

調剤薬局用薬剤売買のためのレコメンドシステム

野地恵里奈¹ 鍛治川晶葉¹ 石川由羽¹ 高田雅美¹ 城和貴¹

概要: 本稿では、調剤薬局における薬剤の消費期限切れによる廃棄を減らすためのシステムの提案を行う。薬剤の消費が減少した場合、事前に仕入れている薬剤が過剰在庫となることがある。この過剰在庫を他店に販売することで、廃棄による損失を必要最低限にとどめるためのシステムを開発する。本システムはクライアントがサーバに送信した在庫データから薬剤が過剰在庫または不足在庫になる割合を予測する。また、予測した結果を利用し過剰在庫と不足在庫のマッチングを行う。通信はRESTを用いる。

A Recommendation System for Trading Medical Materials for Dispensing Pharmacy

ERINA NOJI¹ AKIHA KAJIKAWA¹ YU ISHIKAWA¹ MASAMI TAKATA¹ KAZUKI JOE¹

1. はじめに

一般用医薬品や医療用医薬品には消費期限がある。消費期限のほとんどが2年から4年であり、消費期限を過ぎたものは廃棄される。廃棄された場合、その分の利益は得ることができない。現在、日本では医療用医薬品に関して市場での売値を公定価格として国が定めている。これは我々患者が全国どの調剤薬局でも医療用医薬品を購入する際の一律の価格である。薬局は、医薬品を仕入れる際、問屋から公定価格より安く仕入れる。その価格と公定価格の差が薬局の利益である。そして、公定価格の中には廃棄されなければ得るはずだった利益を補うための値が含まれている。

そこで、消費期限切れまでに消費しきれない在庫を他の薬局に転売することで、廃棄される医薬品を減少させることが考えられる。これによるメリットは、廃棄される医薬品が減少するだけでなく、廃棄による利益損失の減少、公定価格に含まれている廃棄されなければ得るはずだった利益を補うための金額の減少である。さらに、公定価格が値下がりすることで、医薬品にかかる保険料及び医療費の削減も期待できる。

本稿の構成は、以下の通りである。第2章では、調剤薬局用薬剤売買のためのレコメンドシステムを提案し、第3

章では、レコメンドシステムの実装内容について述べる。第4章では、実装したシステムの動作内容について述べる。最後に第5章でまとめを述べる。

2. 調剤薬局用薬剤売買のためのレコメンドシステム

開発するシステムの動きは大きく2つに分けられる。1つ目はクライアントの薬剤の在庫データから在庫の予測データを導き出すシステムである。2つ目は、予測データをもとにどの在庫をどの薬局に売ればよいかを導き出すシステムである。図??は、クライアントのデータベースから在庫データを取得する際のシステムの概略図である。図??はサーバのデータベースの予測データを利用しマッチング結果を表示する際のシステムの概略図である。クライアントデータベースには調剤薬局の在庫データが登録されている。サーバデータベース1は、1週間に1回、クライアントから取得した最新の在庫データを登録するデータベースである。サーバデータベース2は、サーバデータベース1にある在庫データから予測した、予測データを登録するデータベースである。

図??①では、サーバがクライアントのDBに接続するために必要な情報をクライアントがサーバに送信する。必要な情報とはクライアントのIPアドレス、データベース名である。図??②では、サーバがクライアントから取得

