職業能力開発総合大学校紀要　第 42 号（2013 年 3 月）
Bull. Polytechnic University　No.42,　March　2013

プロファイルデータが疎な場合の推薦システムについて

岡島圭介*　遠藤雅樹**　大野成義**

Recommendation System under Sparse Profile Data

Keisuke OKAJIMA*　Masaki ENDOU**　Shigeyoshi OHNO**

The recommendation system is roughly classified in a thing by the contents filtering and a thing by the collaborative filtering. The collaborative filtering is adopted in particular by various systems from the viewpoint of serendipity including amazon.com. However, collaborative filtering method has cold start problems that recommendation precision declines and cannot recommend it when there is few profile data. Therefore we suggest the recommendation system that uses the collaborative filtering for two phases and can recommend even few profile data to. In addition, we inspected the effectiveness by experiments.

Keywords：　Recommendation system, Collaborative filtering, Sparse Profile Data, GroupLens Data Set

1. はじめに

インターネットや情報端末の普及とともに，膨大な量の情報がネットワーク上に流通するようになった．個人やいろいろな組織が簡単にそして安価に情報発信することができるようになったためである．更に，誰もが簡単にネットワークを利用することができるようになり，これらの大量の情報にも容易にアクセスできるようになった．しかし，情報が多すぎることで逆に，ユーザが自分の興味ある情報や商品を探すことが困難になった．情報を参照することのできる状態にあるにもかかわらず，それを利用できないという状況が発生した．このような問題を解決するために情報推薦技術に関する研究が行われてきた．情報推薦技術はユーザの購入履歴や嗜好情報などからユーザの嗜好にあったアイテムを推薦する技術である．[1][2]

情報推薦技術は大きく 2 つに分類される．1 つは内容ベースフィルタリングと呼ばれ，情報内容とユーザの要求を比較・参照することで推薦を行う方式である．もう 1 つは協調フィルタリングと呼ばれ，ユーザの興味や関心について類似する別のユーザの情報から推薦を行う方式である．

2 つの方式のうち既存の推薦システムの多くは協調フィルタリングが使われている．内容ベースフィルタリングでは推薦する情報や商品，一般にアイテムと呼ばれるこれらの内容や特徴が必要になる．更にはユーザが要求していることを明示的に知ることは難しく，内容ベースフィルタリングを実装した推薦システムの構築は限られているからである．一方，協調フィルタリングであれば推薦するアイテムの特徴をシステムが知っておく必要はない．また，システムがユーザの要求を知っておく必要もない．嗜好の類似するユーザを利用して，そのユーザが好むアイテムを推薦する．これは，ユーザが知らなかった意外なアイテム[3]を推薦できる可能性も広がる．しかし，逆に協調フィルタリングではユーザ間で共通の評価アイテムが存在しないと推薦ができないという問題がある．嗜好の類似するユーザが見つけられないからであり，これは cold start 問題として良く知られている．この問題を解決するため間接的な利用者間の相関を利用することで共通の評価アイテムを持たないユーザの推薦を可能にする推薦システムを構築することを提案する．

＊　　吉備高原職業リハビリテーションセンター　Kibi-Kogen Vocational Rehabilitation center
＊＊　職業能力開発総合大学校　情報通信ユニット　Unit of Information and Communication

2. 協調フィルタリングの種類

協調フィルタリングは２つの手法に分類できる．メモリベース法とモデルベース法である．

メモリベース法はユーザデータベースを直接利用して推薦を受けるユーザの嗜好を推定する方法である．推薦システムが利用される以前には何も準備は行わない．それまでのユーザのアイテムに対する評価値をユーザデータベースとして保持しているだけである．推薦するときは，ユーザデータベースの中の嗜好データそのものと対象ユーザの嗜好データを併せて予測する．

一方，モデルベース法は推薦システムが利用される以前にあらかじめモデルを構築する方法である．このモデルとは，ユーザとアイテムの嗜好についての規則性である．推薦をするときは, ユーザデータベースは使わずに，このモデルと対象ユーザの嗜好データとに基づいて予測する．事前にモデルを構築しているのでメモリベース法に比べて推薦時間は速い．しかし，ユーザデータベースが更新されるとモデルを構築し直す必要があり，適応性に劣る．ユーザデータベースを頻繁に変更するため，今回はメモリベース法で検討する．

