入力することで、検索したいワードからYahoo!ショッピングの商品検索結果が10件返ってくる。これが、基本の動作である。

検索結果に条件を付加したい場合は、表の条件文から任意のものを選び、同じく

Python siw.py

の後に続けて記述し、実行することで、対応する条件を商品検索結果に対して付加することができる。また、条件文の記述に順序は影響しない。

|  |  |  |  |
| --- | --- | --- | --- |
| query: | 検索したいワードを  指定できる。※必須  スペースを含まない | offset: | 検索開始位置を指定できる  (default -> 1) |
| sort: | ソートを適応する  -price(価格降順)  +price(価格昇順)  -score(おすすめ降順)  +score(おすすめ昇順)  (default -> -score) | condition: | 商品状態を指定できる  all(すべてを含む)  new(新品のみ)  used(中古のみ)  (default -> all) |
| price\_from: | 最低価格を指定できる  (default -> 0) | price\_to: | 最高価格を指定できる  (default -> 99999999) |

条件例)

python siw.py query:ジャケット offset:11 sort:-price price\_from:7000 price\_to:10000

結果例)

----------------------------------------

-price検索ワード： ジャケット

価格：7,000円から10,000円まで

218,536件中　11～20件

----------------------------------------

11 ジャケット メンズ アウターウェア 男性用ジャケット シンプル 襟折り 四つボタン 無地 上着 メンズウェア カジュアル 紳 士服 新作 LHT

new 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/greenflow/017-bjhj-jk8588656.html

12 ジャケット メンズ アウターウェア 男性用ジャケット シンプル 立ち襟 ブルゾン 無地 上着 メンズウェア カジュアル 紳士 服 ビジネス用 新作 LHT

new 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/greenflow/017-bjhj-jk8581717.html

13 DUHAN バイク メンズ バイクジャケット　バイクウェア ライディングジャケット保護力 通気性 春 秋冬 プロテクター装備 送料無料

new 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/ka-shop/x201709142.html

14 ホンダ HONDA ボルドール Boldor レーシング ジャケット ライデイング ジャケット メッシュ 黒 L ◇MM-8254 ◇02 メンズ【中古】【ベクトル 古着】

used 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/vectorpremium/116-201711210759.html

15 ジョセフ JOSEPH ジャケット ゴート レザー ピーコート 山羊 革ジャン ブルゾン ボタン 茶 36 レディース 【ベクトル 古着】【中古】

used 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/vectorpremium/151-201604210780.html

16 seool's シールズ バイカージャケット 中綿ライナー付 SLB-118 黒 3L 大きいサイズ メンズ【中古】【ベクトル 古着】

used 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/vectorpremium/020-901710190033.html

17 未使用品 ノースフェイス 16SS Compact Jacket コンパクト ジャケット ブルゾン ジャンパー フード 長袖 青系 L アウトド ア NPW71530【中古】【ベクトル 古着】

used 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/vectorpremium/014-901712070003.html

18 パーリーゲイツ PEARLY GATES マスターバニーエディション ゴルフウェア キルティング ジャケット 半袖 ジップ 白 5 ◇02 メンズ【中古】【ベクトル 古着】

used 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/vectorpremium/116-201712150667.html

19 ノースフェイス THE NORTH FACE パープルレーベル ナナミカ ダウンシャツジャケット スナップボタン マスタード M 0906 メンズ【中古】【ベクトル 古着】

used 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： https://store.shopping.yahoo.co.jp/vectorpremium/123-901709060015.html

20 PHENIX （フェニックス）【2017/撥水加工素材商品】 EPIC WindJacket （エピック ウィンドジャケット） PH612WT15

new 10,000円(税込) 平均評価0.00点(0人中)

