

# オンデマンドバス予約簡便化のための利用推薦アルゴリズムの研究

## Recommender algorithms for On-Demand Bus system

杉浦 孝光<sup>\*1</sup> 大和 裕幸<sup>\*1</sup> 坪内 孝太<sup>\*1</sup> ルディー・レイモンド<sup>\*2</sup>  
 Takamitsu Sugiura Hiroyuki Yamato Kota Tsubouchi Rudy Raymond

<sup>\*1</sup> 東京大学大学院 新領域創成科学研究科 <sup>\*2</sup> 日本 IBM 株式会社 東京基礎研究所  
 Graduate School of Frontier Sciences, The University of Tokyo IBM Japan, Ltd. Tokyo Research Laboratory

On-Demand Bus is a Demand Responsive Transit service where users are transported from and to their preferred locations by the vehicles after they reserve their seats on their preferred time, and hence the term 'on-demand'. In contrast to the traditional fixed-route bus, on-demand bus has many advantages but suffers from the difficulty on making reservations. To make reservations, users must input a lot of information (ID, password, origin, destination, time, etc).

Clearly, a recommendation system that can predict the users' origin, destination and time of use with high precision can assist users during the reservation process. This paper presents experiments on applying various recommendation algorithms for On-Demand Bus recommender system. The experiments indicate that a combined collaborative filtering and individual adaption method for recommending origin, destination and time can lower the hurdles in the reservations of on-demand bus.

### 1. 序論

近年、自家用車の普及などの要因により、公共交通としての路線バスは事業縮小や路線廃止などの危機に直面しており、どのように交通弱者の移動手段を確保するかが喫緊の課題となっている。その中で、有効な手段の一つだと考えられているのがオンデマンドバスである。オンデマンドバスは予約・乗合型の交通システムであり、効率的な運行が可能である。東京大学では独自のオンデマンドバスシステムを開発し、これまで多くの自治体において実証実験を行いその有効性を確認してきた[坪内2009]。そのシステムは図1のようになっている。

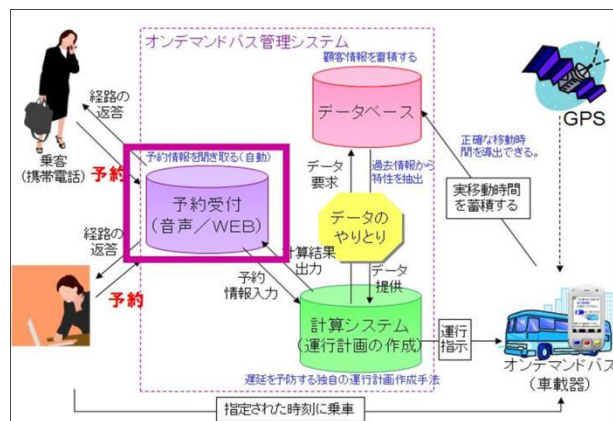


図1: 東京大学が開発したオンデマンドバスシステム

利用者は予約インターフェースにアクセスし、出発地や目的地、時刻などを入力する。入力された情報は運行計画を作成する計算システムに伝えられ、数十μ秒で運行計画を更新しインターフェースを通じて利用者に結果が返され、その運行で問題がなければ利用者が予約をするという仕組みになっている。それと同時に取られた予約のログは逐一データベースに蓄えられ、どのようなプロフィールの人がどのような移動をしたかがすべて

蓄えられている。

このシステムは様々な地域で実証実験を行っているが、その中である問題点が浮き彫りになった。それは、従来の路線バスと違い利用の度に予約をしなければならず、それが手間であるということである。

この問題の解決策の一つとして、ユーザが取りそうな予約情報(日付、場所など)を推薦することで、ユーザは推薦を受け入れるだけで予約を取ることができるようにする方法が挙げられる。

そこで本研究では、オンデマンドバスの利用推薦システムに有効な推薦手法を提案し、その有効性を評価することを目的とする。

本論文の構成を述べる。2章では、本研究で扱った推薦システムを定義する。3章では、本研究で参考にした推薦アルゴリズムについて述べる。4章で提案手法について説明し、5章でその手法の有効性を確認するための実験およびその結果について述べる。6章では実験結果に対する考察を述べ、7章でまとめる。

### 2. 推薦システム

本研究における推薦システムを図2のように定義する。



図2: 推薦システムの定義

システムへの入力情報はオンデマンドバスの利用ログである。利用ログには様々な情報が含まれるが、本研究ではその中でも「ユーザ ID」「曜日」「バス停」「乗った/降りた」の情報を用いた。それを推薦アルゴリズムに入力し、各ユーザへ推薦される「曜日」「バス停」「乗った/降りた」の組合せが出力される。