2.1. 利用者間型メモリベース法の問題点

メモリベース法はユーザデータベースを直接利用して推薦を受けるユーザの嗜好を推定する方法である．メモリベース法は利用者間型とアイテム間型とに分類できるが今回は利用者間型を利用する．利用者間型は，推薦を受けるユーザと嗜好パターンが似ている他のユーザを見つけ，そのユーザの好むものを推薦する．アイテム間型はアイテム間の評価値の類似性から推薦を行う．いろいろなユーザに同じように評価されるアイテムは似ていると考え，関心のあるアイテムに類似のアイテムにユーザは関心を持つという仮定に基づいている．しかし，実験的には特定のアイテムに推薦が偏る傾向が強いという報告があるため[4]，今回は利用しない．

提案するシステムは，利用者間型メモリベース法の中でも特に代表的な手法である GtoupLens の方法[5]をベースとする．この手法では，まず，ユーザデータベース中の各ユーザと推薦を受けるユーザの嗜好の類似度を求める．次に，推薦を受けるユーザが知らないアイテムについて，それを評価している他のユーザの評価値と事前に求めた類似度から，推薦を受けるユーザがどのようにそのアイテムを評価するかを推定する．この際に類似度として以下の式を用いて Pearson 相関で測る．

$$\rho_{ai}=\frac{\sum_{k\in y_{ai}}(s_{ak}-\bar{s}'_a)(s_{ik}-\bar{s}'_i)}{\sqrt{\sum_{k\in y_{ai}}(s_{ak}-\bar{s}'_a)^2}\sqrt{\sum_{k\in y_{ai}}(s_{ik}-\bar{s}'_i)^2}} \tag{1}$$

ここで，y_{ai} はユーザ a と i の二人が共通に評価したアイテム集合を表し，s_{ak} はユーザ a によるアイテム k の評価値，$\bar{s}'_a$ はユーザ a のアイテム集合 y_{ai} に関する評価値の平均である．ρ_{ai} を利用してユーザ a のアイテム j に対する評価値 $\hat{s}_{aj}$ を次式で予測する．

$$\hat{s}_{aj}=\bar{s}_a+\frac{\sum_{i\in y_j}\rho_{ai}(s_{ij}-\bar{s}'_i)}{\sum_{i\in y_j}|\rho_{ai}|} \tag{2}$$

ここで $\bar{s}_a$ はユーザ a が評価済みのアイテム全てに関する評価値の平均，y_j はアイテム j を評価したユーザの集合を表す．

この手法では推薦を受けるユーザと推定するアイテムを評価済みのユーザとの間に最低でも 2 つ以上の共通評価アイテムを必要となる．共通評価アイテム数が 1 以下になると式(1)は計算できないので $\rho_{ai}=0$ となり，ユーザ a に対するアイテム j の予測評価値 $\hat{s}_{aj}$ はアイテムに関係なく計算不能になってしまう．新たにシステムを利用し始めたユーザは評価済みのアイテムも少なく，他のユーザとの類似度を計算することが難しく，適切な推薦をするのが難しい．新たに推薦対象として加わったアイテムに関しても同様に推薦する難しさがある．これが cold-start 問題である．

2.3. 提案システム

本研究では上記の問題を解決するために，協調フィルタリングを使った推薦システムにおいて，対象ユーザ a が知らないアイテム i を評価しているユーザ b と共通評価アイテムがない場合でも，推薦が可能となるようなシステムを提案することを目的としている．

提案するシステムは既存の利用者間型メモリベース法の協調フィルタリングをベースとする．Cold-start 問題に対して，ユーザの購買履歴や評価情報（プロファイルと呼ぶ）が全く

ない状況では対処のしようがない．しかし，プロファイルデータは少ない，疎な状況でも，間接的な利用者間の相関を利用することで推薦する方法を提案する．

ユーザ a のアイテム i の評価値を予測する場合，アイテム i を評価したユーザ b とユーザ a の相関があれば良い．しかし，プロファイルデータが疎であれば，相関のあるユーザが限定され，ユーザ a と相関のある限られたユーザではアイテム i を評価していないことが考えられる．そこで，アイテム i を評価したユーザ b とユーザ a との相関が調べられなければ，別のユーザ c を間に入れて相関を調べる．この場合，推薦精度は落ちるが評価値の予測は可能となる．

