Particionamento Vertical de Dados em Filtragem Colaborativa

Raul Sena Ferreira[1], **Julio C. B. G. Almeida**[1],
Kleyton Pontes Cotta[1], **Filipe Braida**[1,2], **Geraldo Zimbrão**[1]

[1]PESC/COPPE – Universidade Federal do Rio de Janeiro (UFRJ)
Rio de Janeiro – P.O. Box: 68511 – Brazil

[2]DCC/IM – Universidade Federal Rural do Rio de Janeiro (UFRRJ)
Nova Iguaçu, RJ – Brazil

{raulsf, barbieri, kpcotta, filipebraida, zimbrao}@cos.ufrj.br

Abstract. *A collaborative filtering-based recommendation system is used to predict user preferences for certain items. However, it is common that databases have a high dimensionality, e.g., there exists a considerable amount of features assigned to the items. In this work we present a way to partition the data vertically, reducing its dimensionality and increasing the speed of the prediction without damaging their accuracy, thereby, we try to create mechanisms for collaborative filtering that can be automated and efficiently.*

Resumo. *Um sistema de recomendação baseado em filtragem colaborativa serve para prever as preferências dos usuários para determinados itens. Porém, é comum que as bases de dados tenham uma grande dimensionalidade, ou seja, existe ali uma quantidade considerável de atributos atribuídos aos itens. Neste trabalho apresentamos uma forma de particionar os dados de forma vertical, diminuindo sua dimensionalidade e aumentando a rapidez da previsão sem prejudicar sua acurácia, desta forma, tentamos criar mecanismos para que a filtragem colaborativa possa ser automatizada e de forma eficiente.*

1. Introdução

Sistemas de Recomendação são técnicas e ferramentas usadas para sugerir itens personalizados baseados nos interesses dos usuários dentro de um contexto. Essas sugestões referem-se aos vários processos de tomada de decisão, como, que itens comprar ou que música ouvir, ou ainda, que notícias online ler.[Ricci et al. 2011]

Sistemas de recomendação geralmente são usados para tentar resolver o problema de realizar escolhas entre as várias alternativas dentre um grande volume e variedade de dados existentes. Os autores do primeiro sistema de recomendação, denominado Tapestry[Goldberg et al. 1992], criaram a expressão ”filtragem colaborativa”, um tipo de sistema específico no qual a filtragem da informação é realizada pela colaboração entre os grupos interessados.[Reategui and Cazella 2005]

A filtragem colaborativa é uma das abordagens mais bem sucedidas em sistemas de recomendação, essa abordagem visa recomendar um item para um usuário baseado em itens previamente avaliados por outros usuários do sistema. Uma possível abordagem seria a modelagem de uma matriz R de dimensão N x M, onde N representa os usuários e M os itens, e cada posição de R[i,j] contém uma nota ou vazio.[Braida et al. 2015]

Em um sistema comum, geralmente os usuários fornecem as recomendações, essas informações capturadas são utilizadas pelo sistema para apresentá-las para os grupos de indivíduos considerados potenciais interessados para esse tipo de recomendação. Um dos grandes desafios deste tipo de sistema é fazer com que a informação de recomendação previamente inserida seja igual ou muito semelhante às informações de quem está recebendo a recomendação.

Este trabalho tem como objetivo realizar uma abordagem baseada em particionamento vertical dos itens baseados em seus gêneros com o propósito de particionar os itens de forma eficiente, sem se importar com o tipo de conteúdo que estamos recomendando, visando ter bons resultados em um cenário de filtragem colaborativa.

O trabalho é organizado da seguinte forma: A Seção 2 mostra alguns trabalhos relacionados na área, a Seção 3 descreve o problema e a proposta deste trabalho. Já a Seção 4 mostra a metodologia abordada, os experimentos realizados e seus respectivos resultados. E na última seção apresentamos as conclusões extraídas de todo o trabalho realizado.

2. Trabalhos Relacionados

Um dos primeiros a trabalhar com filtragem colaborativa, lidando com uma grande quantidade de itens, foi o grupo Movie Lens no sistema *Usenet news*[Resnick et al. 1994], o trabalho consistiu em usar a correlação de Pearson para identificar similaridade entre os usuários. Outra abordagem de filtragem colaborativa foi explorada em [O'Connor and Herlocker 1999], onde foi usado clusterização dos itens contidos também na base de dados do movie lens, porém algumas restrições foram feitas neste trabalho, como descartar qualquer filme correlacionado com menos de cinco outros filmes ou descartar qualquer filme com menos de dez votos.