¹ 奈良女子大学
Nara Women's University

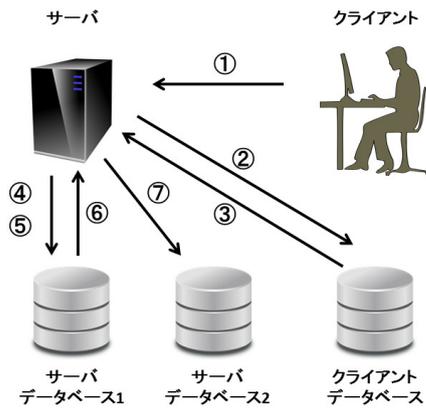


図 1 クライアントのデータベースから
 在庫データを取得する際のシステムの概略図

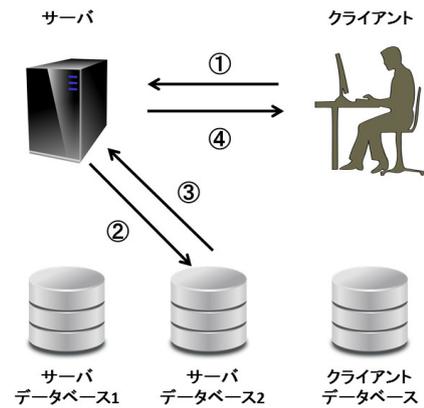


図 2 サーバのデータベースの予測データを利用し
 マッチング結果を表示する際のシステムの概略図

した情報を利用しクライアントデータベースに接続，SQL文を用いてデータを抽出する．図 ?? ③では，クライアントデータベースで抽出したデータをサーバに送信する．図 ?? ④では，サーバがクライアントデータベースから取得した在庫データをサーバデータベース 1 に登録する．以上がクライアントデータベースの在庫データをサーバデータベース 1 に登録する手順である．続いて図 ?? ⑤では，過剰在庫・不足在庫の予測データを作成するために SQL 文を用いてデータを抽出する．抽出するデータは最新の在庫量とその 1 つ前の在庫量である．図 ?? ⑥では，サーバデータベース 1 で抽出したデータをサーバに送信する．図 ?? ⑦では，図 ?? ⑥ で得たデータを用いて予測データを作成し，そのデータをサーバデータベース 2 に保存する．以上がサーバデータベース 1 に保存されている在庫データを用いて過剰在庫・不足在庫を求め，結果をサーバデータベース 2 に登録する手順である．

図 ?? ①では，クライアントが予測を表示するために必要な情報をクライアントがサーバに送信する．必要な情報とは薬局を判別するコードである．図 ?? ②では，薬局を判別するコードを使用し，サーバデータベース 2 にある予測データを SQL 文を用いて抽出する．図 ?? ③では，抽出したデータをサーバで受け取り，配列に格納し，薬剤ごとに他の薬局の予測データと比較する．過剰在庫は正の数，不足在庫は負の数で表されるため，合計した結果が 0 に近ければ近いほど良いマッチングとなる．図 ?? ④では，得られた結果をクライアントに送信する．

平成 26 年度末現在の薬局数は 57,784 か所 ? であり，多くの調剤薬局が参加したシステムを作るためにはサーバ・クライアントともに膨大な数が必要である．このような大規模なシステムは REST を用いて実装する．

REST とは REpresentational State Transfer の略である．2000 年に Roy Fielding 氏が提唱した設計原則，また

は，これを適用したソフトウェアの設計様式のことである？．REST には 4 つの項目がある．1 つ目は，ステートレスであることである．ステートレスの場合は，同じ入力に対する出力は常に同じである．2 つ目は，情報を操作する命令の体系の定義，共有である．たとえば，HTTP の GET や POST が該当する．3 つ目は，すべてのリソースに対する URI は一意である．4 つ目は，情報の内部に，別の情報へのリンクを含められることである．

REST を用いる理由は，スケーラビリティが向上するためである．スケーラビリティとは，システムの利用や負荷が増大した際に，どれだけ柔軟に性能や機能を向上させられるかということを表す．たとえば性能の高い機器に交換したり機器を増やしてシステムの負荷を分散させるというを行う．先ほど述べた通り，将来より多くの調剤薬局が使用する際にはスケーラビリティが重要である．また，ステートレスであれば別のサーバがクライアントからのリクエストに応答できることも重要である？．

3. 実装

3.1 在庫データの取得

本提案システムは，クライアントのデータベースにある調剤薬局の在庫データを取得し，サーバのデータベースに登録する．今回製作したクライアント側のデータベースは，現在多くの調剤薬局で使用されている調剤薬局向け在庫管理システム Fine Justock EX III ? を参考にカラムを作成した．本システムは 1 週間に 1 回，店舗，薬剤ごとに現在の在庫量を手動で取得する．

