PL 4

Algoritmos Probabilísticos

Secção para avaliação [1]

Considere uma aplicação, a desenvolver em Matlab, com algumas funcionalidades de um sistema online de disponibilização de filmes. A aplicação deve considerar um conjunto de utilizadores identificados por um ID e um conjunto de filmes também identificados por um ID (ambos os IDs definidos por um inteiro positivo).

Dados de entrada:
Considere o ficheiro `u.data` do conjunto de dados (release 4/1998) MovieLens 100k, disponível em `http://grouplens.org/datasets/movielens/` e utilize os dados das duas primeiras colunas deste ficheiro para identificar os utilizadores do sistema e os filmes que cada utilizador viu. A terceira coluna do ficheiro contém a avaliação atribuída por cada utilizador.

A informação sobre cada um dos utilizadores encontra-se num segundo ficheiro, `users.txt`, com o seguinte conteúdo:

```
4 ; Carol ; Jesus ; Música ; Fotografia ; Filmes ; Jogos ; Leitura ; ...
49 ; Naísa ; Rodrigues ; Fotografia ; Viagens ; Futebol ;  ...
```

em que os dados de cada coluna estão separados por ";". A linha número n contém a informação do utilizador com o ID n usado no ficheiro `u.data`. A primeira coluna contém o número, a segunda o nome (próprio) e a terceira o apelido. As restantes colunas contém um número variável de interesses do utilizador, como, por exemplo, "Jogos".

NOTA: executando no Matlab a instrução: `dic= readcell('users.txt', 'Delimiter',';');` é criado o cell array `dic` em que a célula `dic{i,j}` contém a informação da linha i e da coluna j do ficheiro.

Descrição da aplicação a desenvolver:
A aplicação deve começar por pedir o ID do filme que se torna o filme actual [2]:

```
Insert Film ID (1 to 1682):
```

certificando-se que o número introduzido é um ID válido (no ficheiro `u.data`, os IDs dos filmes são de 1 até 1682). Depois, a aplicação deve permitir ao utilizador seleccionar uma de 5 opções:

```
1 - Users that evaluated current movie
2 - Suggestion of users to evaluate movie
3 - Suggestion of users to based on common interests
4 - Movies feedback based on popularity
5 - Exit
Select choice:
```

[1] A execução desta secção será objeto de avaliação. Assim, deverá fazer um relatório em PDF com todos os códigos Matlab desenvolvidos devidamente explicados e as opções de desenvolvimento devidamente justificadas. O relatório deverá começar por identificar o ano letivo, a disciplina, a turma prática e os elementos do grupo (nome e No. Mec.) que realizou o trabalho. **Deverá submeter um ficheiro comprimido com o relatório e todos os ficheiros necessários à execução da aplicação desenvolvida. Tenha em atenção os prazos estipulados**

[2] Para introdução de dados pelo teclado, investigue a utilidade da função Matlab `input`

Opção 1: A aplicação lista os nomes dos utilizadores que avaliaram o filme actual. Cada linha deve mostrar o ID e o nome de um utilizador.

Opção 2: A aplicação determina os 2 filmes mais similares ao filme atual (em termos de conjuntos de utilizadores que avaliaram cada filme) e apresenta os IDs e os nomes dos utilizadores que avaliaram pelo menos um dos filmes selecionados, mas que ainda não avaliaram o filme atual.

Opção 3: Para cada utilizador que já avaliou o filme atual, a aplicação seleciona os utilizadores cuja distância de Jaccard estimada (em termos de interesses) seja menor que 0.9 e que ainda não tenham avaliado o filme atual. Isto resulta num conjunto de potenciais utilizadores por cada avaliador do filme atual. No fim, a aplicação apresenta os IDs e os nomes dos dois utilizadores que aparecem no maior número de conjuntos.

Opção 4: O utilizador insere uma string com o nome de um filme (ou parte de um nome). A aplicação devolve os 3 nomes de filmes com os títulos mais similares à string introduzida e, para cada nome, o número de vezes que o filme foi avaliado com nota superior ou igual a 3 (usando um Counting Bloom filter).

Opção 5: A aplicação termina.

Notas sobre a implementação das funcionalidades da aplicação a desenvolver:

A estimativa da similaridade entre conjuntos (i.e., entre utilizadores que avaliaram 2 filmes na Opção 2, entre conjuntos de interesses de utilizadores na opção 3 e entre 2 vectores de caracteres na Opção 4) tem de ser obrigatoriamente implementada por um método *MinHash*.

Na Opção 2, pode reutilizar, com as necessárias alterações, a implementação que efectuou na secção 4.3 deste guião (PL04). O número adequado de funções de dispersão k pode ser escolhido de acordo com as conclusões que retirou nessa altura.

Na Opção 3, deve desenvolver um método *MinHash* adequado à similaridade entre conjuntos de vectores de caracteres (interesses dos utilizadores).

Na Opção 4, deve desenvolver um método *MinHash* adequado a estimar a similaridade entre vetores de caracteres escolhendo de forma fundamentada tanto o tamanho dos *shingles* como o número adequado de funções de dispersão k (sugere-se que experimente tamanhos de *shingle* entre 2 e 5 caracteres).

Requisitos para a implementação em Matlab

É obrigatório desenvolver 2 scripts Matlab.

O primeiro corre uma única vez para ler os dois ficheiros de entrada e guardar em ficheiro todas as estruturas de dados associadas aos utilizadores e aos filmes, incluindo:

- a matriz de assinaturas com os vectores *MinHash* correspondente ao conjunto de utilizadores que avaliaram cada filme (suporte à Opção 2);
- a matriz de asssinaturas com os vectores *MinHash* correspondentes ao conjunto de interesses de cada utilizador (suporte à Opção 3);
- a matriz de assinaturas com os vectores *MinHash* associados aos títulos dos filmes (suporte à Opção 4);
- a(s) estrutura(s) de dados do Counting Bloom filter para armazenamento do número de avaliações com nota superior ou igual a 3 (suporte à Opção 4).

O segundo script começa por ler do disco todas as estruturas previamente guardadas pelo primeiro script e depois implementa todas as interacções com o utilizador descritas anteriormente.

Avaliação do trabalho:

1. Opção 1 a funcionar corretamente **(máximo 2 valores)**
2. Opção 2 a funcionar corretamente **(máximo 4 valores)**
3. Opção 3 a funcionar corretamente **(máximo 5 valores)**
4. Opção 4 a funcionar corretamente **(máximo 6 valores)**
5. Fundamentação/avaliação das opções tomadas na implementação dos métodos probabilísticos (exemplos: número de funções de dispersão, tamanho de *shingles*, dimensionamento dos filtros de Bloom) **(máximo 2 valores)**
6. Qualidade do relatório **(máximo 1)**