Abordagem de recomendação usando o algoritmo IRSVD (*Improved Regularized Singular Value Decomposition*), foi utilizada em [Paterek 2007], neste trabalho a clusterização foi feita usando o K-Means, houve também um pós-processamento com SVD(Singular Value Decomposition) e K-NN, outros algoritmos baseados no SVD também são usados, os resultados apresentados são baseados no MAE (*Mean Absolute Error*) e no RMSE (*Root Mean Squared Error*).

3. Motivação e Objetivo

Este capítulo descreverá o problema de existir uma grande variedade de informações para ser tratada na filtragem colaborativa e em seguida descreverá a solução proposta e sua respectiva metodologia. Inicialmente abordaremos a motivação de se particionar os dados verticalmente e em seguida será mostrado um *workflow* sobre a solução proposta e no final, iremos discutir cada passo desse processo.

A proposta deste trabalho é particionar os dados verticalmente diminuindo consideravelmente a quantidade de dados necessários para o processamento e ao mesmo tempo aumentando a acurácia da previsão e com isso contribuir com formas de automatizar o processo de filtragem colaborativa.

3.1. Definição do problema

Particionar os dados significa reduzir a dimensionalidade de espaço em uma base de dados para pequenos conjuntos com espaços de dimensionalidade menores. Em uma aplicação prática, isso significa considerar menos atributos ligados a um item e processar a recomendação dos dados de forma mais rápida, já que a quantidade de atributos ligados a esses dados que originalmente era muito grande, se torna bem menor.

O problema de termos uma grande quantidade de dados e atributos em um cenário de filtragem colaborativa pode ser melhor entendido se imaginarmos a seguinte situação, que apesar de ser hipotética neste texto, é algo bem corriqueiro na maioria das lojas virtuais de médio e grande porte. Imagine se nesta loja virtual existem 10000 tipos de sapatos diferentes (itens) e existem dados de venda desses sapatos referentes a 1000 usuários, sendo assim, temos então uma matriz esparsa de 1000x10000 para analisar, porém, se tivermos 10 atributos ligados a estes itens (número, cor, tipo, etc) essa matriz teria então 10 dimensões. O que faz com que a quantidade de cálculos a serem considerados seja 1000x10000 elevado a 10.

O caso descrito acima, mostra que para um caso simples, o processo de recomendação pode se tornar algo lento e dispendioso, sem contar que existem inúmeros casos em que o número de itens, usuarios e atributos é muito maior do que o citado no exemplo, o que torna-se um grande problema quando estamos no cenário supervisionado, o que faz surgir outra necessidade, a de se automatizar o processo de recomendação, o que na prática, significa redução de custo.

3.2. Proposta

Se queremos criar um modelo de filtragem colaborativa, precisamos saber lidar com duas coisas importantes, quantidade de dados a ser processada, já que esta pode ser muito grande e com uma quantidade muito grande de atributos ligados a cada item e a acurácia da previsão destes items.

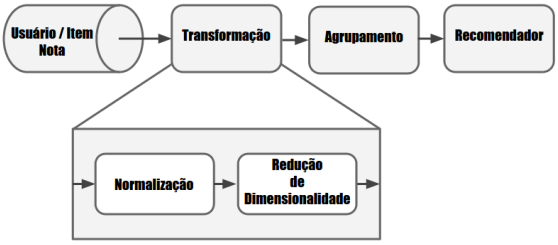

Figure 1. Workflow da proposta.

A figura 1 ilustra a metodologia usada neste trabalho, onde propomos um modelo de particionamento e previsão dos dados, visando diminuir sua dimensionalidade e assim, com poucos itens, poucos usuários e poucos atributos poderemos diminuir consideravelmente o tempo de processamento de uma previsão. Também será usado um algoritmo de clusterização para fazer com que os itens que compartilham um mesmo atributo possam ficar em um mesmo grupo, aumentando as chances de uma recomendação mais precisa e direcionada.