商品ページ： <https://store.shopping.yahoo.co.jp/linkfast/phenix->ph612wt15.html

* 1. システムが利用するリソース

名称：Yahoo!ショッピング情報取得API商品検索

概要；RESTでGETリクエストを送信し、JSONP形式で受け取る

利用目的：Yahoo!ショッピングの商品検索結果を取得する

URL：https://developer.yahoo.co.jp/webapi/shopping/shopping/v1/itemsearch.html

形式：WebAPI

リクエストパラメータ：

appid:アプリケーションID

query:検索キーワード

price\_from:下限価格

price\_to:上限価格

hits:検索結果数

offset:検索結果表示開始位置

availability:在庫絞り込み

sort:反映するソート

condition:商品状態

レスポンス構造：

/”ResultSet”/”firstResultPosition”:最初のデータ位置

/”ResultSet”/”totalResultsAvailable”:検索総Hit数

/”ResultSet”/”0”/”Result”/”Request”/”Query”:検索ワード

/”ResultSet”/”0”/”Result”/”(任意の数字)”/”Condition”:商品状態

/”ResultSet”/”0”/”Result”/”(任意の数字)”/”Review”/”Count”:レビュー総数

/”ResultSet”/”0”/”Result”/”(任意の数字)”/”Review”/”Rate”:レビュー平均点

/”ResultSet”/”0”/”Result”/”(任意の数字)”/”Name”:商品名

/”ResultSet”/”0”/”Result”/”(任意の数字)”/”Price”/”\_value”:価格

/”ResultSet”/”0”/”Result”/”(任意の数字)”/”Url”:商品ページURL

/”ResultSet”/”0”/”Result”/”(任意の数字)”/”\_attributes”/”index:検索結果の順番

* 1. プログラムの構成

本システムのクラス図は以下(図2)で示される。

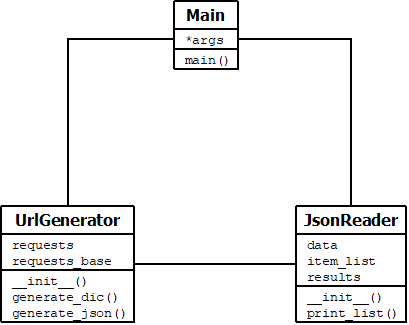


図2　システムのクラス図

Mainクラスはシステムを実行するために各クラスへ指示を行うクラスである。main()を利用して必要なクラスへ指示を行うことができる。

UrlGeneratorクラスは、検索条件からREST形式のURLを生成するクラスである。generate\_dic()は、受け取った検索条件リストをディクショナリに変換して返す。generate\_json()は検索条件からREST形式のURLを生成し、GETリクエストのレスポンスを返す。

JsonReaderクラスは、JSONPデータから必要なデータを取得し、一覧表示させるクラスである。print\_list()は、受け取ったJSONPデータから読み込んだデータをディクショナリで保管し、一覧表示させる。

* 1. データ構造とアルゴリズム

ユーザの指定した条件文をREST形式のURLに組み込むために、渡された条件文を一度リストで受け、ディクショナリを利用して各種リクエストパラメータに対応したkeyのvalueに、条件文から抜き取った条件を保管した。

また、GETリクエストで返ってきたJSONPデータに対しては、ソートされた商品の順番をkey、各種商品データ(商品名、価格等)をvalueにもつディクショナリで管理した。

* 1. その他の工夫点

検索結果のヘッダー部分に、ユーザの指定した条件を表示させるようにした。ユーザの指定した条件はリクエストから生成されたJSONPデータ内にも含まれているものがあり、その場合はJSONPから生成されたディクショナリから、必要なデータを参照すればよいのだが、一部JSONPデータ内には引き継がれないものもあった(商品状態、価格帯等)。そこで、条件から生成されたリクエストをJSONPと一緒に返すことで、ヘッダー部に表示される条件文をより正確にやり取りすることができるようになった。

当初、ユーザ指定の条件文からREST形式のURLを生成することに難航していたが、urllibというライブラリを利用することで、URL生成からGETリクエストを行うまでをスムーズに行うことができた。特にparse.urlencode()は渡されたディクショナリ内の文字列などをパーセントエンコードした上でkeyとvalueを=で結び、各項目を&で繋いでくれるため、あとはAPIリクエスト用のURLと結べば簡単にREST形式のURLが生成されるというものである。

