Research Challenge #2

DCC049 - Sistemas de Recomendação

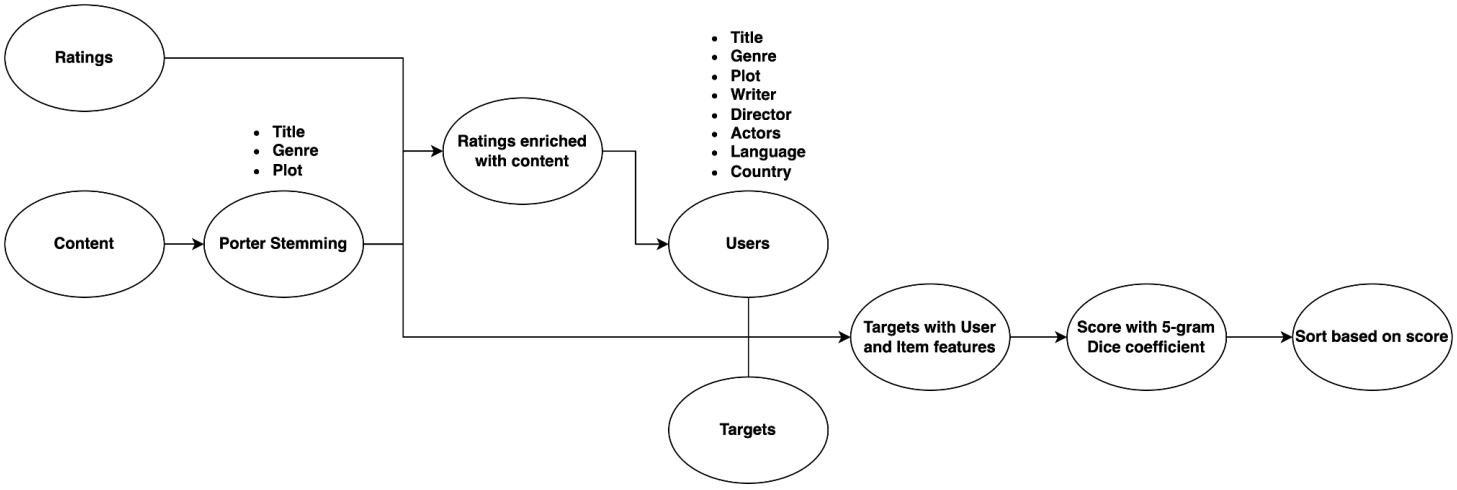
Aluno: Danilo Pimentel de Carvalho Costa
Matrícula: 2016058077

Introdução

O desafio de pesquisa visa aplicar os conceitos aprendidos em aula na implementação de um sistema de recomendação. Os dados a serem usados foram disponibilizados em arquivos .csv e .jsonl de fácil manipulação. O dataset contém 659720 pontuações de 51671 usuários e 29674 itens. As pontuações vão de 1 a 10. Além das pontuações, informações sobre o conteúdo de todos os 38012 itens com pontuação foram disponibilizadas. Objetiva-se produzir 616200 predições, de 6162 para 100 itens cada, sendo que 8338 dos itens são novos, ou seja, há problema de "cold start".

Desenvolvimento

O sistema de recomendação desenvolvido faz recomendações baseadas em conteúdo. Content-based recommendations são oportunas neste cenário, pois há itens que não foram pontuados por nenhum usuário previamente. Em estratégias de filtragem colaborativa, tanto user-based quanto item-based, não haveriam vizinhos para predizer pontuações. A figura abaixo ilustra a estratégia de forma geral:



O algoritmo primeiramente lê os arquivos de entrada. Logo após, é feito um pré-processamento dos dados de conteúdo para cada item, realizando o *stemming* para reduzir as palavras dos atributos *Title, Genre* e *Plot* em sua raíz. Tal estratégia é benéfica, visto que palavras como *bring* e *bringing* possuem a mesma semântica, porém só estão em conjugações verbais diferentes.

Os dados de *ranking* então são enriquecidos com os dados de conteúdo, realizando um *merge* dos dois conjuntos pela coluna *ItemId*. Através deste enriquecimento, agora podemos construir uma representação para usuários, através da agregação dos itens que cada usuário avaliou. A agregação é feita simplesmente através da concatenação dos atributos *Title, Genre, Plot, Writer, Director, Actors, Language, Country* dos itens.

Assim, com a representação de usuários, podemos começar a predição para *targets*. O primeiro passo é realizar o *merge* de *targets* com usuários e itens. Para cada *target*, temos os dados de item e de usuário, e precisamos determinar o quão bom um item é para o usuário. Esta comparação de similaridade é feita através da construção de um *score.* O *score* é o resultado da soma das distâncias de cada atributo de usuário e item. Assim, *Title* do usuário é comparado com *Title* do item, repetindo para *Genre, Plot, Writer, Director, Actors, Language, Country.* O algoritmo de comparação de texto utilizado foi o *5-gram Dice coefficient*.

Por fim, após determinar o *score* para cada *target*, o algoritmo finaliza realizando a ordenação dos *targets* por *score*. Este algoritmo produz nDCG@20 de 0.38050.

Discussão

Durante o desenvolvimento foi observado que quanto mais atributos textuais são utilizados, maior foi a eficácia. Utilizando somente *Title, Plot e Genre*, a submissão pontuou 0.23327. A simples adição de *Writer, Directors, Actors, Language e Country* melhorou a pontuação, passando a ser 0.35362. Além do número de atributos, foi observado que o tamanho do *n-gram* utilizado para cálculo de similaridade via *Dice coefficient* também afeta na eficácia:

- 2-gram: 0.35362
- 3-gram: 0.37050
- 4-gram: 0.37756
- 5-gram: 0.38050

Melhorias são propostas para trabalho futuro. Pode ser feito o uso de TF-IDF nos campos textuais para detectar relevância de termos, como parte do pré-processamento de conteúdo. O algoritmo atualmente também não considera a pontuação dada pelo usuário ao agregar os itens avaliados. A avaliação pode determinar um peso para os termos na agregação, ou ainda pode ser usada para um filtro de itens avaliados positivamente pelo usuário. Collaborative filtering pode ser utilizado também como peso para o *score* gerado para cada *target.* Somente features textuais foram utilizadas pelo recomendador, então é sugerida a exploração de outros atributos como *imdbRating, imdbVotes, Year*, entre outros.

Conclusão

Observa-se as várias possibilidades a serem consideradas na elaboração de algoritmos para sistemas de recomendação baseados em conteúdo. Os vários atributos podem ter relevâncias diferentes para cada usuário. Aspectos como novidade e serendipidade não foram explorados, mas fazem parte da aplicação das técnicas no mundo real. O trabalho introduz o aluno ao meio, e apresenta possíveis caminhos a serem explorados futuramente.