

Exploração de feedback negativo para recomendações top-n

Rafael Torres Souza, Ramon Gonçalves Gonze

13 de junho de 2018

1 Introdução

Recomendações utilizando filtragem colaborativa geralmente são feitas através de feedbacks positivos de usuários. Isto é, tenta-se recomendar os itens mais semelhantes aos itens que um usuário avaliou positivamente. Porém, e se até o momento a maioria das avaliações foram negativas, ou seja, o usuário olhou para itens que não gostou? A recomendação pode ser falha ao tentar se recomendar itens semelhantes aos "itens ruins". Este trabalho propõe uma abordagem alternativa para estes casos onde as avaliações de usuários são, em sua maioria, negativas.

Em especial, o foco é a recomendação de rankings top-n através de filtragem colaborativa. A proposta é utilizar o método de hibridização em cascata, com as abordagens *item-based* e *content-based*. O objetivo é verificar o comportamento da abordagem proposta para com as abordagens usuais item-based e content-based.

2 Trabalhos relacionados

Dentre os trabalhos que foram utilizados como base para esse estudo, podemos destacar [1]. Esse artigo trata sobre os benefícios de se utilizar os feedbacks negativos no contexto de recomendações, especificamente recomendações no domínio de músicas.

Segundo os autores, recomendações negativas são mais fáceis de se obter; induzir o usuário a avaliar negativamente uma música (através de por exemplo, pular uma música) é muito mais simples de se conseguir em comparação com conseguir que o usuário se disponha a avaliar uma música de forma positiva, e além disso, encontrar itens que vários usuário concordem em gostar é mais difícil do que encontrar itens que os usuários concordam em não gostar.

A conclusão do artigo mostra que os feedbacks negativos são bastante úteis no sentido de filtrar resultados. Com apenas dois usuários, 80% da biblioteca já era “excluída” dos possíveis resultados de predição, o que indica que feedbacks negativos filtram de forma eficiente os possíveis resultados para os usuários alvos, de forma que as músicas não excluídas podem até não ser as melhores opções possíveis, mas com certeza não são opções que se encontram dentre os tipos de itens que os usuários definiram como ruins.

Outro artigo que serviu como base para o estudo foi [2], onde, ao contrário do anterior, explora a utilização tanto de feedback negativo quanto positivo. Porém, a abordagem adotada para explorar o contexto foi a fatorização de matrizes.

3 Solução proposta

Um sistema que utiliza preferências negativas age como um filtro que bloqueia itens que o usuário não quer, e permite que todo o resto passe [1]. Se um usuário possui uma quantia considerável de feedbacks negativos, supõe-se que, após o recomendador eliminar itens similares aos negativos, restarão itens relevantes para o usuário. Porém esse conjunto de itens pode ser consideravelmente grande. Então é necessário fazer um refinamento da recomendação, reordenando o ranking e colocando no topo os itens mais relevantes para o usuário. A estratégia é que irão para o topo os itens mais similares aos "itens bons" (avaliados positivamente) do usuário.

A proposta deste trabalho é construir um raking de tamanho n . Considere que o usuário possui um conjunto N de itens avaliados negativamente e um conjunto P de itens avaliados positivamente. A recomendação é feita em dois passos:

1. Utiliza-se a abordagem item-based para construir um ranking de itens menos similares aos itens de N . Ou seja, os itens com o menor valor de similaridade estarão no topo do ranking.
2. São selecionados os n primeiros itens do ranking gerado, e estes são reordenados de forma que os itens mais similares aos itens de P fiquem no topo da lista. Contudo, a similaridade agora é calculada utilizando-se a abordagem content-based.

Ao final dos dois passos tem-se o ranking final. Notas 1, 2 e 3 foram consideradas negativas e 4 e 5 positivas. Foram feitos experimentos para verificar o recomendador obteria melhores resultados com definições diferentes de feedback negativo, como notas negativas 1 e 2 e positivas 3, 4 e 5, contudo, observou-se que o comportamento do recomendador proposto em relação as abordagens usuais de filtragem colaborativa não se alterava.

4 Base de dados

A solução proposta foi testada no domínio de recomendação de livros. A base de dados vêm de um site semelhante ao goodreads.com, mas com termos de uso mais permissivos¹. Ela contém:

- 10.000 livros
- 53.424 usuários
- 5.976.479 notas (de usuários para livros)

Os ratings variam de 1 a 5, sendo 1 o menos relevante e 5 o mais relevante. A base de dados foi separada em dois conjuntos C_1 e C_2 , onde o primeiro corresponde a 70% da base original e o segundo, 30%. Essa separação ocorreu de forma aleatória, sendo que, para cada nota, foi gerado um número aleatório indicando se ela iria para C_1 ou C_2 (com probabilidades 70% e 30% respectivamente). O conjunto C_1 é utilizado como histórico de avaliações dos usuários, e do conjunto C_2 , são extraídas, para cada usuário $u \in C_2$, uma lista de livros b_1, \dots, b_k , sendo que para cada livro b_i da lista do usuário u , existe um par $(u, b_i) \in C_2$. A lista de cada usuário será ordenada em um ranking pelo algoritmo proposto.

A métrica utilizada para avaliar o algoritmo é o nDCG. Portanto, após o algoritmo gerar um ranking para cada usuário, o ranking ideal de cada usuário é gerado utilizando C_2 (os livros com as maiores notas irão para o topo do ranking ideal), e então calculado o valor do nDCG.