連絡先: 杉浦孝光, 東京大学大学院新領域創成科学研究科  
 人間環境学専攻, 千葉県柏市柏の葉 5-1-5 環境棟 283  
 号室, 04-7136-4626, sugiura@is.k.u-tokyo.ac.jp

### 3. 推薦アルゴリズム

本研究では、既存の4つのアルゴリズムを本研究用に修正して用いた。ここではその4つのアルゴリズムについて述べる、

#### 3.1 個別適合アルゴリズム

個人ごとに利用ログを参照し、その中で最も頻度の高いものをそのまま推薦するアルゴリズムである[坪内 2008]。例えば、図3のようになる。

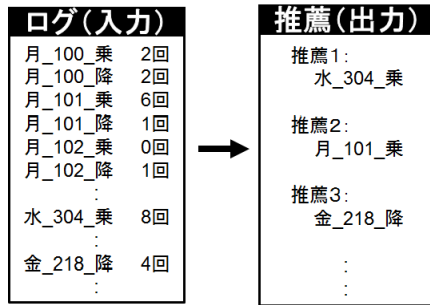


図3: 個別適合アルゴリズムの例

#### 3.2 協調フィルタリング

協調フィルタリングとは、個人のログだけでなく他人のログも用いて推薦を行う手法である。協調フィルタリングはメモリベース手法とモデルベース手法に分類することができる。メモリベース手法とは、ログそのものから推薦を行う手法であり、モデルベース手法とは数学的モデルを仮定した上で推薦を行う手法のことを言う。

##### (1) GroupLens

GroupLens のアルゴリズム[Resnick 1994]による推薦は、①ユーザー間の嗜好の類似度を算出、②未利用アイテムの評価値の推測、③推薦、という手順で行われる。まずすべての組み合わせの2ユーザー間の類似度を算出する。類似度は、各ユーザーの利用回数ベクトルのうち対象のユーザー両者が利用した要素のみを用いて Pearson 相関係数をとる。これは、各ユーザー間のオンデマンドバス利用の傾向がどれだけ似ているかを表す指標となる。定式化すると以下のようになる。

$$r_{ij} = \frac{\sum_{k \in B_{ij}} (V_k^i - \bar{V}^i)(V_k^j - \bar{V}^j)}{\sqrt{\sum_{k \in B_{ij}} (V_k^i - \bar{V}^i)^2} \sqrt{\sum_{k \in B_{ij}} (V_k^j - \bar{V}^j)^2}} \quad (1)$$

ただし、 $V_k^i$ はユーザー  $i$  がアイテム  $k$  を利用した回数、 $\bar{V}^i$ はユーザー  $i$  の各アイテムの利用回数の平均、 $B_{ij}$ は  $i$  と  $j$  が共通して利用したバス停の集合、 $r_{ij}$ はユーザー  $i$  と  $j$  の相関係数を表す。

次に、今後各人がどのバス停をどれだけ利用しようか、の推測をする。これは、該当アイテムの利用回数に類似度を重みとして全員分の加重平均を取る形で行われる。定式化すると以下のようになる。

$$P_n^i = \frac{\sum_{j \in U_{in}} (V_n^j - \bar{V}^j) \times r_{ij}}{\sum_{j \in U_{in}} |r_{ij}|} + \bar{V}^i \quad (2)$$

ただし、 $P_n^i$ はユーザー  $i$  のアイテム  $n$  に対する評価値、 $U_{in}$ はアイテム  $n$  を利用したことのある  $i$  以外のユーザーの集合である。

最後に、ユーザー  $i$  に対して、 $P_n^i$ の値が大きいアイテム  $n$  を推薦する。

##### (2) SVD

特異値分解を用いた推薦アルゴリズム[Sarwar 2001]は、①ユーザー-アイテム行列を特異値分解、②行列の次元削減、③推薦、という手順で行う。まず、行にユーザ、列にアイテム、要素は各ユーザが各アイテムを利用した回数をとったユーザー-アイテム行列  $M$  を特異値分解する。ただし、この際未利用アイテムの要素の値には、そのユーザの利用済みアイテムの利用回数の平均値を入力しておく。

$$M = U \Sigma V^T \quad (3)$$

次に、この行列  $M$  の次元を削減する。次元の削減とは、元の行列  $M$  と削減後の行列  $M'$  の差のフロベニウスノルム  $\|M - M'\|$  を最小にする、 $r = \text{rank}(M') < \text{rank}(M)$  なる行列  $M'$  を探すことである。この次元の削減は、 $M$  を特異値分解して出てくる  $\Sigma$  の要素すなわち特異値のうち、値の大きい方から  $r$  個を除き  $0$  にするという操作を行えばよいことが知られている。この特異値の一部が  $0$  に置き換えられた行列を  $\Sigma'$  とする。そしてこの  $\Sigma'$  と先の分解で算出した  $U$  と  $V$  を用いて、 $M' = U \Sigma' V^T$  を計算することで次元を削減した行列  $M'$  をつくる。