表 ?? は，クライアントデータベースの項目である．調剤薬局 ID は，それぞれの調剤薬局に振り当てられた ID である．薬剤の判別は JAN コードを用いて行う．在庫変動量は入荷を正の数，消費を負の数として扱う．入荷日は，薬剤が入荷された日である．注意しなければいけない点は，

表 1 クライアントデータベースの項目

カラム	名称
id	通し番号
store_id	調剤薬局 ID
store_name	調剤薬局名称
jan_code	JAN コード
m_name	医薬品名称
s.unit	規格単位
act_date	変動日付
act_time	変動時間
pak_variety	包装形態
bef_amount	変動前在庫数
var_amount	変動在庫量
unit_name	単位数
st_date	入庫日
ub_date	消費期限
m_price	薬価
p_price	バラ購入価

表 2 サーバデータベース 1 の項目 (一部)

カラム	名称
id	通し番号
store_id	調剤薬局 ID
store_name	調剤薬局名称
jan_code	JAN コード
act_date	読込日時
amount	在庫量
st_date	入庫日
ub_date	消費期限

消費期限が月数, つまり, この薬剤は入荷してから何ヶ月使用してよいかという値で格納されていることである. サーバデータベース 2 に格納する際に消費期限を迎える日付を計算し, 日付の形式に変換して格納する.

表??は, サーバデータベース 1 の項目の一部である. クライアントデータベースからデータの取得を行うが, いくつか変更点がある. 1 つ目は, 変動日付・時刻の代わりにデータを読み込んだ日時を登録する. 2 つ目は, データを取得する際に, 最新のデータの変動前在庫数と変動在庫量を足して現在の在庫量を計算し, 格納する点である.

表??は, サーバデータベース 2 の項目の一部である. サーバデータベース 1 のデータを用いてサーバで過剰在庫の予測を行い, そのデータをサーバデータベース 2 に格納する. またクライアントデータベースの項でも述べた通り, 消費期限を日付に変換する.

3.2 過剰在庫・不足在庫の予測

サーバデータベース 1 のデータを利用して, ある薬局のある薬剤が消費期限を迎える際にどの程度余っているか, または不足しているかを予測する. 今回は簡易的な予測として最新のデータとその 1 つ前のデータを利用して予測す

表 3 サーバデータベース 2 の項目 (一部)

カラム	名称
id	通し番号
store_id	調剤薬局 ID
store_name	調剤薬局名称
jan_code	JAN コード
act_date	読込日時
pre_amount	予測在庫量
ub_date	消費期限

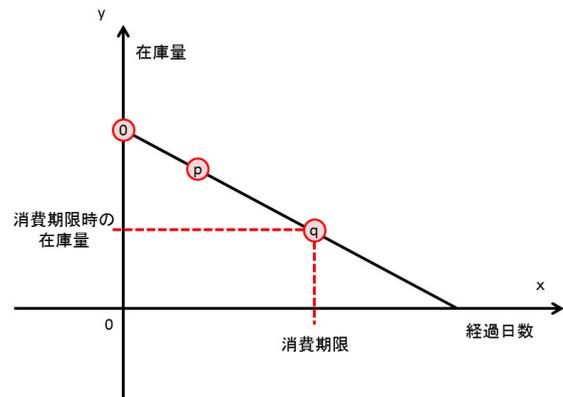


図 3 在庫予測のグラフ

る. 図 ??は, 簡易的な予測をグラフとして図に表したものである. 2 番目に最新のデータを取得した日を基準として, その日からの経過日数を x とする. y は x の時点での在庫数とする. $(x_i, y_i) = (\text{経過日数}, \text{在庫数})$ と置き, $i = 0$ のとき, 2 番目に最新のデータ, $i = p$ のとき, 最新のデータ, $i = q$ のとき, 消費期限時のデータとする. 2 点から一次関数を導き出す式は

$$y = \frac{y_p - y_0}{x_p - x_0} x + y_0$$

であり, $x_0 = 0$ であることから求めたい予測在庫 y_q は

$$y_q = \frac{y_p - y_0}{x_p} x_q + y_0$$

である. 予測在庫 y_3 が正の数ならば過剰在庫と予測され, 負の数ならば不足在庫だと予測される. こうして得られた予測在庫をサーバデータベース 2 の pre.amount, 予測在庫量に格納する.

