

Trabalho 2 - Sistemas de Recomendação

Introdução à Ciência de Computação

Prof. Moacir Ponti / João Batista

PAE: Leonardo Pereira, Matheus Takata

Monitores: George Yoshio Tamanaka, Henry Shinji Suzukawa

Implemente suas atividades sem compartilhar ou olhar código de seus colegas. Procure usar todos os conceitos vistos nas aulas. Documente a sua aplicação por meio de comentários no programa

Descrição do trabalho

Esse trabalho tem como objetivo treinar os primeiros passos em programação, fazendo o aluno pensar em:

1. Quais valores serão armazenados na memória;
2. Que informações deverão ser entradas pelo usuário;
3. Que informações deverão ser exibidas na tela;
4. Como utilizar as estruturas de decisão;
5. Como utilizar as estruturas de repetição;
6. Como processar os dados para obter o resultado desejado;

Tarefa

Sistemas de recomendação são ferramentas automatizadas por plataformas digitais para recomendar itens relevantes para os seus usuários. A utilização dessas ferramentas foram essenciais para alavancar o sucesso de plataformas como Amazon e Netflix ao analisar as preferências dos seus clientes para poder fazer recomendações de itens mais personalizadas. Neste trabalho, o aluno irá implementar um recomendador de filmes simples que utiliza as notas dos usuários com gostos mais parecidos como base para prever as notas de um determinado usuário em relação a um filme, e utilizar essas notas previstas para criar uma lista de recomendação de filmes ordenada por essas notas preditas.

No processo de recomendação, o sistema pode gerar uma matriz de notas de usuário x nota conforme os usuários deixam suas avaliações para os filmes. Nessa matriz, cada linha representa o conjunto de avaliações dadas por um usuário, e cada coluna representa um item, um filme no caso da Tabela 1, presente na base de dados de uma aplicação.

Tabela 1: Exemplo de matriz de notas entre usuários e filmes de uma base de dados. As notas dadas estão entre 1 e 5.

	Minions	Up!	Ex-Machina	300	It	Procurando Dory	Tropa de Elite
Ana	5	3	2	3	4	4	4
João	4	5	2	1	5	4	2
Maria	1	4	4	2	3	5	1
José	5	2	3	4	2	1	5
Mauro	1	0	2	3	2	3	0

Para encontrar os usuários com gostos mais parecidos é preciso utilizar uma métrica que ponderem os valores de cada nota dada. Para este trabalho, utilizaremos a função cosseno, que determina a similaridade entre os usuários de acordo com o ângulo formado pelos vetores representados pelas notas dos usuários. Dados dois usuários u e v , a fórmula da função cosseno é dada por:

$$Sim(u, v) = \cos(u, v) = \frac{u \cdot v}{||u|| \cdot ||v||}$$

Pelo exemplo da Tabela 1, a similaridade entre Ana e João seria calculada por:

$$\cos(u, v) = \frac{5 \times 4 + 3 \times 5 + 2 \times 2 + 3 \times 1 + 4 \times 5 + 4 \times 4 + 4 \times 2}{\sqrt{5^2 + 3^2 + 2^2 + 3^2 + 4^2 + 4^2 + 4^2} \times \sqrt{4^2 + 5^2 + 2^2 + 1^2 + 5^2 + 4^2 + 2^2}} \approx 0,92$$

Feito isso, o próximo passo será prever as notas que o usuário ainda não deu para determinados filmes. Neste trabalho, as avaliações que ainda não foram feitas são representadas por 0 na matriz. Uma abordagem comum na área de recomendação é fazer a prever a nota de um usuário baseando-se nos usuários com gostos mais similares a ele. Nesta atividade, utilizaremos um limiar para montar um conjunto N de usuários com as notas mais similares ao usuário em questão como base para prever sua nota. Sendo assim, para prever a nota que um usuário u daria para um filme i é utilizar as notas dadas pelo conjunto N de usuários cuja similaridade ultrapassou o limiar determinado, fazendo uma ponderação em relação ao grau de similaridade. Essa fórmula é dada por:

$$Nota(u, i) = \frac{\sum_{v \in N_i(u)} Similaridade(u, v) Nota(v, i)}{\sum_{v \in N_i(u)} |Similaridade(u, v)|} \quad (1)$$

Voltando ao exemplo da Tabela 1, para predizer a nota que Mauro daria para o filme Up!, temos calcular primeiro as similaridades entre Mauro e Ana, Mauro e João, Mauro e Maria, Mauro e José. Utilizando a função cosseno, esses valores são, respectivamente, 0.75, 0.66, 0.81 e 0.62. Utilizando um limiar de 0.65, o conjunto N deste caso seria formado por Maria, Ana e João. Sendo assim, a nota predita seria:

$$Nota(u, i) = \frac{0.750 \times 3 + 0.666 \times 5 + 0.816 \times 4}{0.750 + 0.666 + 0.816} \approx 3.96 \quad (2)$$