O modelo também aplica transformações na matriz, que é naturalmente esparsa, para reduzir sua dimensionalidade, transformando-a em outra matriz mais densa, através de decomposições. Existem vários métodos para decompor e reduzir a dimensionalidade de uma matriz, como PCA, SVD e outros[Koren et al. 2009].

Em resumo, a proposta deste trabalho é particionar os dados de maneira vertical, aplicar um algoritmo de clusterização, agrupar pelos itens e utilizar um método para predição em cima dos grupos gerados por essa clusterização.

4. Experimentos

Para realizar os experimentos utilizamos o K-fold validation, com K = 10, este tipo de validação consiste em criar K divisões da amostra original de forma aleatoria, comumente chamada de fold, onde cada fold alternadamente é usado como teste e os demais folds restantes são usados como treinamento. O critério de medida usado foi o MAE (Mean Absolute Error)[Goldberg et al. 2001] que é o calculo da média da diferença absoluta entre as previsões e as reais notas dadas e o RMSE (Root Mean Squared Error), que é a raiz quadrada do MAE ao quadrado.

4.1. Base de dados

A base de dados usada neste trabalho foi a coletada do recomendador online do Movie Lens[1], contendo 100 mil notas dadas por 943 usuários para 1682 filmes, onde cada usuário votou em pelo menos 20 filmes e estas notas poderiam variar de 1 a 5, onde 1 é ruim e 5 é excelente.

4.2. Metodologia e Organização dos Experimentos

Para avaliar a proposta deste trabalho foram realizados dois experimentos:

O primeiro consiste em separar os dados por gênero, treinar um modelo para cada gênero em cada fold, selecionar o fold que obteve o melhor resultado segundo seu MAE e no fim, utilizar o IRSVD (Improved Regularized Singular Value Decomposition) para recomendar os itens.

O segundo consiste em utilizar o PCA (Principal Component Analysis) na base de dados, reduzindo a sua dimensionalidade, seguido por uma clusterização dos itens utilizando o algoritmo K-Means e para cada grupo, é realizado então a predição do filme utilizando o IRSVD.

A base de dados foi convertida para uma matriz esparsa, onde as notas faltantes foram primeiramente preenchidas com 0 e em seguida, normalizada pela média das notas.

4.3. Resultados

Nesta seção serão apresentados os resultados obtidos nos experimentos citados anteriormente. Em seguida, será feita uma análise dos resultados, avaliando os ganhos na utilização da proposta em detrimento dos resultados obtidos originalmente, sem a aplicação do modelo.

4.3.1. Experimento

Para o experimento, utilizamos o k de 2 até 25 para a entrada do algoritmo K-Means, em seguida, para cada modelo criado, foi aplicado 10 vezes o IRSVD, calculando a predição e utilizando o MAE e RMSE como métricas. Os resultados podem ser vistos através da figura 2.

[1]https://movielens.org/

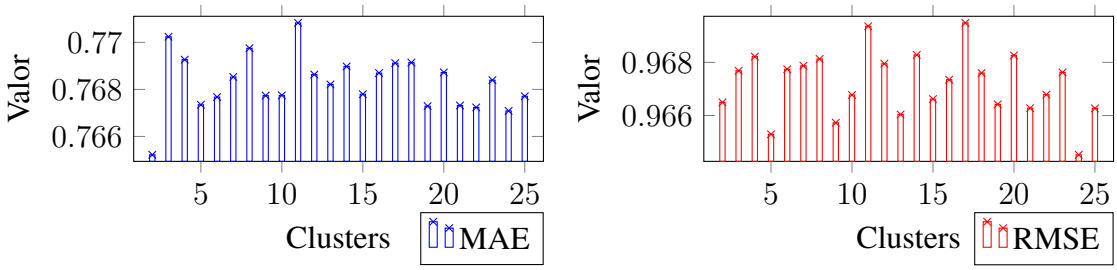

Figure 2. Resultado obtido do MAE e RMSE por Cluster.

Para meio de comparação, foi aplicado 10 vezes o IRSVD para cada gênero da base original obtendo o MAE de **0.968** e RMSE de **1.226**. Deste modo, pelos dados apresentados na tabela 1 em comparação com o *baseline*, vemos que os modelos propostos obtiveram uma melhora em média de **25.99%** para o MAE e **26.75%** para RMSE, podendo ser visto na figura 3.