最後に、ユーザー  $i$  に対して  $M'_{ij}$  の値が大きいアイテム  $j$  を推薦する。

##### (3) Random Walk with Restart (RWR)

RWR による推薦[Tong 2008]は、①ユーザー-アイテム二部グラフ上のランダムウォークの定常状態の計算、②収束した確率による推薦、という手順で行う。まず、ユーザー-アイテム二部グラフを作る。エッジの重み  $w_{ij}$  には、ユーザー  $i$  がアイテム  $j$  を利用した回数をとる。ここで、ある時刻  $t-1$  にノード  $i$  に存在するとき、次の時刻  $t$  に、確率  $(1-c)$  でその場に留まり、移動する場合は

$$P_{ij} = \frac{M_{ij}}{\sum_k M_{ik}} \quad (4)$$

の確率でノード  $j$  に移動するような点を考えると、ノード  $i$  を出発したこの点が定常状態においてノード  $j$  に存在する確率は、

$$Q_{ij} = \sum_k Q_{ik} \frac{M_{kj}}{\sum_l M_{kl}} + (1-c)\delta_{ij} \quad (5)$$

と表せる。これを行列形式で書いて整理すると、

$$Q = (1-c)(I - cW)^{-1} \quad (6)$$

となる。 $Q_{ij}$ の値は、ノード  $i$  と  $j$  の「近さ」だと解釈できることから、ユーザー  $i$  に  $Q_{ij}$ の値が大きいアイテム  $j$  を推薦する。

### 4. 提案手法

本論文では、以下のように状況によって推薦アルゴリズムを使い分けるという手法を提案する。

#### 4.1 全ログ件数によるエリアの分類による使い分け

あるエリアに関してログが何件集まっているかに応じて適用する協調フィルタリングのアルゴリズムを変える。メモリベースは頼るものがログデータのみであるため、全体でログがどれだけ集まっているかに精度が大きく左右され、逆にメモリベースはデータ数が少なくても影響が少ないと考えられる。よって、ログが十分集まっていればメモリベース手法、またそうでなければモデルベースのほうが有効であると考えられる。

#### 4.2 利用回数によるユーザの分類による使い分け

次に、各ユーザーに対し利用回数ごとに適用するアルゴリズムを変える。利用回数の多いユーザーに対しては、本人の履歴を見れば十分なので個別適合が有効であり、利用回数の少ないユ

ーザには他人のデータも用いる協調フィルタリングが有効であると考えられる。

### 4.3 推薦の形式による使い分け

推薦には、プッシュ型とプル型の 2 つの形式がある。プッシュ型の推薦とは、電話やメールのように利用者が何もしなくても受動的に受け取る推薦であり、プル型の推薦とは予約インターフェースへの表示など、利用者が能動的に受け取る推薦である。プッシュ型推薦は正解でない推薦の数が多いと利用者にとって逆に煩わしくなってしまうが、プル型はそのような制約が少ないため、正確さを犠牲にしても利用者の需要の可能性を網羅する方が望ましいと考えられる。3.2(3)で述べたように、RWR は利用者アイテムの間接的な接続を直接的に考慮しているため、推薦の多様性が増し、利用者の需要を網羅する必要のあるプル型推薦に有効であると考えられる。

## 5. 実験と結果

### 5.1 実験の概要

実験の流れを図 4 に示す。

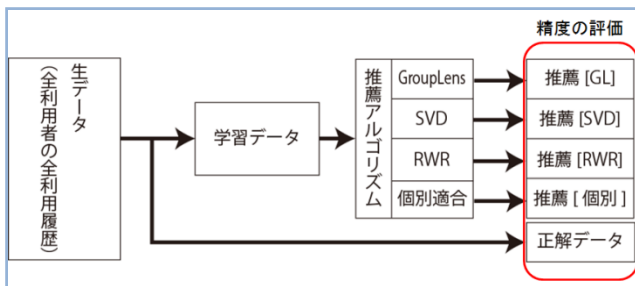


図 4: 実験の流れ

まず、実験に用いるログデータを、学習データと正解データに分割する。今回は、時系列に前半 8 割を学習データ、後半 2 割を正解データとした。次に、学習データを各推薦アルゴリズムに入力し、推薦を出力する。最後に出力された推薦データと正解データを比較し、各アルゴリズムの精度を評価する。このとき、推薦数が 1 個の場合と 5 個の場合と 10 個の場合の 3 つの場合について評価を行った。