3. 実験方法

3.1. 使用プロファイルデータ

本研究を進めるにあたって，プロファイルデータとして GroupLens[6]が公開している MovieLens Data Sets 100k を利用する．このデータは映画推薦システムである MovieLens のユーザ情報をまとめたもので，評価アイテム数は 1,682（映画タイトル），評価ユーザ数は 943，評価データ数は 100,000，映画のジャンル数 19 で構成されている．

評価データは 1 件ごとに，ユーザ番号，アイテム番号，どう評価しているのか（評価値 1～5），評価した日付時刻がいつであるかが Tab で区切られている．評価データは改行で区切られており，これら評価データ 100,000 件を含むファイルを実験に使用した．

3.2. 実験用データの作成

MovieLens Data Sets では，評価ユーザは最低でも 20 個のアイテムの評価を行なっている．実際，評価ユーザ 943 人中の 10%にあたる 94 人は 245 本以上の映画を評価している．最も多くの映画を評価したユーザは 737 本もの映画を評価している．従って，利用者間の相関のとれないようなデータが疎な場合とはいえない．ユーザ a のアイテム i に対する評価値を予測する場合，アイテム i を評価済みでユーザ a と類似度を計算できるユーザが必ず見つけることができてしまう．そこで，類似度が計算できないように評価データを削除して，実験データ用データを作成する．

例えば，図 1 のようにユーザ a のアイテム i の評価値を予測する場合を考える．図で○は評価済みを表し，×は未評価であることを示す．ユーザ a はユーザ b_1 ともユーザ b_2 とも共通評価アイテムが 2 つ以上あるため，類似度

ユーザ＼アイテム			i				
a	○	○	○	△	×	×	×
b_1	○	×	○	○	○	○	○
b_2	○	○	×	○	○	×	○

図 1　類似度を計算できる場合の例

ユーザ＼アイテム			i				
a	○	○	○	△	×	×	×
b_1	×	×	×	○	○	○	○
b_2	×	×	×	○	○	×	○

図 2　類似度を計算できない場合の例

ユーザ＼アイテム			i				
a	○	○	○	△	×	×	×
b_1	×	×	×	○	○	○	○
b_2	×	×	×	○	○	×	○
c_1	○	×	○	×	○	○	×
c_2	○	○	×	×	○	○	○

図 3　間接的に類似度を計算できる場合の例

を計算することができる．そこで図 2 のように評価済みのデータを一部削除し，類似度を計算できないようにする．

3.2. 実験用データからの推定

データを削除することで，対象ユーザ a とアイテム i を評価したユーザ $b_{j(=1,2,\ldots)}$ は共通評価アイテムを持たないため相関を調べることができない．そこで図 3 のように，間接的に類似度を計算して推定を行うためにユーザ a とユーザ b_j それぞれと 2 つ以上の共通評価アイテムをもつ別のユーザ $c_{k(=1,\ldots)}$ を探す．ユーザ c_k はアイテム i を評価していないので，ユーザ c_k のアイテム i への評価データをユーザ b_j との相関から推定する．推定したユーザ c_k のアイテム i への評価データを利用してユーザ a のアイテム i への評価データを推定する．

3.3. 評価方法と環境

元のプロファイルデータから推定する箇所の評価データを削った状態で推定を行い，元の評価値と推定した評価値との差を求める．この推定を 100,000 件の評価データについて行い，絶対平均誤差（MAE）を求めることで推薦システムの精度を評価する．評価を参考にして，より精度の高いシステムの構築と考察を行う．

Eclipse 開発環境下で Java を用いてプログラム作成を行い検証した．PC スペック，使用バージョンは表 1 の通りである．

表 1　使用した実験環境

OS	Windows 7 Ultimate
プロセッサ	Intel®Core™i7 3.2GHz
メモリ	16GB
Eclipse SDK	Version 3.7.0
Java	SE6 U27

4. 実験結果

実験の結果を表 2 に示す．比較のためベースラインとして，各ユーザの評価値の相加平均と各アイテムの評価値の相加平均を乗算し平方根をとったもの，相乗平均を推定値としたときの MAE も計算した．被覆率とは間接的に相関をとることで評価値を推定できた割合である．ほとんど差はなく，間接的に推定を行ってもあまり意味がないことがわかる．