3.3 過剰在庫・不足在庫のマッチング, レコメンド

上記で得た過剰在庫・不足在庫データを利用してマッチングを行う. クライアントが指定した調剤薬局 ID を持つ予測データを薬剤ごとにそれぞれ他の調剤薬局 ID を持つデータと比較する. クライアントが指定した調剤薬局 ID を持つ薬剤 A とそれ以外の調剤薬局 ID を持つ薬剤 A をそれぞれ B, C と定義した際に, 予測在庫が正の数ならば過

剰在庫, 負の数ならば不足在庫であることから, $B+C$ が 0 に近ければ近いほど良いと考えられる. クライアントがレコメンドの表示を求めた場合, 指定された調剤薬局 ID を持つ薬剤について表示する. ユーザはレコメンド結果を元に在庫を入手, 転売する.

3.4 開発環境

このシステムの主な使用言語は, Java8 である. REST の原則に則り開発する際にアノテーションを利用するため, アノテーションに関しての機能拡張が行われた最新バージョンとしている. また, ライブラリのダウンロードのために Maven を使用する. Maven とは, Java ベースのプロジェクト管理ツールである. 登録された外部のライブラリをプロジェクトに自動でダウンロード, インストール, クラスパスの追加をする. REST での開発を手助けするものとして, jersey をサーブレットとして登録する. このサーブレットからクライアントデータベースの接続に必要なクライアントの IP アドレスなどの情報を得ることができる. 使用しているサーブレットコンテナは, Apache Tomcat v7.0 である.

4. 動作内容

動作内容は 3 つに分けることができる. まず, クライアントが接続するためのトップページにあたる html ファイルを作成する. ここにいくつかボタンを設置する. それぞれ GET や POST などの HTTP メソッドを指定する. 最初にクライアントはこのページにまず接続し, 次の動作を選択する. 図 ?? はトップページである. 図 ?? ではデータベース名を入力する欄にクライアントのデータベース名が入力されている. データベース名のフォームにデータベース名を入力し, 下の「データ送信」のボタンを押すと, フォーム内のデータとともにメソッドが POST で送信先 URL に送信される. また, その下の調剤薬局 ID フォームに調剤薬局 ID を入力し, 「予測データ表示」ボタンを押すと, フォーム内のデータとともにメソッドが POST で送信先 URL に送信される. そうして送信されたデータは, 処理を行うクラスで受信される. GET アノテーションや POST アノテーション, Path アノテーションを用いて使用する java メソッドを選択することができる.

次に図 ?? に表されている, クライアントデータベースから在庫データを取得などを行うメソッドについて説明する. java を使用する場合, データベースに接続するためには, 接続するデータベース分の JDBC ドライバが必要である. そして, JDBC ドライバを用いてデータベースに接続する際は, 接続する IP アドレスとデータベース名, ユーザ名とそのパスワードの情報が必要である. このシステムはサーブレットを使用しているため, クライアントの IP アドレスは getRemoteAddr にて取得可能である. クライ

アントのデータベース名はフォームに入力された値を利用する. 今回はユーザ名とパスワードはサーバが知っているものとする. データベースに接続後 select によりデータを抽出する. SQL 文を実行させるためには, String 型の変数に SQL 文を入れ, その変数を executeQuery もしくは executeUpdate を用いて実行させる. 欲しいクライアントデータベースの在庫データは, 一番新しいデータの変動前在庫量と変動在庫量を足したものである. したがって, 取得したデータをそのままサーバデータベース 1 に insert するのではなく, 変動前在庫量と変動在庫量を足したものを求めてから, 求めた結果を insert しなければならない. また, データを取得した日時も取得し, 同時に insert する. 次にサーバデータベース 1 にあるデータを用いて消費期限切れの場合にいくら在庫が余るか, もしくは不足するかの予測をし, 予測された結果をサーバデータベース 2 に格納する. 予測の前に, 消費期限がいつかを具体的に求める. 入庫日に消費期限までの月を加算した結果の月数の最終日が消費期限日である. ??項で述べた通り, 最新のデータとその 1 つ前のデータを用いて計算する. こうして得たデータを予測した時の日時とともにサーバデータベース 2 に insert する.

