Collaborative Competitive Filtering:

Learning Recommender Using Context of User Choice

Постановка задачи

- есть множество пользователей $U = \{1, 2, \ldots, U\}$ и множество товаров $I = \{1, 2, ..., I\}$
- пользователь $u \in U$ посещает сайт
- система рекомендует u множество товаров $O = \{i_1, \ldots, i_l\}$
- u выбирает подмножество $D \subseteq O$ и предпринимает соответствующие действия
- полагаем: $D = \{i^*\}$, т.е. $|D| = 1$
- цель: составить O_t так, чтобы максимизировать удовлетворение u_t

Collaborative Filtering

- принцип: похожие пользователи сходным образом реагируют на похожие товары
- определяется отображение $(u, i) \longrightarrow y_{ui}$: y_{ui} – например, оценка товара пользователем или указание на то, предпринял ли u действия по поводу i
- рассматривается множество $\{(u, i, y_{ui})\}$
- 2 подхода в CF

CF: Neighborhood model

- идея: распространение ответов среди пользователей и товаров-«соседей»
- шаг 1: определение похожих пользователей и товаров (вычисление т.н. меры подобия)
- шаг 2: паре (u, i) сопоставляется значение, каким-то образом полученное из значений, соответствующим «соседним» пользователям и товарам

CF: Latent factor model (1)

- Matrix factorization models:
  - каждому пользователю и товару сопоставляется вектор из k независимых переменных (factor vector):

    $u \longrightarrow \varphi_u \in \mathrm{R}^k$

    $i \longrightarrow \varphi_i \in \mathrm{R}^k$
  - заинтересованность пользователя *u* в товаре *i* – значение функции выгоды $r(u, i) = \varphi_u^{\mathrm{T}} \varphi_i$
  - полагаем $p(y_{ui}|u, i) = p(y_{ui}|r_{ui}, \Theta)$, где Θ – множество т.н. гиперпараметров
  - проблема – вычисление φ_u и φ_i

CF: Latent factor model (2)

- два наиболее популярных подхода:
  - L2 regression

$$\min_{\phi} \sum_{(u,i)\in\Omega} (y_{ui} - \phi_u^\top \phi_i)^2 + \lambda_{\mathcal{U}} \sum_{u\in\mathcal{U}} ||\phi_u||^2 + \lambda_{\mathcal{I}} \sum_{i\in\mathcal{I}} ||\phi_i||^2$$

здесь Ω – множество пар (u, i), для которых известно y_{ui}, λ_U и λ_I – веса для регуляризации

  - Logistic regression

$$\min_{\phi} \sum_{(u,i)\in\Omega} \log\left[1 + \exp(-\phi_u^\top \phi_i)\right] + \lambda_{\mathcal{U}} \sum_{u\in\mathcal{U}} ||\phi_u||^2 + \lambda_{\mathcal{I}} \sum_{i\in\mathcal{I}} ||\phi_i||^2$$

Недостатки CF – мотивация для CCF

- используются только пары (u, i^*), остальные же $\{(u, i) | i \in O, i \neq i^*\}$ не рассматриваются, отсюда $\forall\ (u, i)\ y_{ui}$ равен 1 либо не определен
- в результате система ставит высокую оценку соответствия практически всем парам (u, i)
- обучение проходит путем обработки либо приближения ответов $\{y_{ui^*}\}$
- CF не моделирует поведение пользователя в процессе выбора и принимает во внимание только совершенные им действия

Collaborative Competitive Filtering (1)

- идея: необходимо рассматривать контекст принятия решения пользователем (т.е. множество O_t)
- решение пользователя зависит от контекста (разные O_t – разные решения)
- пользователь u всегда принимает *рациональное* (локально оптимальное) *решение* :

  $i^* = \arg\max_i \pi_{ui}$ по всем $i \in O$,

  где $\pi_{ui} = r_{ui} - c_{ui}$ при $c_{ui} = \max\{r_{uj} : j \in O \setminus i\}$
- на шаге t пользователю доступны только товары из O_t
- CCF, в отличие от CF, различает понятия «было предложено и не было выбрано» и «не было предложено»

CCF (2)

- принцип локальной оптимальности:

$$\forall i^* \in \mathcal{D}_t, \quad r_{ui^*} \geqslant \max\{r_{ui} | i \in \mathcal{O}_t \setminus \mathcal{D}_t\}$$

$$\text{or} \quad P(i^* \text{ is taken}) = P(r_{ui^*} \geqslant \max\{r_{ui} | i \in \mathcal{O}_t \setminus \mathcal{D}_t\})$$

- минусы такой формулировки:
  - неравенство определяет функцию выгоды с точностью до монотонности
  - решение не единственно
  - формулировка неэффективна с точки зрения вычислений из-за оператора max
- 2 удобные формулировки