5 Experimentos

Os experimentos foram executados variando a porcentagem mínima de feedback negativo de um usuário e o tamanho do ranking retornado. Para os dois testes são comparados os comportamentos do item-based e content-based comuns com a abordagem proposta, assinalada como "Heurística", nos gráficos das figuras 1 e 2.

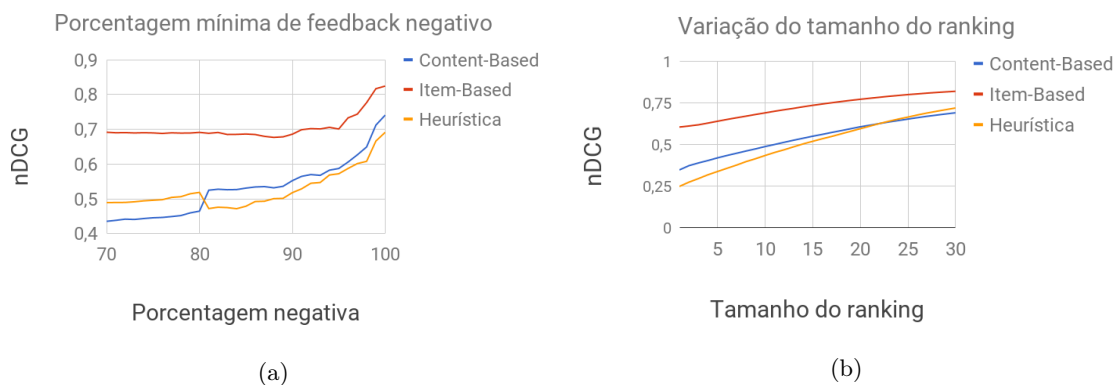


Figura 1: Experimentos que variam o tamanho do ranking e a porcentagem mínima de feedback negativo de um usuário.

¹<http://fastml.com/goodbooks-10k-a-new-dataset-for-book-recommendations/>

Um dos parâmetros para se testar é a proporção de feedbacks negativos e positivos do usuário. A definição da proposta é aplicar a recomendação para usuários em que a **maioria** dos seus feedbacks sejam negativos. Foi testado empiricamente qual o valor da "maioria", e através da Figura 1 (a) percebe-se que a heurística sempre é menos eficiente que a abordagem item-based, porém é superior ao content-based quando a porcentagem mínima de feedback negativo considerada está entre 70% e 81%.

Em relação ao tamanho do ranking, a heurística tem resultados abaixo do item-based e content-based para rankings pequenos, mais especificamente de tamanho 1 a 22. Rankings de tamanho acima de 22 a heurística registra um valor de nDCG maior que o content-based, mas ainda assim, abaixo do item-based.

O código-fonte utilizado encontra-se disponível no Github².

6 Conclusão

A proposta do trabalho foi melhorar a recomendação de rankings top-n em cenários onde a maioria de feedbacks do usuário fosse negativo, pelo fato de que ao utilizar a filtragem colaborativa clássica, seria recomendado itens similares aos itens que o usuário avaliou (neste cenário, itens semelhantes aos itens considerados ruins para os usuários). Contudo a heurística aplicada não se mostrou muito eficiente, e não obteve resultados melhores que a filtragem colaborativa clássica já utilizada.

Uma das suposições do motivo da recomendação não estar alcançando o objetivo, é que o ranking inicialmente gerado pelo item-based (Seção 3, passo 1) esteja trazendo para o topo do ranking itens muito aleatórios que não tenham nenhuma relevância para o usuário. Logo, um ranking inicial com carência de itens relevantes, produzirá também um ranking ruim após o reranqueamento (Seção 3, passo 2).

Um ponto não analisado foi a diversidade trazida pela heurística. Segundo [1], ao realizarmos a filtragem colaborativa usual, recomenda-se itens semelhantes aos itens que o usuário mais gostou, ou seja, mais itens com as mesmas características, gerando uma falta de diversidade. Porém, ao recomendar considerando-se os feedbacks negativos, estamos eliminando itens semelhantes aos itens que o usuário já avaliou, logo, recomenda-se itens com características diferentes, introduzindo mais diversidade no ranking retornado. Este é um ponto interessante a ser analisado em trabalhos futuros.

Por fim, entende-se que a utilização do feedback negativo também pode ser utilizada de outras formas não tratadas neste trabalho. Como por exemplo, na abordagem user-based, ao se recomendar um ranking para um usuário u , verificar quais usuários não gostaram dos itens que u não gostou (ou seja, usuários com "desgostos" iguais) e construir um ranking de acordo com os itens que estes mesmos usuários gostaram.

Referências

- [1] Dennis L. Chao, Justin Balthrop, and Stephanie Forrest. Adaptive radio: Achieving consensus using negative preferences. In *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work*, GROUP '05, pages 120–123, New York, NY, USA, 2005. ACM.
- [2] Evgeny Frolov and Ivan Oseledets. Fifty shades of ratings: How to benefit from a negative feedback in top-n recommendations tasks. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys '16, pages 91–98, New York, NY, USA, 2016. ACM.
- [3] Zygmunt Zajac. Goodbooks-10k: a new dataset for book recommendations. *FastML*, 2017.

²https://github.com/rafaatsouza/recsys_finalproj