Uma questão a ser levantada em relação a essa heurística é a de que os usuários avaliam filmes de maneiras diferentes. Certos usuários costumam dar apenas notas altas, outros dão apenas notas medianas, ou então, no caso do Mauro, dão somente avaliações baixas (sua maior nota é 3). Talvez uma nota 4 seria uma nota muito alta para os padrões de Mauro. Uma maneira de melhorar essa predição é ponderar as notas em relação a média de notas dadas por cada usuário. Chamando essa variação de s , temos

$$s(usuario, nota) = nota(usuario, filme) - media_notas(usuario) \quad (3)$$

e assim, a nova estimativa seria dada por:

$$Nota(u, i) = media_notas(u) + \frac{\sum_{v \in N_i(u)} Similaridade(u, v) \times s(v, i)}{\sum_{v \in N_i(u)} |Similaridade(u, v)|} \quad (4)$$

Com essa estimativa, a nota de Mauro para Up! será:

$$Nota(u, i) = 2.2 + \frac{0.750 \times (3 - 3.571) + 0.666 \times (5 - 3.285) + 0.816 \times (4 - 2.857)}{0.750 + 0.666 + 0.816} \approx 2.94 \quad (5)$$

O que é uma nota mais coerentes para os padrões de avaliação de Mauro.

O trabalho do aluno então, estará em prever o quanto os usuários podem gostar dos filmes que ainda não avaliou (possivelmente porque ainda não os viu) estimando a nota que eles dariam para esses filmes. Para isso, seu programa deverá ler:

- n , o número de usuários da base de dados

- i , o número de itens da base de dados
- t , o limiar para selecionar os usuários mais similares
- a matriz de notas usuário \times item

Como saída, imprimir, para os filmes ainda não avaliados por um usuário, as suas notas estimadas na mesma linha. Caso não seja possível fazer essa estimativa, imprimir "DI"(Dados Insuficientes) ao invés da nota. Quando o usuário tiver avaliado todos os filmes, ou seja, não há notas para serem estimadas, apenas desconsidere o mesmo sem fazer qualquer tipo de impressão. Observação: Na etapa de predição de nota são consideradas apenas as avaliações existentes para o item.

Para saber mais sobre sistemas de recomendação que utilizam avaliações de usuários para gerar recomendações, os alunos podem pesquisar sobre:

- Filtragem Colaborativa
- Correlação Pearson como medida de similaridade
- Modelos de vizinhança (*user-knn* e *item-knn*)
- Modelos de fatoração de matrizes

Entrada

A primeira linha da entrada é formada por dois inteiros n e i e um double t , sendo que:

- $4 \leq n \leq 15$
- $\leq i \leq 15$
- $0.5 \leq t \leq 1.0$

As notas serão valores inteiros em que $0 \leq nota \leq 5$. Exemplo:

```
5 4 0.7
5 3 2 0
1 3 4 3
0 3 0 2
0 0 0 3
1 3 2 4
```

Saída

Seu programa deve imprimir na saída apenas:

1. As notas preditas com precisão de duas casas decimais
2. DI para uma nota que não estimada

Exemplo:

```
DI
0.75 3.75
DI DI DI
2.00
```

Não exiba mais nenhuma informação de saída, apenas as especificadas acima. **Não deverá**, por exemplo, ser feito pedido ao usuário como “digite o número”, nem mesmo especificado na tela “o menor valor é”, ou “o resultado é”. A saída do seu programa deverá ser unicamente os valores elencados acima.

Por exemplo, para a entrada:

```
5 4 0.7
5 3 2 0
1 3 4 3
0 3 0 2
0 0 0 3
1 3 2 4
```

A saída será apenas:

```
DI\n
0.75 3.75\n
DI DI DI\n
```

note os símbolos `\n` denotando uma quebra de linha na função `printf()`

Instruções

O trabalho será avaliado levando em consideração:

1. Realização dos objetivos
2. Representação correta da entrada e saída dos dados
3. Uso de comentários e estrutura no código (e.g. indentação, legibilidade)
4. Número de acertos no sistema Run.Codes
5. Uso da memória

ATENÇÃO:

- O projeto deverá ser entregue apenas pelo (<http://run.codes>) no formato de **código fonte**, ou seja apenas o código C.
- O prazo está no sistema run.codes
- Em caso de projetos **copiados** de colegas ou da Internet, todos os envolvidos recebem nota zero. Inclui no plágio a cópia com pequenas modificações, cópia de apenas uma parte ou função. Portanto programe seu próprio trabalho.