CCF: Softmax model (1)

- функция выгоды: r_{ui} + e_{ui} , где:
  - r_{ui} характеризует непосредственный интерес пользователя u к товару i : r_{ui} =

= $\varphi_u^T \varphi_i$ (см. latent factor models)
  - e_{ui} отражает неточность и сложность процесса выбора
  - полагаем:

$$\Pr(e_{ui} \leqslant \epsilon) = e^{-e^{-\epsilon}}$$

CCF: Softmax model (2)

- отсюда:

$$p(i^* = i | u, \mathcal{O}) = \frac{e^{r_{ui}}}{\sum_{j \in \mathcal{O}} e^{r_{uj}}} \text{ for all } i \in \mathcal{O}$$

- на обучающем множестве $\{(u_t, O_t, i_t^*)\}$ скрытые факторы оцениваются следующим образом:

$$\min_{\phi} \sum_t \log \Big[\sum_{i \in \mathcal{O}_t} \exp(\phi_{u_t}^\top \phi_i) \Big] - \phi_{u_t}^\top \phi_{i_t^*}$$

$$+ \lambda_{\mathcal{U}} \sum_{u \in \mathcal{U}} \|\phi_u\|^2 + \lambda_{\mathcal{I}} \sum_{i \in \mathcal{I}} \|\phi_i\|^2$$

CCF: Hinge model (1)

- модель основана на простом следствии из принципа локальной оптимальности:

$$P(i = i^*|u, \mathcal{O}) = P((r_{ui^*} - r_{ui}) > (e_{ui} - e_{ui^*}),\ \forall i \in \mathcal{O})$$
$$\leqslant P((r_{ui^*} - \bar{r}_{u\bar{i}}) > (\bar{e}_{u\bar{i}} - e_{ui^*})),$$

где $\bar{r}_{u\bar{i}} = \frac{1}{|\mathcal{O}|-1} \sum_{i \in \mathcal{O} \setminus i^*} r_{ui}$

является средней возможной выгодой, которую пользователь *u* мог бы получить от не выбранных им товаров

- пользователь принимает решение, сравнивая между собой возможные варианты и оценивая разницу потенциальных значений выгоды

CCF: Hinge model (2)

- получаем:

$$\min_{\theta,\xi} \sum_t \xi_t + \lambda_{\mathcal{U}} \sum_{u\in\mathcal{U}} \|\phi_u\|^2 + \lambda_{\mathcal{I}} \sum_{i\in\mathcal{I}} \|\phi_i\|^2$$

так, чтобы $r_{ui_t^*} - \frac{1}{|\mathcal{O}_t| - 1} \sum_{i\in\mathcal{O}_t\setminus\{i_t^*\}} r_{ui} \geq 1 - \xi_t$

и $\xi_t \geq 0$

- цель – максимизировать разницу между выбором пользователя и средним арифметическим невыбранных товаров

Пара слов о сложности

- описанные выше алгоритмы линейны: $O(|I| \times |O|)$
- в большинстве случаев $|O| \leq 10$
- CCF работает быстрее большой части подходов CF
- некоторые подходы CF имеют такую же сложность, как описанные модели CCF

Обучение (1)

- Стохастическая оптимизация (stochastic gradient descent algorithm)
  - для всех $i \in O_t$

$$\phi_i \leftarrow \phi_i - \eta \left[l'(\phi_u^\top \phi_i)\phi_u + \lambda_{\mathcal{I}} \phi_i \right]$$

  - для всех u

$$\phi_u \leftarrow \phi_u - \eta \left[\sum_{i \in \mathcal{O}_t} l'(\phi_u^\top \phi_i)\phi_i + \lambda_{\mathcal{U}} \phi_u \right]$$

  - градиенты:

$$l'_{\mathrm{Softmax}}(r_{ui}) = \frac{\exp(r_{ui})}{\sum_{j \in \mathcal{O}} \exp(r_{uj})} - \delta_{i,i^*}$$

$$l'_{\mathrm{Hinge}}(r_{ui}) = -\frac{|\mathcal{O}|\,\delta_{i,i^*} - 1}{|\mathcal{O}| - 1} H(1 - r_{ui^*} + \bar{r}_{u\bar{i}})$$

Обучение (2)

- $H(x) = 1$ при $x > 0$ и $H(x) = 0$ в остальных случаях
- для удобства используется приближение $H(x)$ непрерывной функцией

$$\frac{1}{1+e^{-100x}}$$

- хеширование фактор-векторов

CCF: дополнения (1)