### 5.2 評価指標

精度の評価指標としては以下の 2 つを用いた。

$$\text{precision} = \frac{|\text{推薦} \cap \text{正解}|}{|\text{推薦}|} \quad (7)$$

$$\text{recall} = \frac{|\text{推薦} \cap \text{正解}|}{|\text{正解}|} \quad (8)$$

なお、|推薦| は推薦の個数、|正解| は正解の個数、|推薦 ∩ 正解| は推薦と正解の両者に含まれるものの個数である。precision は「出した推薦がどれだけ正解か」の指標であり、プッシュ型の推薦に対する有効性の評価に適する。一方で recall は「利用者の実際の需要のうちどれだけ推薦が網羅したか」の指標なので、プル型の推薦に対する有効性の評価に適する。

### 5.3 利用したデータ

本実験では、北杜市と玉城町の 2 つのエリアで行っている実証実験のログデータを用いた。北杜市のデータは全部で 5060 件、玉城町のデータは 7275 件ある。ログを全件使う以外に、データセットの大きさの影響を調べるためデータセットを時系列で前から 1000 件、2000 件など部分的に使用した実験も行った。

よって、北杜市は 5 パターン、玉城町は 7 パターンの合計 12 パターンの実験を行ったことになる。

### 5.4 実験結果

実験結果を示す。図 5 から図 8 は北杜市・玉城町の全データに対し、横軸に利用回数、縦軸に precision の平均値をとり、その利用回数以下のユーザの precision の平均をプロットしたものである。結果を示す。表 1 は、それぞれのデータセットに対し、推薦数が 1 個の時の precision と推薦数が 10 個の時の recall の値が最も高かったアルゴリズムを表したものである。

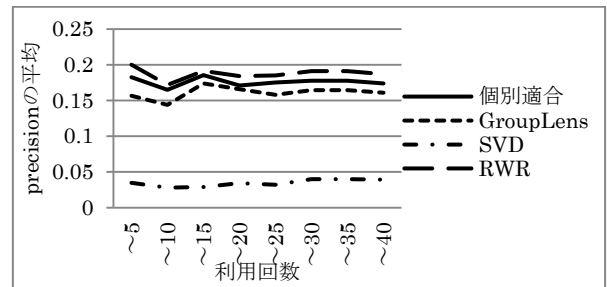


図 5: 利用回数と precision の関係(北杜・2000 件)

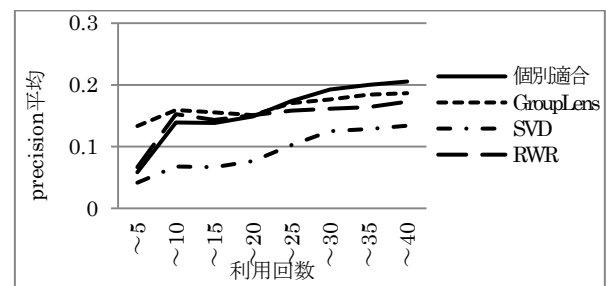


図 6: 利用回数と precision の関係(北杜・5060 件)

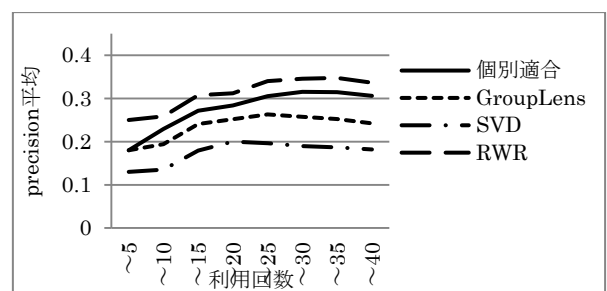


図 7: 利用回数と precision の関係(玉城・4000 件)