そして図 ?? に表されている, サーバのデータベースの予測データを利用しマッチング結果を表示するシステムについて説明する. ??項で述べたようにマッチングを行うには, 異なる調剤薬局同士かつ同じ薬剤同士で行う必要がある. 本来ならば, 自動的に全ての調剤薬局においてマッチングが行われなければならないが, 今回はクライアントが指定した調剤薬局 ID を持つ薬剤に合うデータをマッチングさせる. まずは, クライアントがフォームで指定した調剤薬局 ID を取得する. 次に, 先ほどと同じように JDBC ドライバを用いてサーバデータベース 2 に接続し, 指定した調剤薬局 ID を持つデータを取得し, 配列に格納する. 抽出するデータは, 指定した調剤薬局 ID 以外の調剤薬局 ID を持つ同じ薬剤のデータである. select で抽出する際には条件をつけて抽出する. たとえば指定した調剤薬局 ID の薬剤 A の予測在庫が 10 の場合, マッチングする際に必要な予測在庫の値は -10 以上 0 未満である. SQL 文の時点で条件をつけて抽出するデータを少なくすることで, マッチングにかかる時間を減らすことができる. そして, さらにその中で 1 番小さい値が最も適したマッチング相手のデータである. 対して薬剤 A の予測在庫が負であれば, この逆の求め方で求めればよい. 最後に, こうして求めたデータを, ??項で述べたようにレコメンドとして指定した調剤薬局 ID を持つ薬剤のデータと一緒に表示する.

5. まとめと今後

本稿では, 調剤薬局における薬剤の消費期限切れによる廃棄を減らすためのシステムを提案し, 開発した. 本提案



図 4 トップページ

システムは、クライアントのデータベースから薬剤の在庫データを取得し、取得したデータを利用して薬剤が消費期限を迎えた際に過剰在庫となるか不足在庫となるかを予測する。そして、その結果をもとに過剰在庫で不足在庫を補うようにマッチングを行い、結果をレコメンドとして通知する。

本提案システムの今後の課題について述べる。1つ目は、予測が簡易的なものであるということである。これは複数の調剤薬局の今までの膨大な在庫データを利用した本格的な予測にしたいと考えている。2つ目は、クライアントからのデータの取得が手動である点である。これを自動的にすることで煩わしさを解消する。たとえば、閉店後の深夜の指定された時間に取得するようにする。3つ目は、予測のマッチング方法である。クライアントがデータを要求してからマッチングを行なっているが、実際の調剤薬局では1つの店舗でおよそ1000種類もの薬剤を扱っているため、要求してから表示されるまで時間がかかる。課題2つ目にて提案したデータの時間指定での受信の後にマッチングを行い、結果を保存することでデータを要求されてからすぐに表示させることが可能であると考えられる。現在のシステムでは、マッチング方法も簡易的なものであるため、探索アルゴリズムを利用したものにすべきである。そのためには、多くのアルゴリズムの比較を検討する必要がある。また、過剰在庫と不足在庫のマッチングの際に消費期限の配慮をしていないため、これも改善するべき点である。そして、最終的にはクライアントがサーバにアクセスするのではなく、メールにて結果を送信するようにしたい。今後はそれを改善し、実際に調剤薬局にて使用し、評価する必要があると考えている。

参考文献

- [1] 厚生労働省：平成26年度衛生行政報告例の概況結果の概要5 薬事関係(オンライン), 入手先(http://www.mhlw.go.jp/toukei/saikin/hw/eisei_houkoku/14/dl/kekka5.pdf) (参照2016-11-13).
- [2] Leonard Richardson and Sam Ruby: RESTful Web サービス, オライリー・ジャパン(2007).
- [3] Bill Burke: JavaによるRESTfulシステム構築, オライリー・ジャパン(2010).

- [4] アルフレッサ株式会社：情報機能—会社案内—アルフレッサ株式会社入手先(<http://www.alfresa.co.jp/business/it/>) (参照2016-11-13).