表 2　ベースラインとの比較

	被覆率(%)	MAE
間接的推定	99.859	0.7937
ベースライン	100.000	0.7939

そこで，精度をあげるため，ユーザ a とユーザ c_k の共通アイテム数, ユーザ b_j とユーザ c_k との共通アイテム数の下限を閾値として設けて，比較を行う．共通アイテムが多いほど相関の信頼度が高まると考えられる．

共通アイテム数の下限を 2, 5, 10, 15 として実験を行った結果は表 3 の通りである．共通アイテム数の下限値を大きくすると MAE が減少するが，大きくしすぎると逆に精度が低下することを確認できる．

表 3　共通アイテム数を制限

最低共通アイテム数	被覆率（%）	MAE
2	99.859	0.7937
5	99.858	0.7916
10	99.843	0.7904
15	99.278	0.7915

更に，精度を上げるため予測に使用する類似ユーザに関して，類似度の高いユーザに限定する. 類似度の絶対値で 0.1 以上, 0.2 以上, 0.3 以上，0.4 以上，0.5 以上に制限を行って評価値の予測計算を行った．その結果を図 4 に示す．ここでは先の共通アイテム数を制限する方法も併用し，共通アイテム数の下限を 2 に制限した場合と 10 にした場合をグラフ化した．共通アイテム数の下限を 10 にし，類似度が絶対値で 0.3 以上のユーザに制限して，評価値の推定を行うと精度が最も良くなることが確認できた．グラフの縦軸は MAE，横軸は制限する類似度の絶対値である．

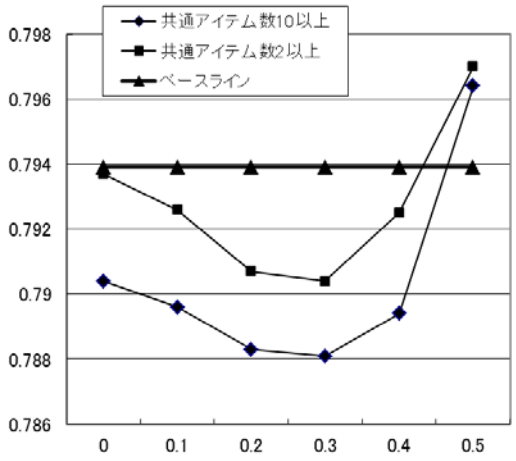

図 4　類似度の制限と評価値予測の関係

5. まとめと今後の課題

あるアイテムの評価値を推定するためには，そのアイテムを評価したユーザと相関がとれなければならない．そのようなユーザが存在しなくても，間接的に評価を推定する方法を提案し，実験を行った．単純に 2 段階の推定を行っても計算が可能になるだけで，精度は良くない．しかし，共通評価アイテム数の下限を設け，類似度の高いユーザのみを推定に使うことで精度を上げることができるのを確認した．

今回の実験では推薦システムに関する多くの研究で使われている MoveiLens Data Sets を用いたが，他のデータでも同様のことが言えるのか実験して確かめることが今後の課題である．

参考文献

[1]　神嶌敏弘，“推薦システムのアルゴリズム (1)-(3)”，人口知能学会誌, 22(6) pp.826-837, 23(1) pp.89-103, 23(2) pp.248-263，2008.
[2]　土方嘉徳, “嗜好抽出と情報推薦技術”，情報処理 48(9), pp.957-965, 2007.
[3]　Ta Son Tung，奥健太，服部文夫，“利用者の潜在的嗜好を予測する協調フィルタリングの検討”, DEIM Forum 2011 F7-5，2011.
[4]　McNee,S.M., Riedl,J. and Konstan,J.A., “Accurate is not always good: How Accuracy Metrics have hurt Recommender System”, Proc. SIGCHI Conf. on Human Factors in Computing System, pp.1097-1101, 2006.
[5]　Rensnick,P., Iacovou,N., Suchak,M., Bergstrom,P. and Riedl,J., “GroupLens: An open architecture for collaborative filtering of netnews”, Proc. Conf. on Computer Supported Co-operative Work, pp.175-186, 1994
[6]　GroupLens Research, the Univerity of Minnesota, MovieLens Data Sets, http://www.grouplens.org/ (2011/8/24).