
配信品質と嗜好を考慮した コンテンツ推薦システム

立命館大学 情報理工学部

落合翔大 上山憲昭

研究の背景

- 扱うコンテンツ数が膨大なサービスの普及
 - 例: 動画配信サービス, オンラインショッピング
- ユーザは嗜好に合ったコンテンツを探し出すのは困難
⇒ 推薦システムの需要が増加
- 推薦システムはユーザとサービス運営者の双方に利点
 - ユーザ: 観たい動画や欲しい商品にアクセスできる
 - 運営者: 広告収入やテナント料などの収益が増加
- コンテンツ配信遅延や低ビットレートはサービス離脱率上昇
⇒ CDNの仕組みを利用して配信時間やトラフィック量を削減

CDNの仕組みを利用した推薦システムの課題

- ユーザやネットワークの観点で望ましいコンテンツが異なる
 - ユーザの観点
 - 個々のユーザの嗜好に合ったコンテンツ
 - ネットワークの観点
 - 配信コスト削減のため、高人気コンテンツ

⇒ これらの両方を考慮しないと性能が低下する可能性

研究の目的

■ 既存研究

- 推薦対象をキャッシュに限定した推薦システム
⇒ キャッシュ性能の低下を回避

■ 既存研究の課題

- 推薦対象を限定 ⇒ ユーザの満足度が低下
- 静的な環境を想定

■ 本研究の目的

- ユーザやコンテンツの数, コンテンツ人気度が変化し続ける動的な環境で, ユーザの満足度とキャッシュヒット率の両方の向上を目指す

提案手法の概要

- 強化学習を推薦システムに適用
 - 動的な環境では, 単一時点での最適化が困難
 - ⇒ 試行錯誤による学習で最適な解を見つける強化学習が有効的
- 学習テーブルを作成
 - ユーザが生成する評価値の予測値
 - 実際の評価値などから報酬を付与
- コンテンツ人気度の変化量を用いて動的な環境へ適用

$$T_i = |P_i^{pre1} - P_i^{pre2}| + |P_i - P_i^{pre1}|$$

学習テーブル

- コンテンツに対する評価値の予測値をテーブルで管理
- 加重平均で予測値を計算

$$V(x, i) = \frac{\sum_k S_{kx} \cdot E_{ki}}{\sum_k S_{kx}} + P_i + T_i$$

S : ユーザ間の類似度

E : 類似するユーザの評価値

	商品1	商品2	商品3	商品4	商品5	商品6	商品7	商品8	商品9	商品10	Xとの相関係数
X	—	4	2	—	—	—	—	1	1	5	1.000
A	5	3	4	—	—	—	—	1	2	1	0.063
B	—	—	—	1	1	2	4	5	4	0	-0.982
C	2	3	2	1	—	4	3	2	1	4	0.941
D	2	—	—	2	4	5	1	2	5	5	0.500
E	—	5	0	—	5	1	—	1	1	2	0.615
推薦度	2.00	4.00	1.00	1.50	4.50	3.33	2.00	1.66	2.33	3.66	

報酬の付与

- コンテンツ視聴の有無, 評価値, キャッシュ有無

$$R = w_1 D + w_2 E + w_3 C$$

D : コンテンツ視聴の有無

E : ユーザが生成した評価値

C : キャッシュの有無

w : 重み

性能評価条件

- ユーザからの推薦要求に対してコンテンツ(映画)を推薦する
- データセット
 - Movielens
- CDN
 - 単一のキャッシュサーバのみを考慮
 - キャッシュ置換方式: LRU(Least Recently Used)
- コンテンツ視聴の判断
 - コンテンツ嗜好度 > 閾値であれば視聴
- 視聴の有無に関わらず評価値を生成