1. 評価・考察
   1. 機能

本システムで実装した機能は以下の通りである。

・ワードによる商品検索

・検索結果への任意の条件(商品状態、価格帯等)付加

・検索結果にソートを適用

・検索結果を一覧表示

構想段階で実装予定だった機能はすべて実装することができた。開発者本人としては表示されるデータも好みに近づけることができた。

しかし、上記実装済みの機能は商品検索システムとして便利な機能であるとともに、必須レベルである部分もいくつかあるため、もう1つ2つ、独自の機能を実装できればなお良いシステムに仕上がるだろう。

現時点で考えられる改良点としては、過去の検索条件から、検索条件についてある程度の推薦が行えるようになると、より便利なシステムになると考えられる。例えば、価格帯3000円以下、2000円以下を頻繁に条件指定するユーザに対して、次回検索の際にprice\_to:3000と入力する手間を省ける選択肢をユーザに提案できるような機能である。

* 1. 実現方法

本システムの構造は大まかに説明すると、Mainクラスで受けたリクエストからUrlGeneratorクラスでJSONPを取得して、JsonReaderクラスで結果を表示させるといった流れである。Pythonを使い始めてまだ日が浅いが、Pythonのリスト構造とディクショナリ構造が以外にも使いやすく、特にREST形式のURLを生成するためにディクショナリが大きく役に立ってくれた。Yahoo!ショッピングで提供されているAPIも構造が大変わかりやすく、PHP形式ではあるが利用例も掲載されていたため開発する上でとても参考になった。

1. おわりに

以上により、ユーザの指定した条件から、特定のECサイトにより提供されているAPIを利用して商品情報を検索し、表示させるシステムを作成した。本システムはYahoo!ショッピングから商品情報を取得する形式であるが、Amazonや楽天等のECサイトも開発者用にAPIを公開しており、それらも併せて利用することができれば、より多くの商品情報を取得することが出来るだろう。

感想

講義内ではJavaを主に利用していましたが、いい機会だと思い、最近学び始めたPythonを利用して課題に取り組ませていただきました。Pythonは記述が簡単で分かりやすいことでもよく耳にしますが、開発中「え、そんなに簡単でいいの！？」と戸惑うことも多々ありました。しかし、使ってみると本当に便利で、リファレンスも見やすく、想像していたよりも快適にコードが書けた気がしました。

課題の内容に関しては、構想段階から「ECサイトのAPIを利用してみたい」という好奇心が強かったため、その方針でシステムを開発していてとても楽しかったです。しかし、一部「ECサイトによる商品検索」に執着し過ぎて、課題のテーマに沿わせることに難航した部分もあり、もう少し構想段階で練っておくべきだったと痛感しました。現時点ではYahoo!ショッピングからの情報取得で完成としましたが。個人的にAmazon等の他ECサイトのAPIもいい感じに合わせられないかと興味が残っているため、Pythonに慣れるついでにもうしばらくこのシステムを弄っていきます。

課題とはあまり関係ない部分になりますが、今回の開発を進める上でGitHubを利用したコード管理にも挑戦してみて、こちらもとても使い勝手がよく、もっと早いう段階から大学内でもこういったサービスに触れる機会があってもよかったかなと感じました。

今回の機会で今まで聞いただけだった技術に触れることが出来、全体を通してとても楽しい時間でした。

参考文献

・Webサイト　Yahoo!JAPANデベロッパーネットワークHP

URL　<https://developer.yahoo.co.jp/>

・Webサイト　Python-izmHP

　URL　<https://www.python-izm.com/>

・Webサイト　Qiita「Amazon、楽天、Yahoo!のショッピングAPIをJavaから使う」

　編集者：KeigoHattori氏

　URL　<https://qiita.com/keigohtr/items/d50cdbcb062743e9b9e9>

1. 総務省「情報通信白書」

   http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h27/html/nc122400.html [↑](#footnote-ref-1)