CLUSTERS	MAE	RMSE	CLUSTERS	MAE	RMSE
2	0.7652	0.9665	14	0.769	0.9683
3	0.7702	0.9677	15	0.7678	0.9666
4	0.7693	0.9682	16	0.7687	0.9673
5	0.7673	0.9653	17	0.7691	0.9695
6	0.7677	0.9677	18	0.7691	0.9676
7	0.7685	0.9679	19	0.7673	0.9664
8	0.7698	0.9681	20	0.7687	0.9683
9	0.7677	0.9657	21	0.7673	0.9663
10	0.7677	0.9668	22	0.7672	0.9668
11	0.7708	0.9694	23	0.7684	0.9676
12	0.7686	0.9679	24	0.7671	0.9645
13	0.7682	0.966	25	0.7677	0.9663

Table 1. Tabela de resultados da aplicação do modelo.

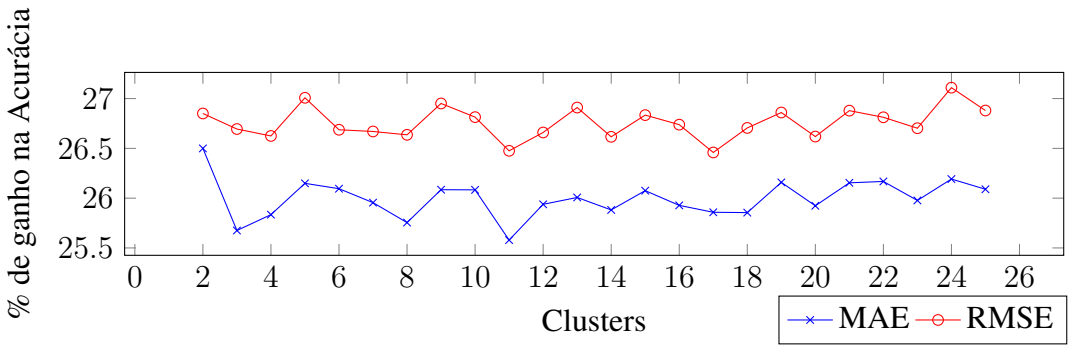

Figure 3. Relação de Melhoria do MAE e RMSE em relação ao Baseline.

5. Conclusão e Trabalhos Futuros

Neste artigo, elucidamos uma visão sobre sistemas de recomendação e seus desafios para gerar resultados eficientes e rápidos, levando em conta a dimensionalidade de informações. Apresentamos a dificuldade de trabalhar com filtragem colaborativa em um ambiente com grande quantidade de dados e atributos, sendo necessário automatizar o processo de recomendação para reduzir os custos sem piorar a predição.

Através da metodologia e organização dos experimentos propostos deste trabalho, podemos concluir que o modelo de particionamento e clusterização dos dados obteve melhor resultado na previsão em detrimento da abordagem original, que considera toda a base separada por gênero. Como trabalho futuro, tentaremos encontrar formas de determinar qual ou quais as melhores divisões em k-grupos e testar outros algoritmos de clusterização que possam trazer resultados melhores.

References

Braida, F., Mello, C. E., Pasinato, M. B., and Zimbrão, G. (2015). Transforming collaborative filtering into supervised learning. *Expert Systems with Applications*, 42(10):4733–4742.

Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70.

Goldberg, K., Roeder, T., Gupta, D., and Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2):133–151.

Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, (8):30–37.

O'Connor, M. and Herlocker, J. (1999). Clustering items for collaborative filtering. In *Proceedings of the ACM SIGIR workshop on recommender systems*, volume 128. Citeseer.

Paterek, A. (2007). Improving regularized singular value decomposition for collaborative filtering. In *Proc. KDD Cup Workshop at SIGKDD'07, 13th ACM Int. Conf. on Knowledge Discovery and Data Mining*, pages 39–42.

Reategui, E. B. and Cazella, S. C. (2005). Sistemas de recomendaçao. In *XXV Congresso da Sociedade Brasileira de Computação. Universidade do Vale do Rio dos Sinos (UNISINOS). São Leopoldo*. Citeseer.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186. ACM.

Ricci, F., Rokach, L., and Shapira, B. (2011). *Introduction to recommender systems handbook*. Springer.