比較対象

■ CF

- 従来手法の協調フィルタリング

■ RL0

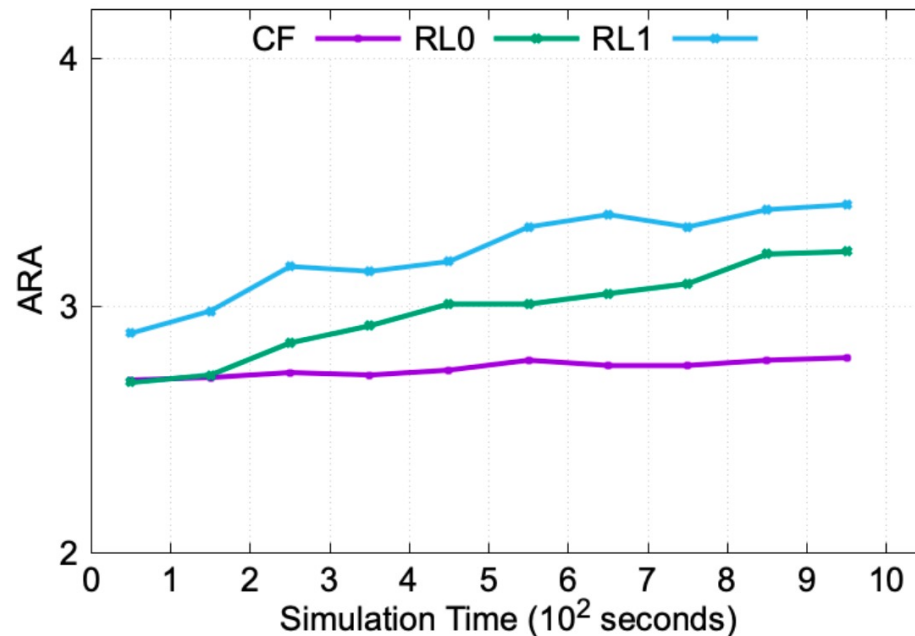
- 提案手法
- 強化学習

■ RL1

- 提案手法
- 強化学習
- 事前学習: 10,000エピソード
 - シミュレーションとは別のMovielensのデータで学習

ユーザの満足度とキャッシュヒット率

- 満足度: ユーザが視聴したコンテンツに対する評価値
 - ARA(Average Recommendation Accuracy)
 - 分割された期間ごとの評価値の平均