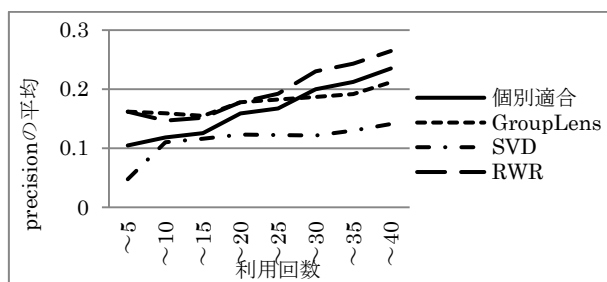


図 8: 利用回数と precision の関係(玉城・7275 件)

## 6. 考察

### 6.1 ログ件数と精度の関係

利用回数の少ない部分(グラフの左側)では、GroupLens の精度が最も高く、利用回数が増えると精度が逆転するという様子が図 6 や図 8 から見て取れる。一方、これ以下のデータ数を用いた実験(図 5・図 7)ではこの傾向は見られなかった。よって、全ログ件数の多寡によって場合分けをするのは有効であるといえる。

表 1:各データセット・指標で最も値の大きかったアルゴリズム

	推薦数 1 個	推薦数 10 個
	precision	recall
北杜 1000 件	RWR	RWR
北杜 2000 件	個別適合	RWR
北杜 3000 件	RWR	RWR
北杜 4000 件	GroupLens	RWR
北杜 5060 件	個別適合	個別適合
玉城 1000 件	個別適合	RWR
玉城 2000 件	個別適合	個別適合/RWR
玉城 3000 件	個別適合	RWR
玉城 4000 件	個別適合	RWR
玉城 5000 件	RWR	RWR
玉城 6000 件	個別適合	RWR
玉城 7275 件	RWR	RWR

### 6.2 利用回数と精度の関係

図 6 を見ると、利用回数 20 回までの部分では GroupLens が個別適合や RWR に勝っているが、20 回を超えた部分から精度が逆転していつている。図 8 でも同じ、精度の逆転が見られる。よって、これらの全ログ件数が多い実験においては利用回数に応じて使用するアルゴリズムを変えるのは有効であるといえる。

### 6.3 推薦の形式と精度の関係

表 1 を見ると、推薦数 1 個の precision は個別適合が優勢で、推薦数 10 個の recall では RWR が優勢であることが分かる。よって、推薦数を絞って precision の高い推薦をする必要があるプッシュ型推薦には個別適合が、数多くの推薦で recall の高い推薦をする必要があるプル型推薦には RWR が有効であるといえる。

### 6.4 考察のまとめ

以上の考察をまとめると、オンデマンドバスの推薦に有効なアルゴリズムの使い分けは表 2 のようになる。

表 2:アルゴリズムの使い分け手法のまとめ

		利用者の利用回数	
		少ない	多い
全ログ件数	少ない	プッシュ型:個別適合 プル型:RWR	
	多い	GroupLens	プッシュ型:個別適合 プル型:RWR

## 7. 結論

本論文では、オンデマンドバス予約簡便化のための利用推薦システムの開発のために有効な推薦手法として、様々な推薦アルゴリズムをデータ数や利用者の利用回数、推薦の形式によって使い分ける手法を提案した。さらに、オンデマンドバス実証実験によって集められた実際のデータを用いて、提案手法が有効であることを確かめた。

ただし、まだ課題も多く残されている。今回は 2 つの地域におけるデータのみを用いたため、一般的にこれらの結論が言えるかは未知である。また、実際に推薦システムを導入した際に、利用者の行動が変わる可能性を考慮していない。今後はこれらの点も考慮しながらさらに研究を進めていく。

### 参考文献

- [坪内 2009] 坪内孝太: オンデマンドバスシステムの開発と地域への導入設計の研究, 東京大学大学院新領域創成科学研究科人間環境学専攻学位論文 2009.
- [坪内 2008] 坪内孝太, 大和裕幸, 稗方和夫, 杉本千佳: オンデマンドバスにおける予約提案サービスの開発. 第 26 回日本ロボット学会学術講演会, 2F2-04, 2008
- [Resnick 1994] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom and John Riedl, "GroupLens: An Open Architecture for Collaborative Filtering of Netnews" In Proceedings of the 1994 Computer Supported Collaborative Work Conference, pp.175.186, 1994
- [Sarwar 2000] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John T. Riedl: Application of dimensionality reduction in recommender systems-a case study, In ACM WebKDD Workshop, 2000
- [Tong 2008] Hanghang Tong, Christos Faloutsos, and Jia-Yu Pan : Random walk with restart: fast solution and applications, Knowledge and Information Systems, An international Journal(KAIS), vol.14, No.3, pp.327-346, 2008