- сессии без ответа (реакции) пользователя
  - может быть: $D_t = \varnothing$
  - Softmax :
    - добавляем скаляр θ_u для обозначения порога действия (*action threshold*) пользователя *u*

$$p(i^* = i | u, \mathcal{O}) = \frac{\exp(\phi_u^\top \phi_i)}{\exp(\theta_u) + \sum_{j \in \mathcal{O}} \exp(\phi_u^\top \phi_j)}$$

    - соответственно, вероятность остаться без ответа пользователя равна

$$\frac{\exp(\theta_u)}{\exp(\theta_u) + \sum_{j \in \mathcal{O}} \exp(\phi_u^\top \phi_j)}$$

CCF: дополнения (2)

- Hinge model :
  - добавление константы C > 0

$$\min_{\phi,\xi,\varepsilon,\theta} \sum_t \xi_t + C \sum_t \varepsilon_t + \lambda_{\mathcal{U}} \sum_{u\in\mathcal{U}} ||\phi_u||^2 + \lambda_{\mathcal{I}} \sum_{i\in\mathcal{I}} ||\phi_i||^2$$

  - причем:

при $D_t \neq \varnothing$ для всех $i^* \in D_t$ $\quad r_{u_t i_t^*} - \bar{r}_{u_t \bar{i}} - \theta_u \geqslant 1 - \xi_t$

$\forall\, i \in O_t$ при $D_t = \varnothing$ $\quad \theta_u - r_{u_t i} \geqslant 1 - \varepsilon_t$

а также

$$r_{ui} = \phi_u^\top \phi_i$$

$$\xi_t \geqslant 0 \qquad \varepsilon_t \geqslant 0$$

CCF: дополнения (3)

- свойства контента
  - формула $r(u, i) = \varphi_u^\top \varphi_i$ не подходит для новых (отсутствовавших в обучающем множестве) данных
  - идея: CF + Content Filtering
  - помимо скрытых переменных φ рассматриваются явные свойства пользователей и товаров $x_u \in R^m$ (e.g. введенные при регистрации данные) и $x_i \in R^n$ (e.g. текстовое описание товара) соответственно
  - тогда $r_{ui} \sim p(r_{ui} | \phi_u^\top \phi_i + x_u^\top M x_i)$

  где $M \in R^{m \times n}$ определяет билинейную форму для вычисления пользы i для u

Результаты экспериментов (1)

- первый этап тестирования:
  - обработка только таких пар (*u*, *i*), в которых *u* предпринял какие-либо действия касательно *i* (action dyads)
  - такие пары взяты из наборов данных CF
  - контекст выбора синтетический
  - Netflix, данные из соц. сети
- второй этап тестирования:
  - обработка реальных данных, полученных в результате взаимодействия пользователей с коммерческой системой рекомендаций новостей

Результаты экспериментов (2)

- оценки эффективности:
  - средняя полнота

$$\text{average recall} = \frac{|\text{Drel} \cap \text{Dretr}|}{|\text{Drel}|}$$

  - средняя точность

$$\text{average precision} = \frac{|\text{Drel} \cap \text{Dretr}|}{|\text{Dretr}|}$$

  - Normalized Discounted Cumulative Gain

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{\text{rel}i}{\text{log2(i)}}$$

$$nDCG_p = DCG_p \,/\, IDCG_p$$

Результаты экспериментов (3)

- CCF превосходит CF по всем оценкам эффективности:
  - первый этап

Model		AP@5	AR@5	nDCG@5
Social				
CF	ℓ_2	0.448	0.230	0.475
CF	Logistic	0.449	0.230	0.476
CCF	Softmax	**0.688**	**0.261**	**0.704**
CCF	Hinge	0.686	0.260	0.702
Netflix-5star				
CF	ℓ_2	0.135	0.022	0.145
CF	Logistic	0.135	0.023	0.146
CCF	Softmax	**0.186**	**0.033**	**0.189**
CCF	Hinge	0.185	0.032	0.188

Результаты экспериментов (4)

- второй этап

Model		AP@4	AR@4	nDCG@4
30% Training				
CF	ℓ_2	0.245	0.261	0.255
CF	Logistic	0.246	0.263	0.257
CCF	Softmax	**0.262**	**0.278**	**0.274**
CCF	Hinge	0.261	**0.278**	0.273
50% Training				
CF	ℓ_2	0.250	0.273	0.268
CF	Logistic	0.252	0.276	0.269
CCF	Softmax	**0.266**	**0.285**	**0.278**
CCF	Hinge	0.265	**0.285**	0.277
70% Training				
CF	ℓ_2	0.253	0.275	0.271
CF	Logistic	0.253	0.276	0.274
CCF	Softmax	**0.267**	**0.287**	**0.280**
CCF	Hinge	**0.267**	0.286	**0.280**