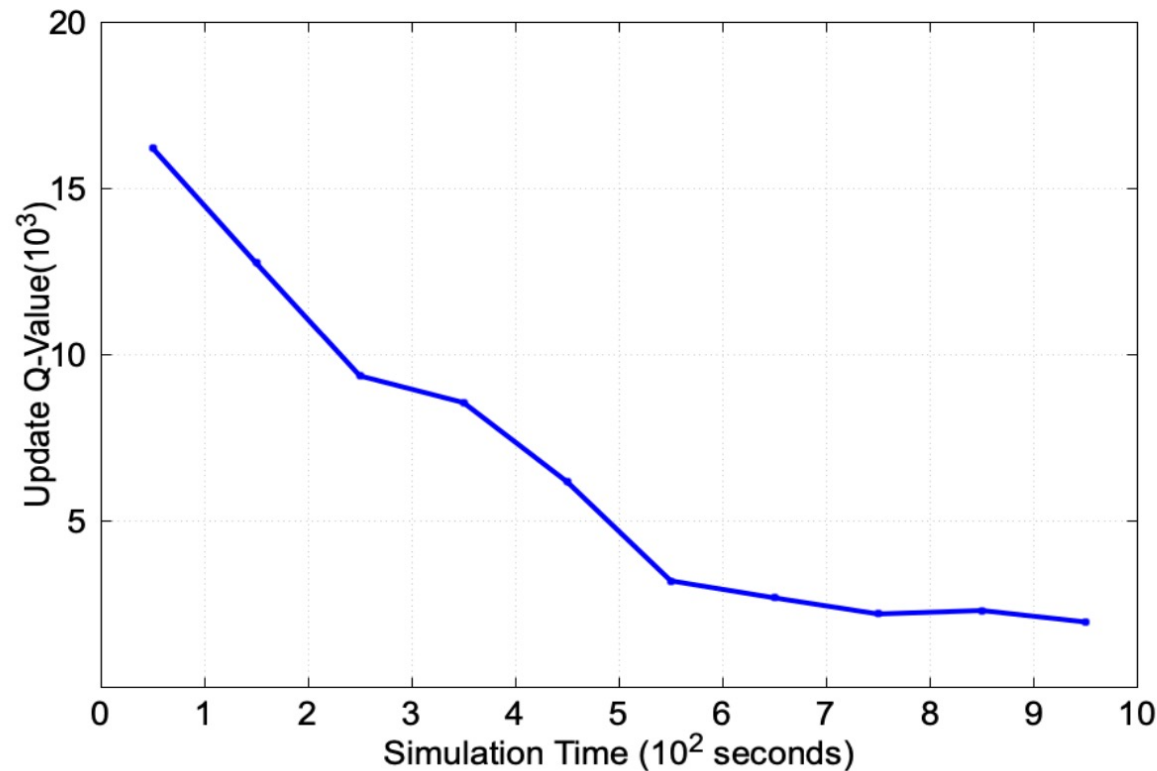
キャッシュヒット率の比較

CF	RL0	RL1
0.0325	0.1153	0.1545

- 提案手法は既存手法と比較して、満足度とキャッシュヒット率が両方向上
- 事前学習を行うことで、開始時から高い推薦精度

学習レベルの推移

■ 強化学習でテーブルの最大値が更新される回数



■ 時間経過で更新回数が減少

⇒ 強化学習による遅延の影響が減少

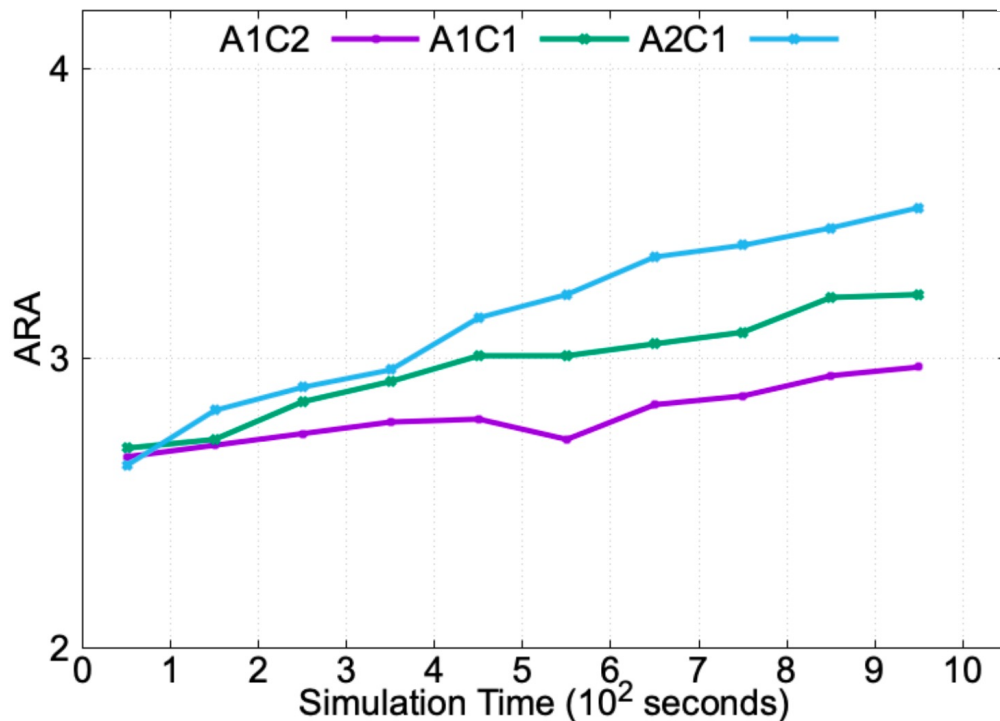
■ 動的な環境ではコンテンツの人気度が常に変化

⇒ 一定数は残り続ける

報酬の重みを変更

- A1C2: キャッシュヒット率の報酬が2倍
- A1C1: 1対1
- A2C1: 評価値の報酬が2倍

$$R = w_1 D + w_2 E + w_3 C$$



キャッシュヒット率の比較

A1C2	A1C1	A2C1
0.1723	0.1153	0.0576

- 重みを変更することで学習の比重を調整可能
- 全てのシナリオでCFより高性能

まとめ, 今後について

■ まとめ

- 動的な環境で強化学習を用いることで, ユーザの満足度とキャッシュヒット率の両方を考慮した方式を提案
- ユーザの満足度とキャッシュヒット率の両方の向上が確認
- 強化学習による遅延の影響は時間経過で減少
- 報酬の重みを変更することで調整が可能

■ 今後について

- 複数のキャッシュサーバを考慮した数値評価
- 提案手法の性能向上